Anomaly searches in jet Physics:

Self-supervision for anomaly detection

Siegen - CRC Young Scientist Meeting

Luigi Favaro - 18/10/2023

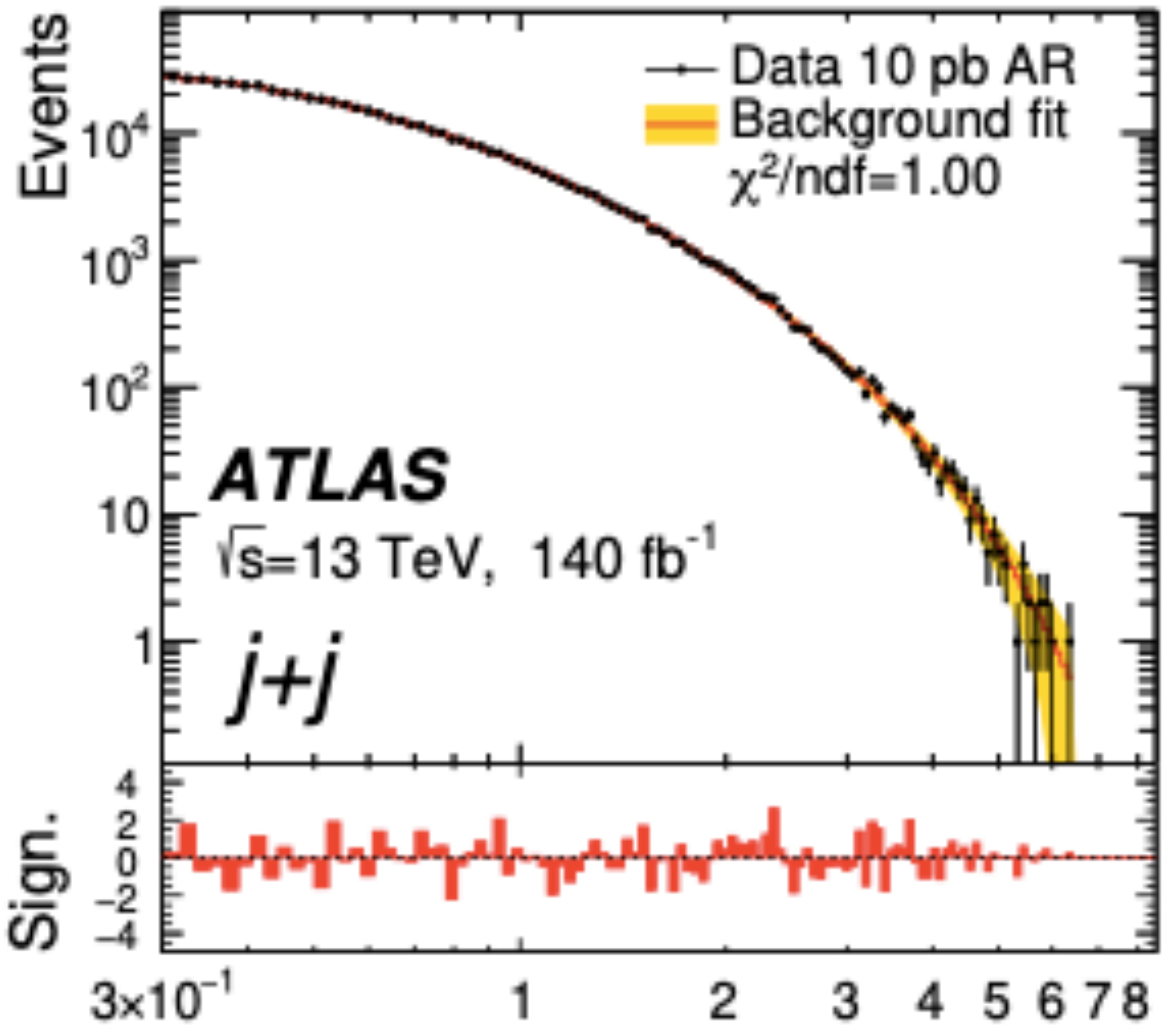
Funded by
DFG Deutsche Forschungsgemeinschaft
German Research Foundation

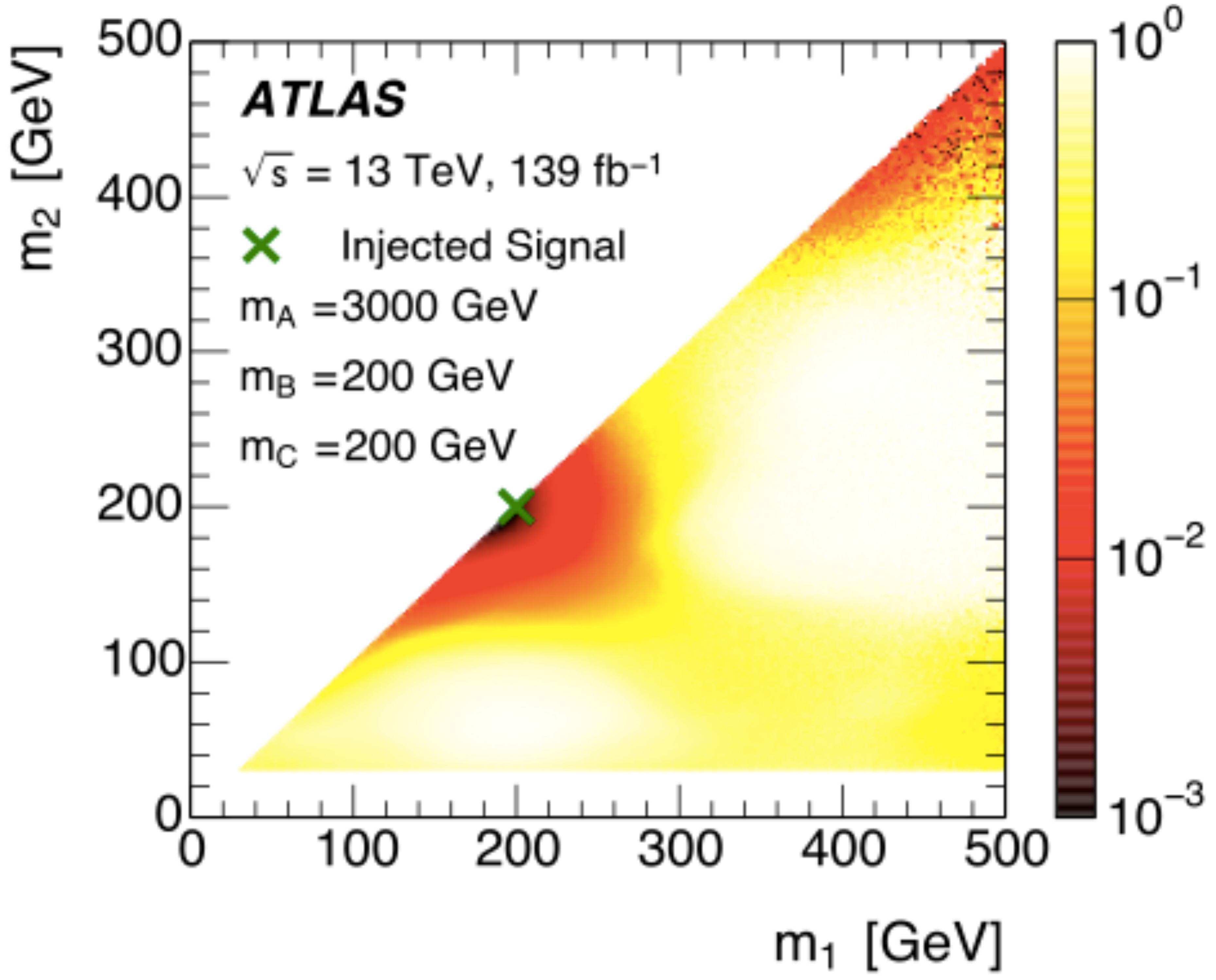
Universität Heidelberg
Zukunft. Seit 1386.

Model-agnostic searches & ML

Deep-learning has demonstrated potential for analyses with...

- no restriction to a few, or even high-level, observables
- the ability to process high-dimensional datasets
- less reliance on simulation





Open questions:

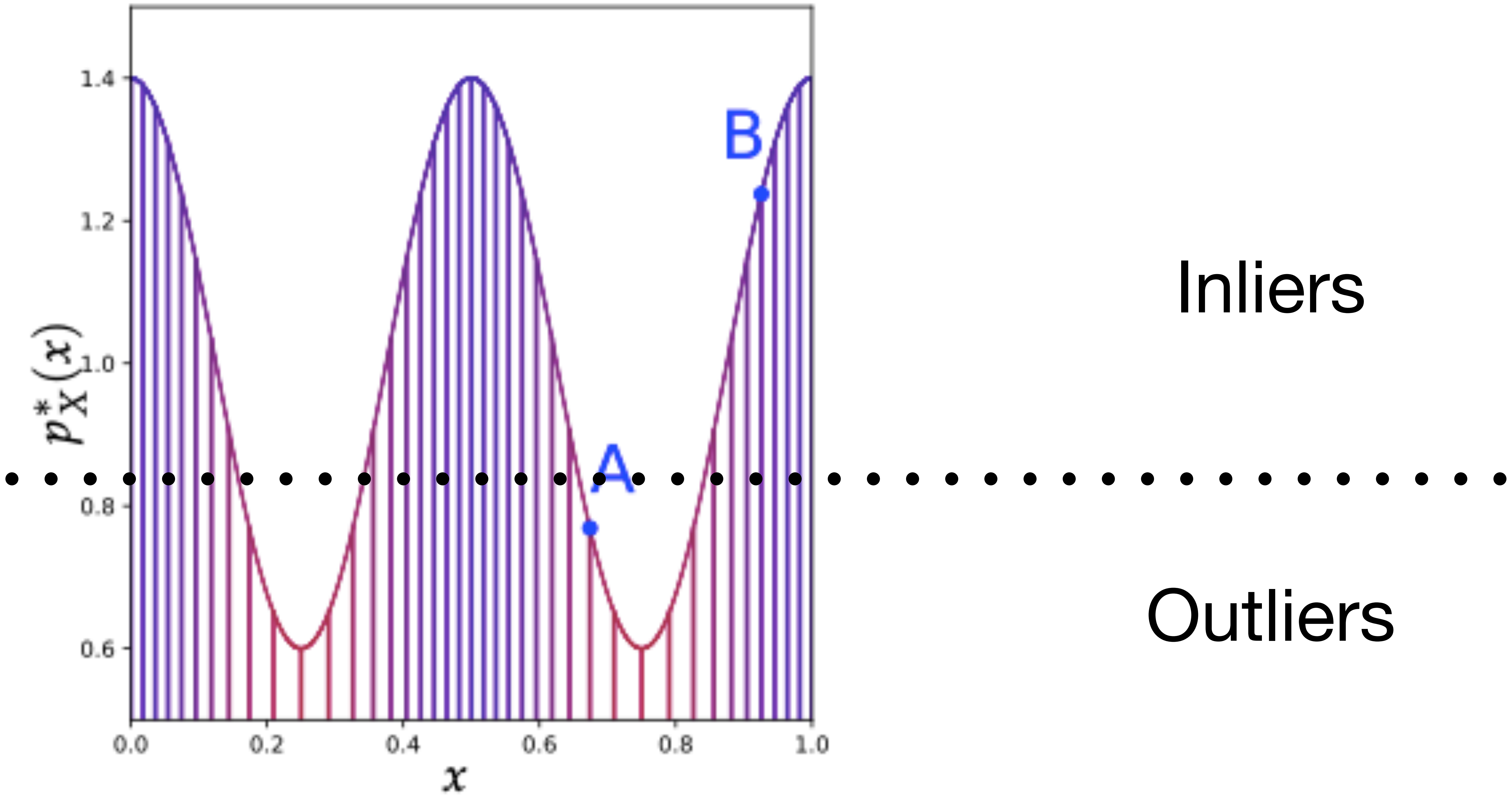
- Physics bias in deep-learning approaches?
- How model-agnostic are the approaches?
- What would an analysis with these tools look like?

Unsupervised anomaly detection

Unsupervised: Ideally no assumptions on signals

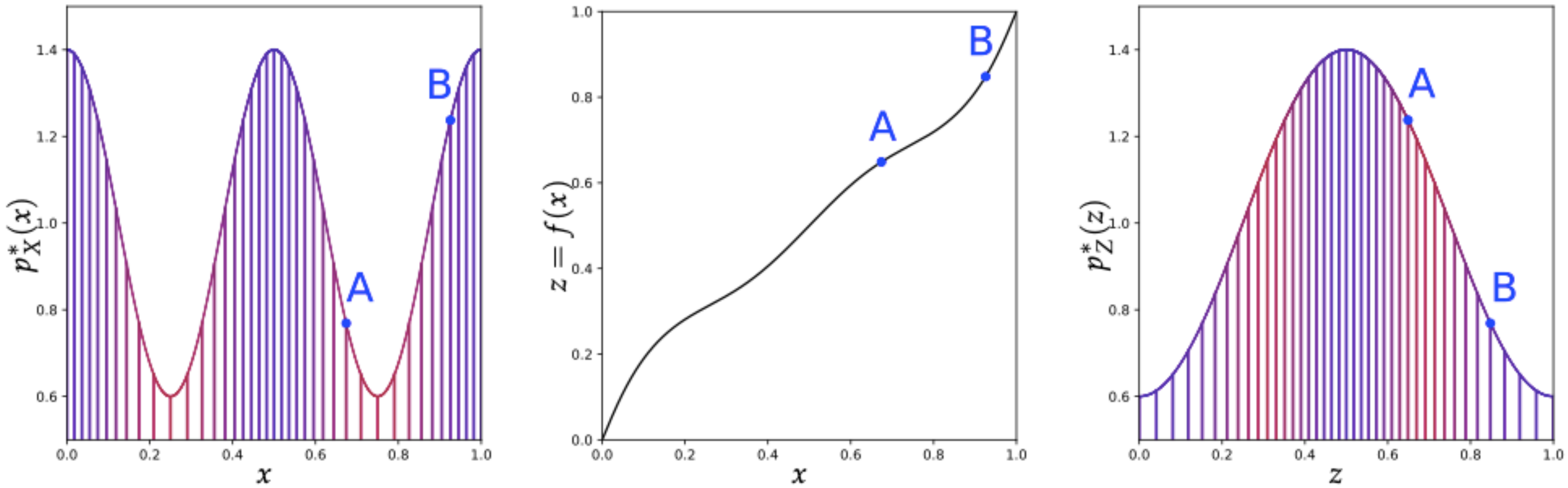
Definition based on density estimate:

$$OOD_\epsilon(x) := \{x \,|\, p(x) < \epsilon\}$$



Unsupervised anomaly detection

However, ordering is not invariant under coordinate transformations



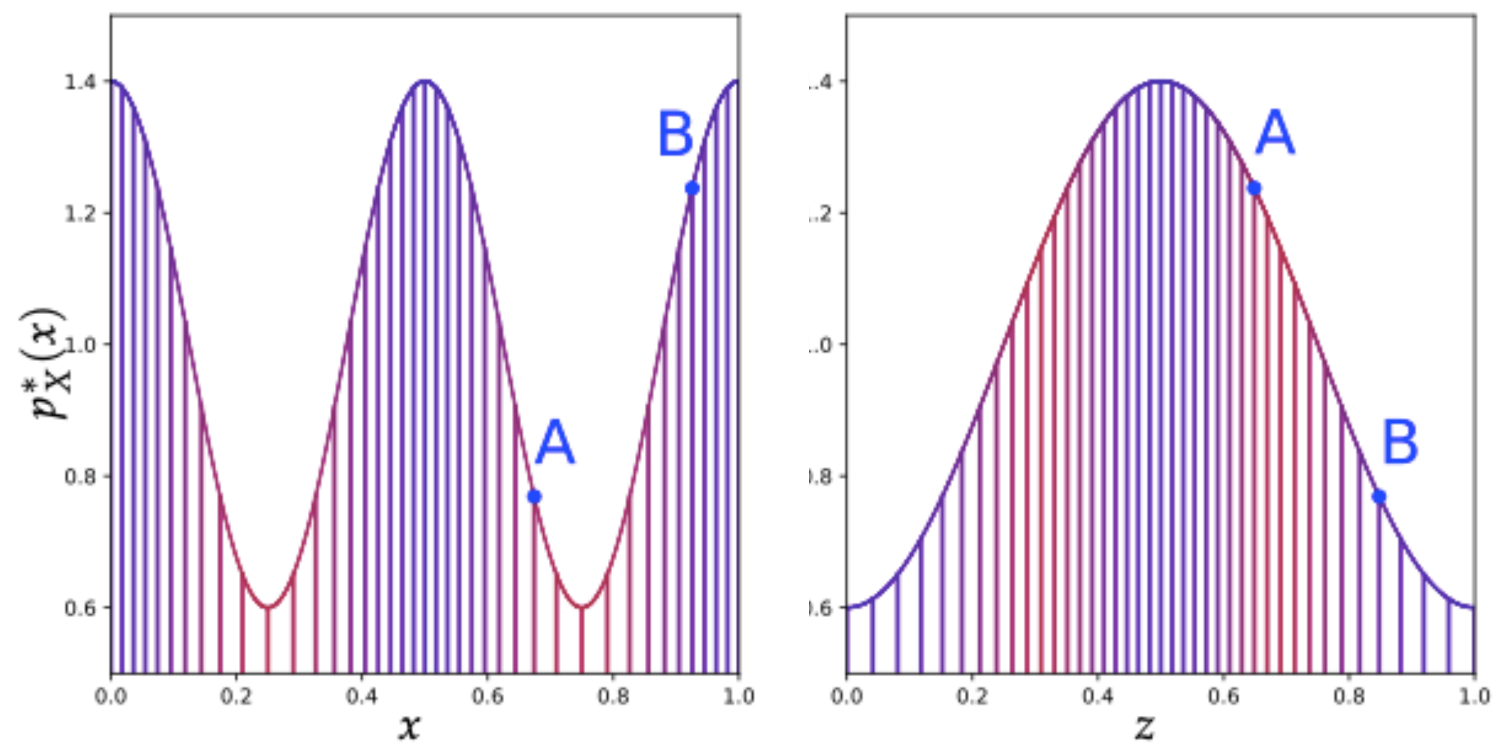
[Perfect density models cannot guarantee anomaly detection - Le Lan C., Dinh L.]

Preprocessing

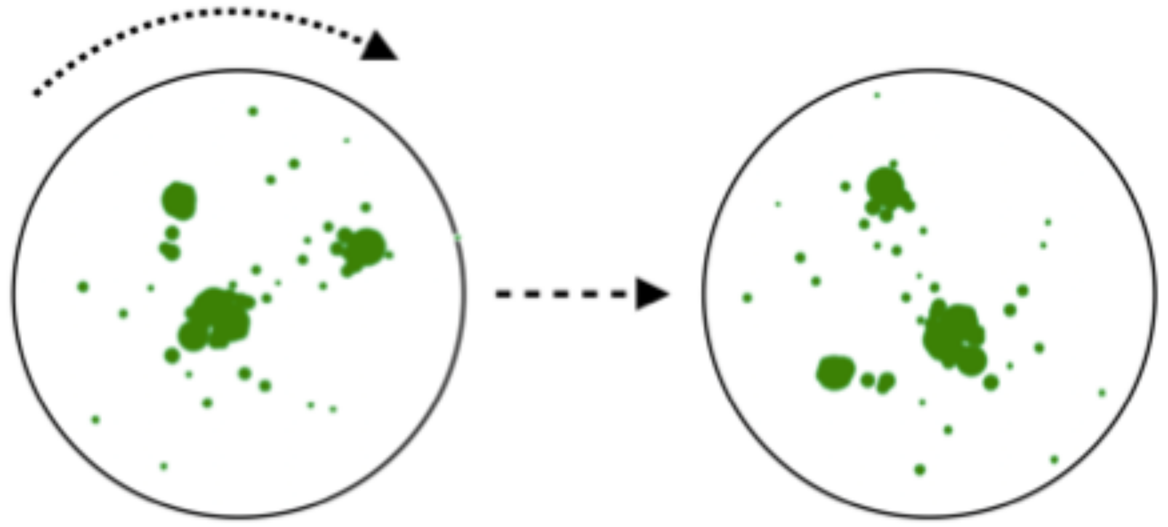
Preprocessing is an important step in any machine learning application

- Guarantees numerical stability;
- Introduce symmetries.

Model dependence



Powerful representations



Self-supervision

Self-supervision

Self-supervision

- Neural Networks are not invariant to physical symmetries in data
- Typically solved through “pre-processing”
- Self-supervision: during training we use pseudo-labels, not truth labels

Self-supervision

- Neural Networks are not invariant to physical symmetries in data
- Typically solved through “pre-processing”
- Self-supervision: during training we use pseudo-labels, not truth labels

Our goal: control the training to ensure we learn physical quantities

What the **representations** should have:

- invariance to certain transformations of the jet/event
- discriminative power

Self-supervision

- Neural Networks are not invariant to physical symmetries in data
- Typically solved through “pre-processing”
- Self-supervision: during training we use pseudo-labels, not truth labels

Our goal: control the training to ensure we learn physical quantities

What the **representations** should have:

- invariance to certain transformations of the jet/event
- discriminative power

- CLR: map raw data to a new representation/observables

JetCLR

[Symmetries, safety, and self-supervision, Dillon B. et al. arXiv:2108.04253]

JetCLR

Dataset: mixture of top and QCD jets

Contrastive Learning paradigm:

- positive pairs: $\{(x_i, x_i')\}$ where x_i' is an augmented version of x_i
- negative pairs: $\{(x_i, x_j) \cup (x_i, x_j')\}$ for $i \neq j$

Augmentation: any transformation (e.g. rotation) of the original jet

[Symmetries, safety, and self-supervision, Dillon B. et al. arXiv:2108.04253]

JetCLR

Dataset: mixture of top and QCD jets

Contrastive Learning paradigm:

- positive pairs: $\{(x_i, x_i')\}$ where x_i' is an augmented version of x_i
- negative pairs: $\{(x_i, x_j) \cup (x_i, x_j')\}$ for $i \neq j$

Augmentation: any transformation (e.g. rotation) of the original jet

Train a Transformer-encoder network to map the data to a new repr. space, $f : \mathcal{J} \to \mathcal{R}$

Loss function:

$$\mathcal{L} = -\log \frac{exp(s(z_i, z_i')/\tau)}{\sum_{x \in batch} \mathbb{I}_{i \neq j}[exp(s(z_i, z_j)/\tau) + exp(s(z_i, z_j')/\tau)]}$$

[Symmetries, safety, and self-supervision, Dillon B. et al. arXiv:2108.04253]

Invariances

Invariances

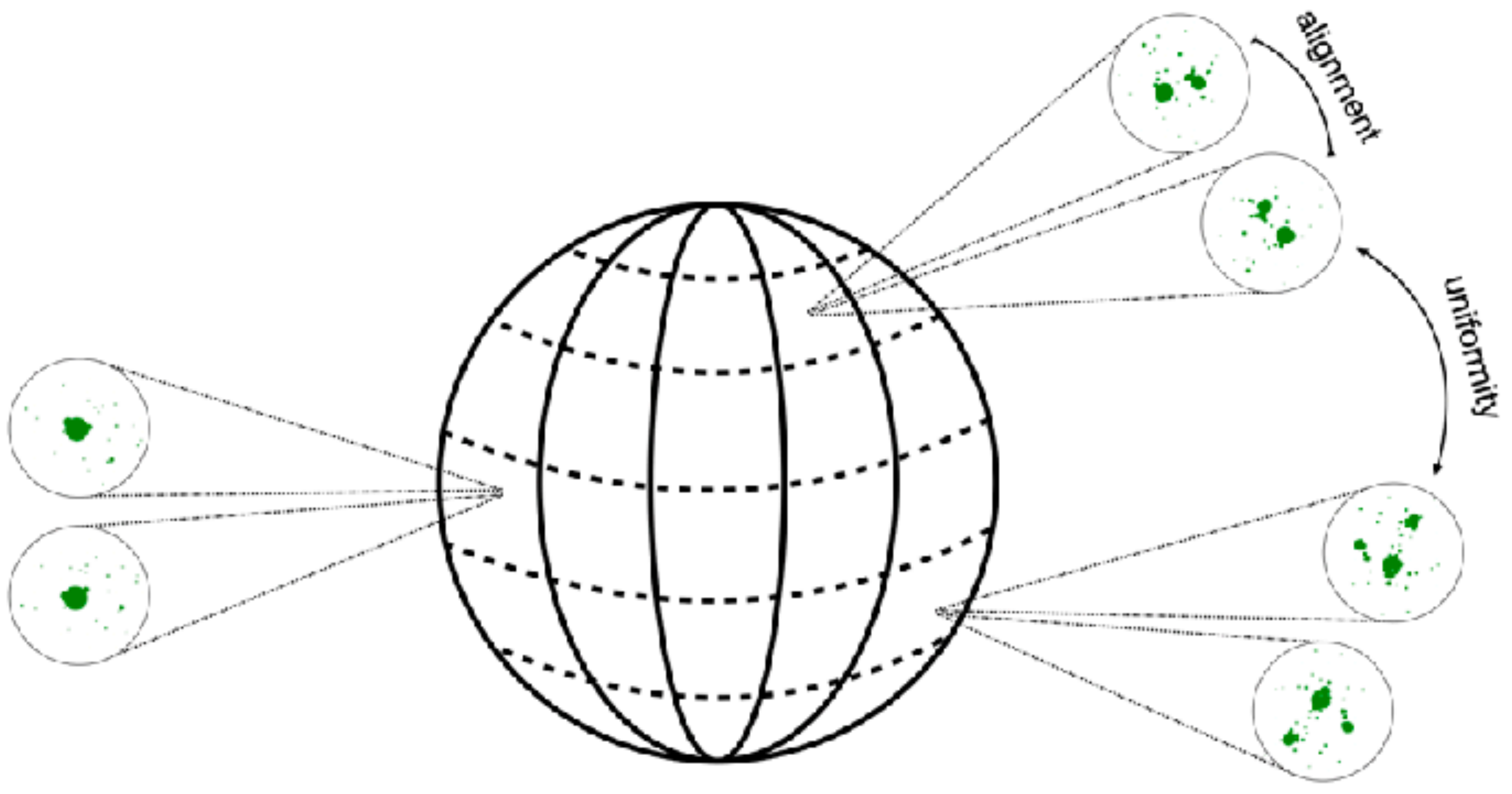
$$\mathscr{L} = -\log \frac{exp(s(z_i, z_i')/\tau)}{\sum_{x \in batch} \mathbb{I}_{i \neq j}[exp(s(z_i, z_j)/\tau) + exp(s(z_i, z_j')/\tau)]}$$

Similarity measure:

$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|}, \qquad z_i = f(x_i)$$

Invariances

$$\mathcal{L} = -\log \frac{exp(s(z_i, z_i')/\tau)}{\sum_{x \in batch} \mathbb{I}_{i \neq j}[exp(s(z_i, z_j)/\tau) + exp(s(z_i, z_j')/\tau)]}$$

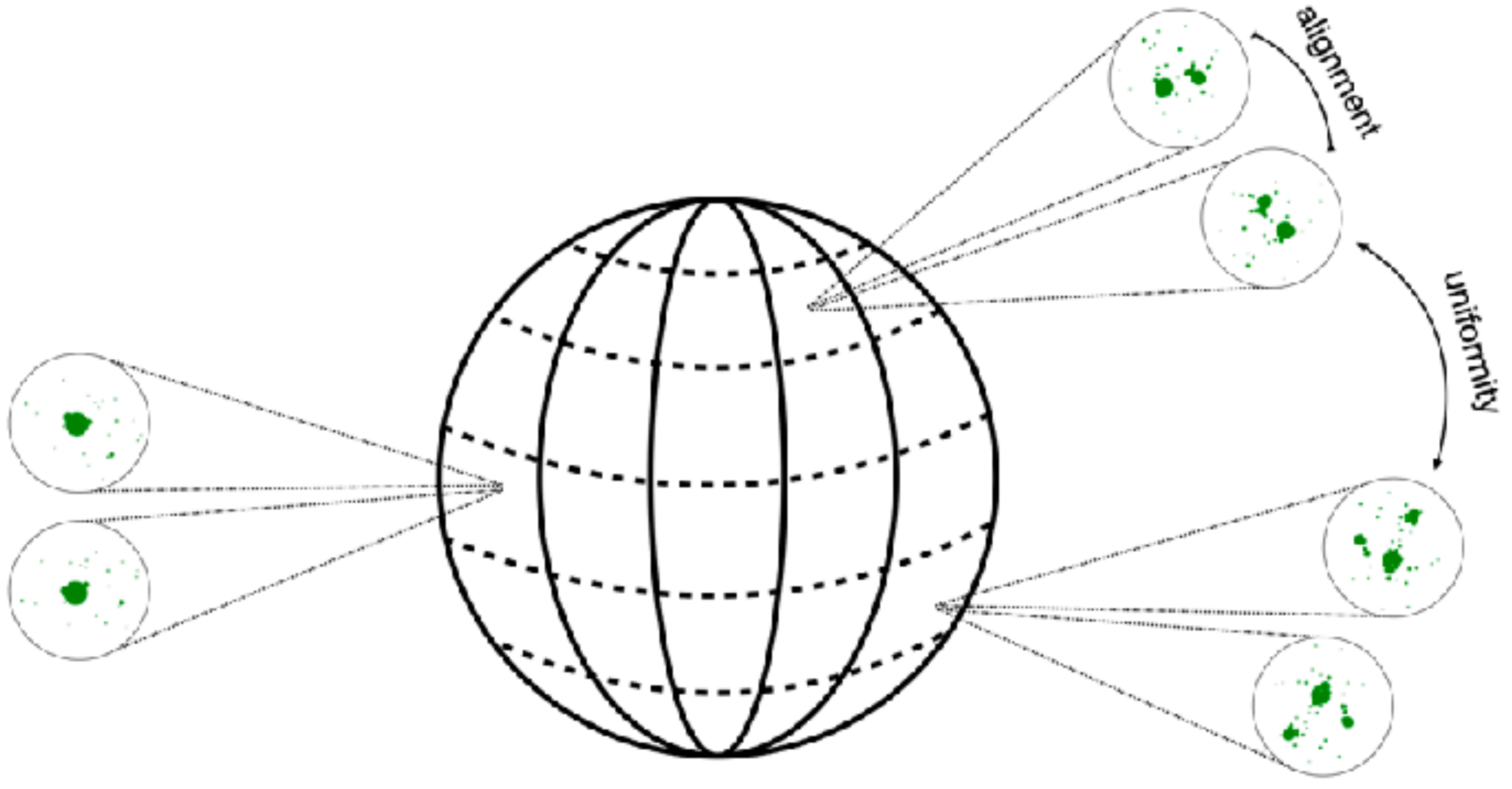


Similarity measure:

$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|}, \qquad z_i = f(x_i)$$

Invariances

$$\mathscr{L} = -\log \frac{exp(s(z_i, z_i')/\tau)}{\sum_{x \in batch} \mathbb{I}_{i \neq j}[exp(s(z_i, z_j)/\tau) + exp(s(z_i, z_j')/\tau)]}$$



Similarity measure:

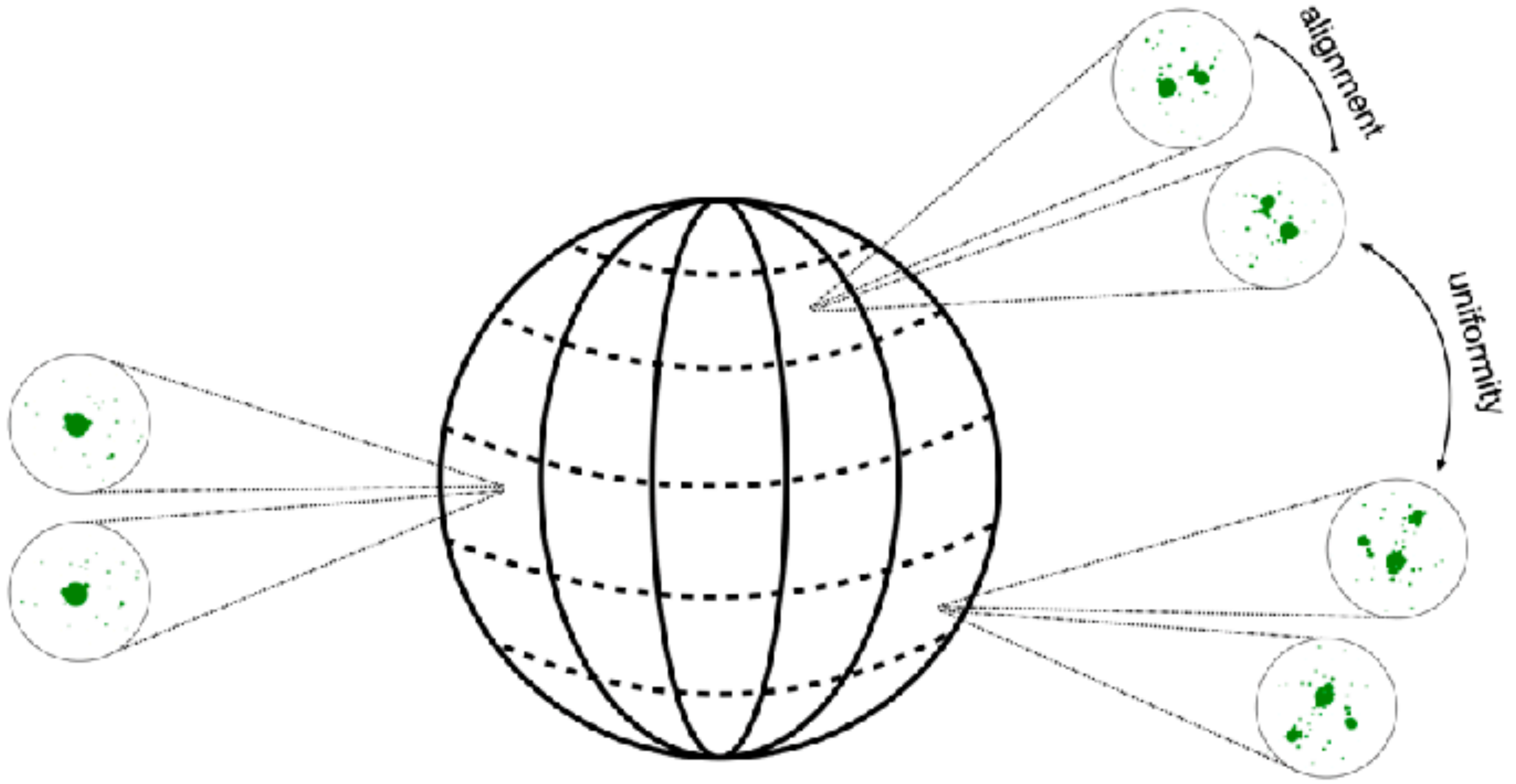
$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|}, \qquad z_i = f(x_i)$$

Applied augmentations:

- rotations
- translations
- collinear splittings
- low p_T smearing

Invariances

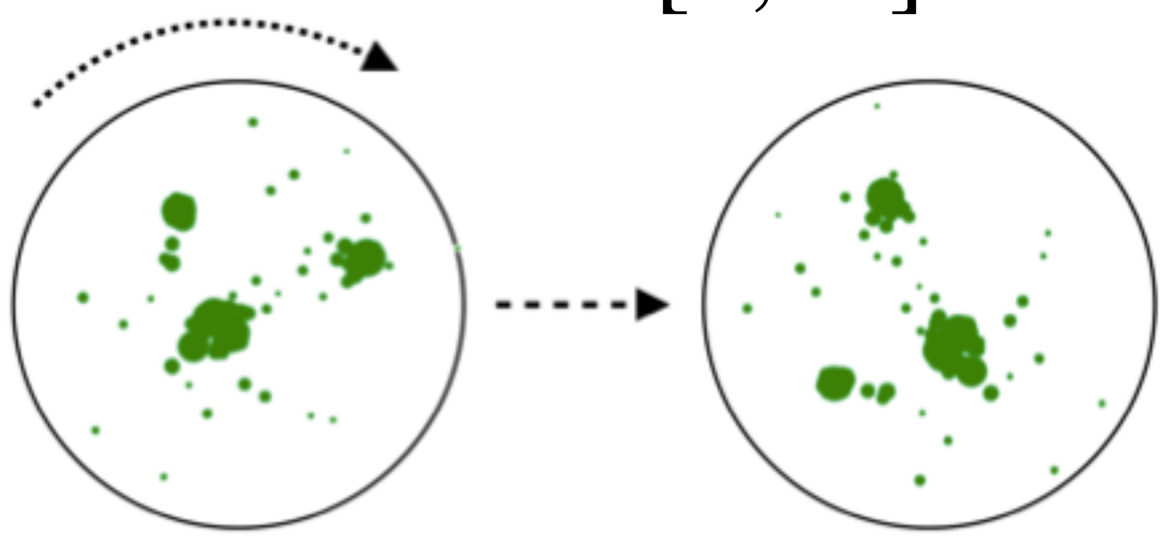
$$\mathscr{L} = -\log \frac{exp(s(z_i, z_i')/\tau)}{\sum_{x \in batch} \mathbb{I}_{i \neq j}[exp(s(z_i, z_j)/\tau) + exp(s(z_i, z_j')/\tau)]}$$



Similarity measure:

$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|}, \qquad z_i = f(x_i)$$

rotation in $[0,2\pi]$

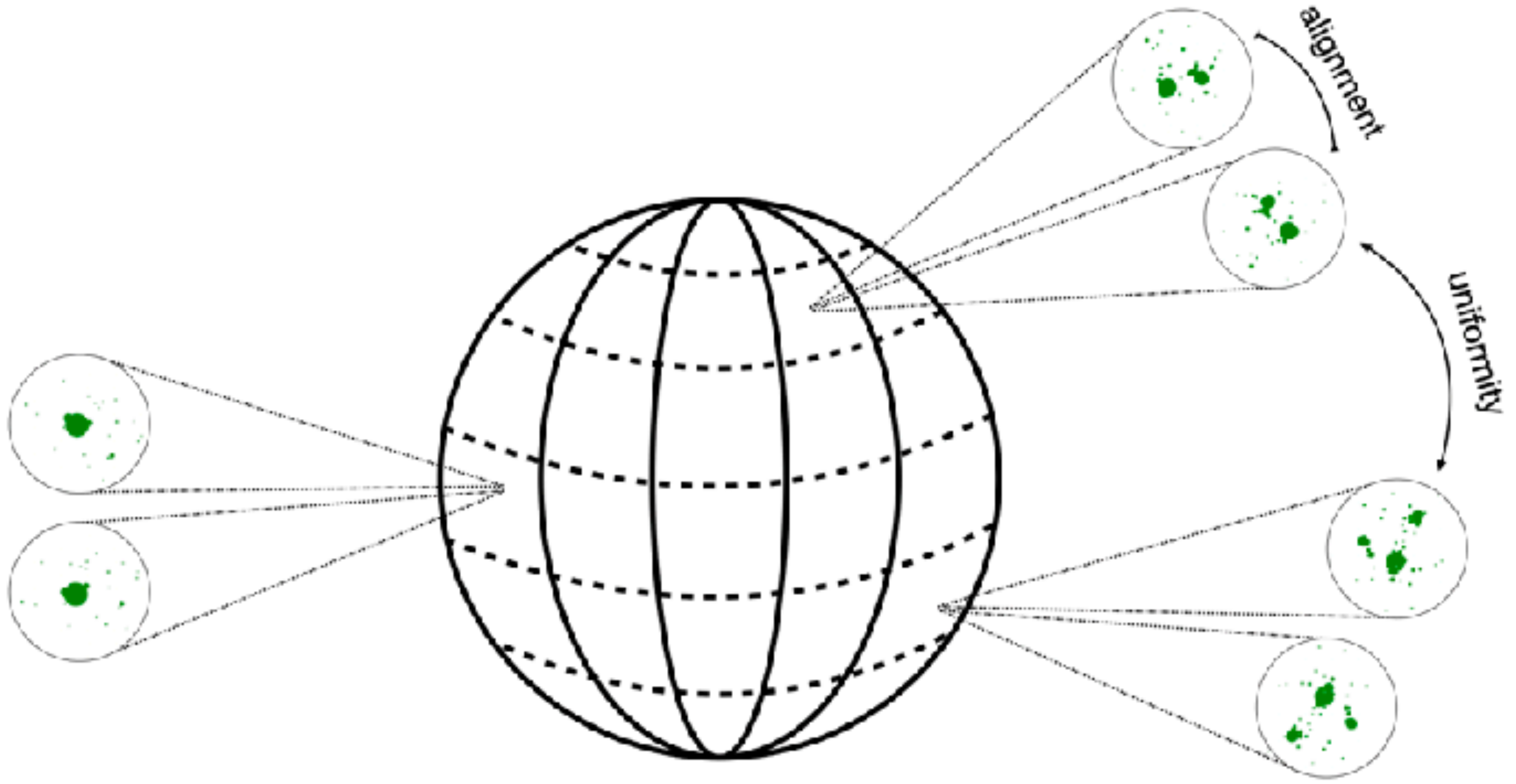


Applied augmentations:

- rotations
- translations
- collinear splittings
- low p_T smearing

Invariances

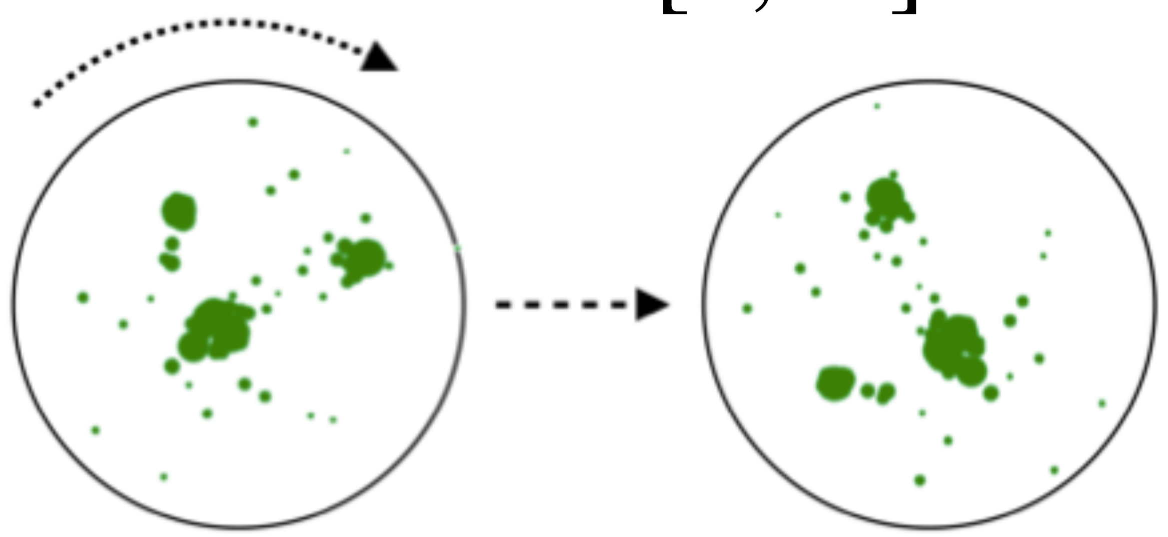
$$\mathcal{L} = -\log \frac{exp(s(z_i, z_i')/\tau)}{\sum_{x \in batch} \mathbb{I}_{i \neq j}[exp(s(z_i, z_j)/\tau) + exp(s(z_i, z_j')/\tau)]}$$



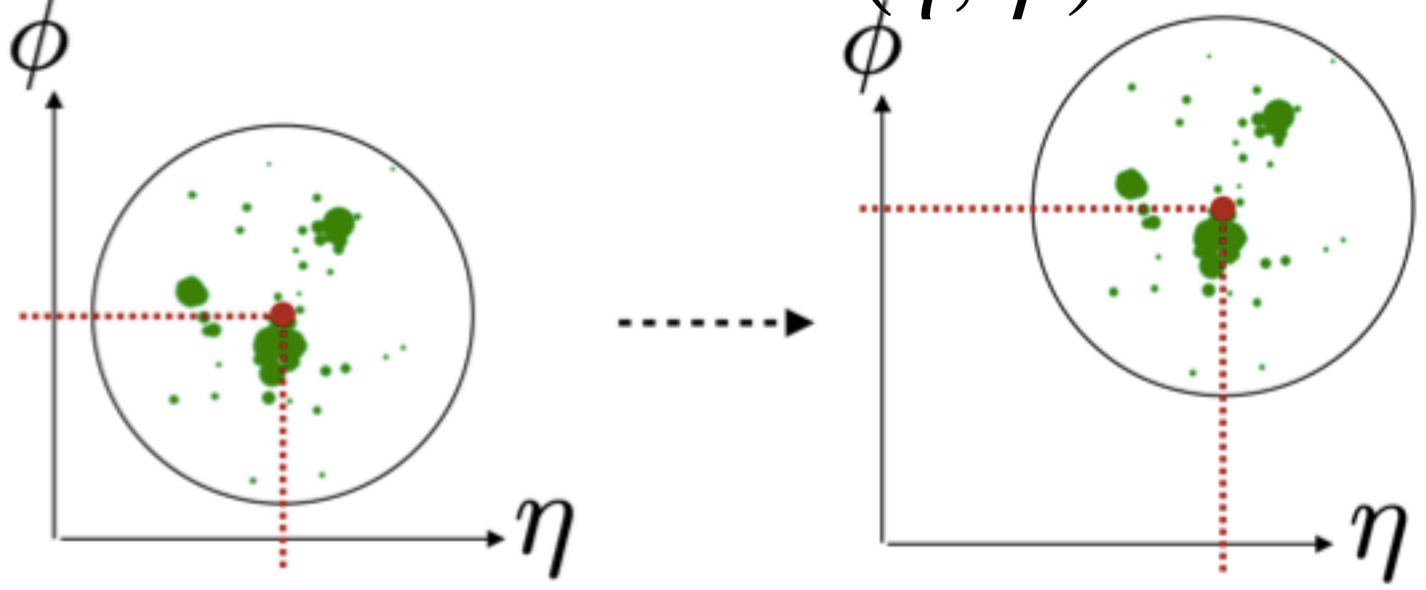
Similarity measure:

$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|}, \qquad z_i = f(x_i)$$

rotation in $[0,2\pi]$



translation in (η, ϕ)

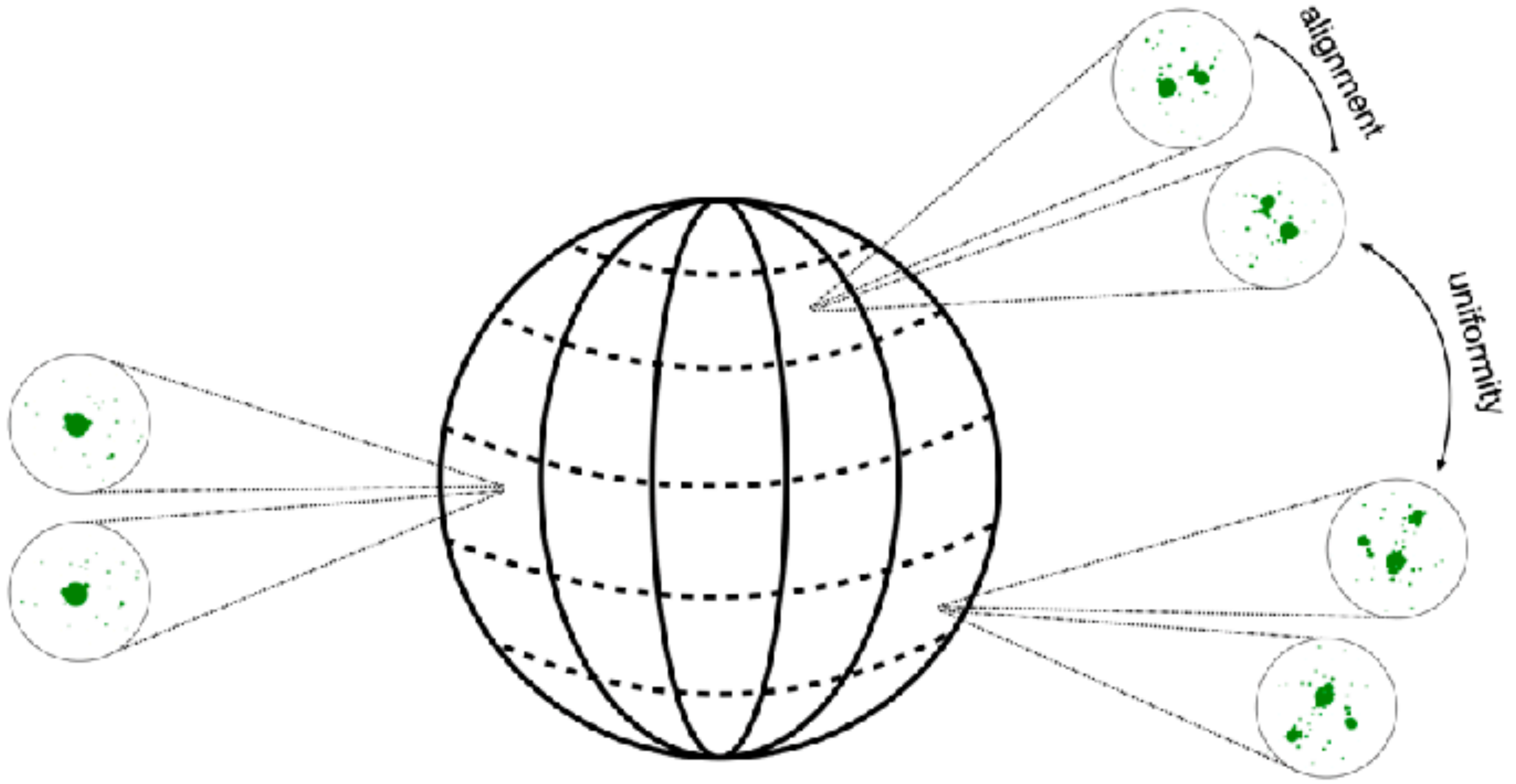


Applied augmentations:

- rotations
- translations
- collinear splittings
- low p_T smearing

Invariances

$$\mathcal{L} = -\log \frac{exp(s(z_i, z_i')/\tau)}{\sum_{x \in batch} \mathbb{I}_{i \neq j}[exp(s(z_i, z_j)/\tau) + exp(s(z_i, z_j')/\tau)]}$$



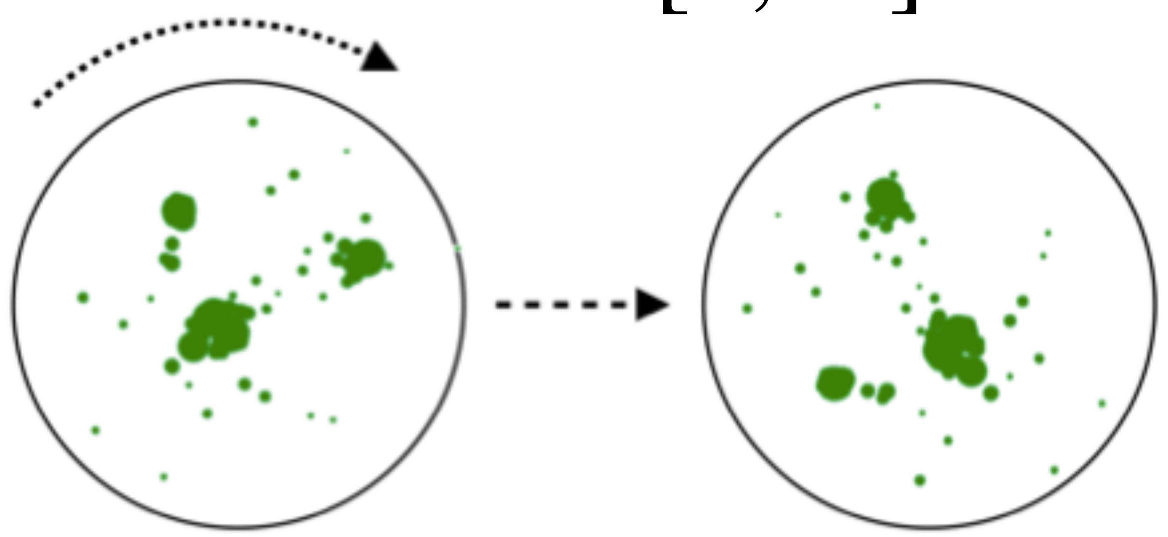
Similarity measure:

$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|}, \qquad z_i = f(x_i)$$

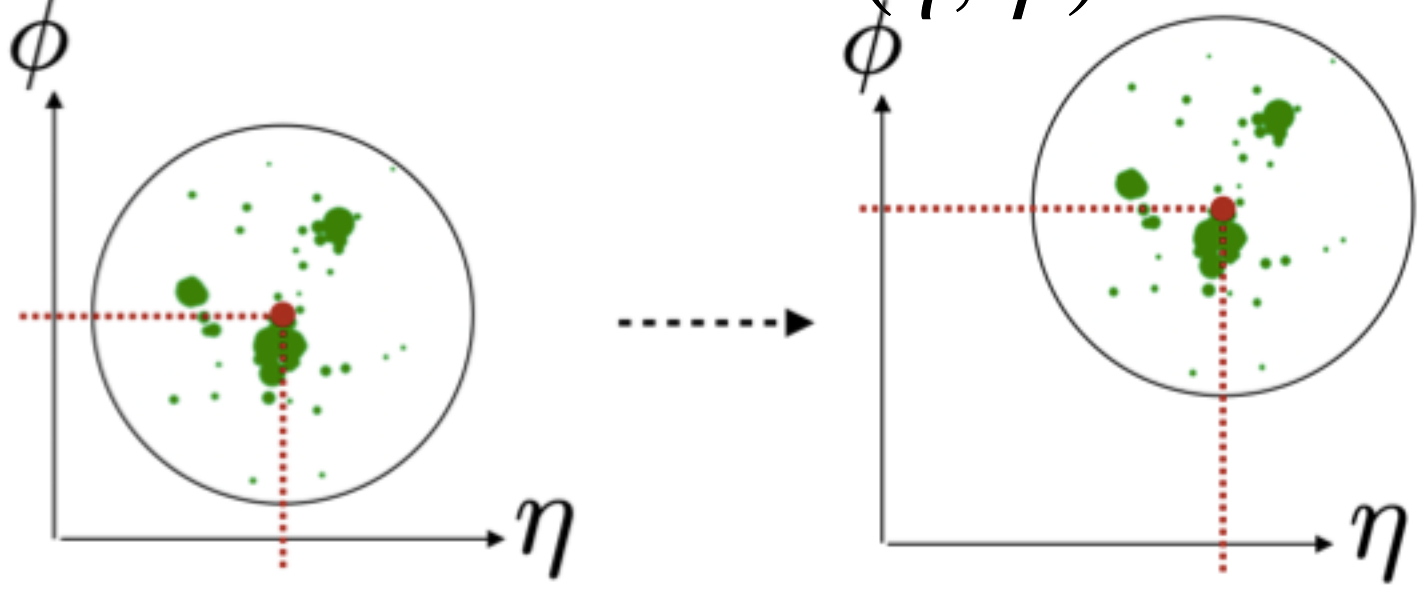
Applied augmentations:

- rotations
- translations
- collinear splittings
- low p_T smearing

rotation in $[0,2\pi]$



translation in (η, ϕ)



random split of constituents

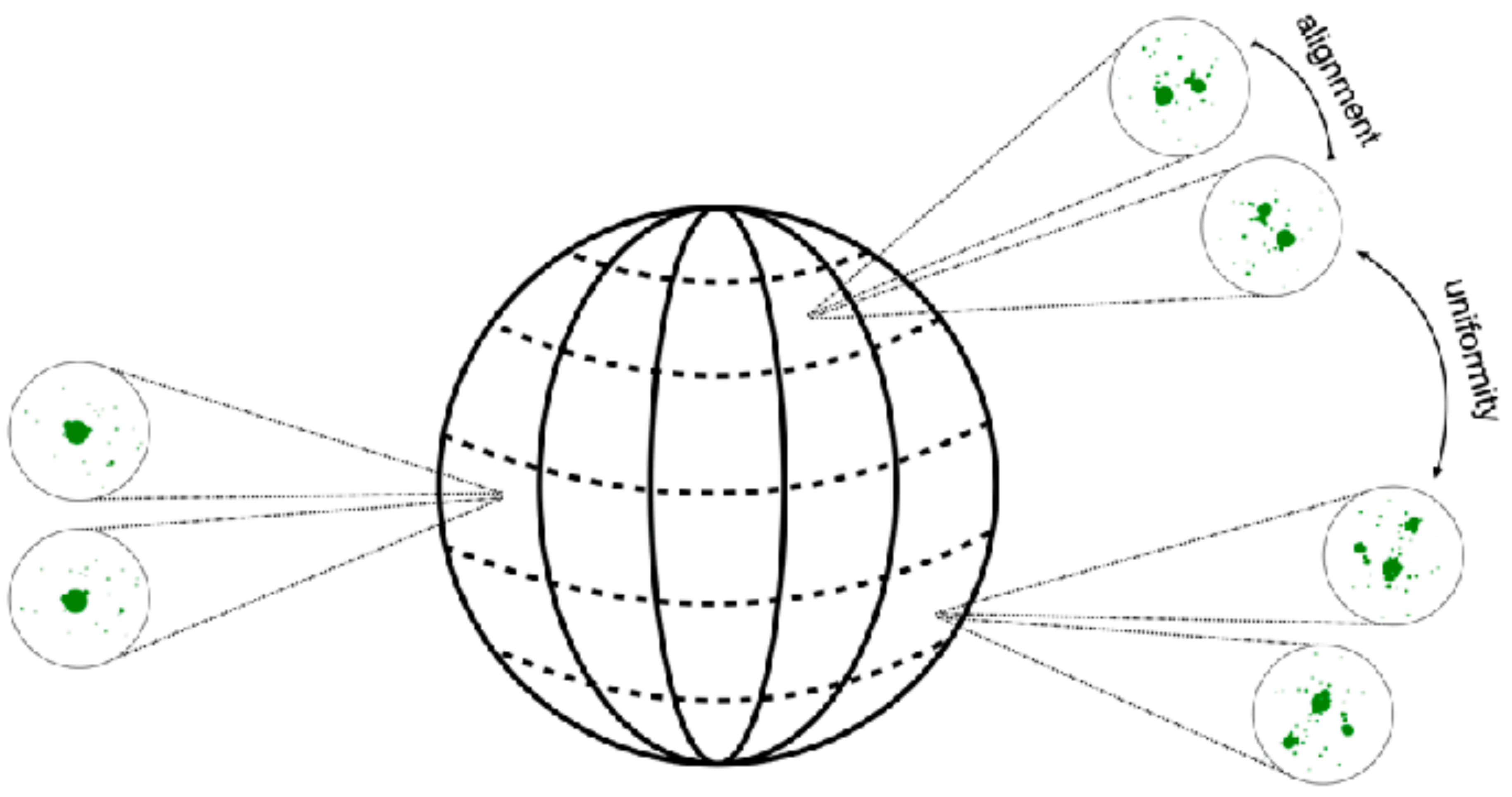
$$p_{T,a} + p_{T,b} = p_T$$

$$\eta_a = \eta_b = \eta$$

$$\phi_a = \phi_b = \phi$$

Invariances

$$\mathcal{L} = -\log \frac{exp(s(z_i, z_i')/\tau)}{\sum_{x \in batch} \mathbb{I}_{i \neq j}[exp(s(z_i, z_j)/\tau) + exp(s(z_i, z_j')/\tau)]}$$



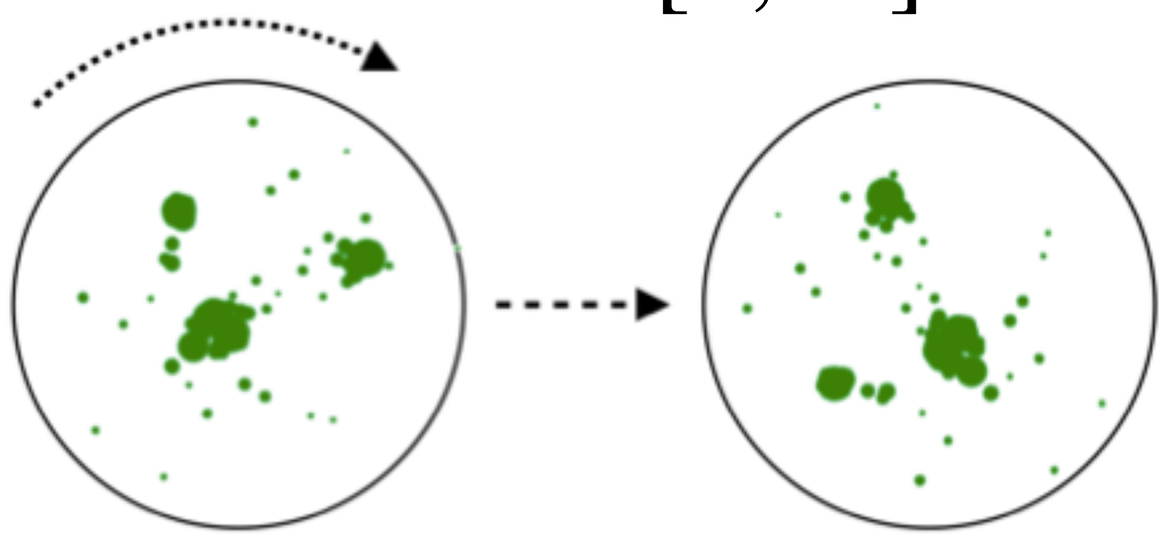
Similarity measure:

$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|}, \qquad z_i = f(x_i)$$

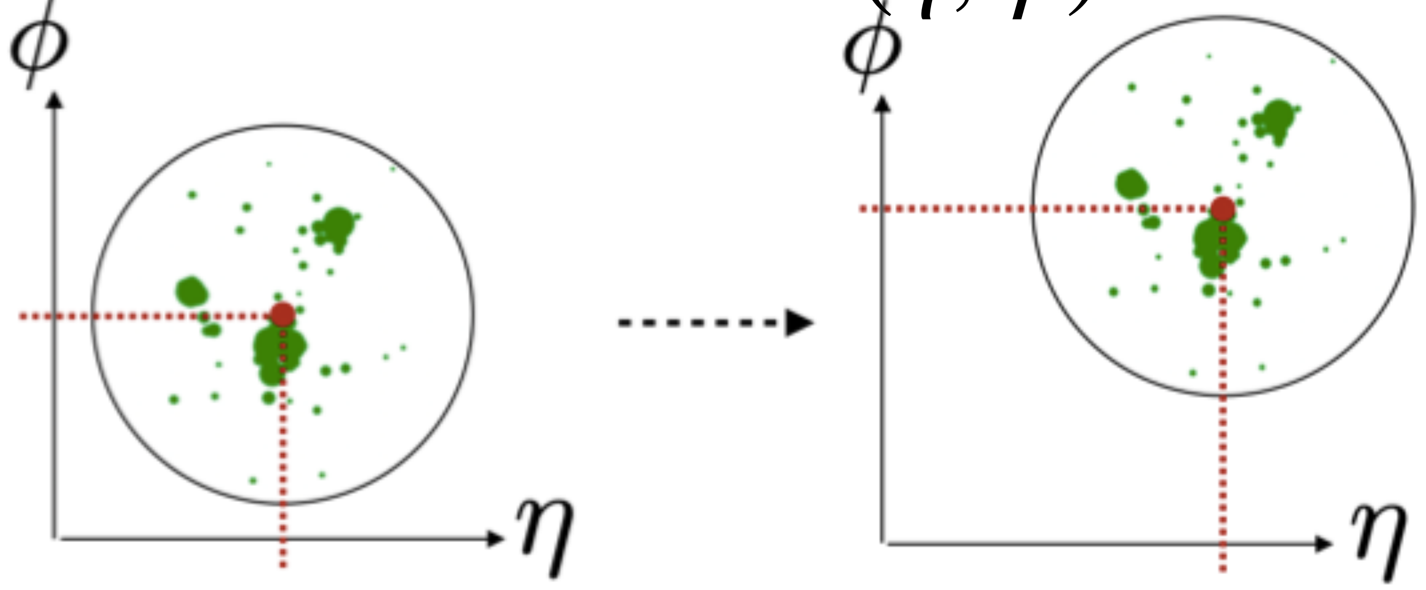
Applied augmentations:

- rotations
- translations
- collinear splittings
- low p_T smearing

rotation in $[0,2\pi]$



translation in (η, ϕ)



random split of constituents

$$p_{T,a} + p_{T,b} = p_T$$

$$\eta_a = \eta_b = \eta$$

$$\phi_a = \phi_b = \phi$$

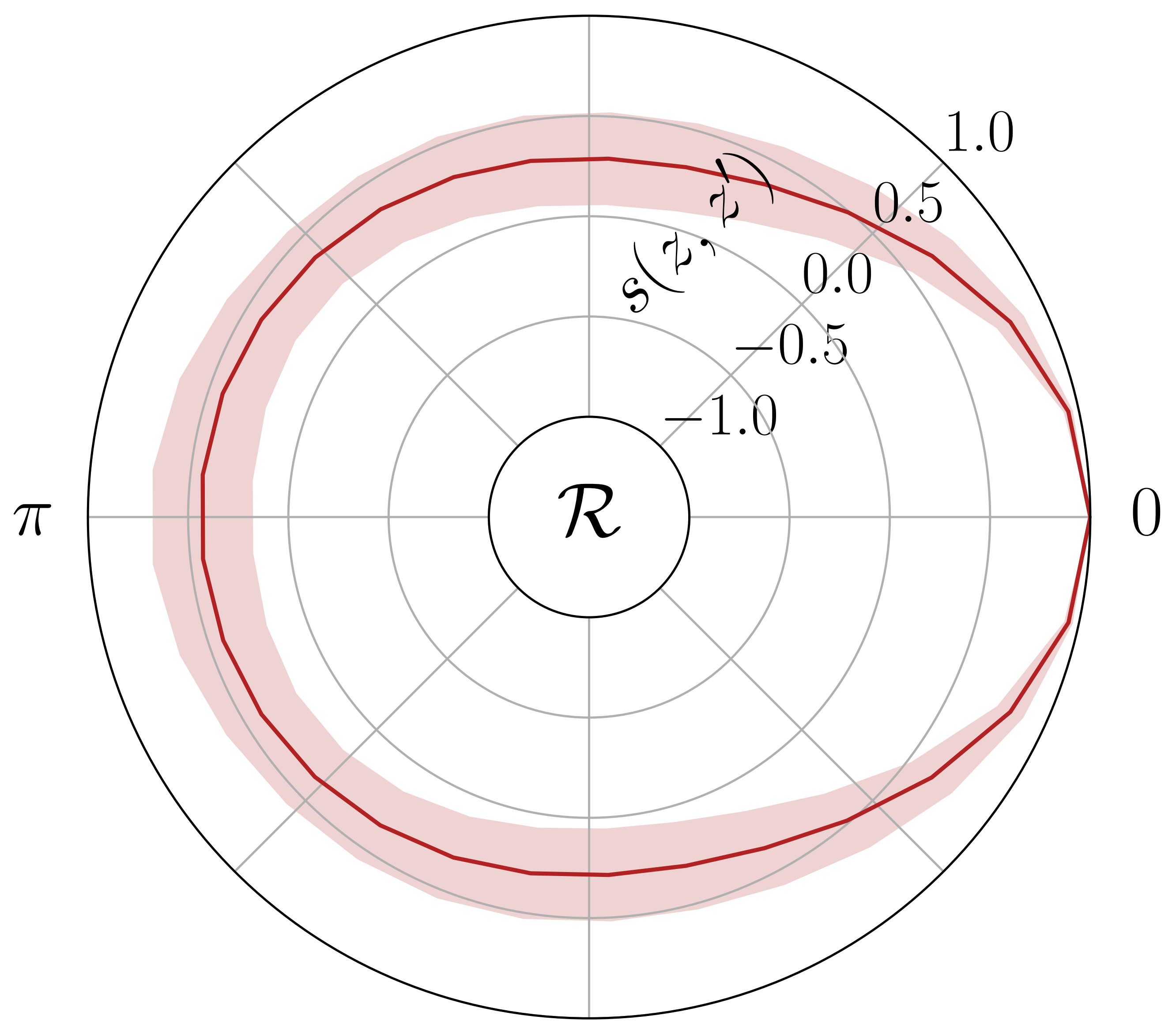
re-sampling of (η, ϕ)

$$\eta' \sim \mathcal{N}\left(\eta, \frac{\Lambda_{soft}}{p_T}\right)$$

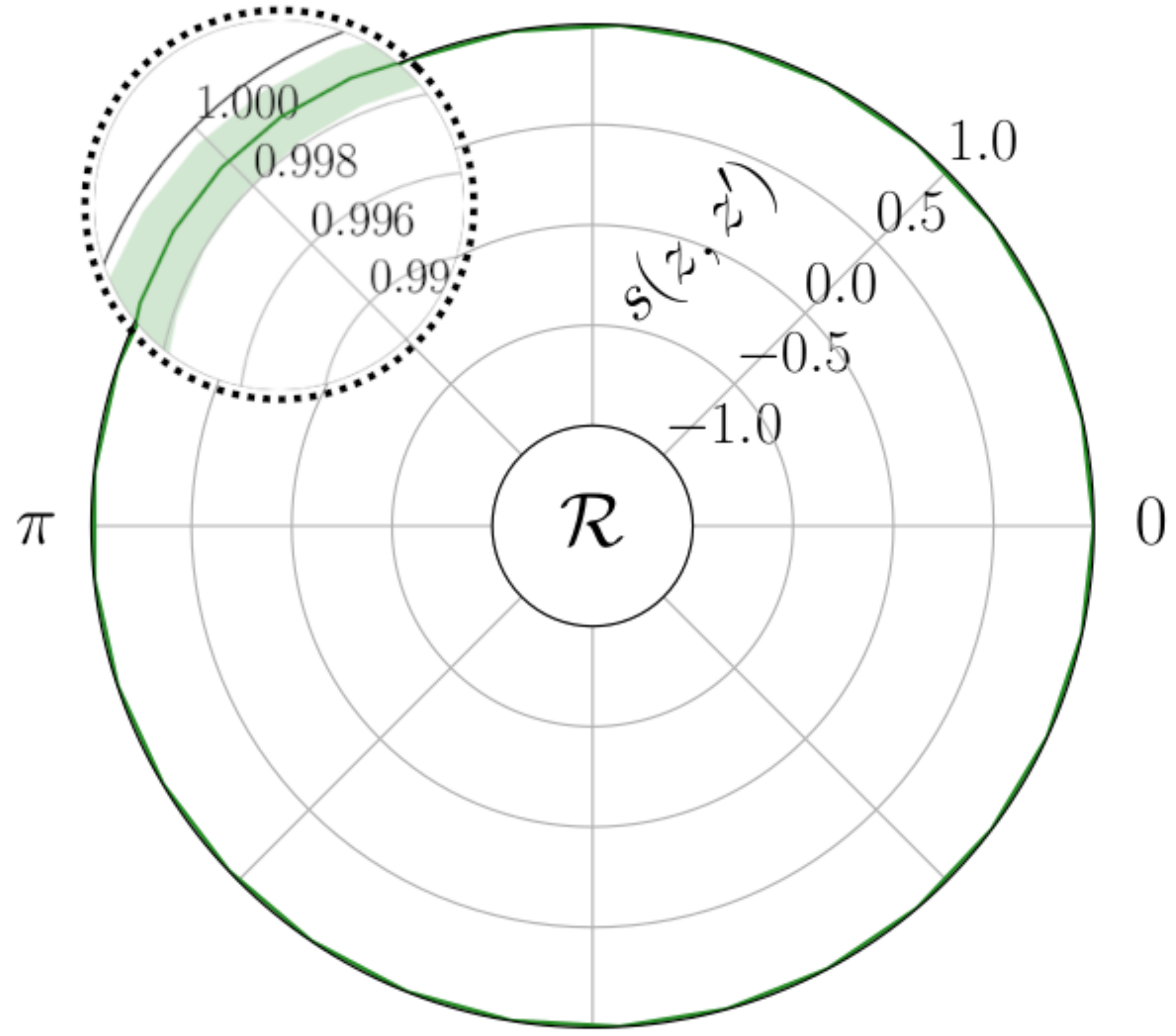
$$\phi' \sim \mathcal{N}\left(\phi, \frac{\Lambda_{soft}}{p_T}\right)$$

Are we learning invariances?

without rotational invariance

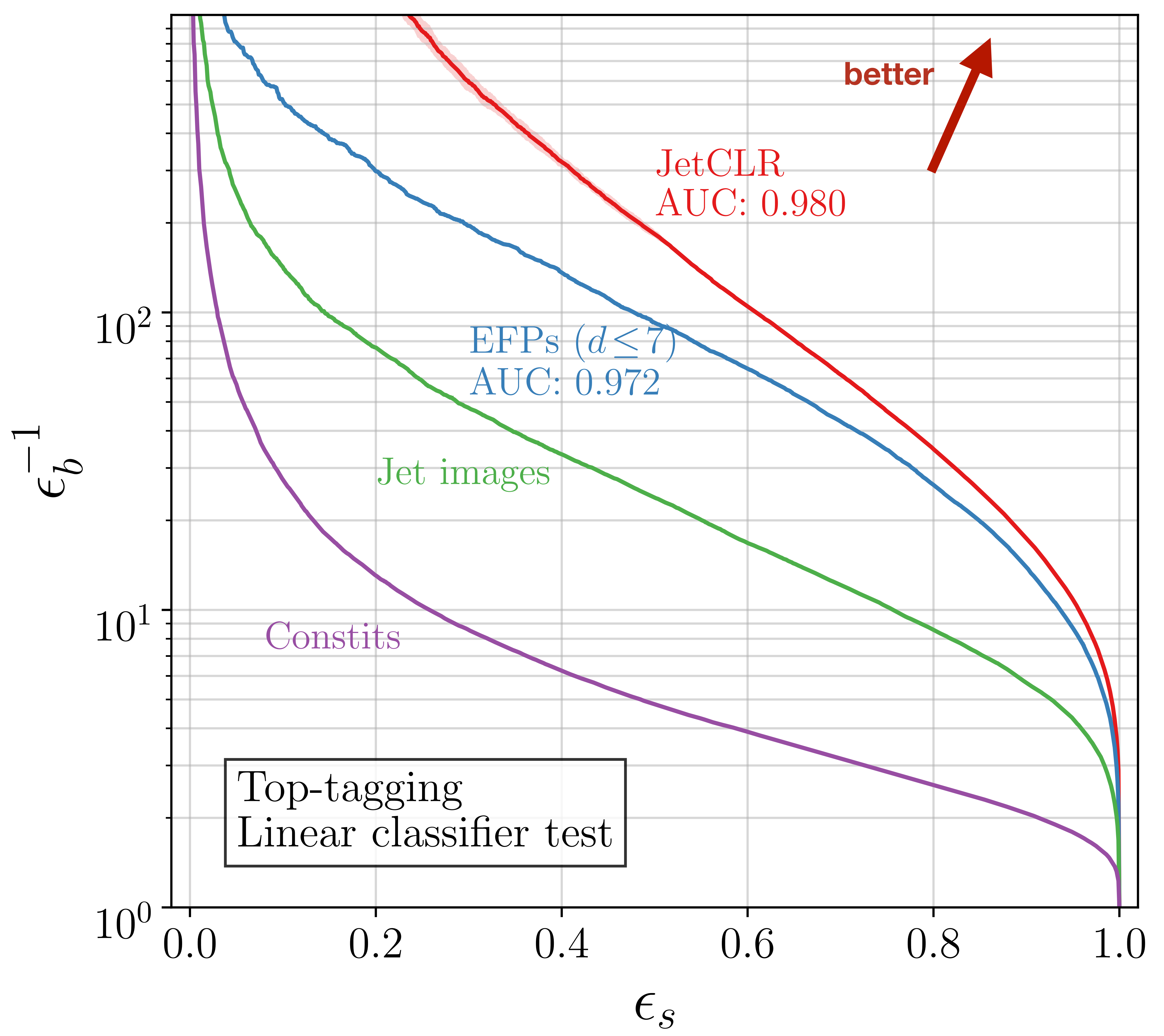


with rotational invariance



The network $f(\mathbf{x})$ is approximately rotationally invariant

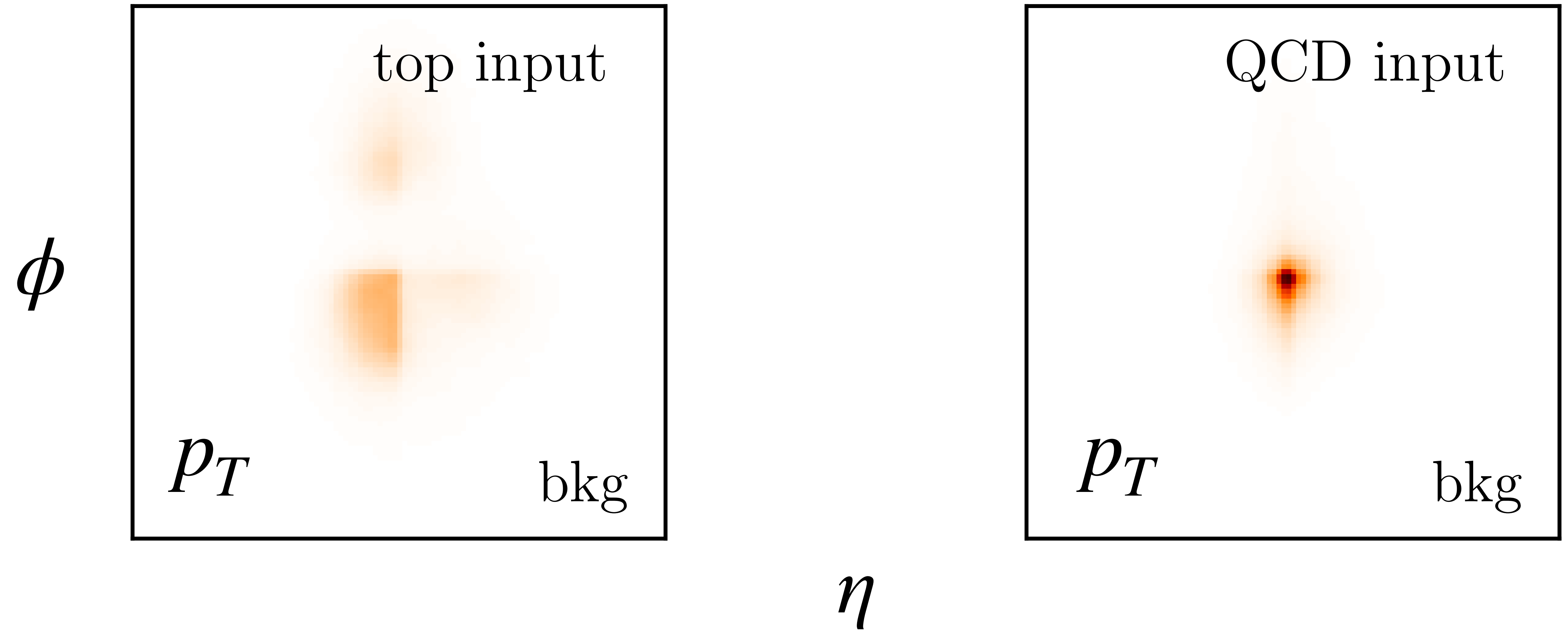
Linear Classifier test



Self-supervision for anomaly detection

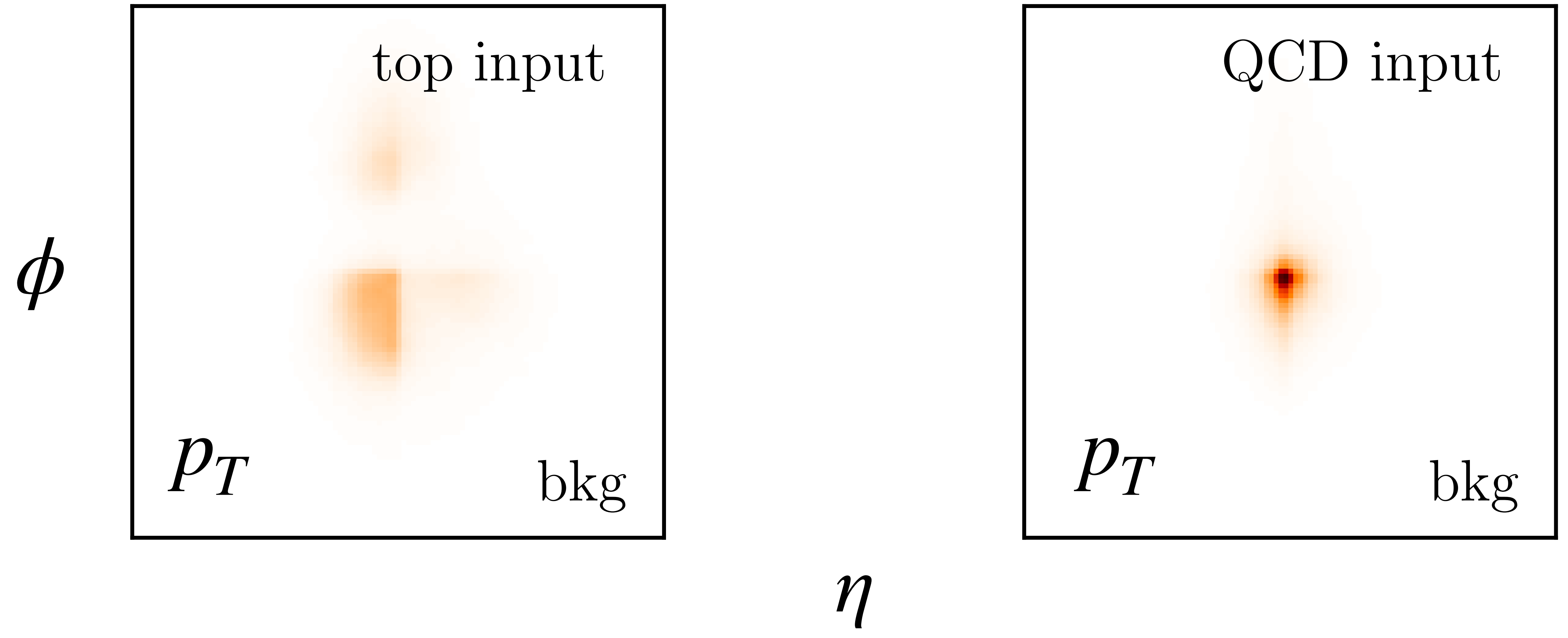
Self-supervision for anomaly detection

JetCLR uses a mixed dataset ⟶ not possible for AD



Self-supervision for anomaly detection

JetCLR uses a mixed dataset ⟶ not possible for AD

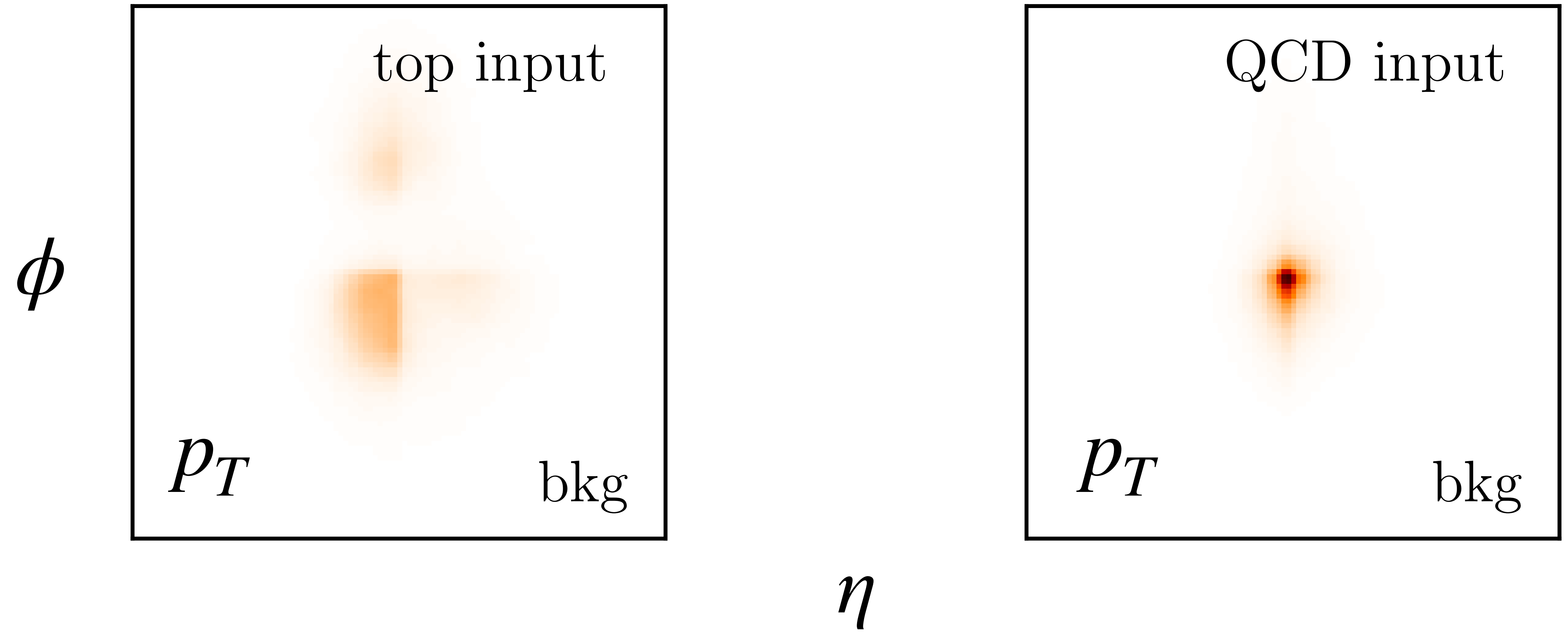


Can we train a transformer-encoder only on background data?

Possible, with no guarantee to learn representations sensitive to new physics

Self-supervision for anomaly detection

JetCLR uses a mixed dataset ⟶ not possible for AD



Can we train a transformer-encoder only on background data?

Possible, with no guarantee to learn representations sensitive to new physics

Introduce z^*, anomaly-augmented point

Self-supervision for anomaly detection

[Anomalies, representations, and self-supervision, Dillon B. et al. arXiv:2301.04660]

Self-supervision for anomaly detection

Loss function:

[Anomalies, representations, and self-supervision, Dillon B. et al. arXiv:2301.04660]

Self-supervision for anomaly detection

Loss function:

"anomalous" augmentation

$$\mathscr{L}_{AnomCLR} = -\log \frac{exp(s(z_i, z_i') - s(z_i, z_i^*)/\tau)}{\sum_{x \in batch} \mathbb{I}_{i \neq j}[exp(s(z_i, z_j)/\tau) + exp(s(z_i, z_j')/\tau)]}$$

[Anomalies, representations, and self-supervision, Dillon B. et al. arXiv:2301.04660]

Self-supervision for anomaly detection

Loss function:

"anomalous" augmentation

$$\mathcal{L}_{AnomCLR} = -\log \frac{exp(s(z_i, z_i') - s(z_i, z_i^*)/\tau)}{\sum_{x \in batch} \mathbb{I}_{i \neq j}[exp(s(z_i, z_j)/\tau) + exp(s(z_i, z_j')/\tau)]}$$

$$\mathcal{L}_{AnomCLR+} = -\log e^{(s(z_i, z_i') - s(z_i, z_i^*))/\tau} = \frac{s(z_i, z_i^*) - s(z_i, z_i)}{\tau}$$

only anomalous loss

[Anomalies, representations, and self-supervision, Dillon B. et al. arXiv:2301.04660]

DarkCLR

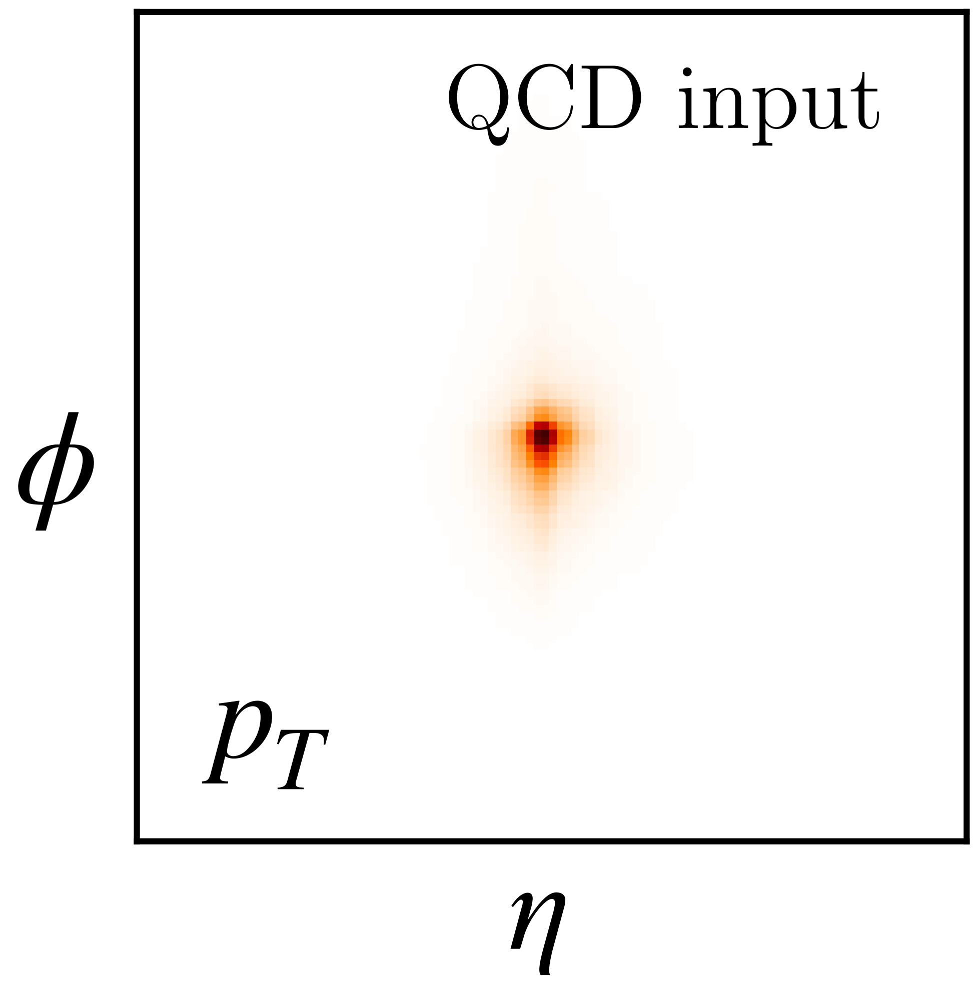
preliminary

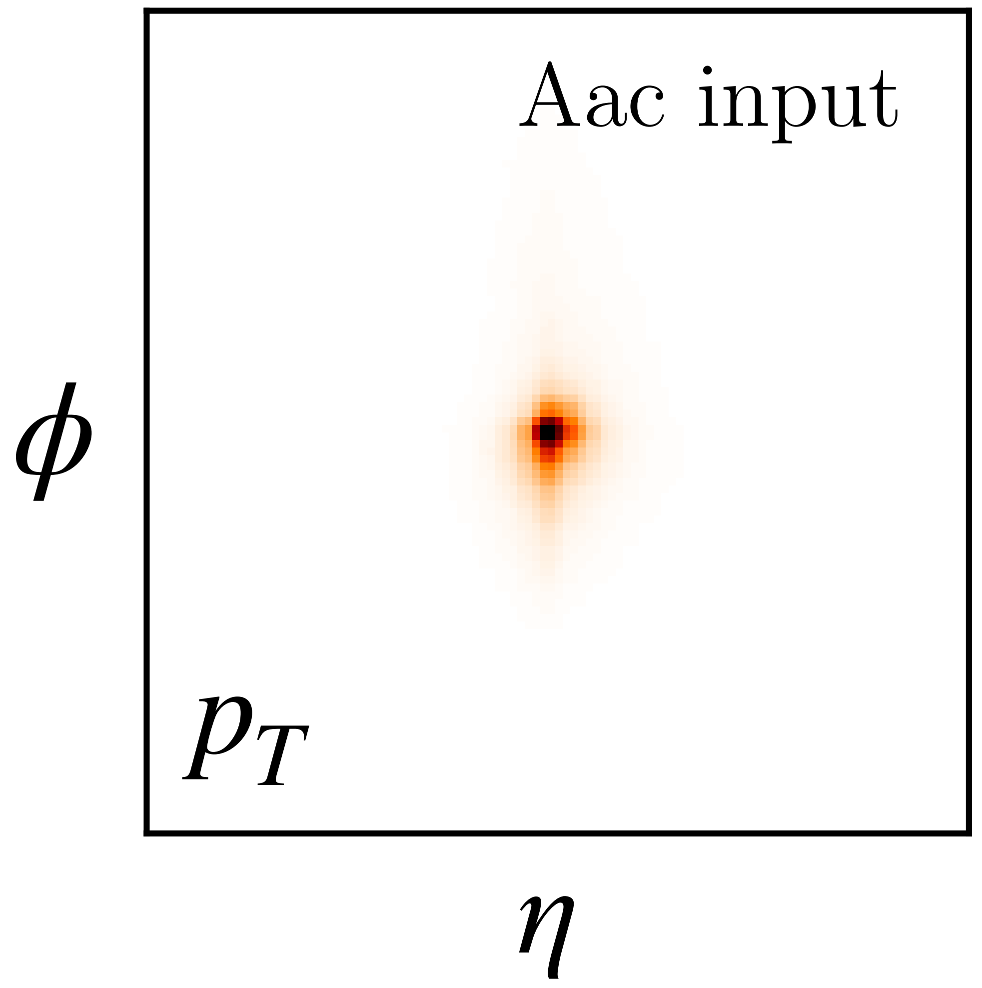
Create a representation space sensitive to dark jets

Benchmark signal: semi-visible jets

$Z' = 1.4$ TeV dark sects mediator

q_d dark quarks charged under $SU(3)_d$





QCD-like showers with fraction of invisible dark particles

DarkCLR

preliminary

Create a representation space sensitive to dark jets

- Avoid preprocessing besides global rescaling;

control preprocessing through augmentations $\longrightarrow$ based on physics bias

rescale p_T: $p_T \longrightarrow p_T / \bar{p}_T$

Transformer network: $f : \mathbb{R}^N \to \mathbb{R}^D$

DarkCLR

preliminary

Create a representation space sensitive to dark jets

- Avoid preprocessing besides global rescaling;
- Representations are informative regardless of the embedding dimension;

Work In Progress

QCD and semivisible jets look more separable regardless of the embedding dimension

CLR can perform dimensionality reduction

$\longrightarrow$ get observables we can use to study simulations

DarkCLR

preliminary

Create a representation space sensitive to dark jets

- Avoid preprocessing besides global rescaling;
- Representations are informative regardless of the embedding dimension;
- Evaluate for anomaly detection.

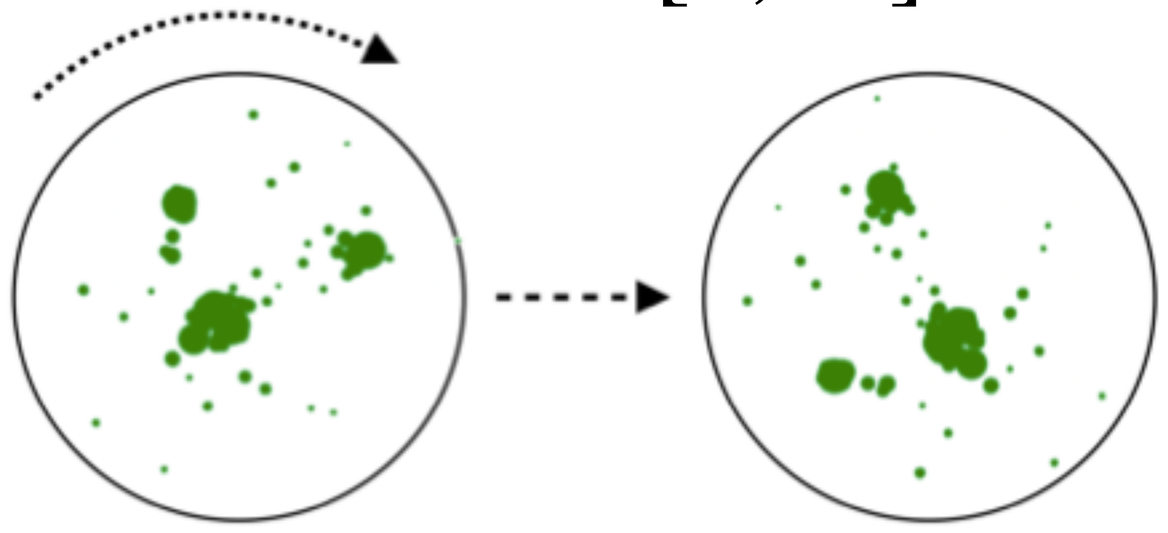
define a new anomaly score in the CLR representation space…

… and train a Normalized AutoEncoder

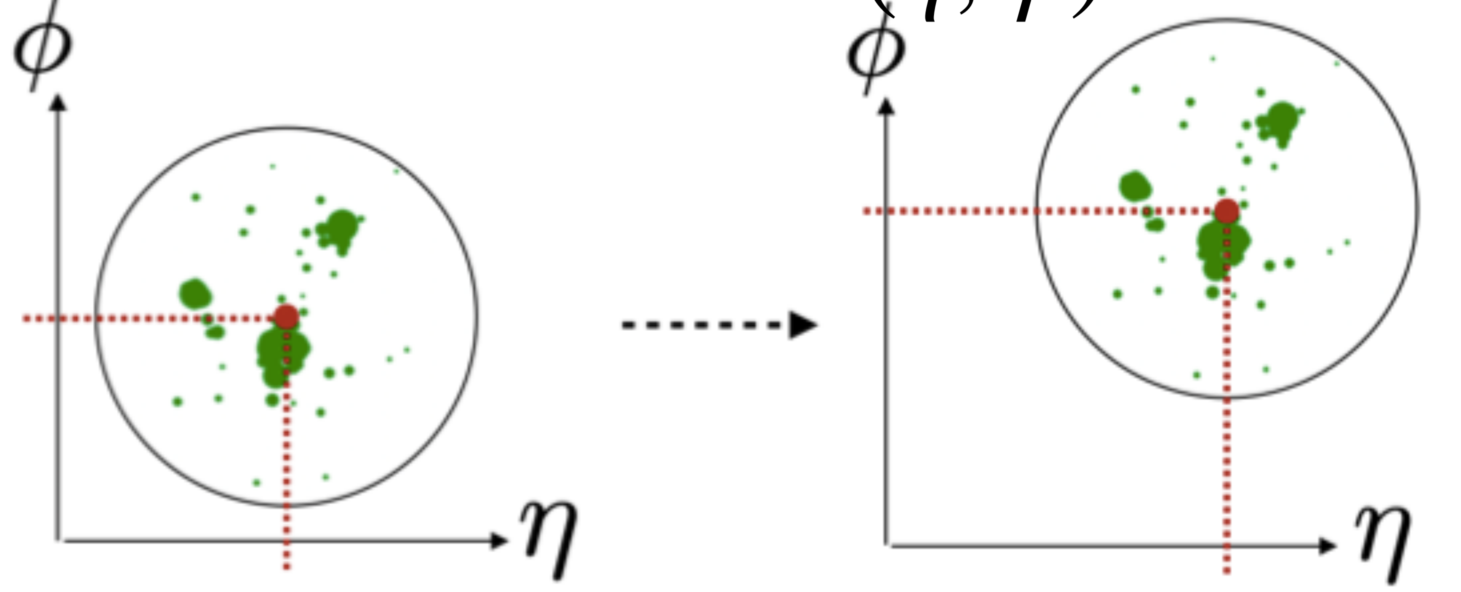
Invariances

preliminary

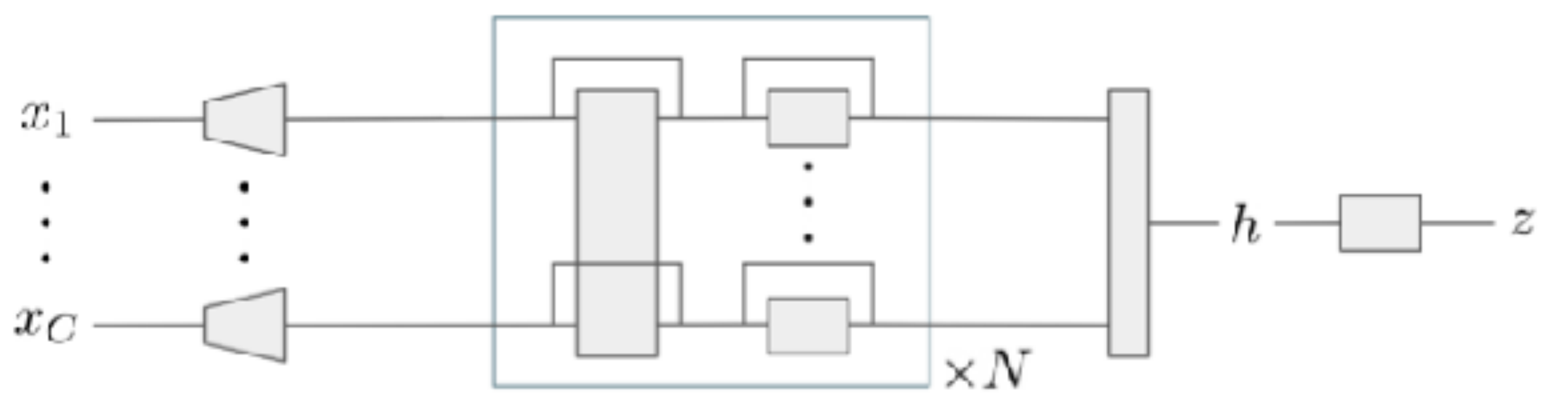
rotation in $[0, 2\pi]$



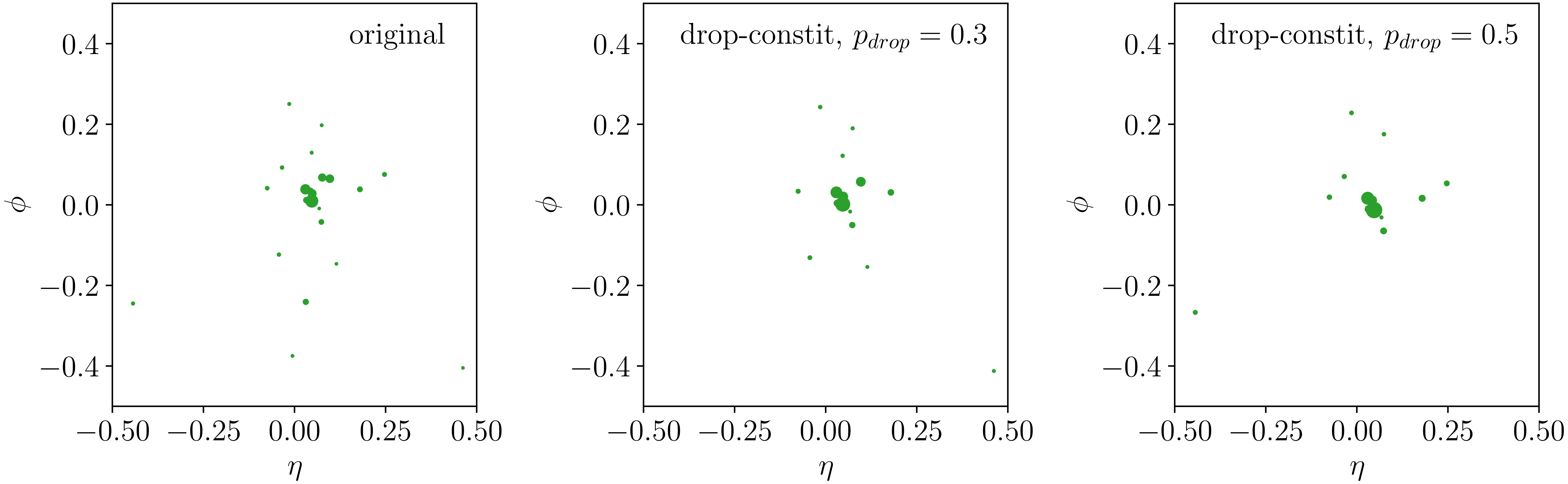
translation in (η, ϕ)



permutation between constituents



Anomalous augmentation: drop constituents with probability p_{drop}



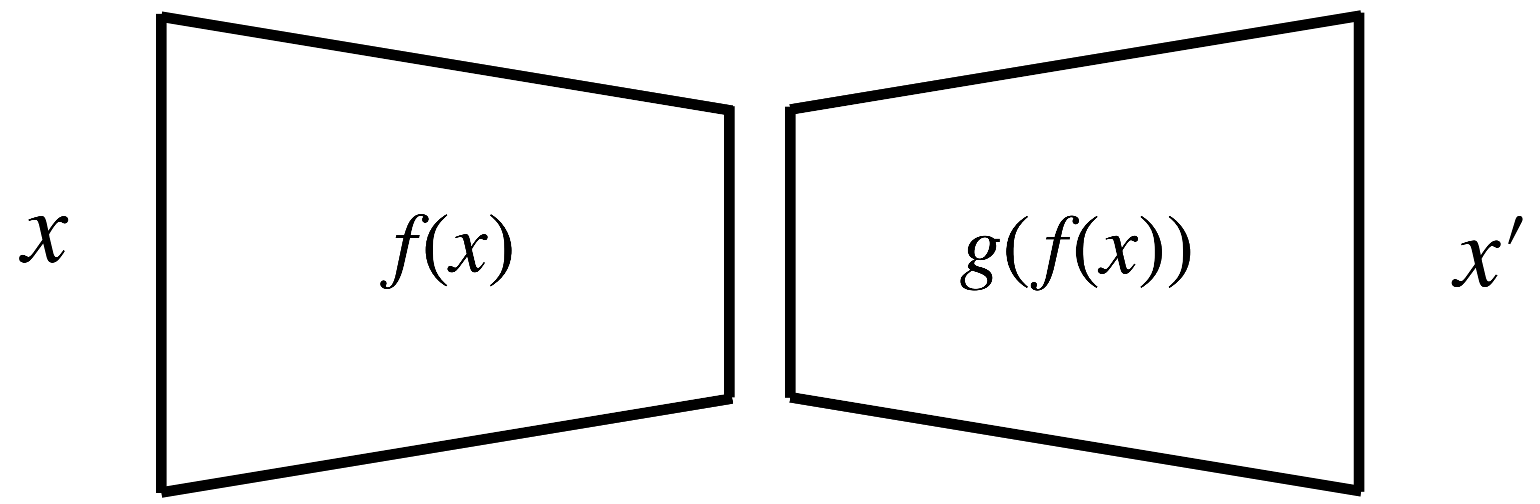
Anomaly scores

preliminary

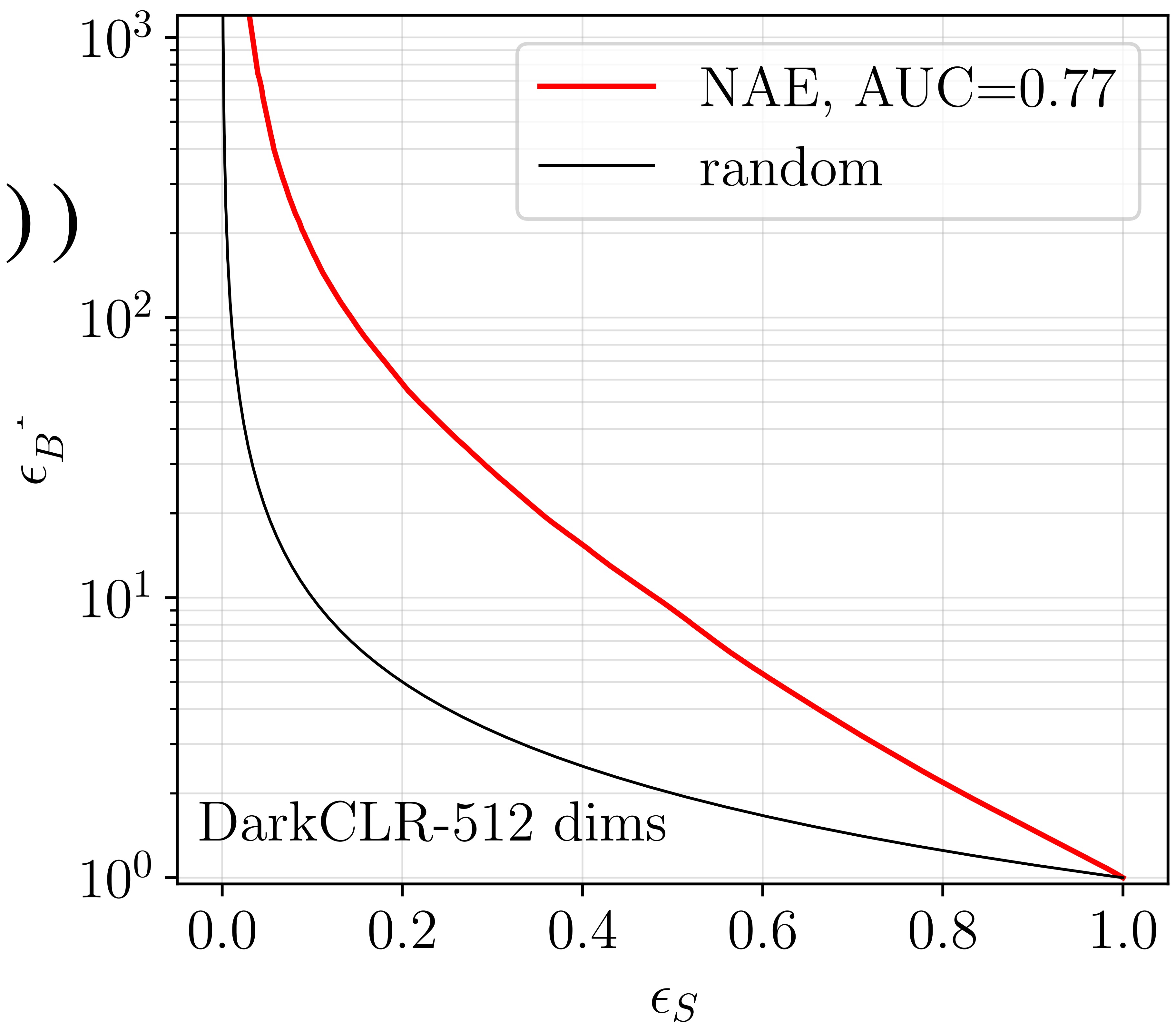
Anomaly scores

preliminary

(N)AutoEncoder based anomaly score: $MSE(x, g(f(x)))$



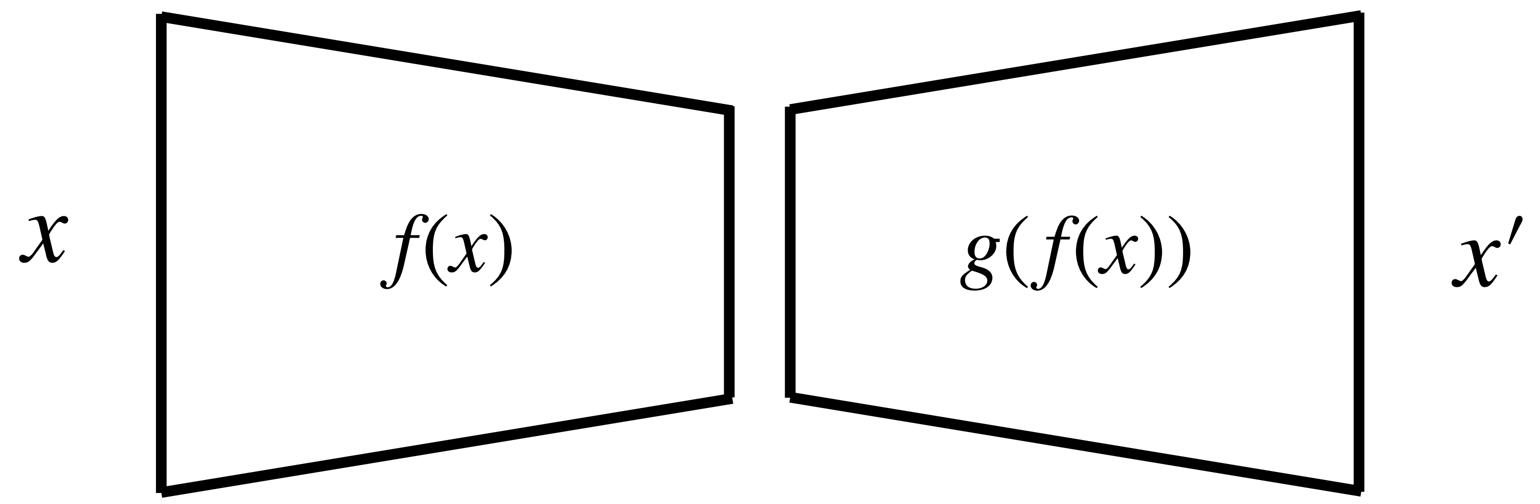
$$p_\theta(x) = \frac{e^{-E_\theta}}{\Omega} \qquad E_\theta = MSE(x, g(f(x)))$$



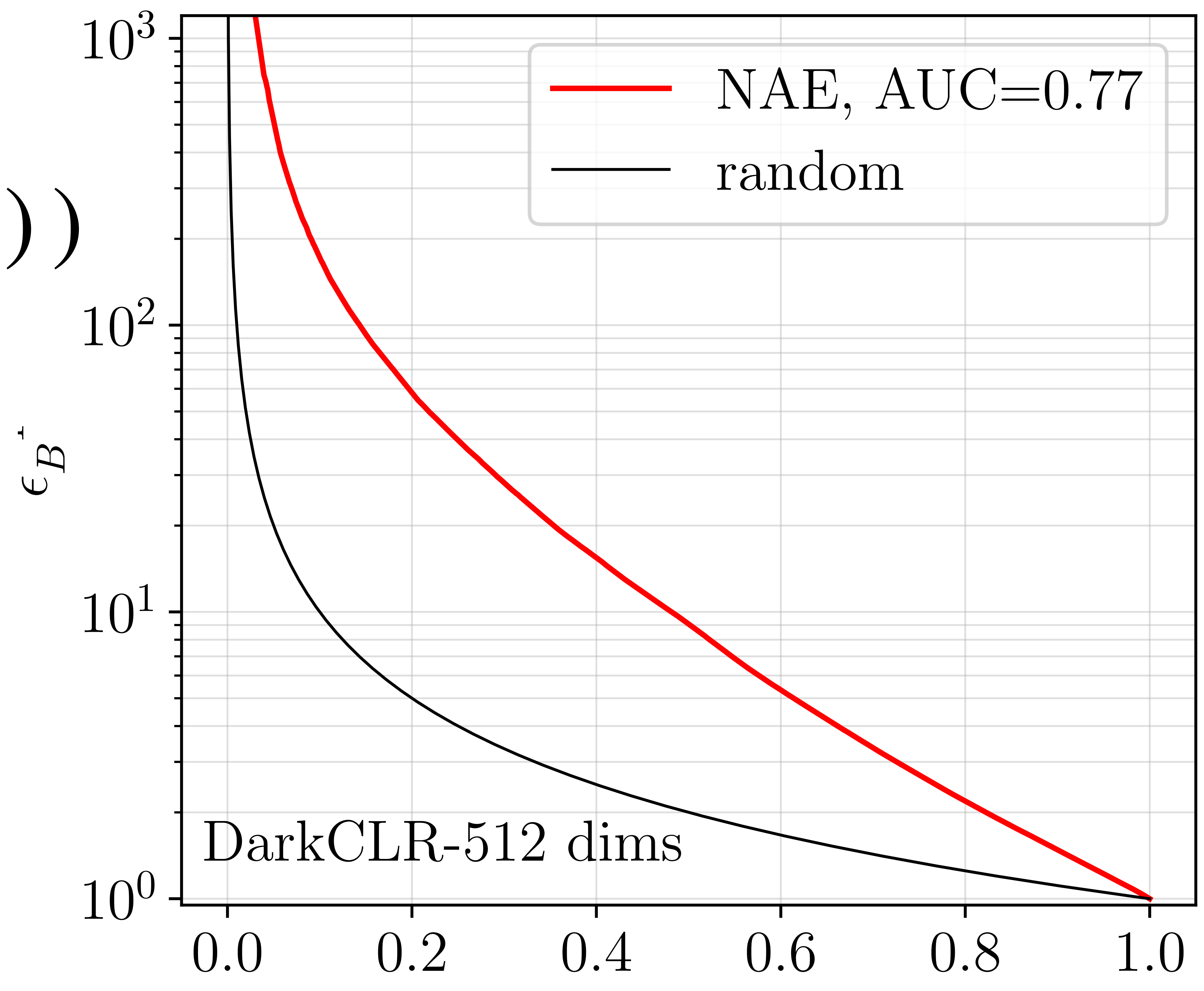
Anomaly scores

preliminary

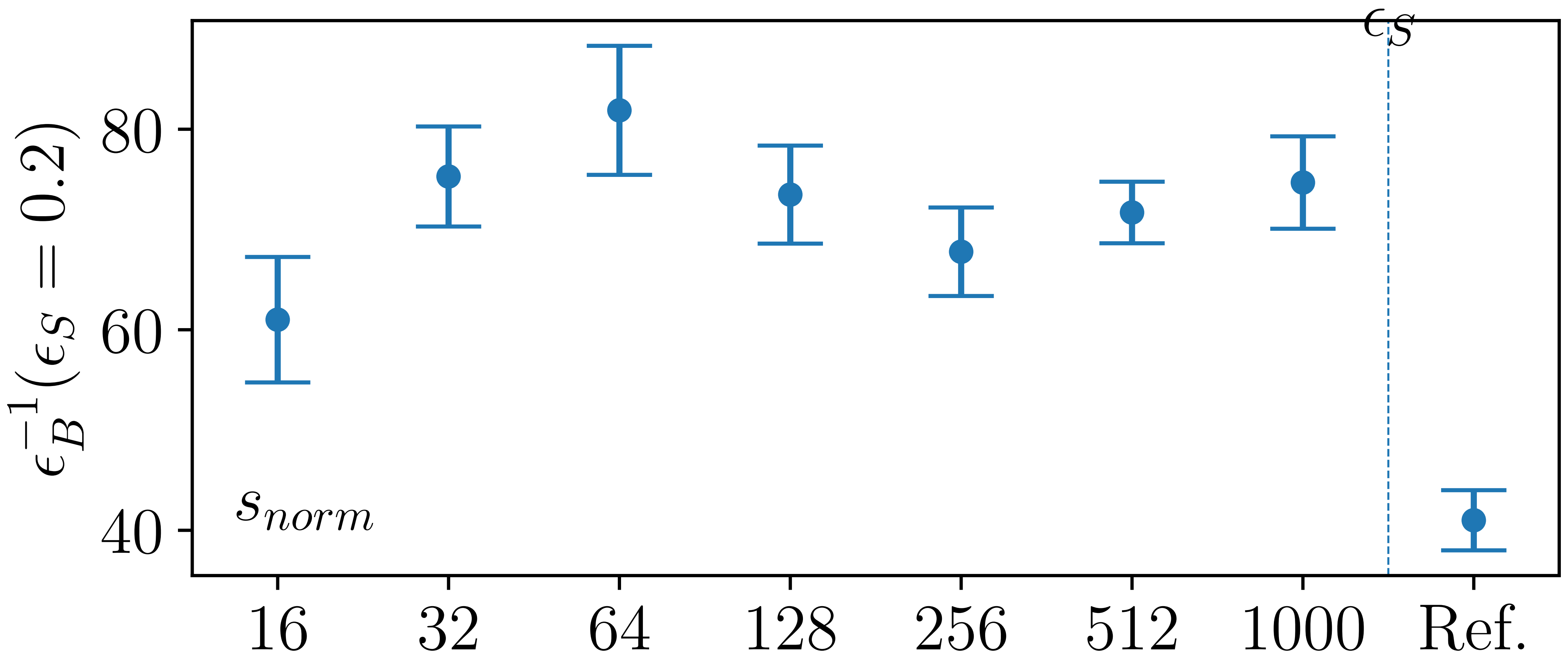
(N)AutoEncoder based anomaly score: $MSE(x, g(f(x)))$



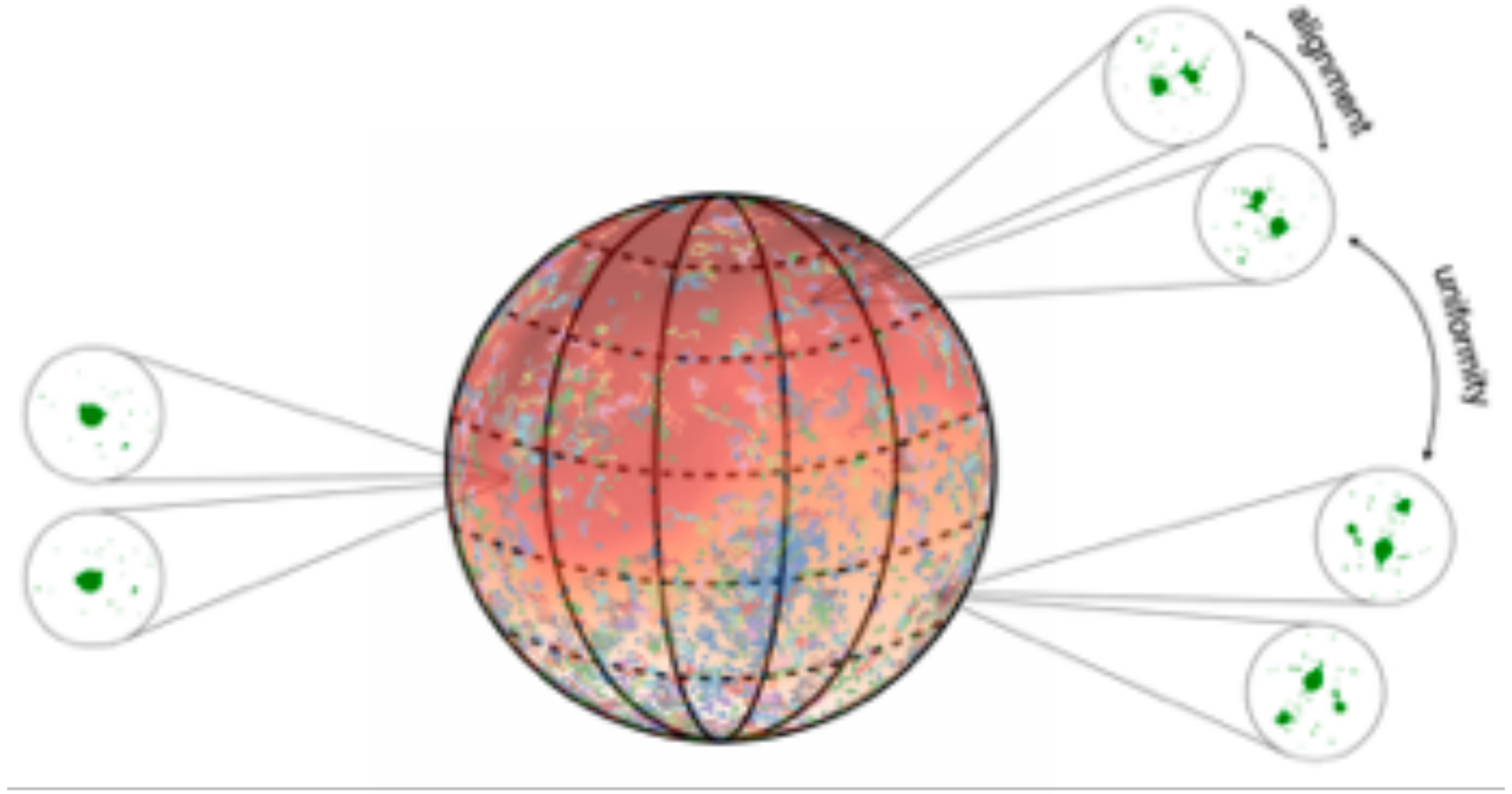
$$p_\theta(x) = \frac{e^{-E_\theta}}{\Omega} \qquad E_\theta = MSE(x, g(f(x)))$$



Latent anomaly score: $||z||_{L_2}$



Conclusions/Outlook



Conclusions/Outlook

Self-supervision: extracting features from unlabelled data through pseudo-tasks

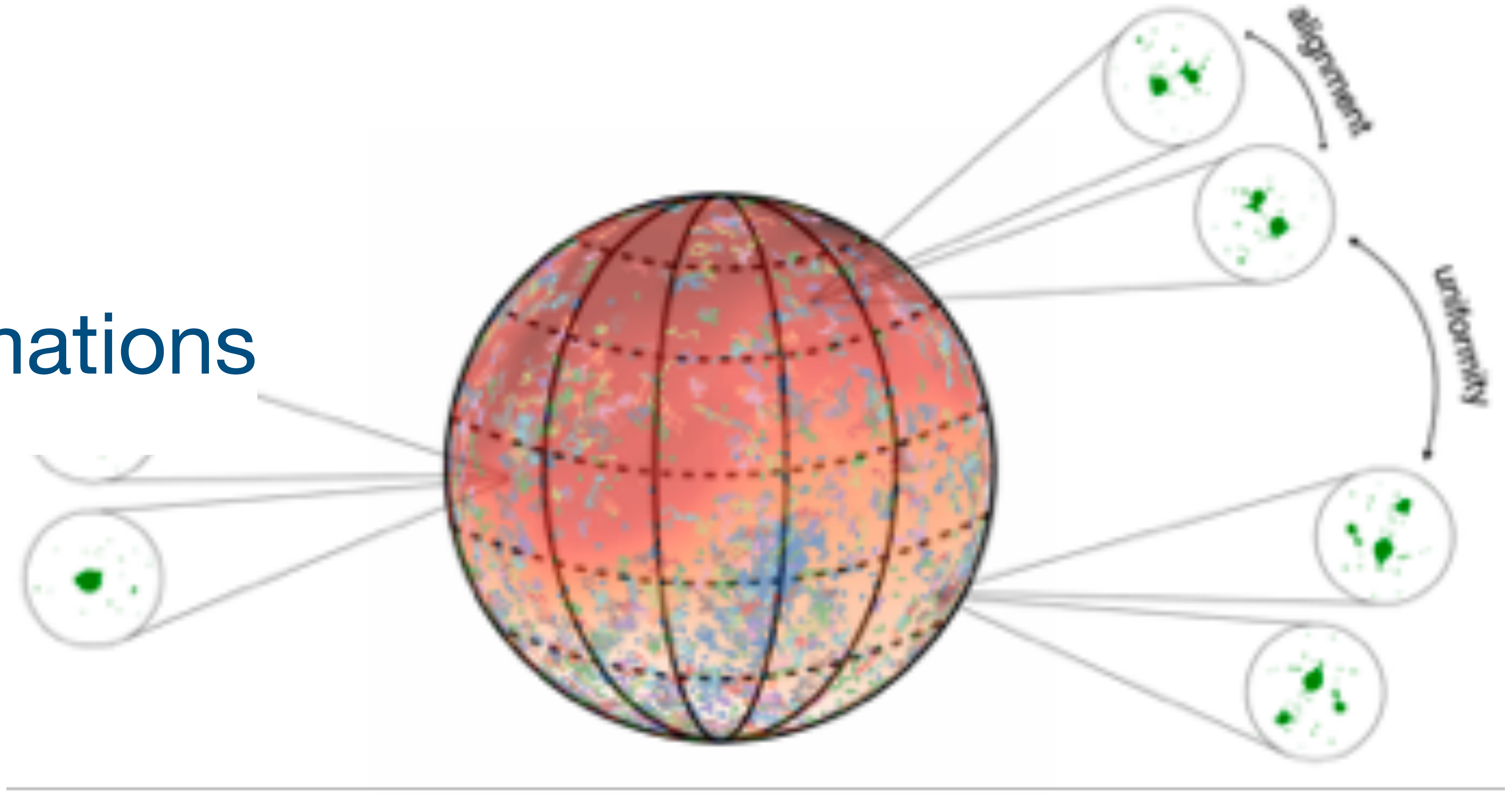
- Allows us to build highly expressive physical representations

JetCLR $\longrightarrow$ learn invariances, and provide simpler representations

- Can be used for anomaly detection tasks

DarkCLR $\longrightarrow$ CLR for semi-visible jet detection

- Apply preprocessing via invariances to transformations
- Downstream task: Anomaly detection



Thanks for your attention!

Backup

Model-agnostic searches & ML

Model-agnostic searches & ML

Deep-learning has demonstrated potential for analyses with...

- no restriction to a few, or even high-level, observables
- the ability to process high-dimensional datasets
- less reliance on simulation

Model-agnostic searches & ML

Deep-learning has demonstrated potential for analyses with...

- no restriction to a few, or even high-level, observables
- the ability to process high-dimensional datasets
- less reliance on simulation

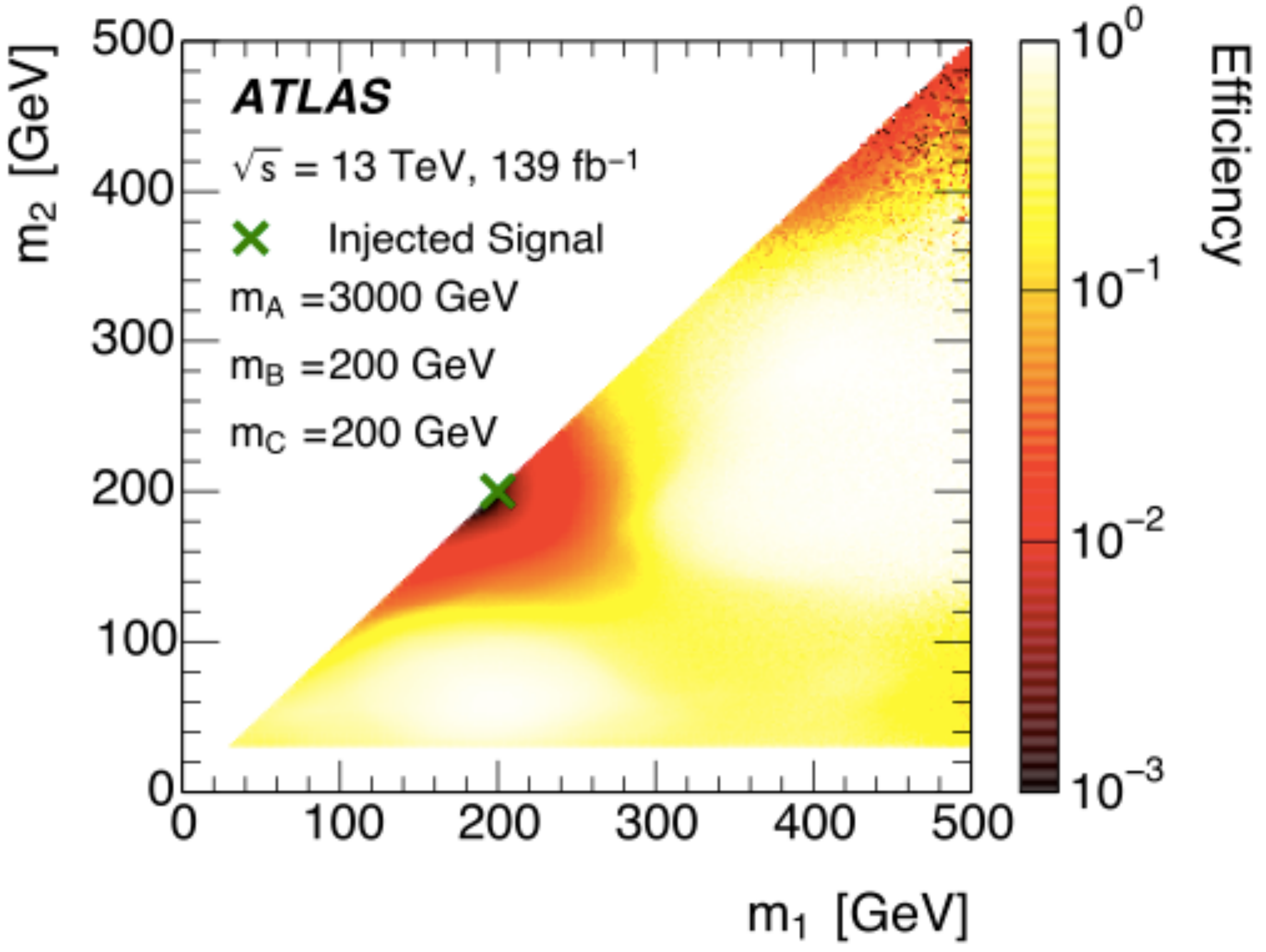
Open questions:

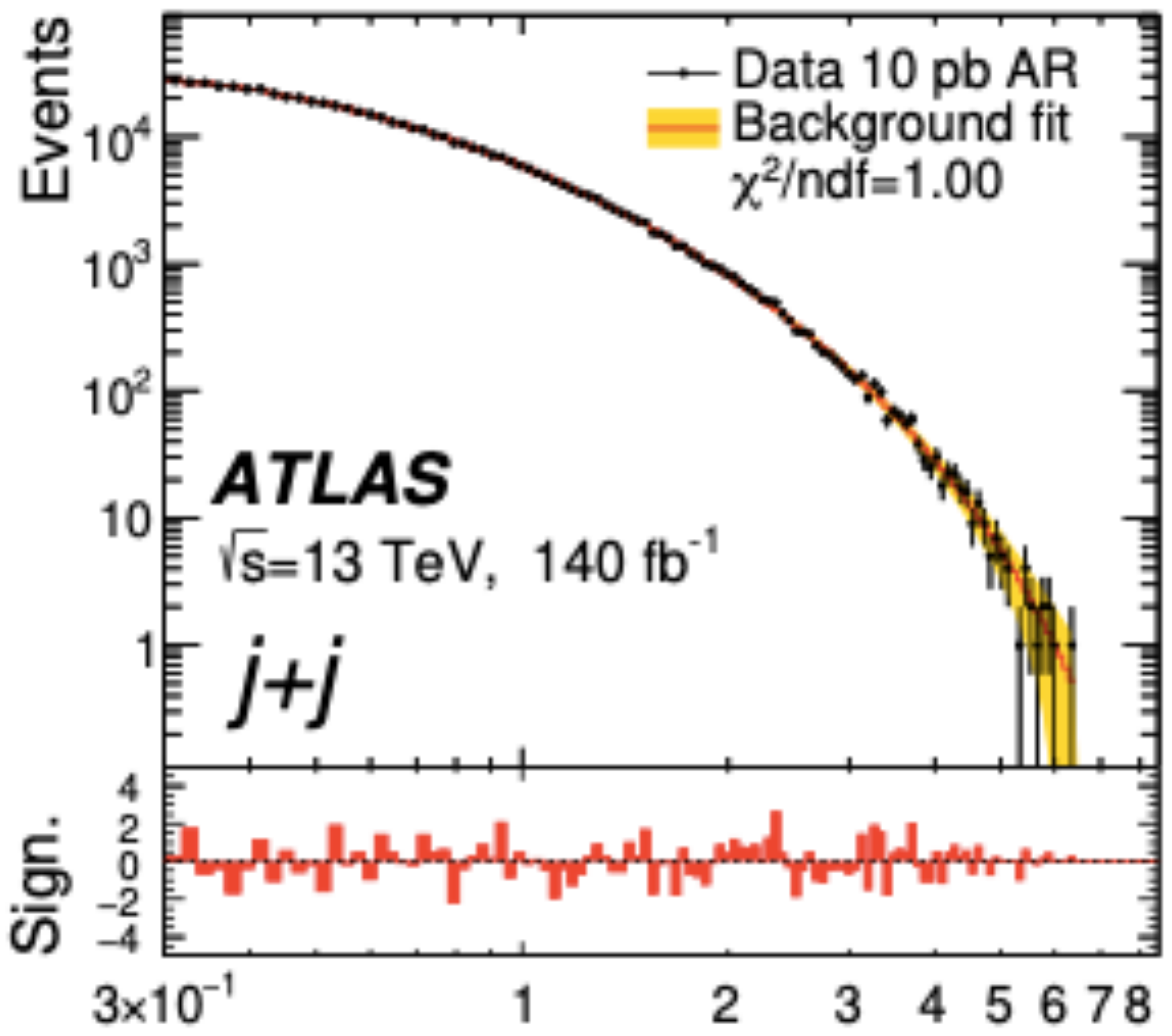
- Physics bias in deep-learning approaches?
- How model-agnostic are the approaches?
- What would an analysis with these tools look like?

Model-agnostic searches & ML

Deep-learning has demonstrated potential for analyses with...

- no restriction to a few, or even high-level, observables
- the ability to process high-dimensional datasets
- less reliance on simulation

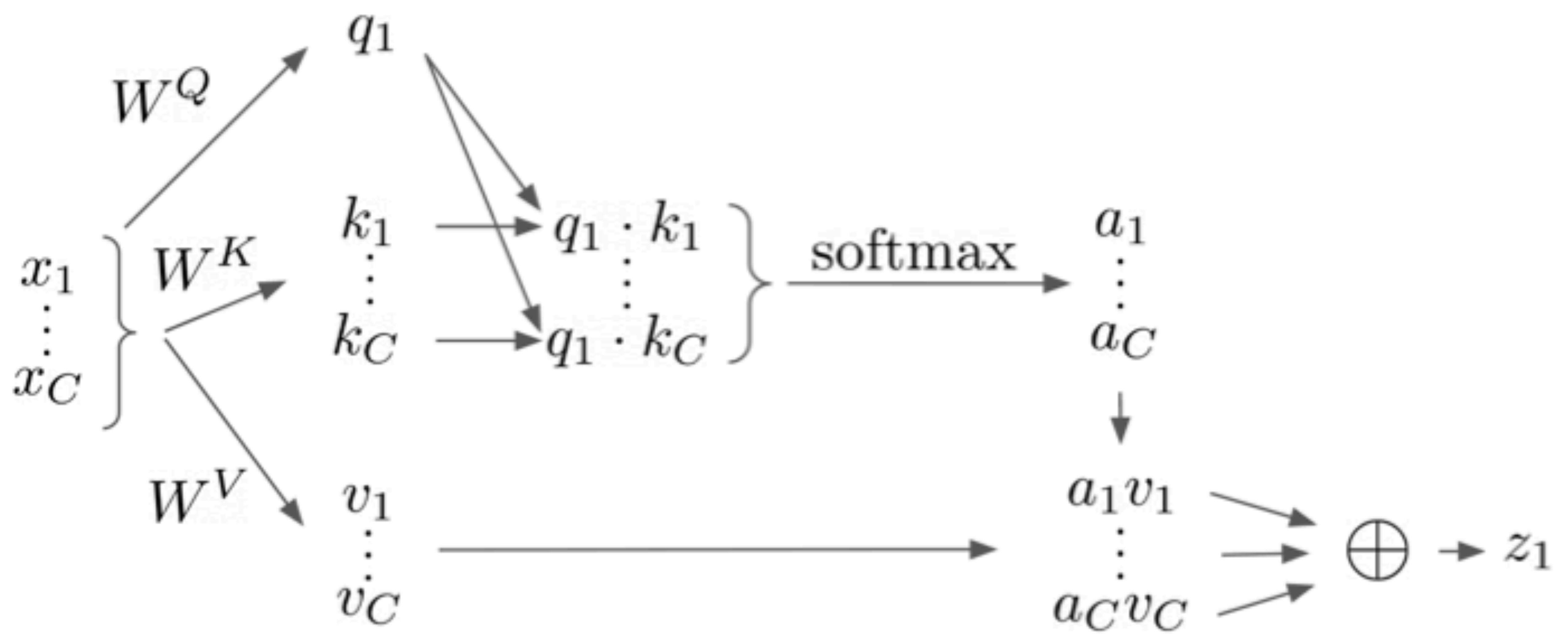


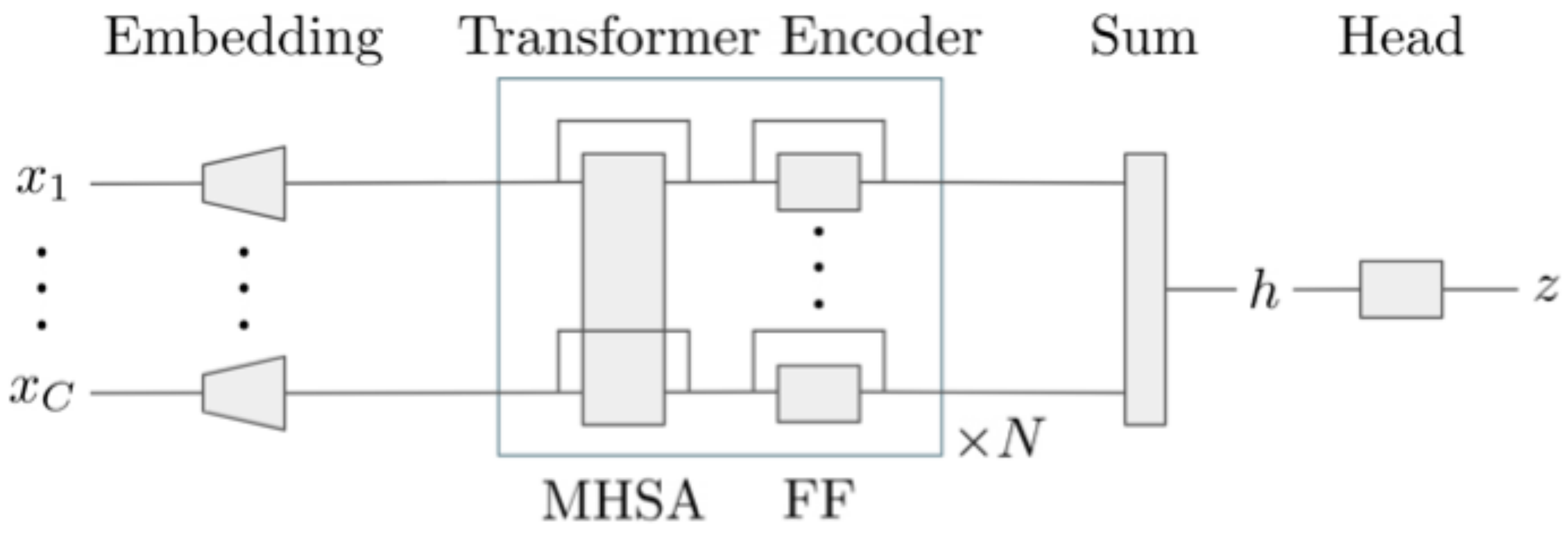


Open questions:

- Physics bias in deep-learning approaches?
- How model-agnostic are the approaches?
- What would an analysis with these tools look like?

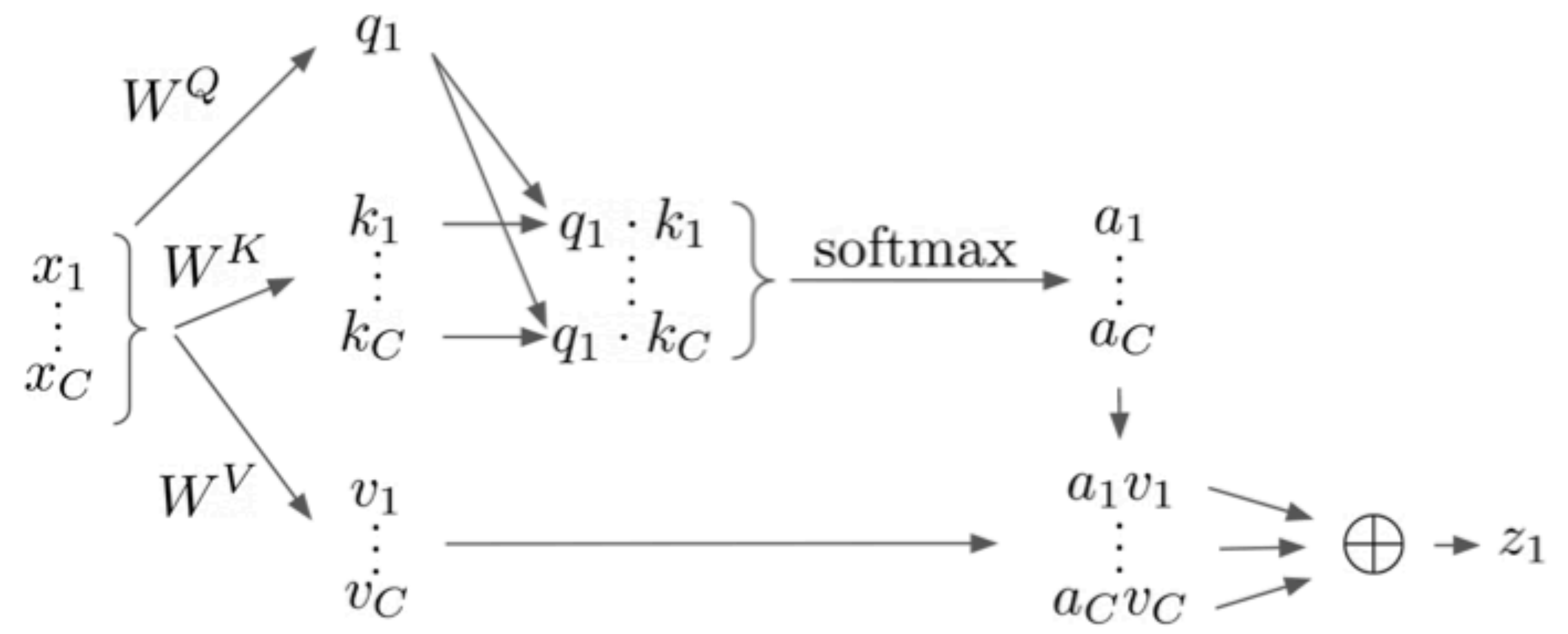
Transformer Encoder

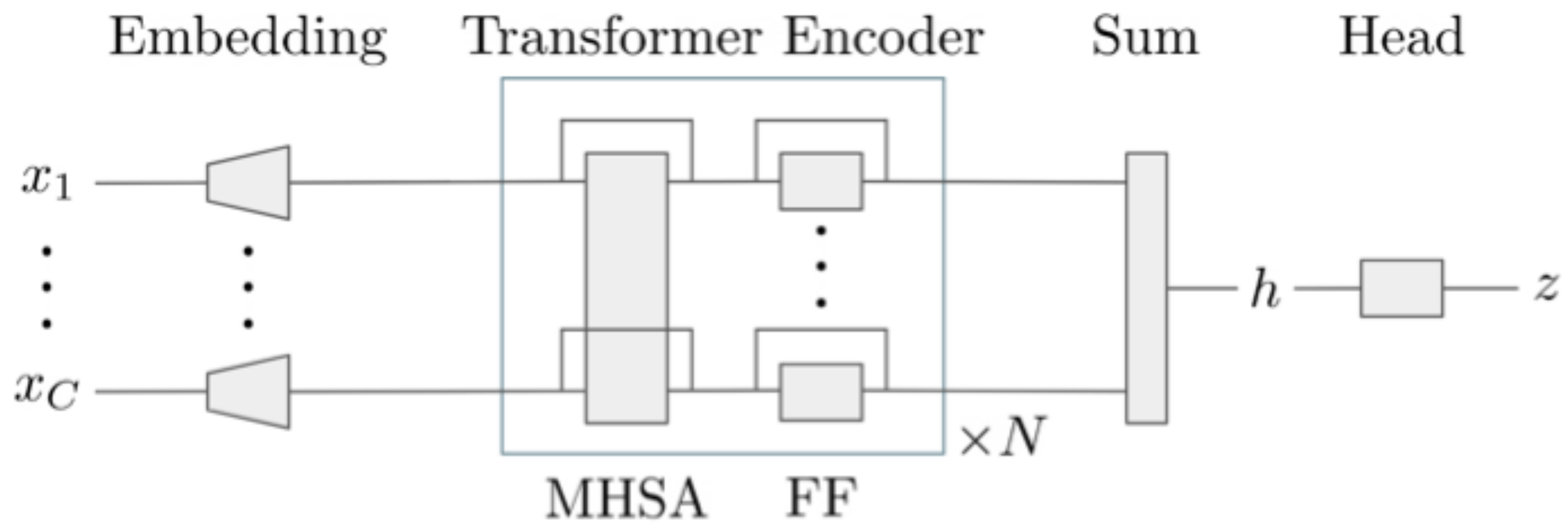




Transformer Encoder

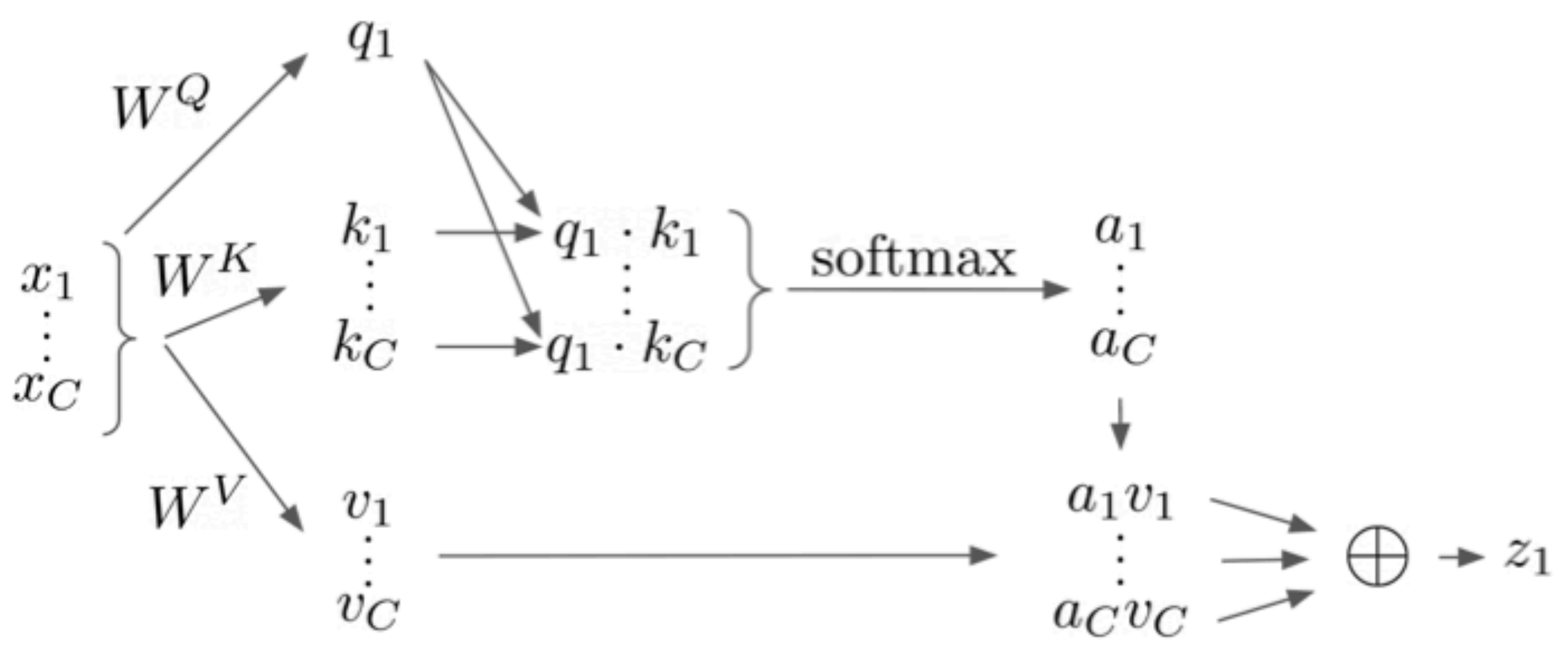
Self-Attention:



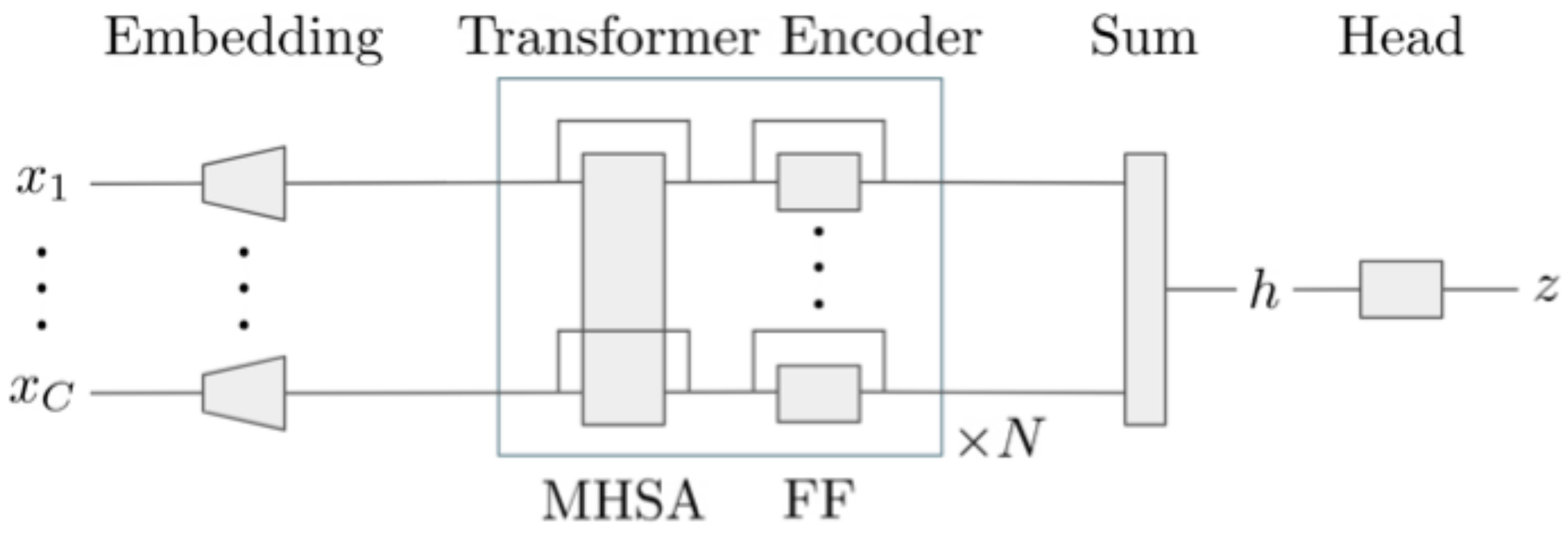


Transformer Encoder

Self-Attention:



Network:



AnomalyCLR on Jets

preliminary