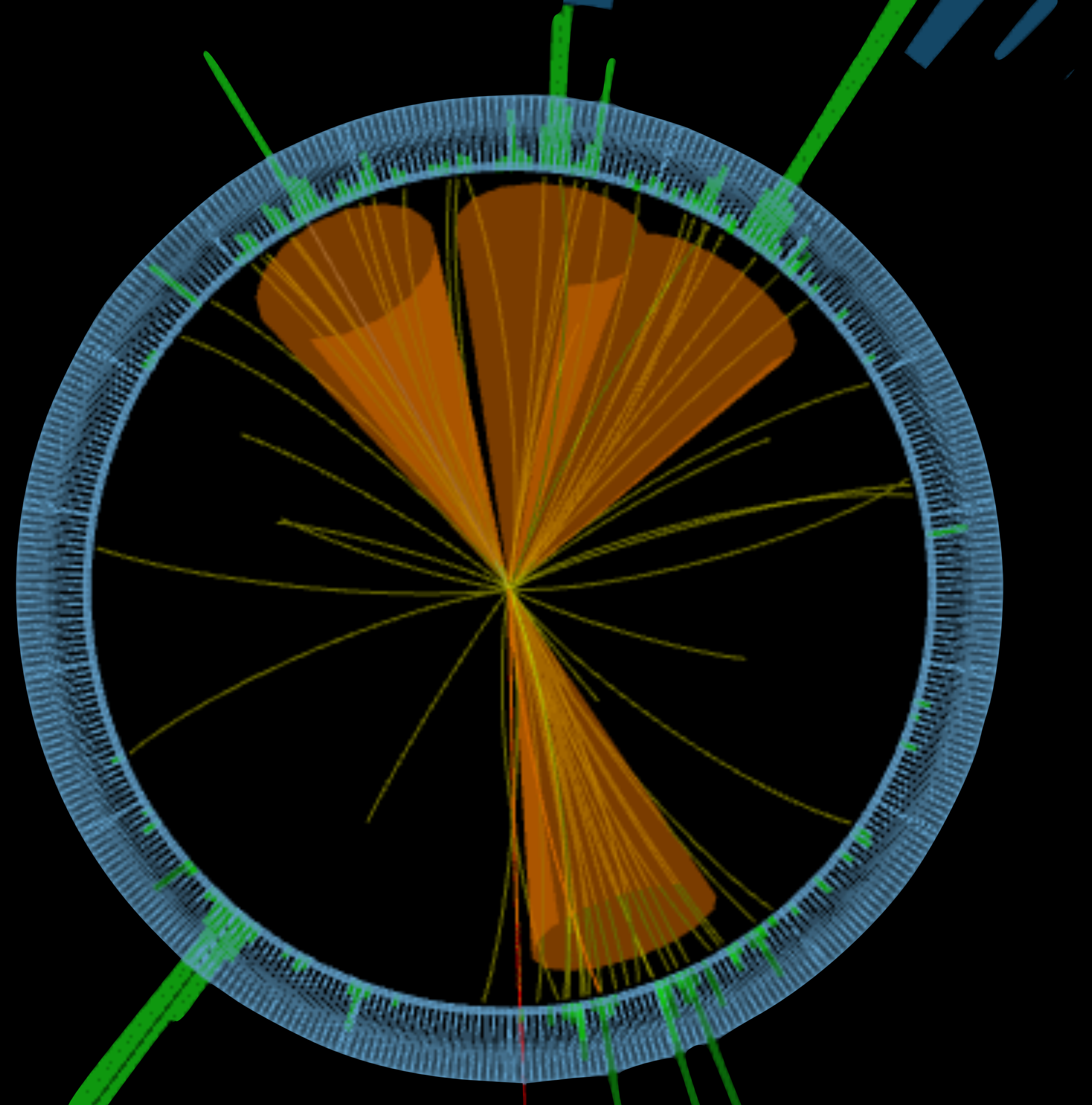
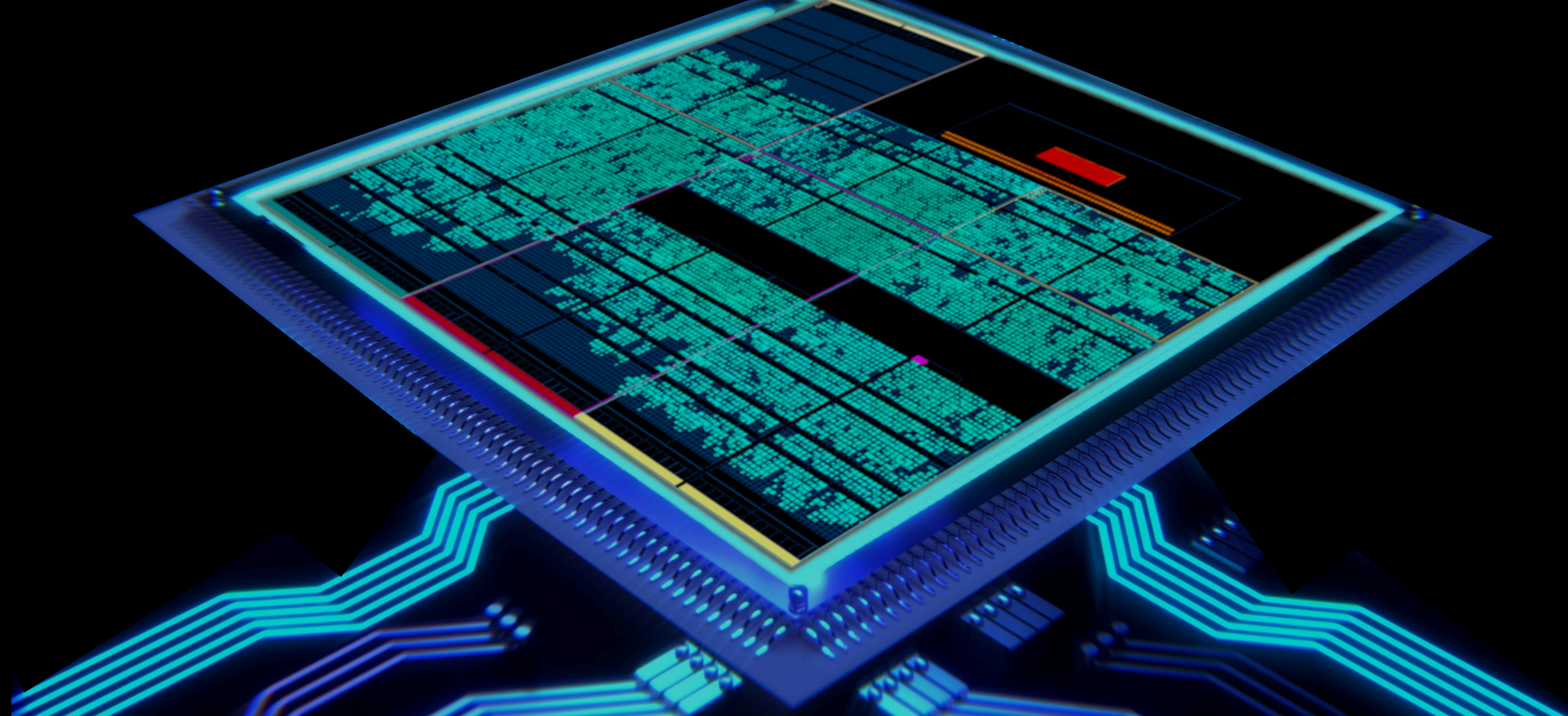


Machine Learning in High Energy Physics



Thea K. Årrestad (ETH Zürich)
thea.aarrestad@cern.ch
thaarres.github.io

QCD School 2024

ML-powered anomaly detection exercise on CMS Open Data

When: Today at 17:00

Material: github.com/thaarres/qcd_school_ml

ML Exercise: Anomaly detection in high energy physics

In this notebook we will demonstrate how to design a tiny autoencoder (AE) that we will use for anomaly detection in particle physics. More specifically, we will demonstrate how we can use autoencoders to select potentially New Physics enhanced proton collision events in a more unbiased way!

We will train the autoencoder to learn to compress and decompress data, assuming that for highly anomalous events, the AE will fail.

Dataset

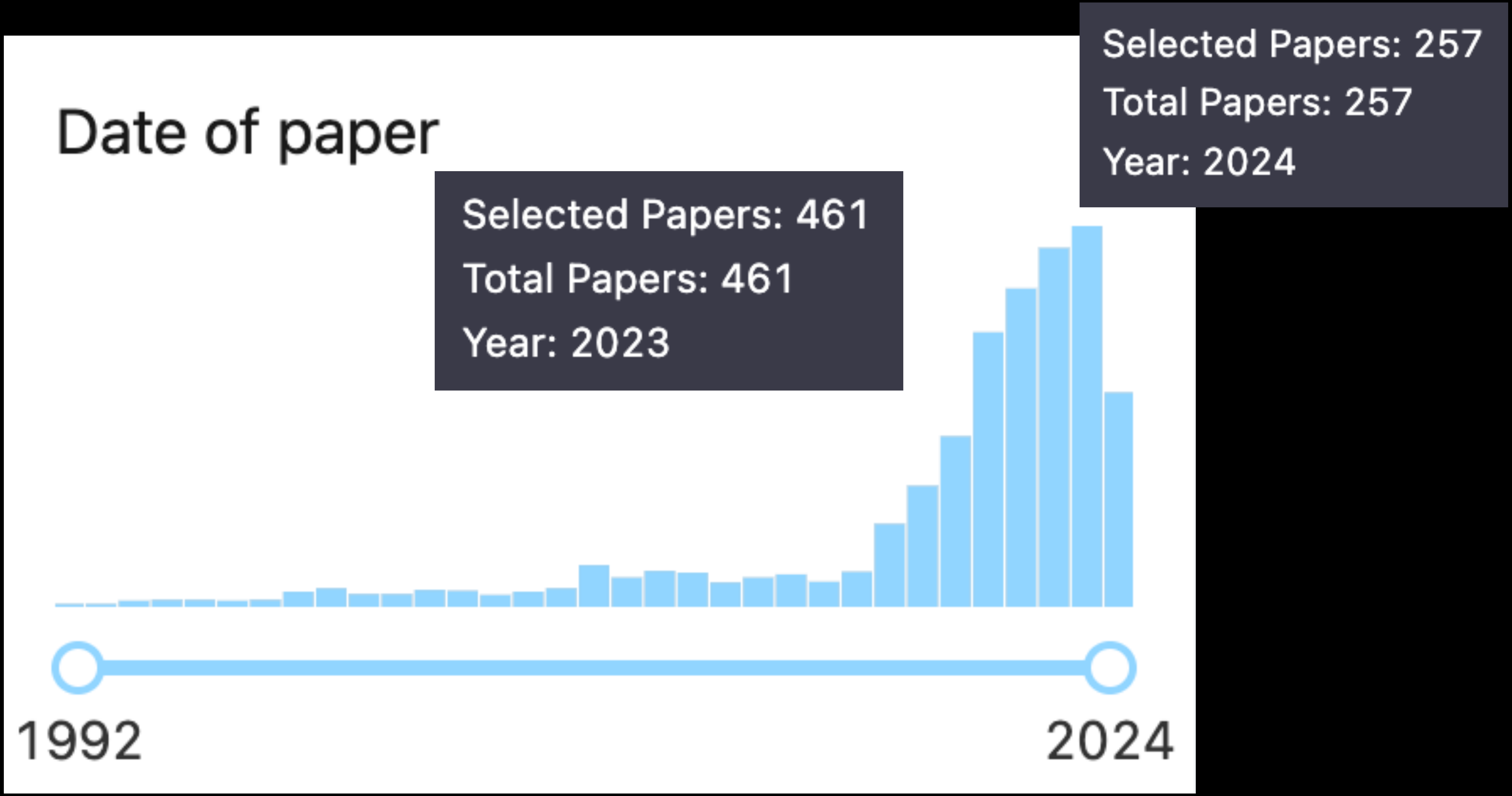
As a dataset, we will use the CMS Open data that you have been made familiar with already. Our dataset will be represented as an array of missing transverse energy (MET), up to 4 electrons, up to 4 muons and 10 jets each described by pT, η, ϕ and particle ID (just from knowing whether it is a muon/electron/jet)--recid 63168 --protocol xrootd. The particles are ordered by pT. If fewer objects are present, the event is zero padded.

We will train on a QCD MC dataset (we could also train directly on data), and evaluate the AE performance on a New Physics simulated sample: A Bulk graviton decaying to two vector bosons: G(M=2 TeV) → WW

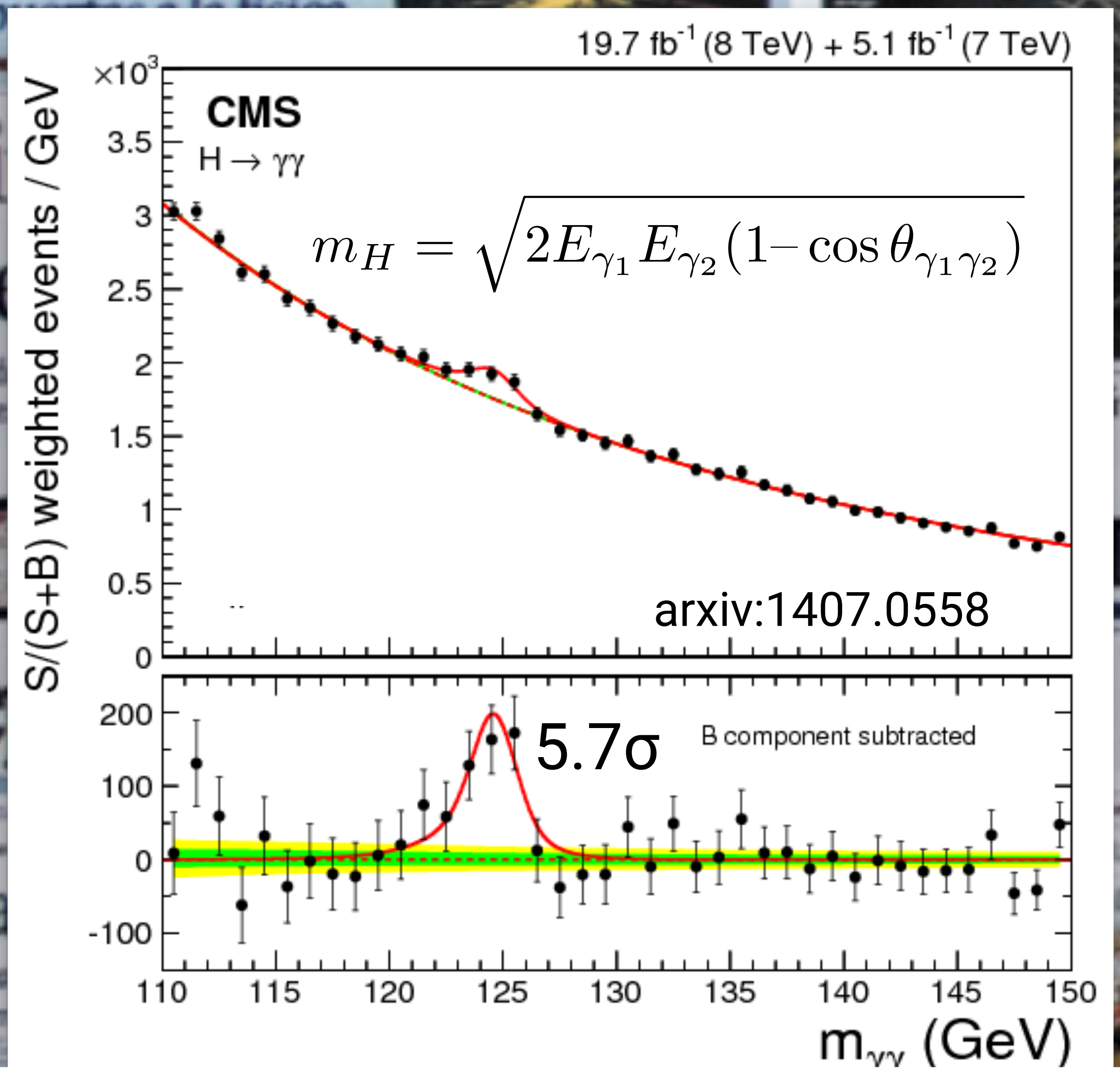
We'll train using background data only and test using both background and the Graviton sample. Let's fetch them! The background data are available here (recid = 63168) and the signal data here (recid = 33703). The signal consists of 1,37M events and the background 19,279M events. We will use roughly 500K for each process.

We will use the docker client to print all the file names. You can then use this list to concatenate data from all the files.

("machine learning" or "deep learning" or neural) and (hep-ex or hep-ph or hep-th)

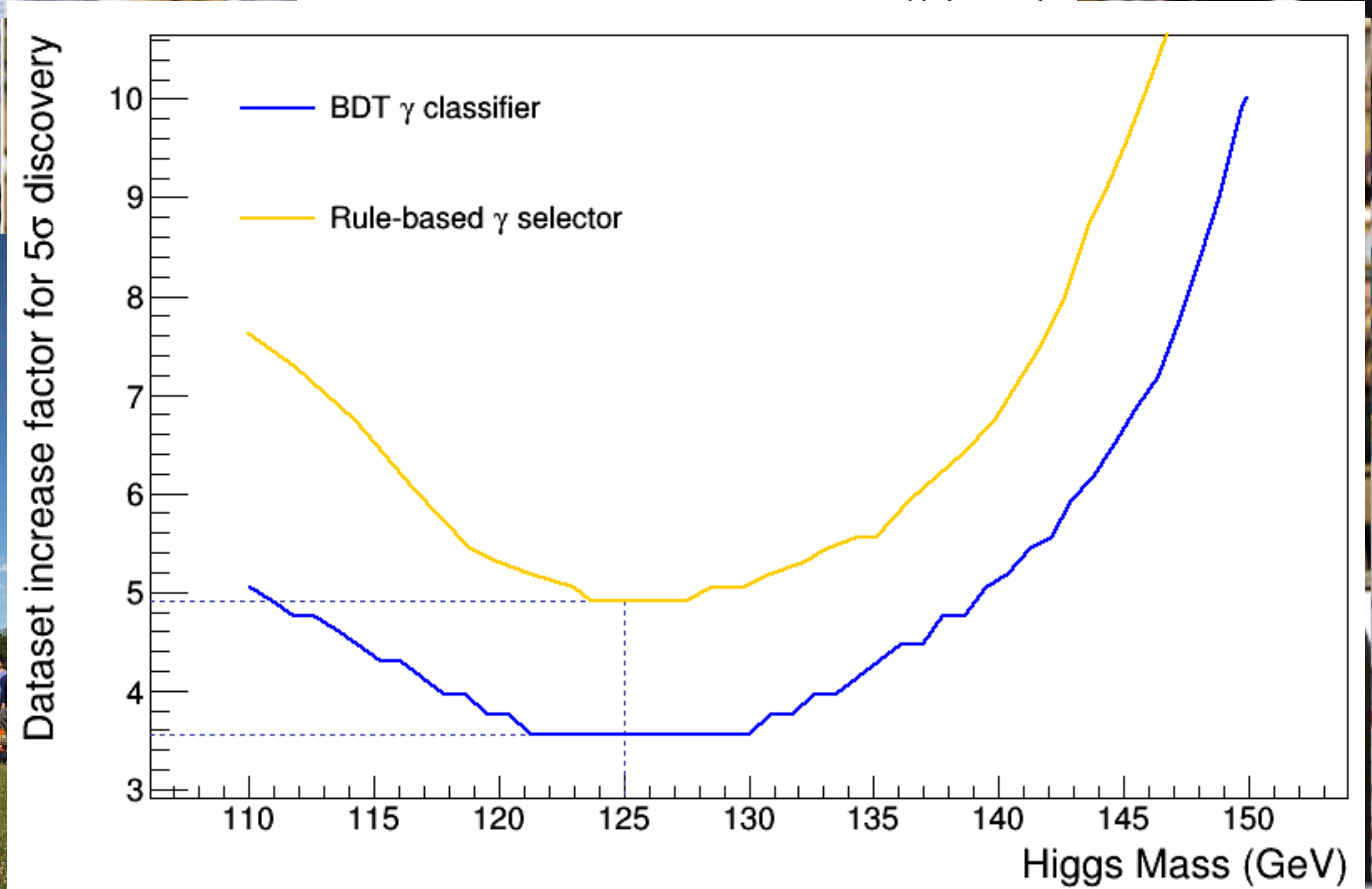


Spain bank debacle becomes legal tangle
DIE WELT
Union will Überschüsse in Sozialkassen horten
El juez i
32 cons
"Perché l'Italia
la Re
THE HINDU



19.7 fb^{-1} (8 TeV) + 5.1 fb^{-1} (7 TeV)
CMS
H → γγ
$m_H = \sqrt{2E_{\gamma_1}E_{\gamma_2}(1-\cos\theta_{\gamma_1\gamma_2})}$
arxiv:1407.0558
S/(S+B) weighted events / GeV
×10^3
5.7σ
B component subtracted
$m_{\gamma\gamma}$ (GeV)

ПОСЛЕДНИЙ КИРПИЧ В СТЕНУ МИРОЗДАНИЯ
МЕТРО СПУСТИТ НА ВОДУ
The Gazette
THE JERUSALEM POST
Mofaz rejects



BDT γ classifier
Rule-based γ selector
Dataset increase factor for 5σ discovery
Higgs Mass (GeV)

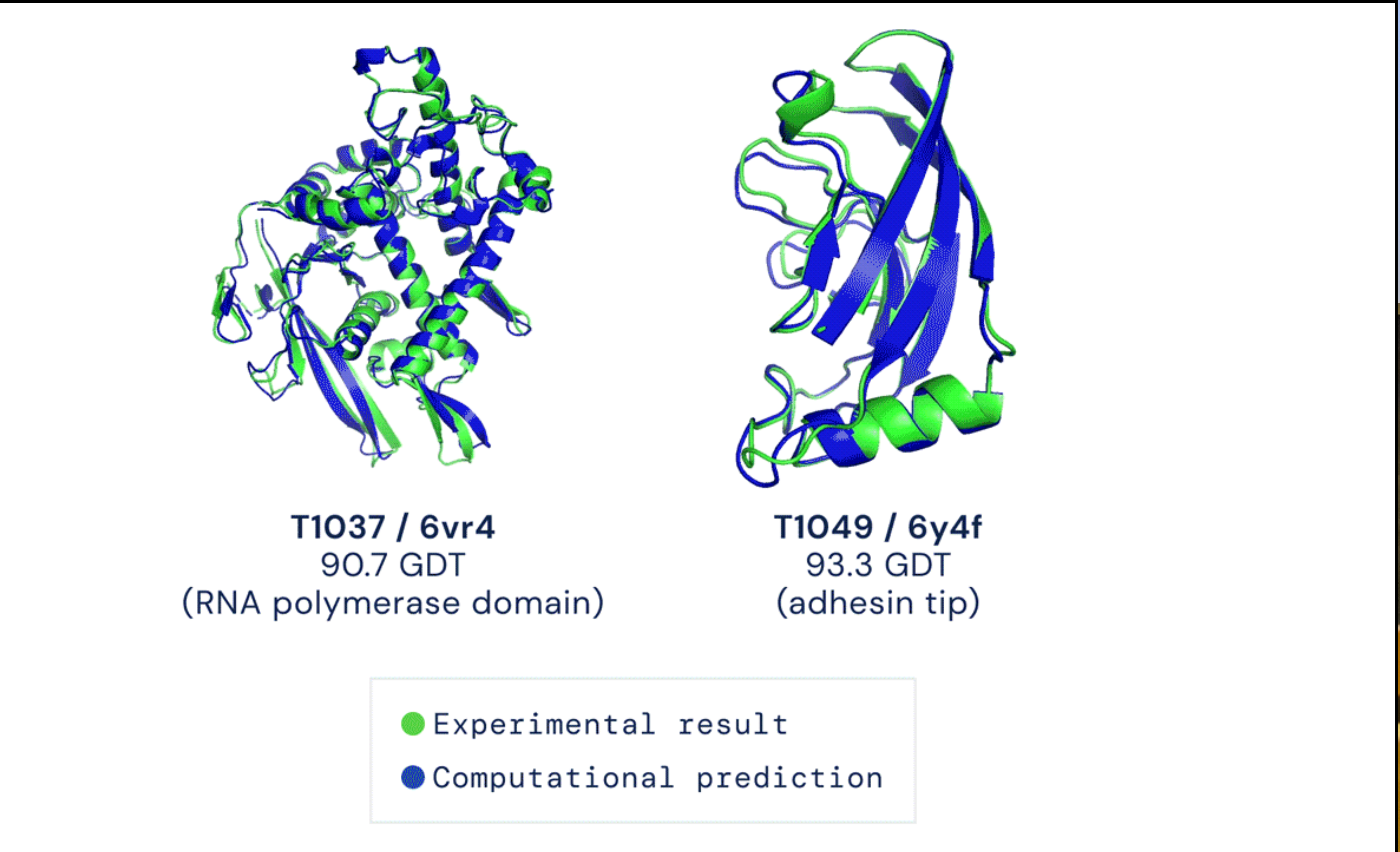
CERN Summer student 2012

Nature Review

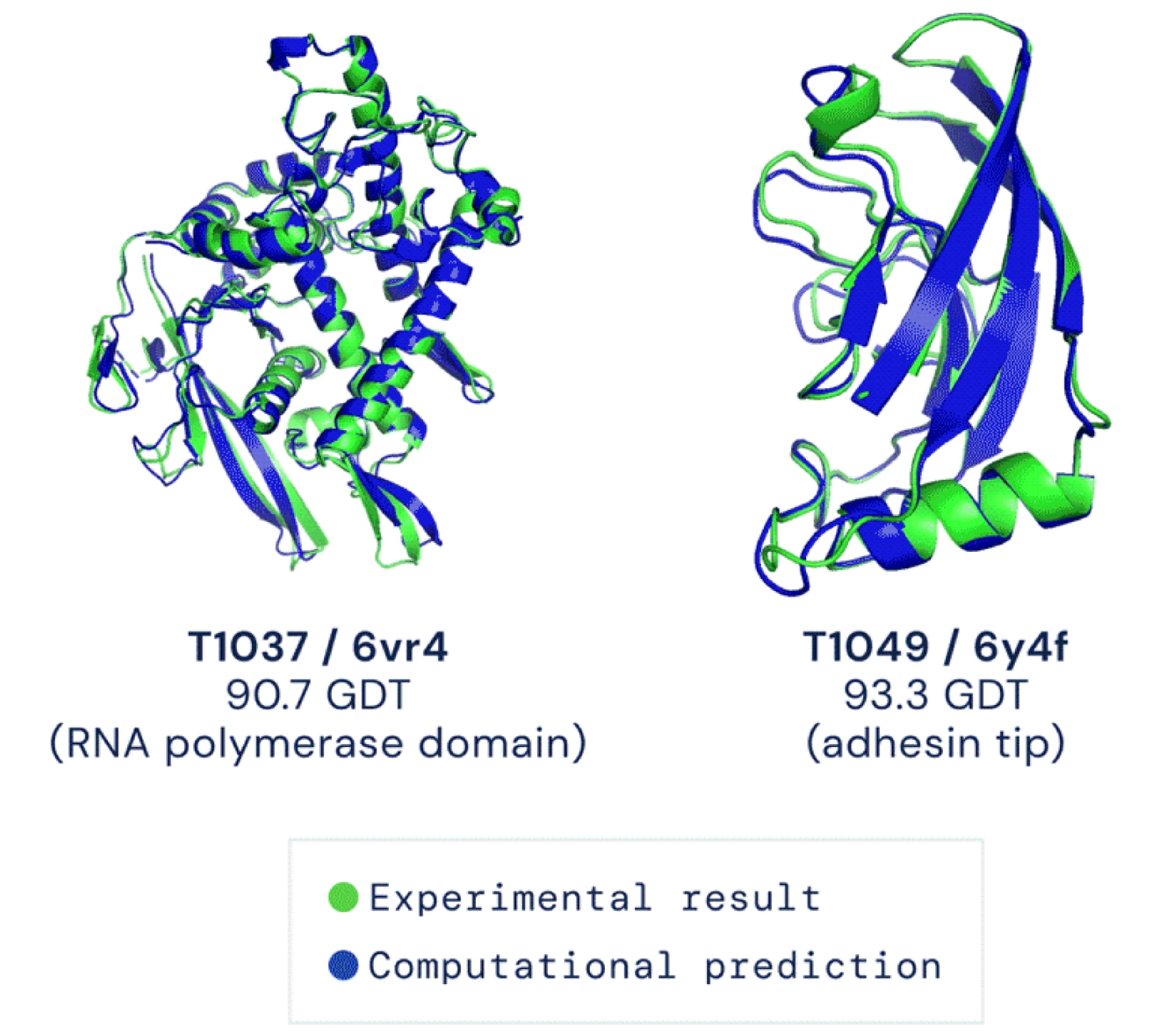
Analysis	Years of data collection	Sensitivity without machine learning	Sensitivity with machine learning	Ratio of P values	Additional data required
CMS[24] $H \to \gamma\gamma$	2011–2012	2.2σ, $P = 0.014$	2.7σ, $P = 0.0035$	4.0	51%
ATLAS[43] $H \to \tau^+\tau^-$	2011–2012	2.5σ, $P = 0.0062$	3.4σ, $P = 0.00034$	18	85%
ATLAS[99] $VH \to bb$	2011–2012	1.9σ, $P = 0.029$	2.5σ, $P = 0.0062$	4.7	73%
ATLAS[41] $VH \to bb$	2015–2016	2.8σ, $P = 0.0026$	3.0σ, $P = 0.00135$	1.9	15%
CMS[100] $VH \to bb$	2011–2012	1.4σ, $P = 0.081$	2.1σ, $P = 0.018$	4.5	125%

We were using ML for discovery very early on

CERN Summer student 2012

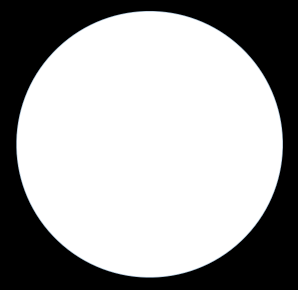


sequence—the structure prediction component of the 'protein folding problem'[8]—has been an important open research problem for more than 50 years[9]. Despite recent



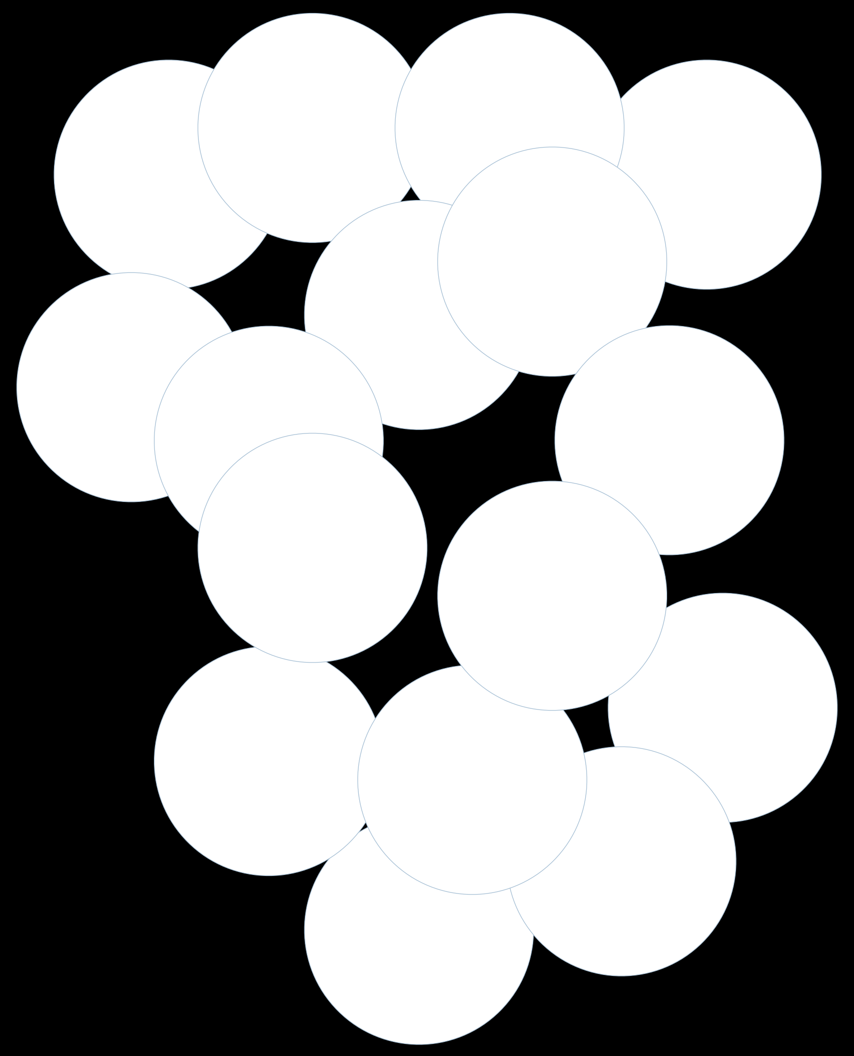
sequence—the structure prediction component of the 'protein folding problem'[8]—has been an important open research problem for more than 50 years[9]. Despite recent

GPT-3



175,000,000,000
(0.16% of neurons in your brain)

GPT-4 (MoE)



1,800,000,000,000
(1.6% of neurons in your brain)

semianalysis 2023

Train (GPT-4):
- 2.15^{25} floating point operations
- ~25,000 A100 GPUs
- 90-100 days
- $63 million
- Trained on 13 trillion tokens

Inference (GPT-4):
- Multiple clusters of 128 GPUs
- Model carefully mapped onto hardware

AI and Memory Wall
Transformer Size: 240x / 2 yrs
AI HW Memory: 2x / 2 yrs
10TB Baidu RecSys
2TB Baidu RecSys
GShard
GPT-3
Microsoft T-NLG
A100-80 (80GB)
Megatron LM
A100 (40GB)
V100 (32GB)
TPUv3 (32GB)
TPUv2 (16GB)
P100 (12GB)
GPT-2
BERT
ALBERT
ELECTRA
GPT-1
Transformer
Inception V4
ResNext101
ResNet50
DenseNet
Parameter Count (Billion)
10000
1000
100
10
1
0.1
0.01
2016
2017
2018
2019
2020
2021
YEAR

Computer vision: 10–100M trainable parameters (10^{18} –10^{19} floating point operations for training)
LLMs: 100M to 100Bs trainable parameters (10^{20}–10^{23} floating point operations for training)

Kaplan et al. (2020)

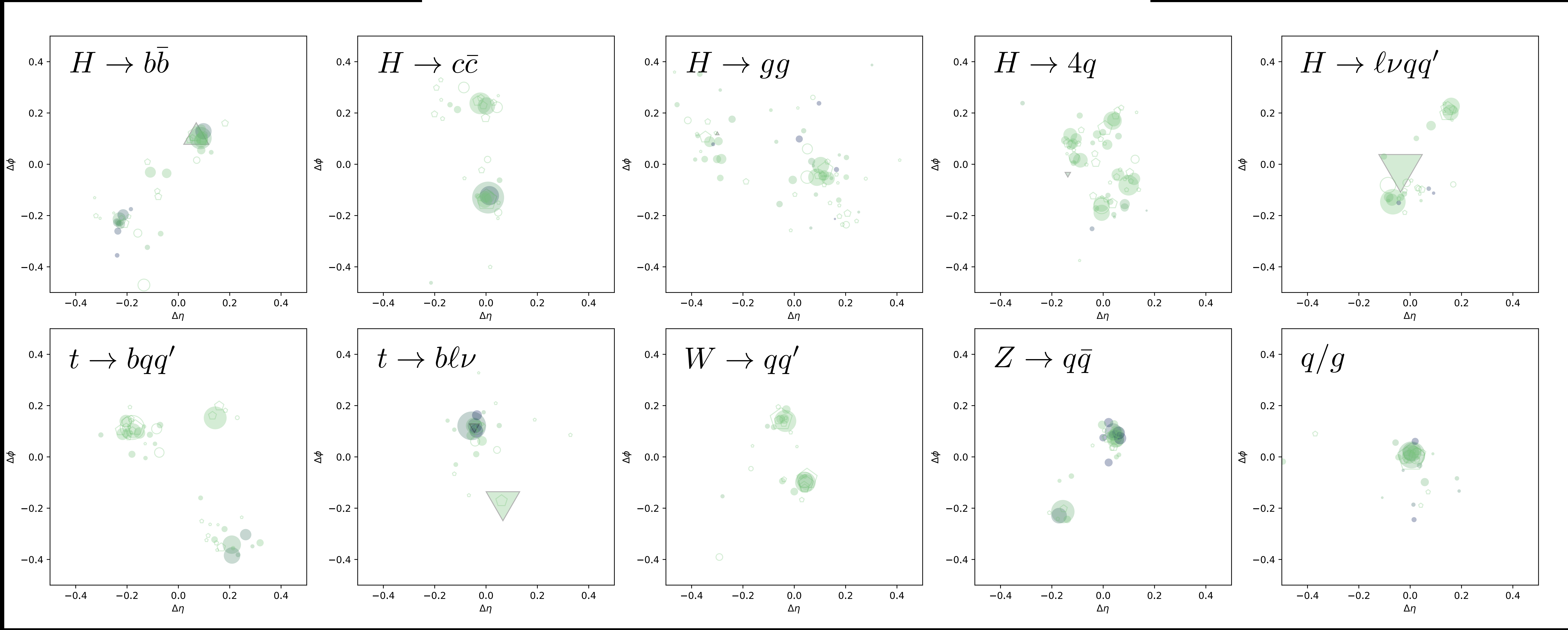
What is deep learning?

- **innovations in network structures**
- **strategies to train them**
- **dedicated hardware**

Kaplan et al. (2020)

100 million jets for training

	Accuracy	# params
PFN	0.772	86.1 k
P-CNN	0.809	354 k
ParticleNet	0.844	370 k
ParT	**0.861**	2.14 M
ParT (plain)	0.849	2.13 M

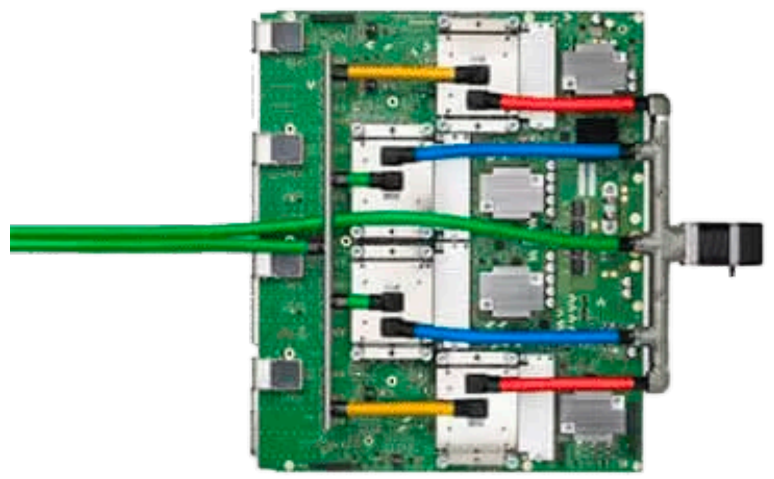


"Particle Transformer For Jet Tagging" H. Qu, C. Li, S. Qian

What has changed?

Krizhevsky et al. [2012]:

Artificial Neural Network with a **simple structure**
(known for >20 years [LeCun et al., 1989]),
Beat complex SOTA image recognition methods by huge margin
How? x100 larger and trained on a data set x100 larger



TPU

GPU

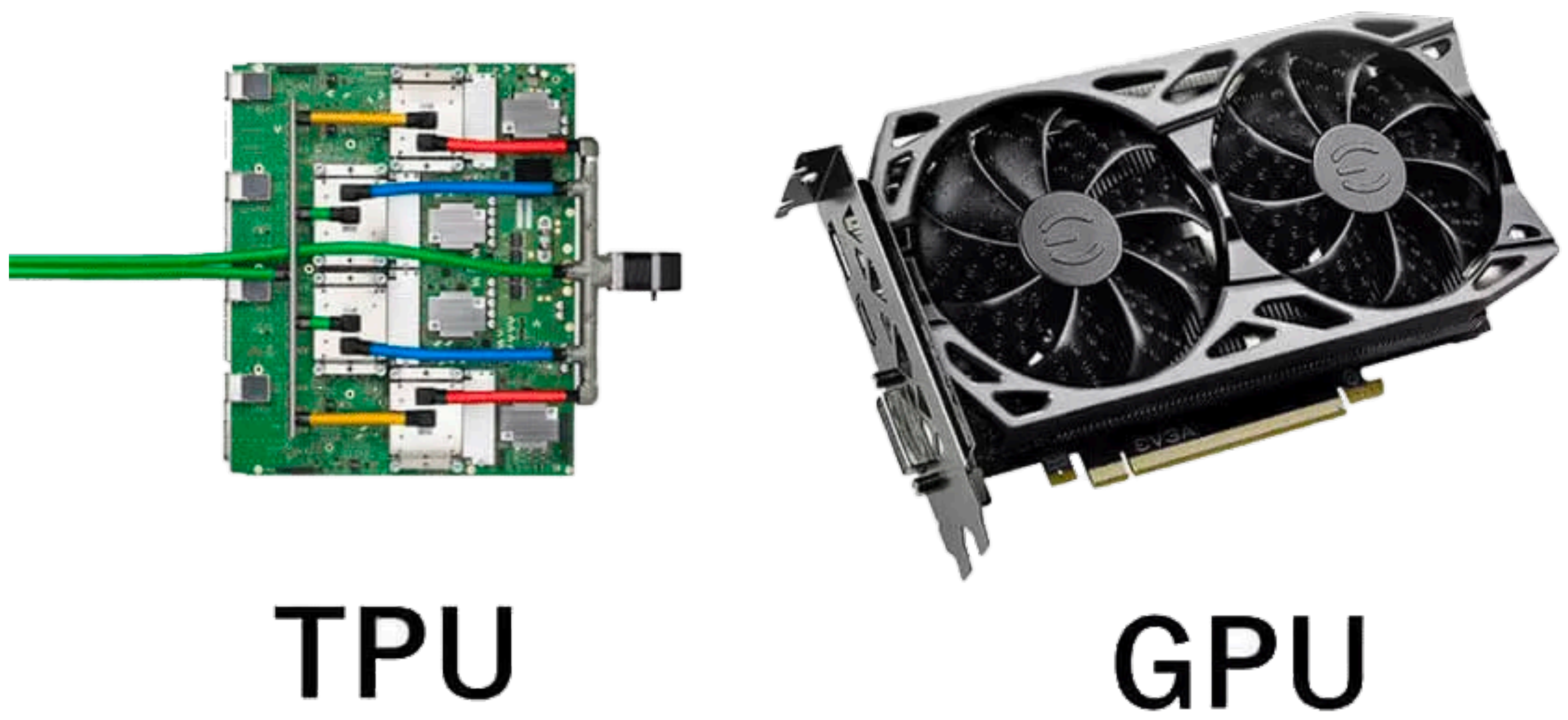
What has changed?

Krizhevsky et al. [2012]:

Artificial Neural Network with a **simple structure**
(known for >20 years [LeCun et al., 1989]),
Beat complex SOTA image recognition methods by huge margin
How? x100 larger and trained on a data set x100 larger

Made possible due to

Graphical Processing Units (GPUs)
Data, data and data!



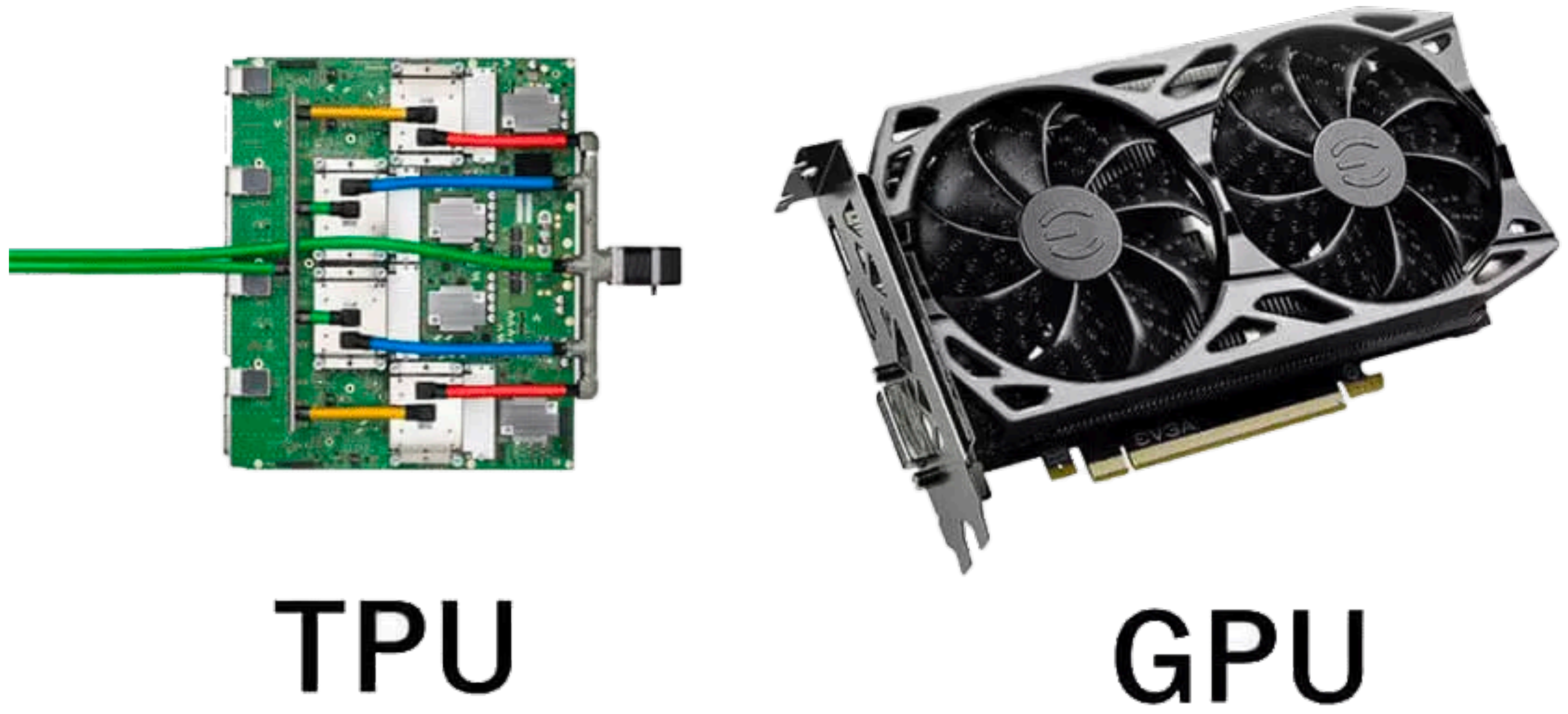
What has changed?

Krizhevsky et al. [2012]:
Artificial Neural Network with a **simple structure**
(known for >20 years [LeCun et al., 1989]),
Beat complex SOTA image recognition methods by huge margin
How? x100 larger and trained on a data set x100 larger

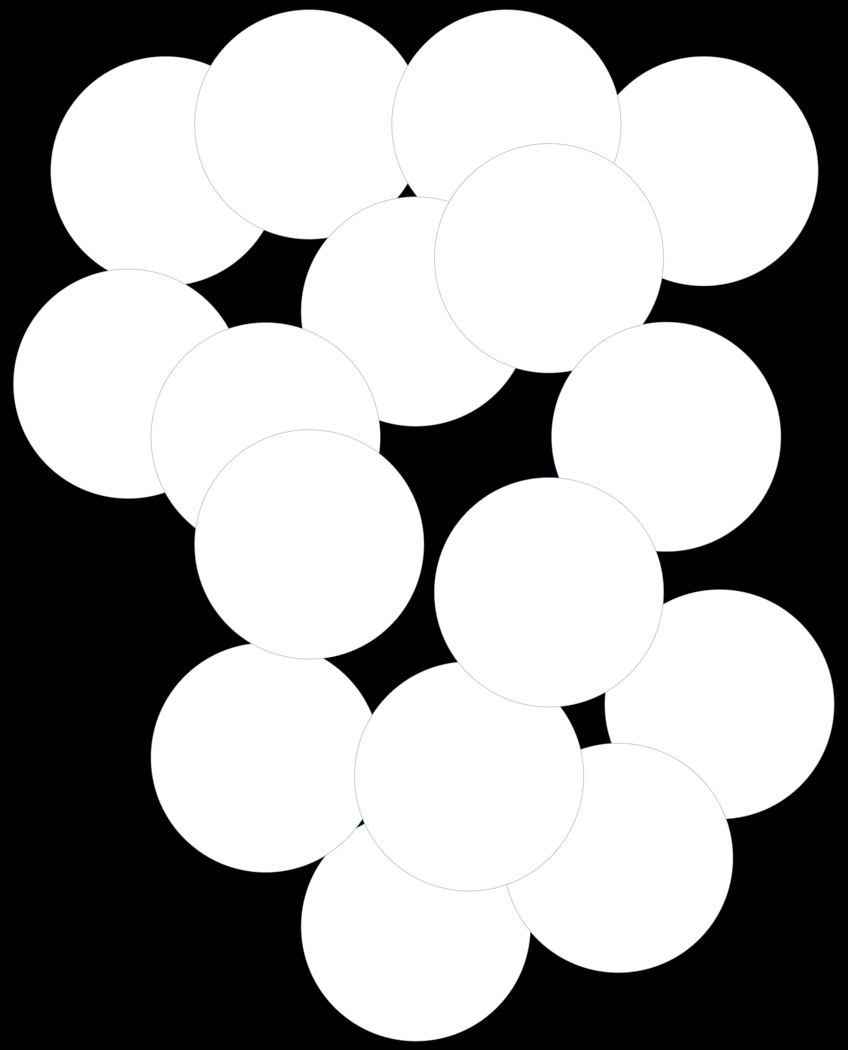
Made possible due to
Graphical Processing Units (GPUs)
Data, data and data!

Deep Learning:
innovations in network structures,
strategies to train them,
and dedicated hardware

Exponential increase in size and quantity of training data [Sevilla et al., 2022]!

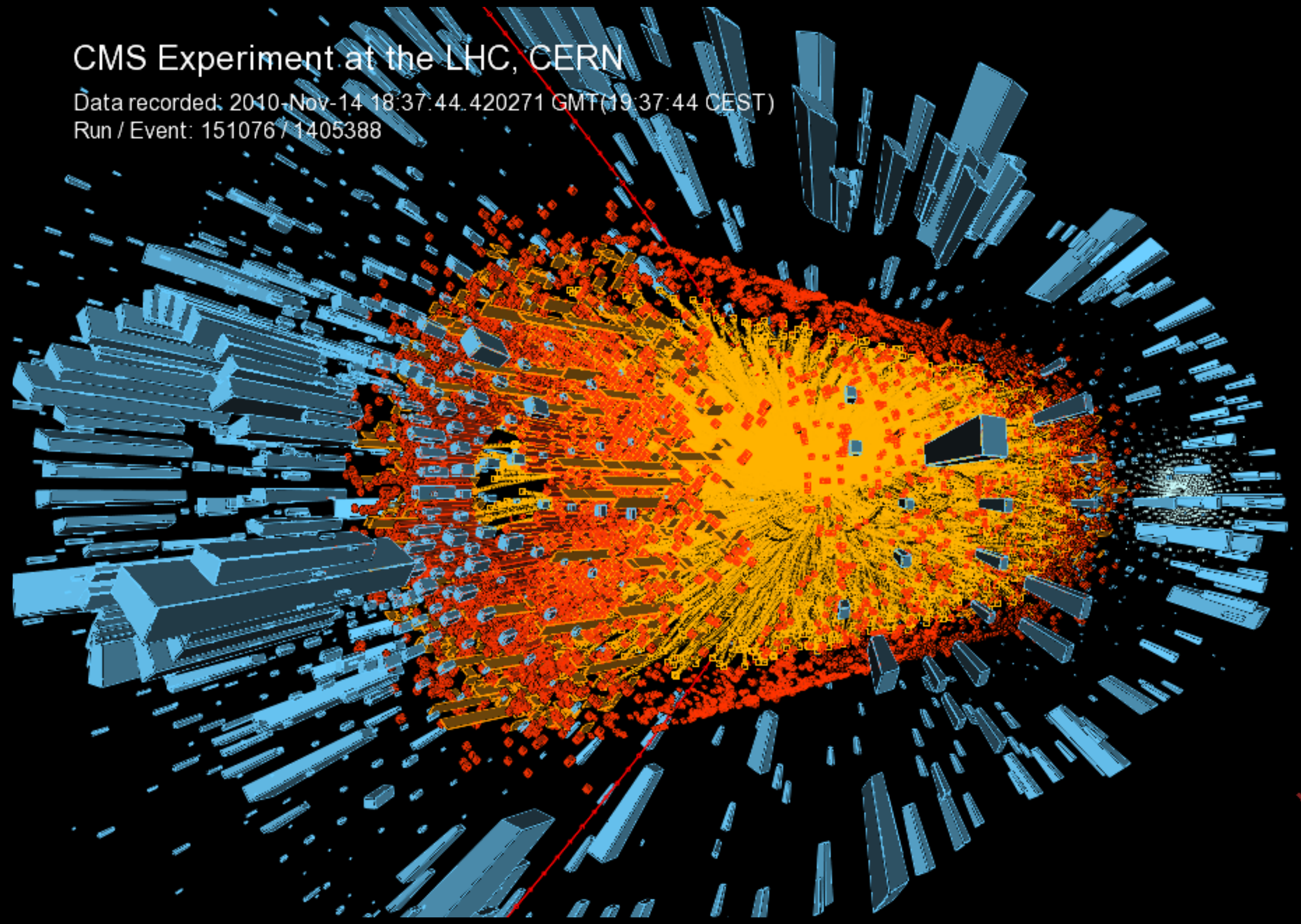


GPT-4(omni?)



↔

?


CMS Experiment at the LHC, CERN
Data recorded: 2010-Nov-14 18:37:44.420271 GMT(19:37:44 CEST)
Run / Event: 151076 / 1405388

Where would you say Machine Learning has the potential to make the largest contributions within high energy physics?

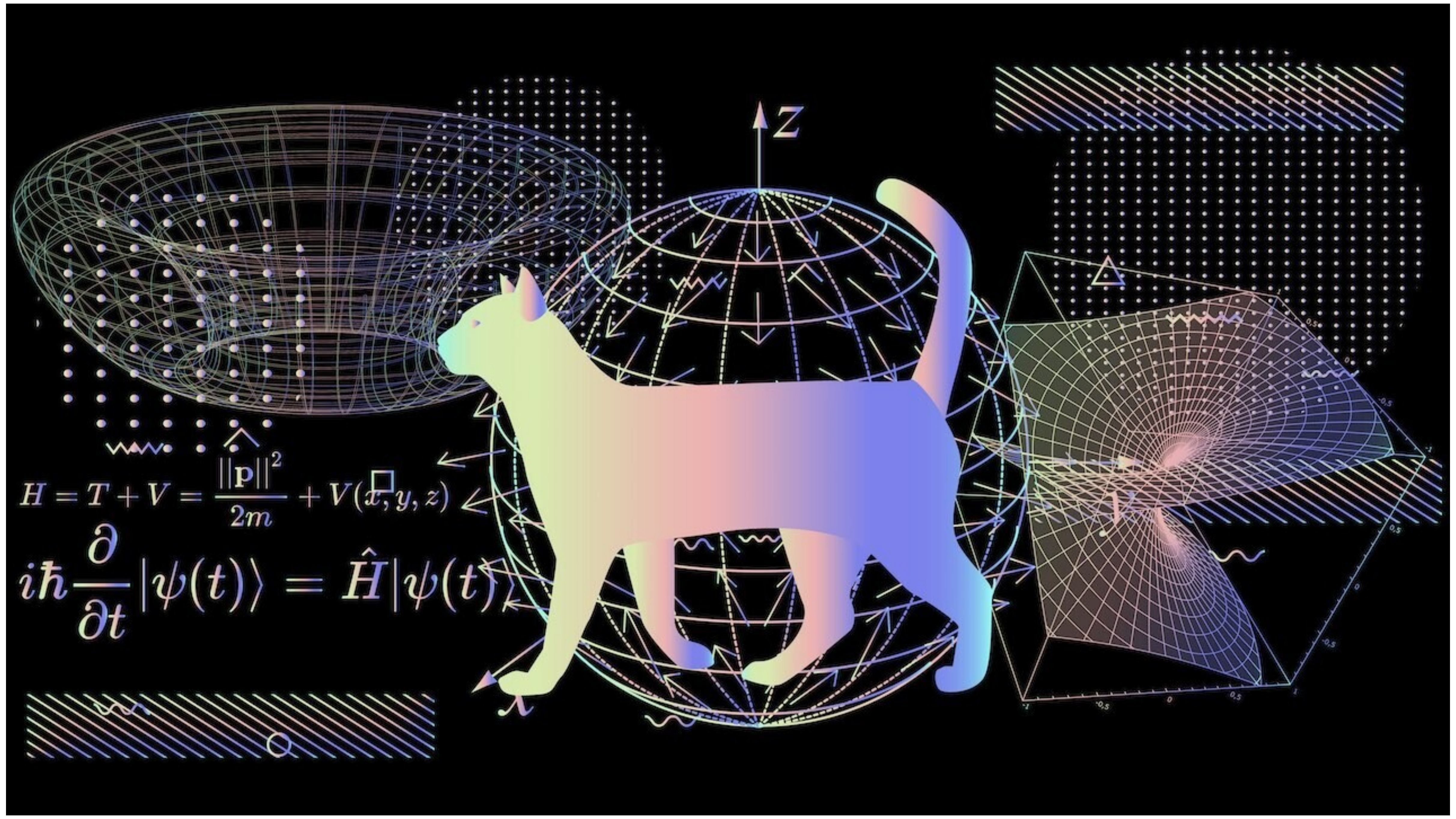
Machine learning has the potential to make significant contributions within high-energy physics in various areas. Here are some of the key areas where machine learning can have a substantial impact:

1. **Data Analysis and Event Classification:** High-energy physics experiments generate massive amounts of data, and sifting through this data to identify interesting events and classify them is a challenging task. Machine learning algorithms can help automate this process by efficiently identifying and categorizing particle collisions, reducing the workload on physicists.
2. **Anomaly Detection:** Machine learning can be used to detect anomalies or rare events in experimental data. This is crucial for identifying unexpected or new physics phenomena that may deviate from existing theories. Algorithms like autoencoders and anomaly detection models can help in this regard.
3. **Particle Tracking and Reconstruction:** Accurate reconstruction of particle trajectories and interactions is essential in high-energy physics experiments. Machine learning can improve the precision and efficiency of particle tracking, helping physicists better understand the paths and properties of particles.
4. **Simulation and Generative Models:** Generative models like Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) can be used to simulate high-energy physics events and particle interactions. These models can generate synthetic data that can be compared with experimental data, aiding in background estimation and calibration.
5. **Detector Calibration and Optimization:** Machine learning can assist in calibrating and optimizing particle detectors. Algorithms can automatically adjust detector parameters to improve the accuracy of measurements and reduce systematic errors.
6. **Rare Signal Searches:** High-energy physics experiments often involve searching for rare signals amidst a vast amount of background noise. Machine learning techniques, including advanced statistical analysis and deep learning, can enhance the sensitivity of these searches.
7. **Data Compression:** Storing and transmitting large datasets from particle collisions can be resource-intensive. Machine learning can be used for data compression techniques, allowing for more efficient storage and transmission of experimental data.
8. **Dark Matter and Beyond the Standard Model Physics:** Machine learning can help in the search for dark matter and other particles beyond the Standard Model. Advanced algorithms can assist in designing experiments, analyzing data, and identifying potential signatures of new physics.
9. **Optimizing Experimental Design:** Machine learning can assist in optimizing the design of high-energy physics experiments. It can help determine the most informative configurations and setups for detectors, accelerators, and data collection strategies.
10. **Reducing Human Bias:** Machine learning can help reduce human bias in data analysis by providing automated and unbiased methods for data interpretation.

It's important to note that the application of machine learning in high-energy physics often requires collaboration between physicists and machine learning experts to develop specialized algorithms and models tailored to the unique challenges of the field. Additionally, the interpretation of results and the validation of machine learning techniques remain crucial aspects of ensuring the reliability of the findings in high-energy physics experiments.

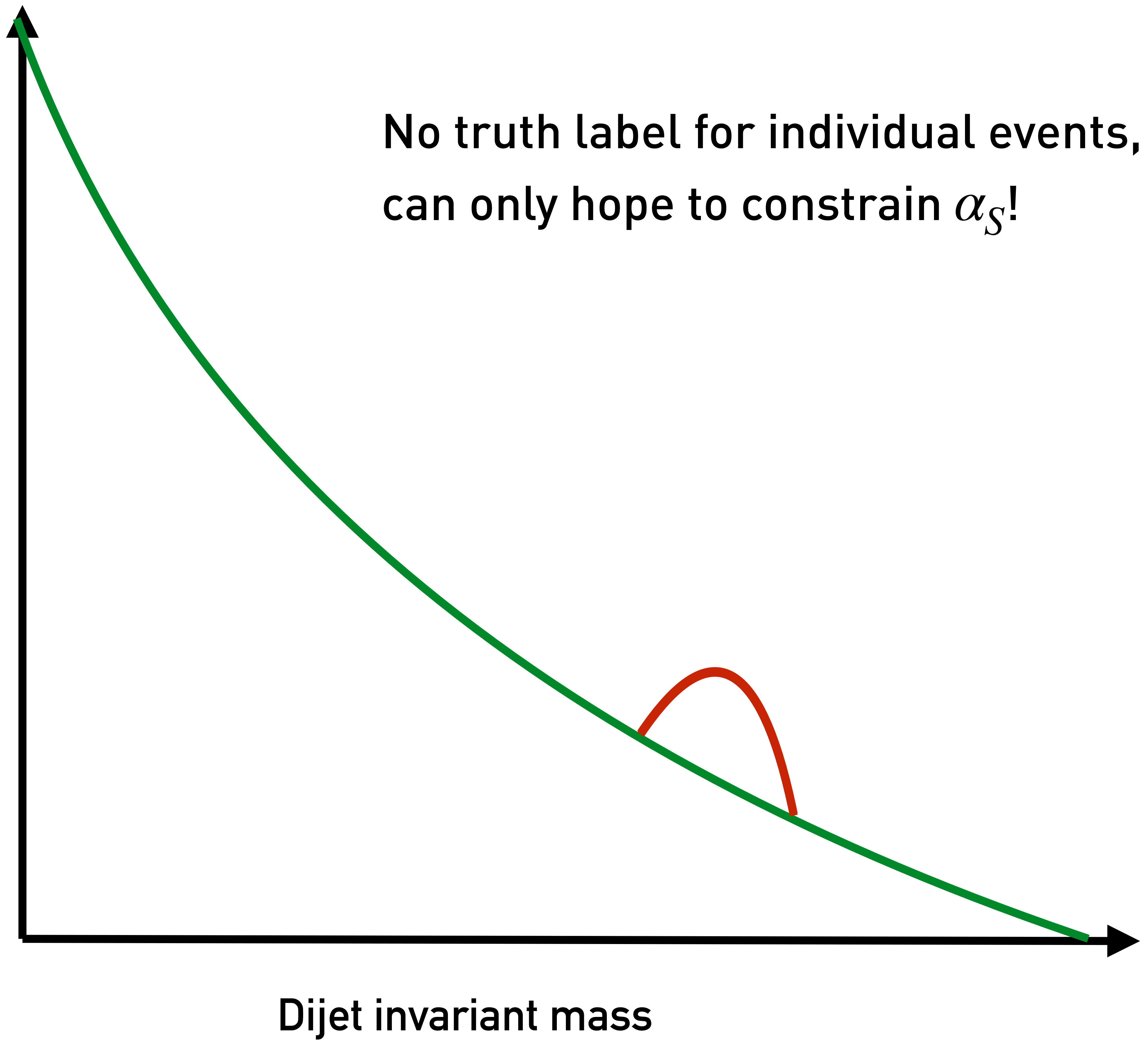
What makes particle physics special?

$$dP^n_{data} = |M_S + M_B|^2 dp_1 dp_2 \ldots dp_n$$



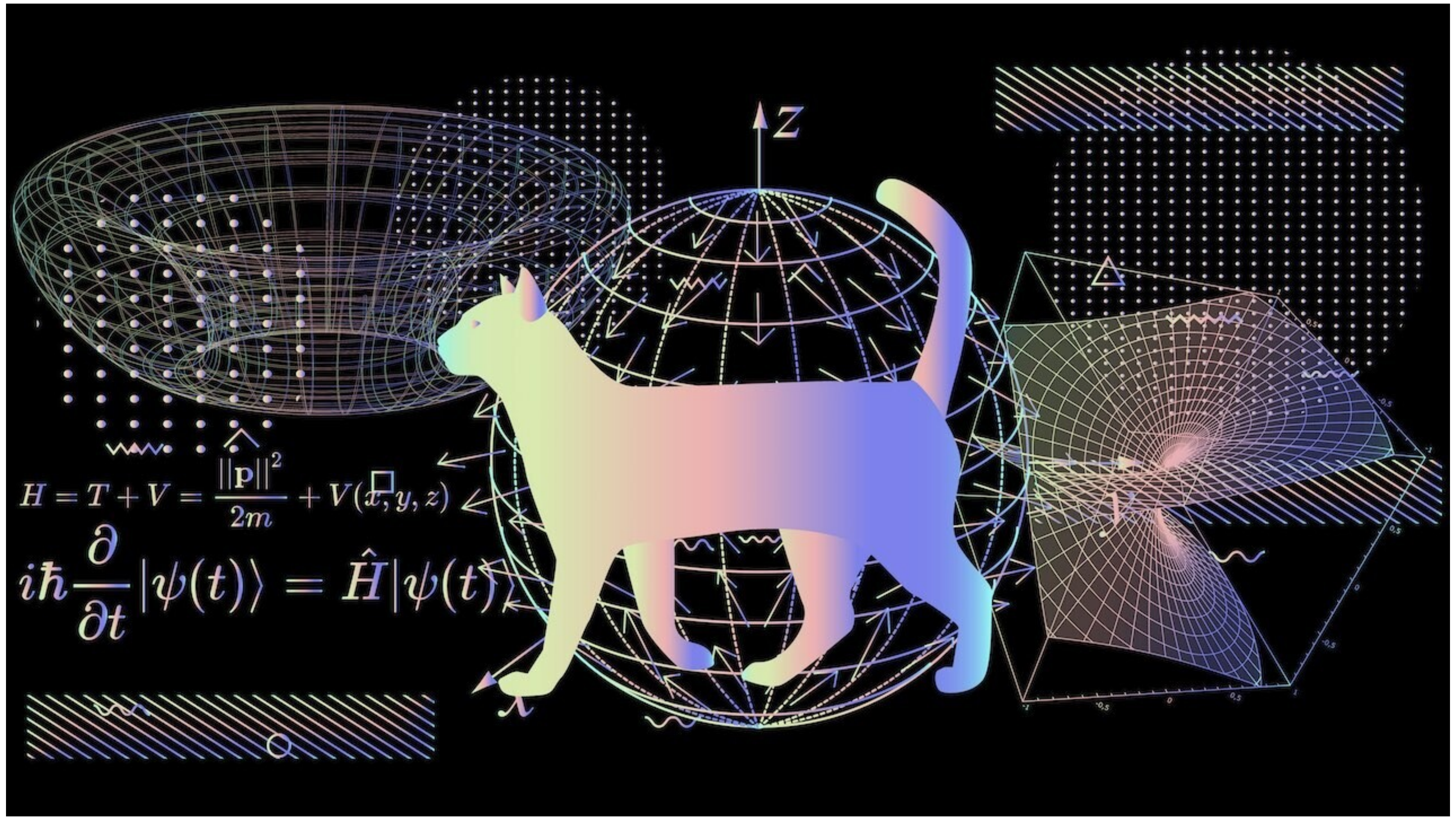
$$M_S M_B{}^* + M_B M_S{}^*$$

$$P_{data} = \alpha_S P_S + \alpha_B P_B$$



It's against physical law to annotate our data!

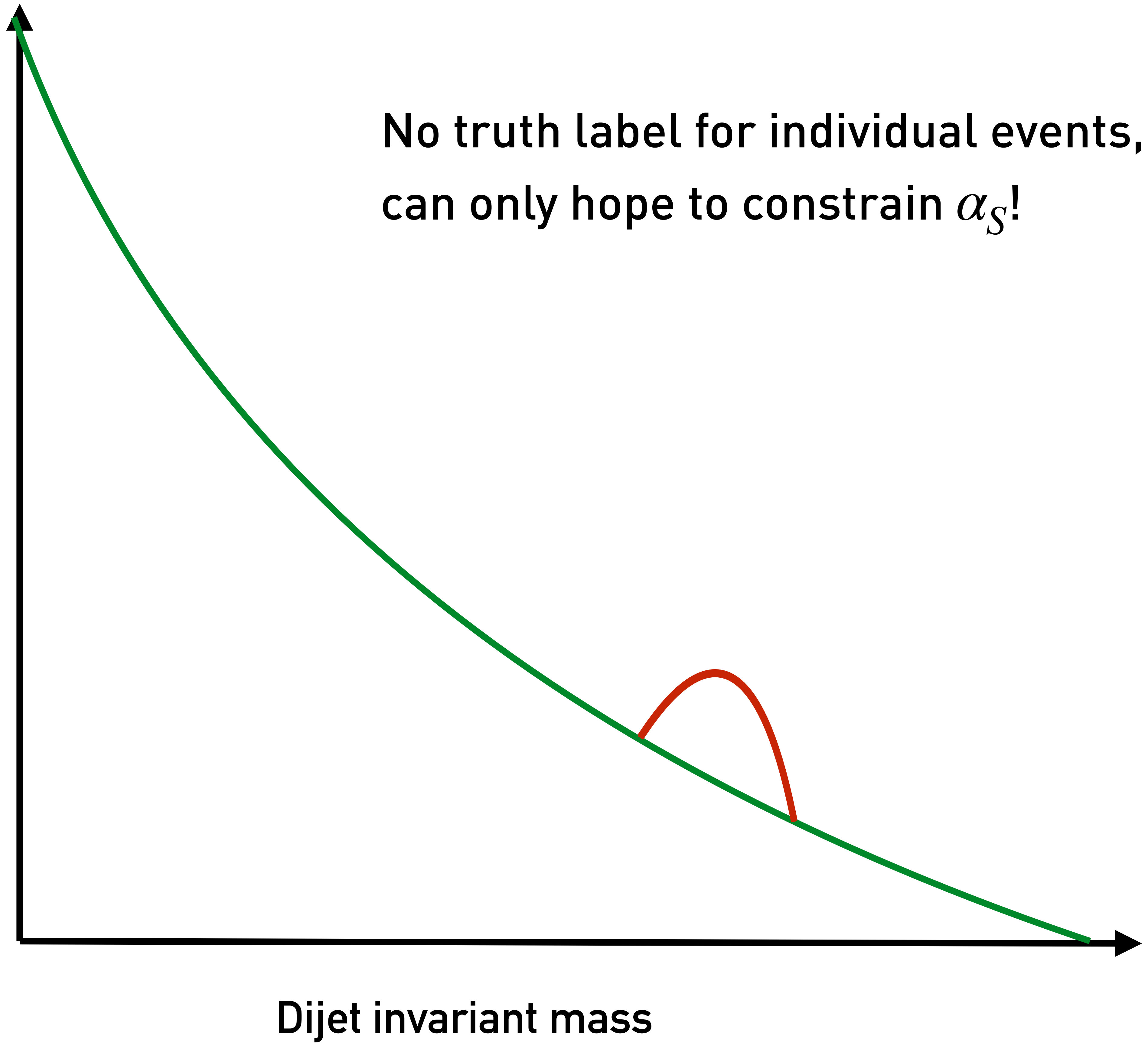
$$dP^n_{data} = |M_S + M_B|^2 dp_1 dp_2 \ldots dp_n$$

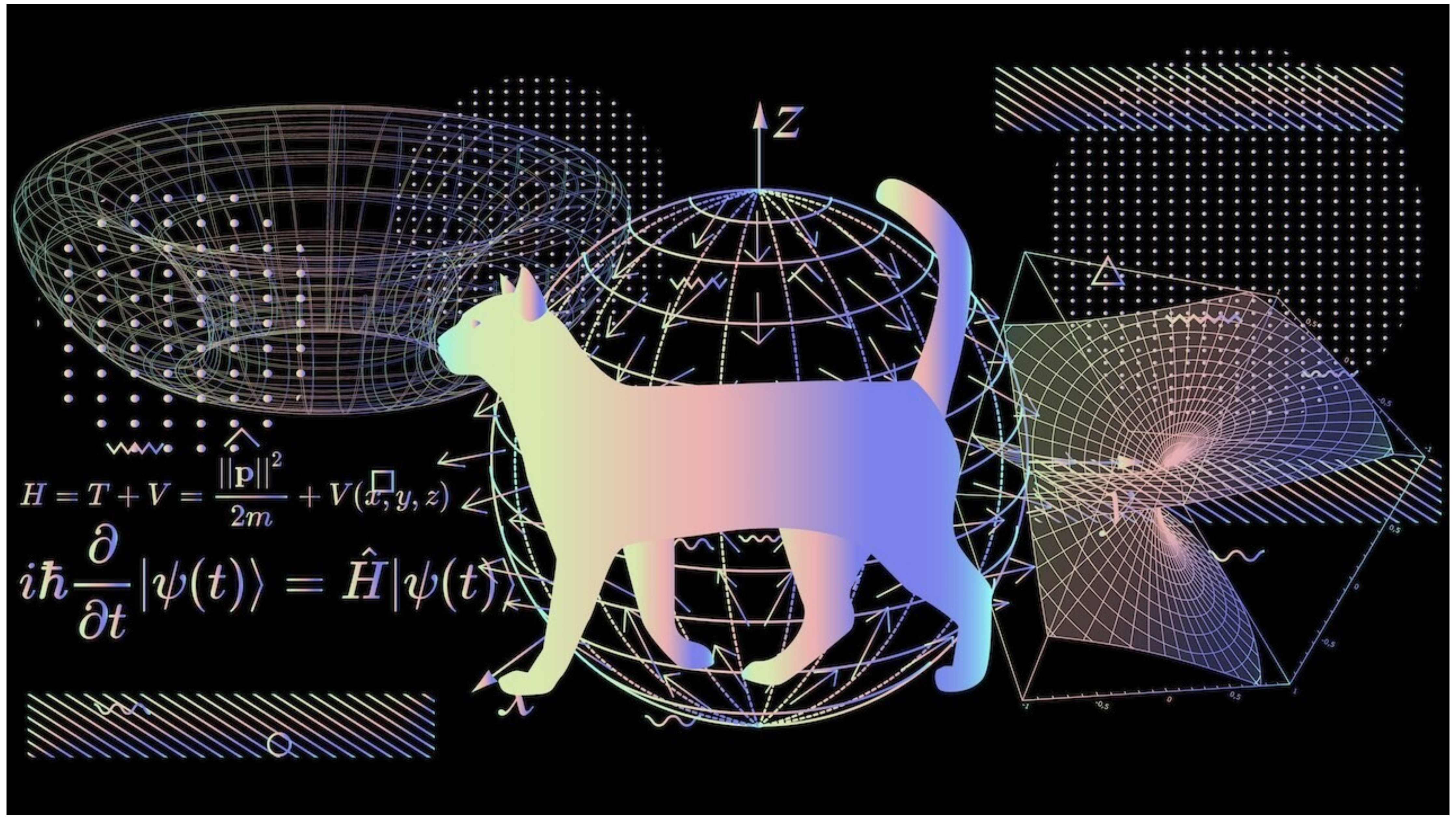


$$M_S M_B{}^* + M_B M_S{}^*$$

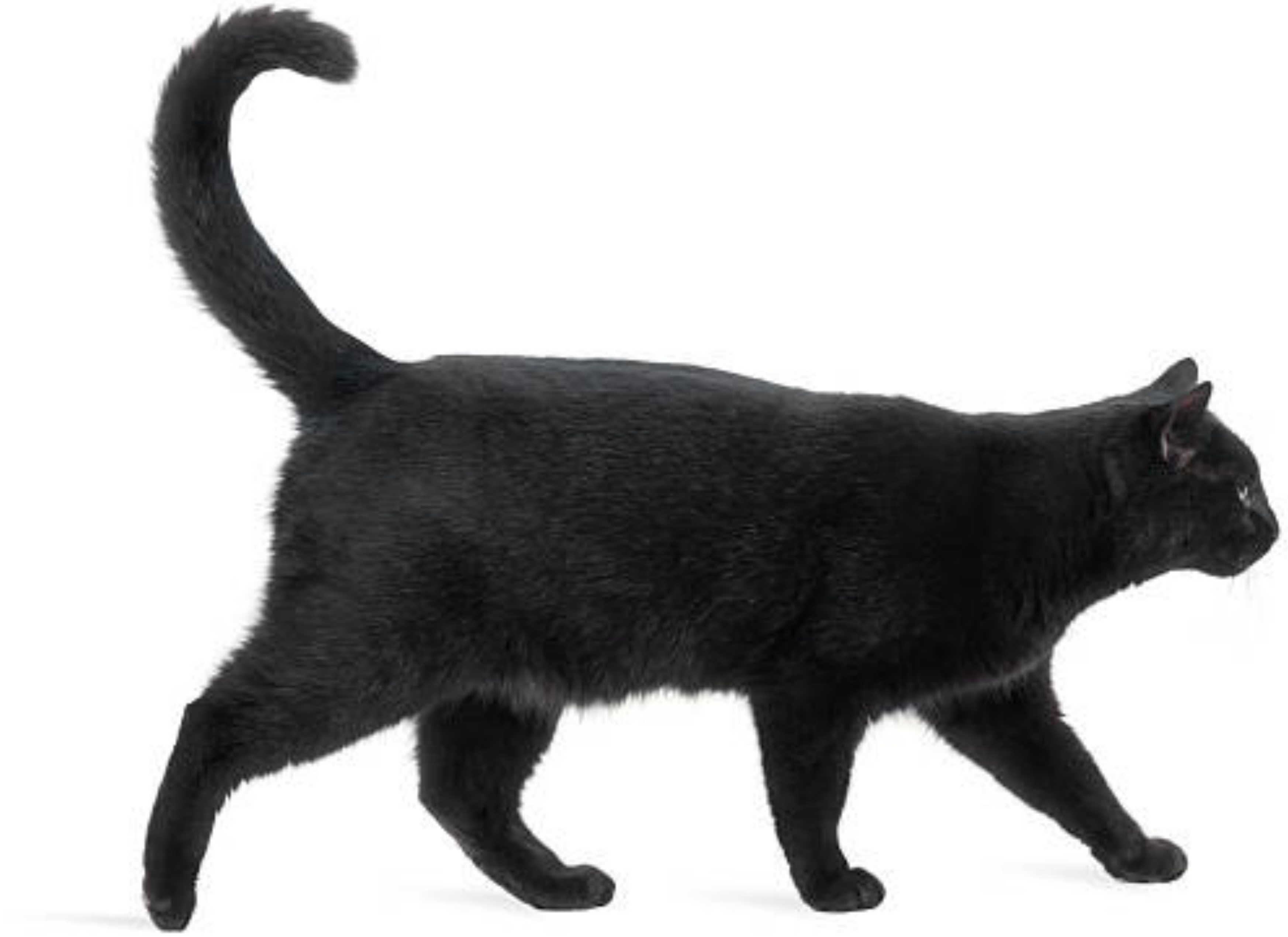
$$P_{data} = \alpha_S P_S + \alpha_B P_B$$

No truth label for individual events, can only hope to constrain α_S!




Z
$H = T + V = \frac{\|\mathbf{p}\|^2}{2m} + V(x, y, z)$
$i\hbar \frac{\partial}{\partial t}|\psi(t)\rangle = \hat{H}|\psi(t)\rangle$

!=



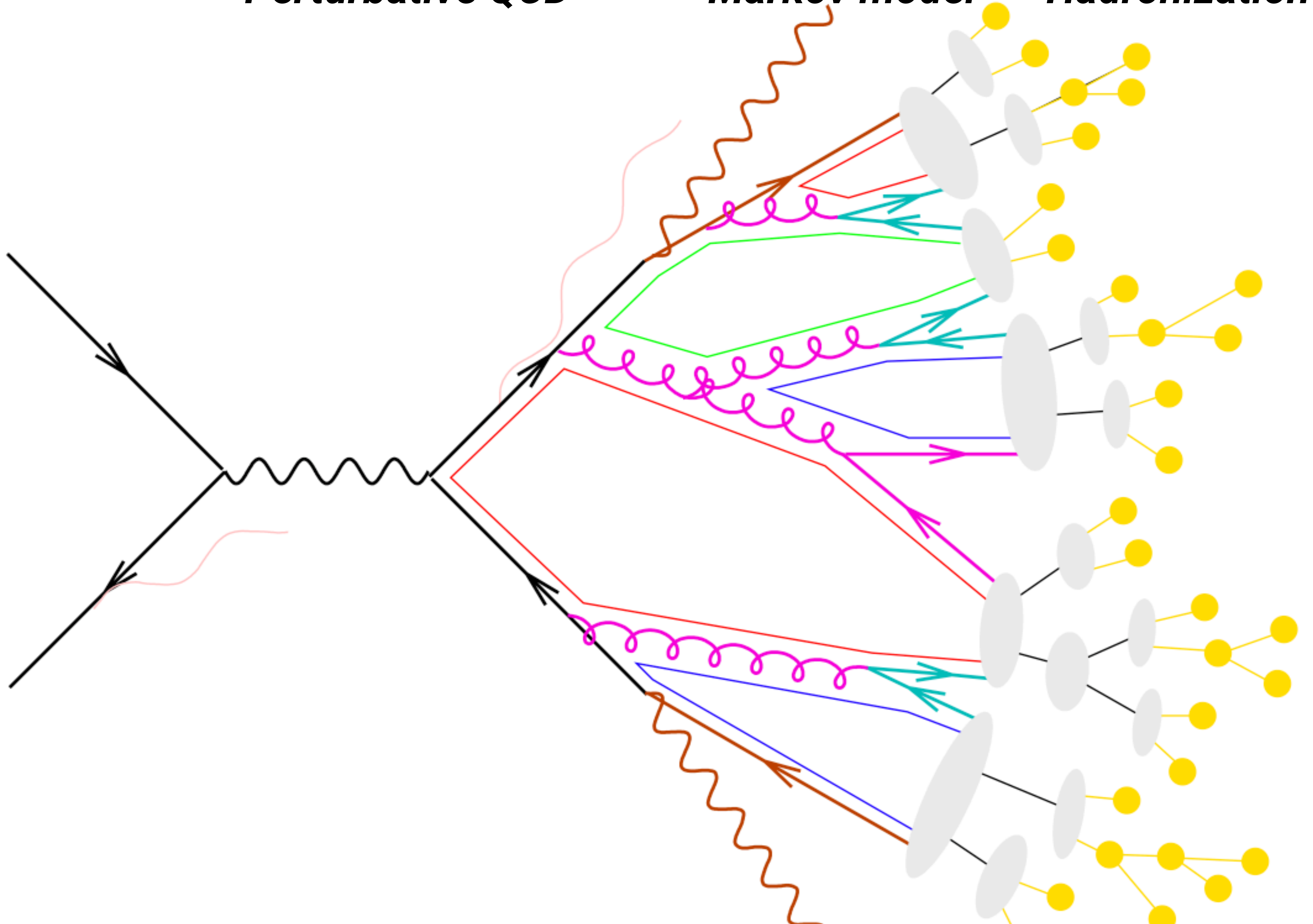
Monte Carlo Simulation

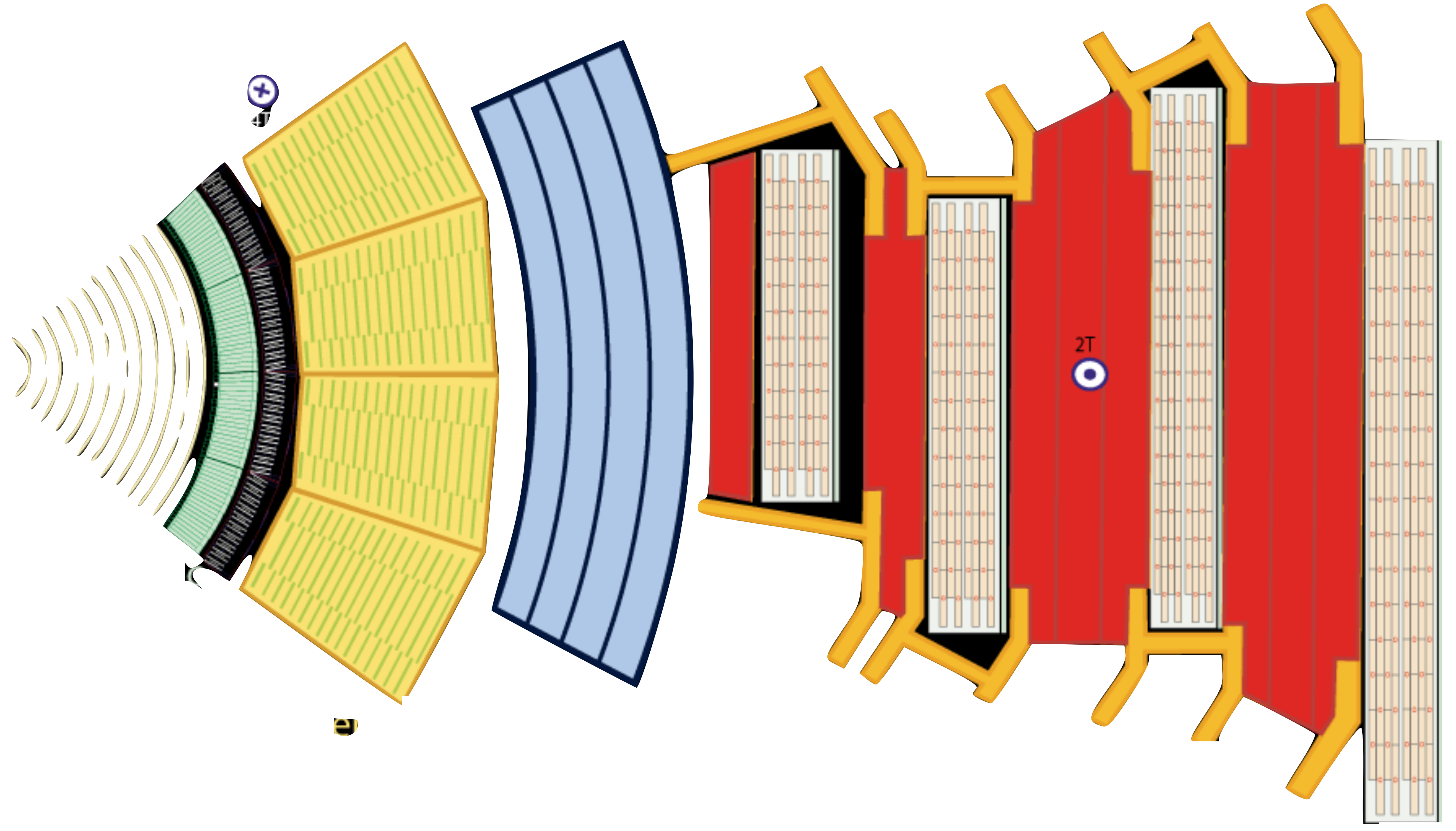
Dimensions

$O(10)$ $O(10^3)$ $O(10^{10})$

10^{-18}m 10^{-15}m 10^{-6}m 100m

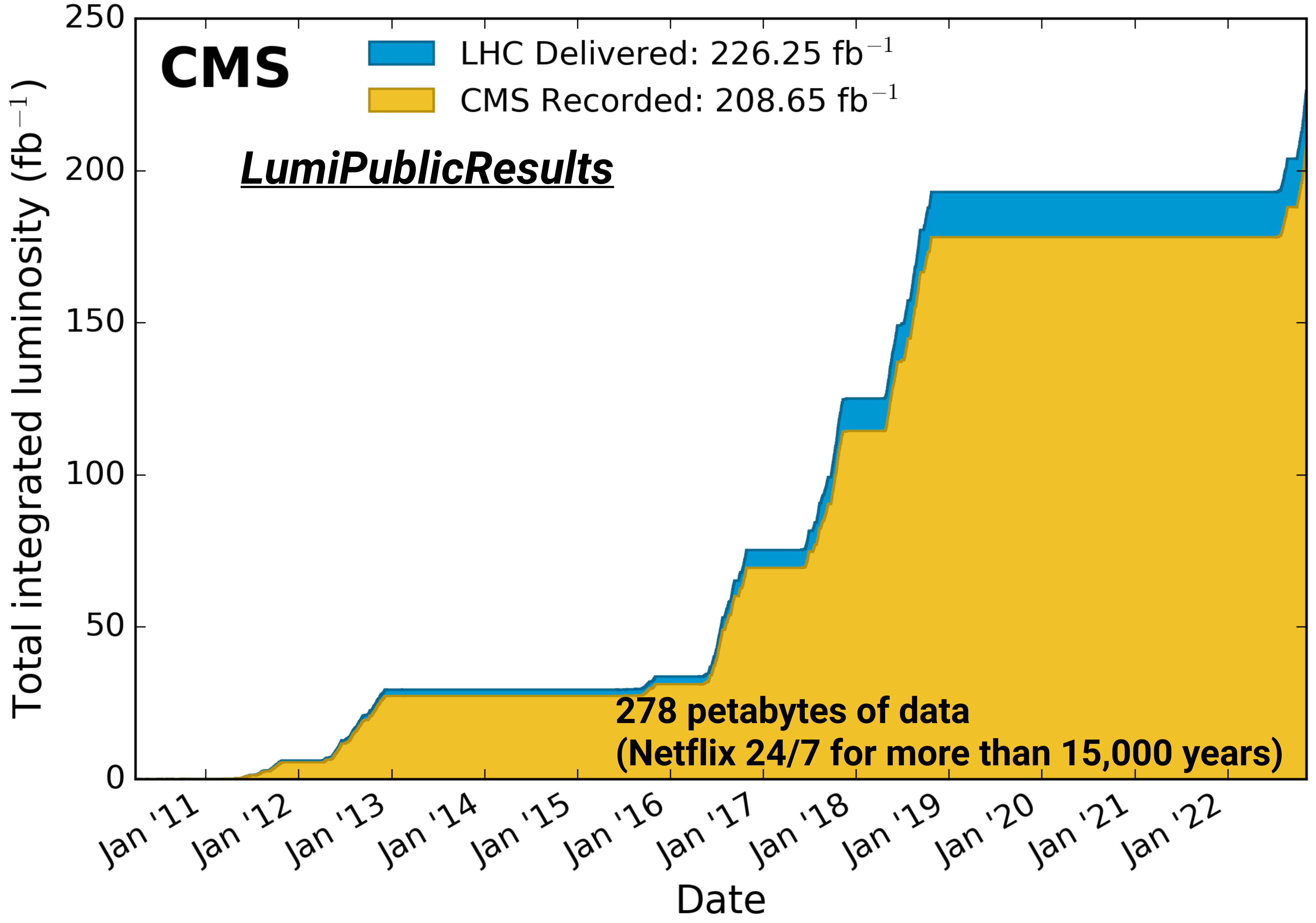
Perturbative QCD ***Markov model*** ***Hadronization*** ***Detection***



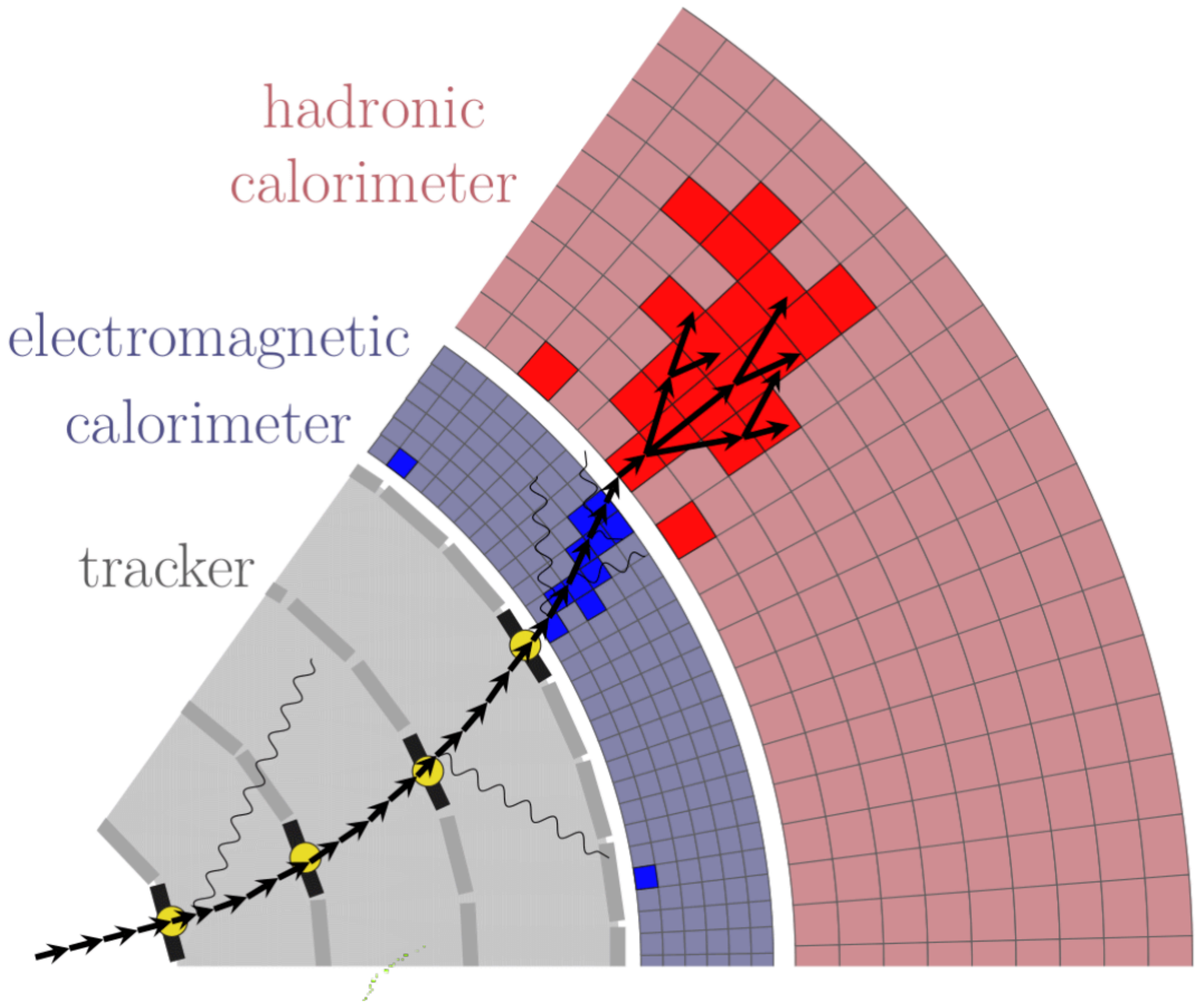


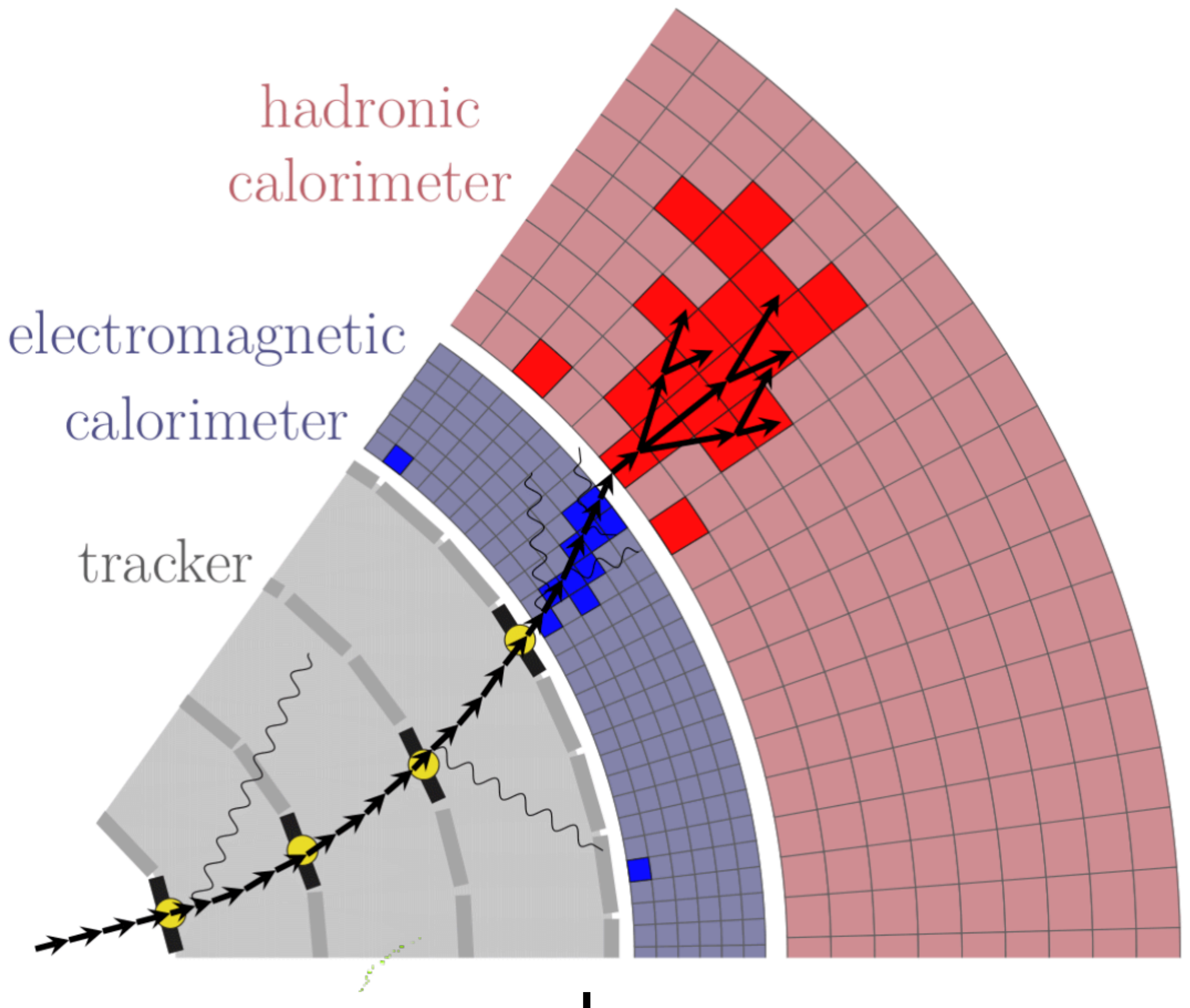
CMSOfflineComputingResults

~40 quadrillion collisions recorded at LHC
(1 fb^{-1} ~ 100 trillion collisions)



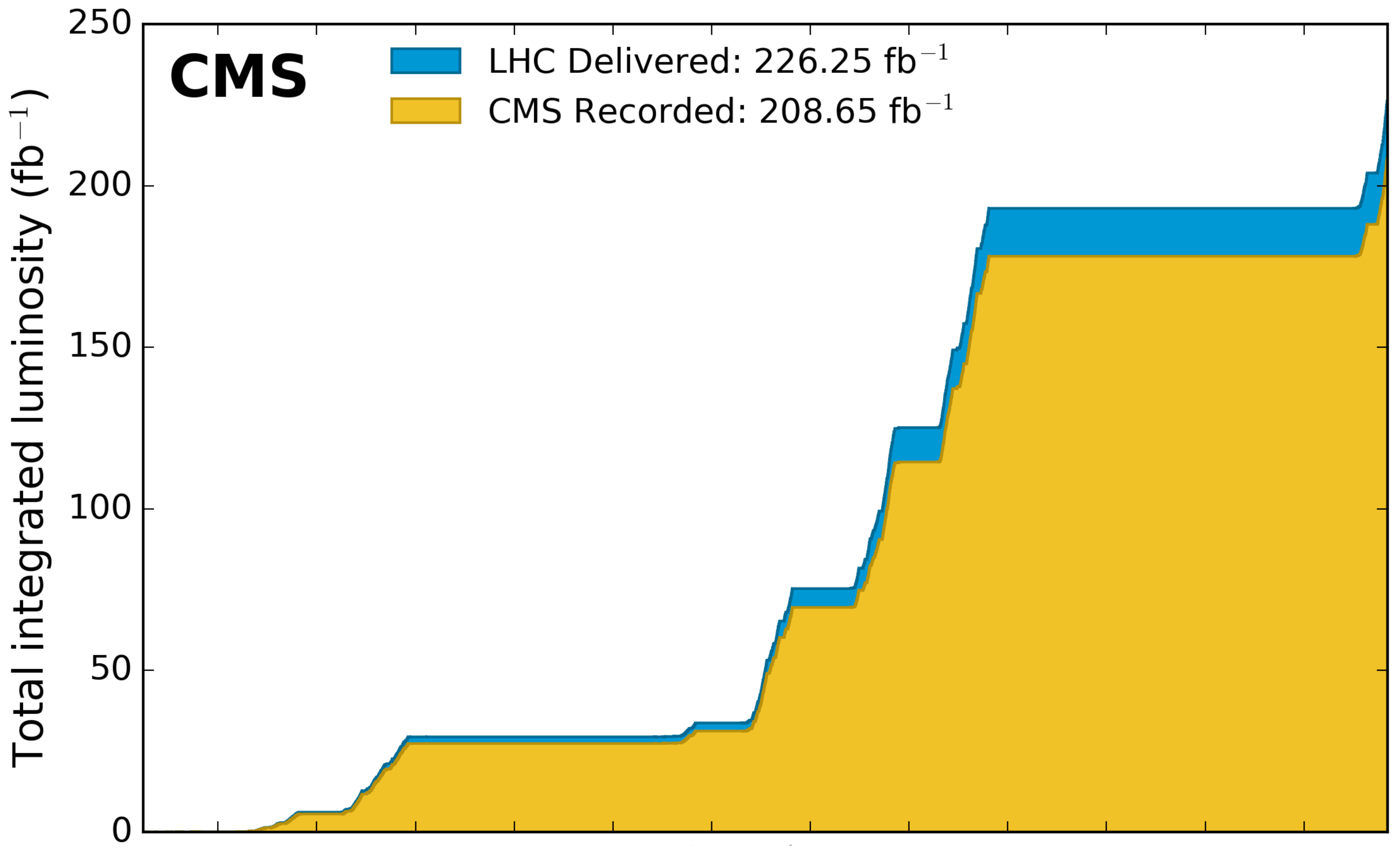
O(1) trillion simulated events





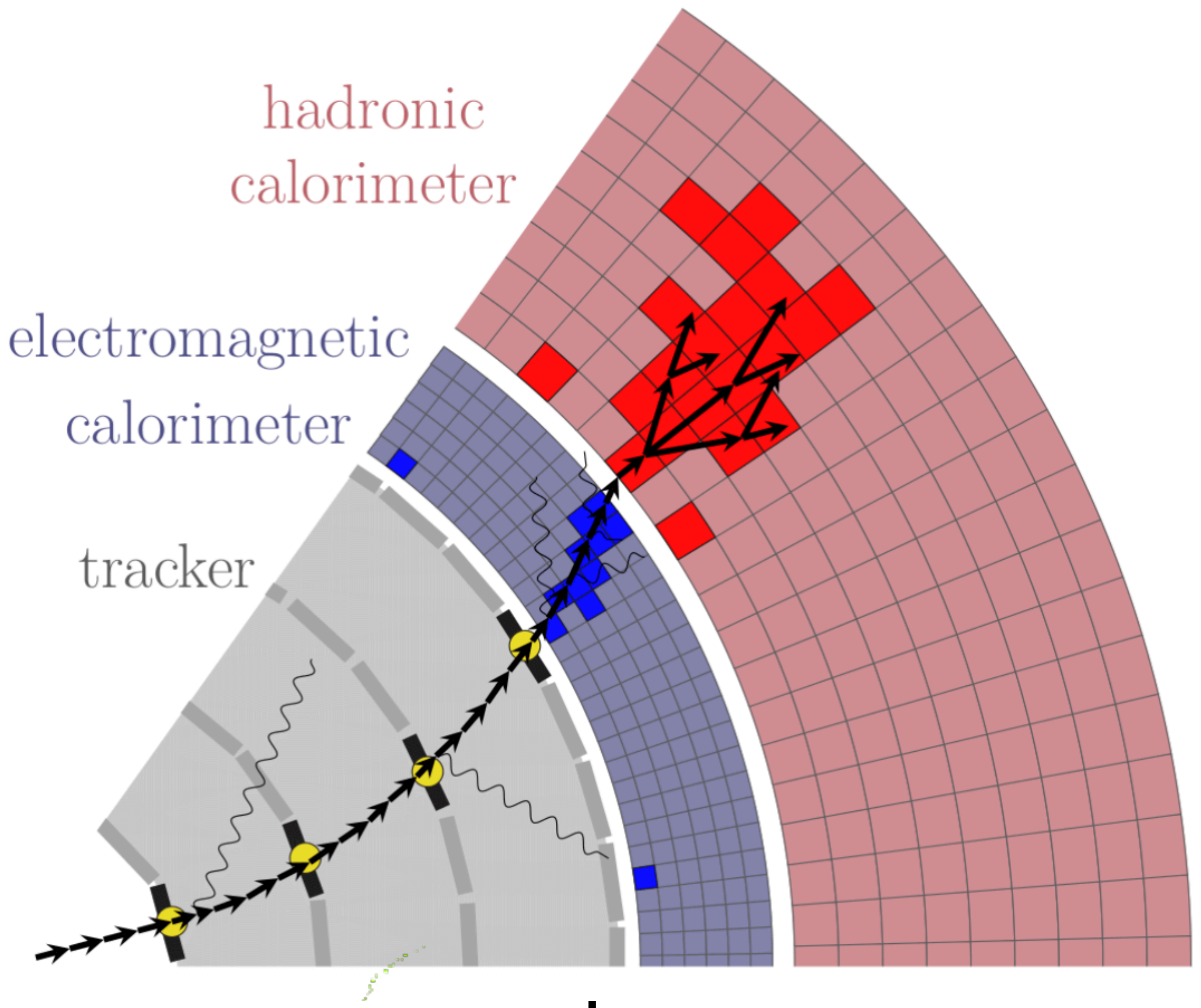
Fully supervised
- Requires truth labels
- Only possible using simulation

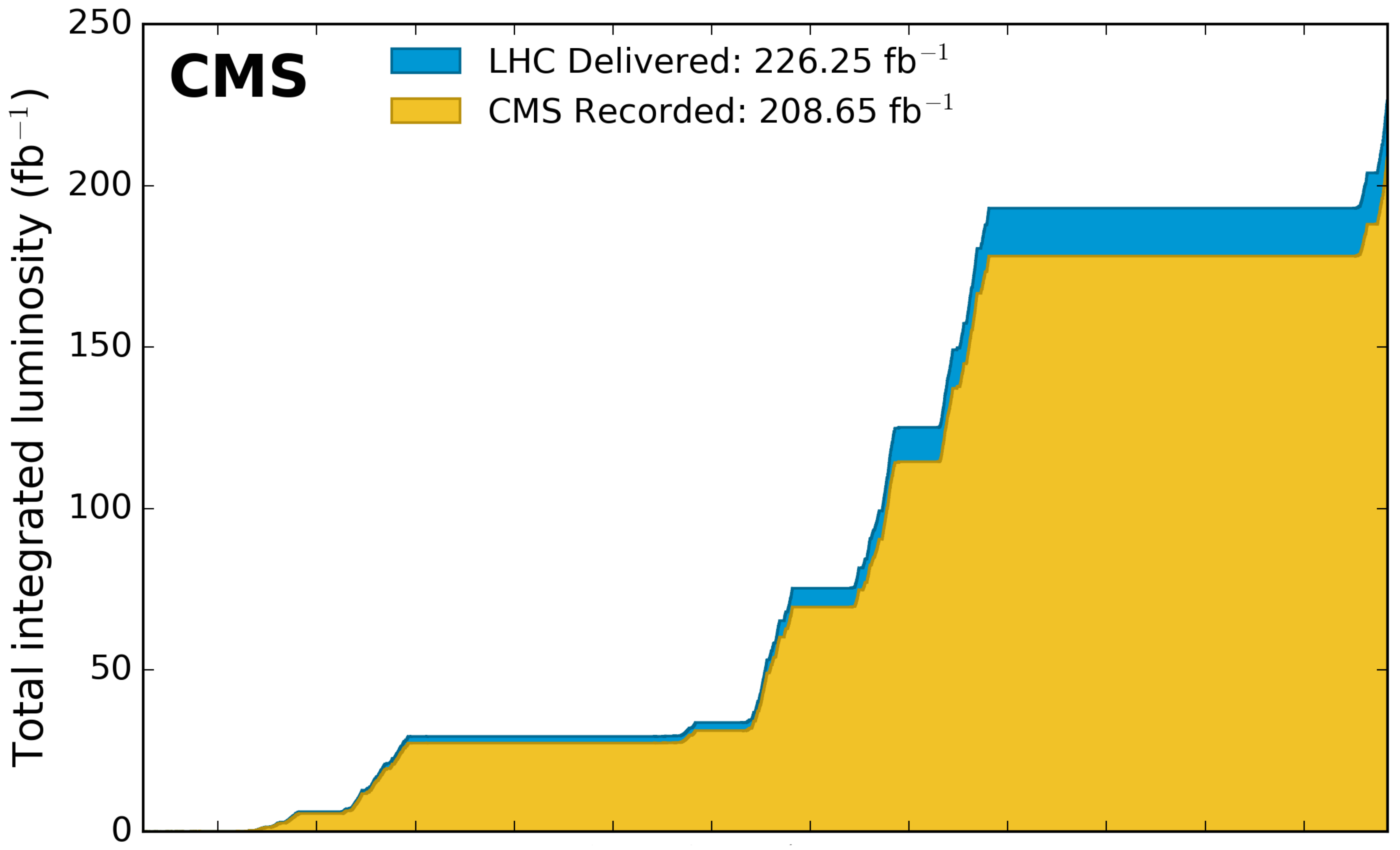
We have a lot of high quality simulated data that we want to use



Unsupervised/SSL
- No labels, completely data driven

We are also very keen on using this!





Simulation != test data

Fully supervised

- Requires truth labels
- Only possible using simulation

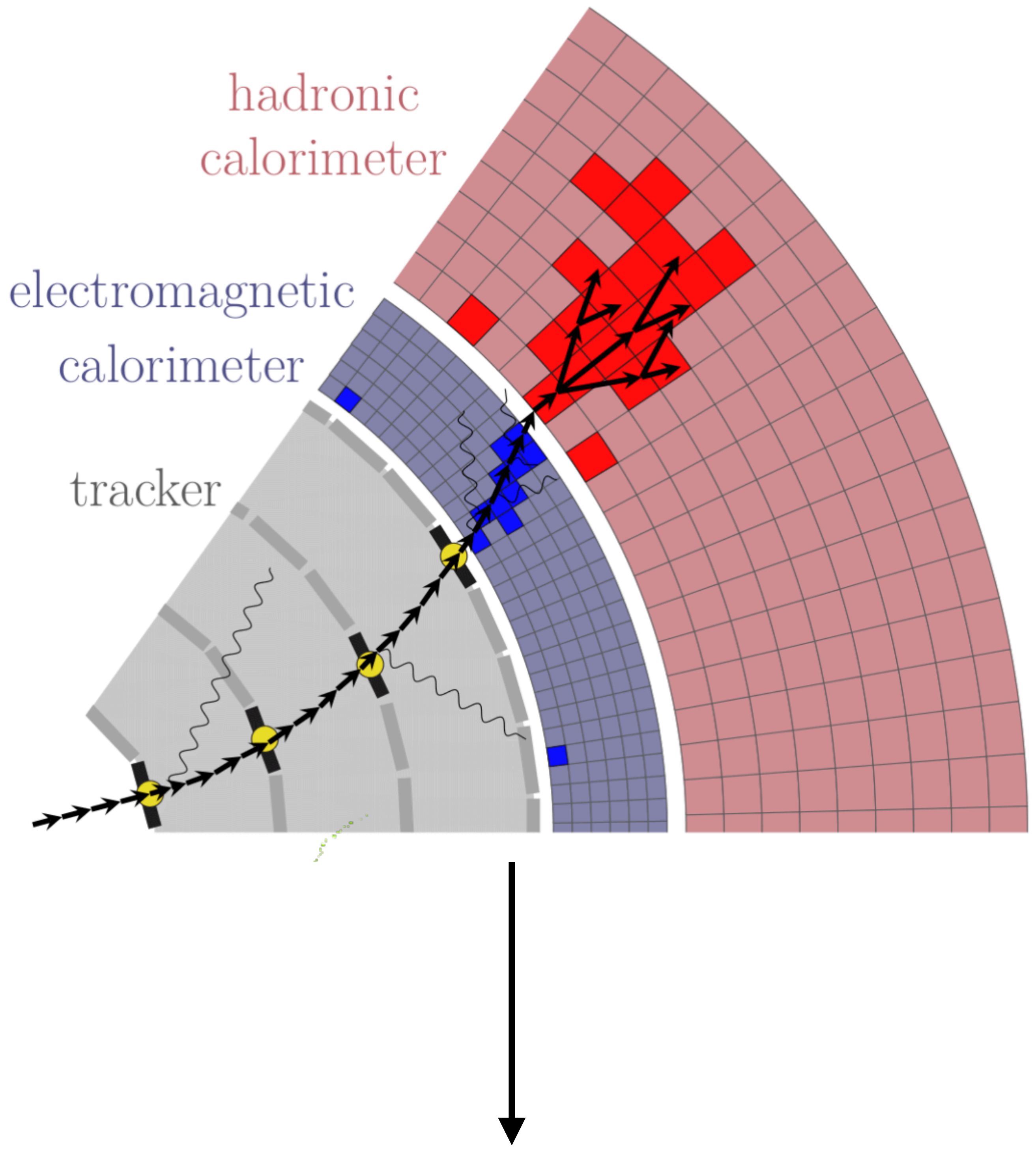
Mostly (SM)background samples, small signal datasets

Unsupervised/SSL

- No labels, completely data driven

We have a lot of high quality simulated data that we want to use

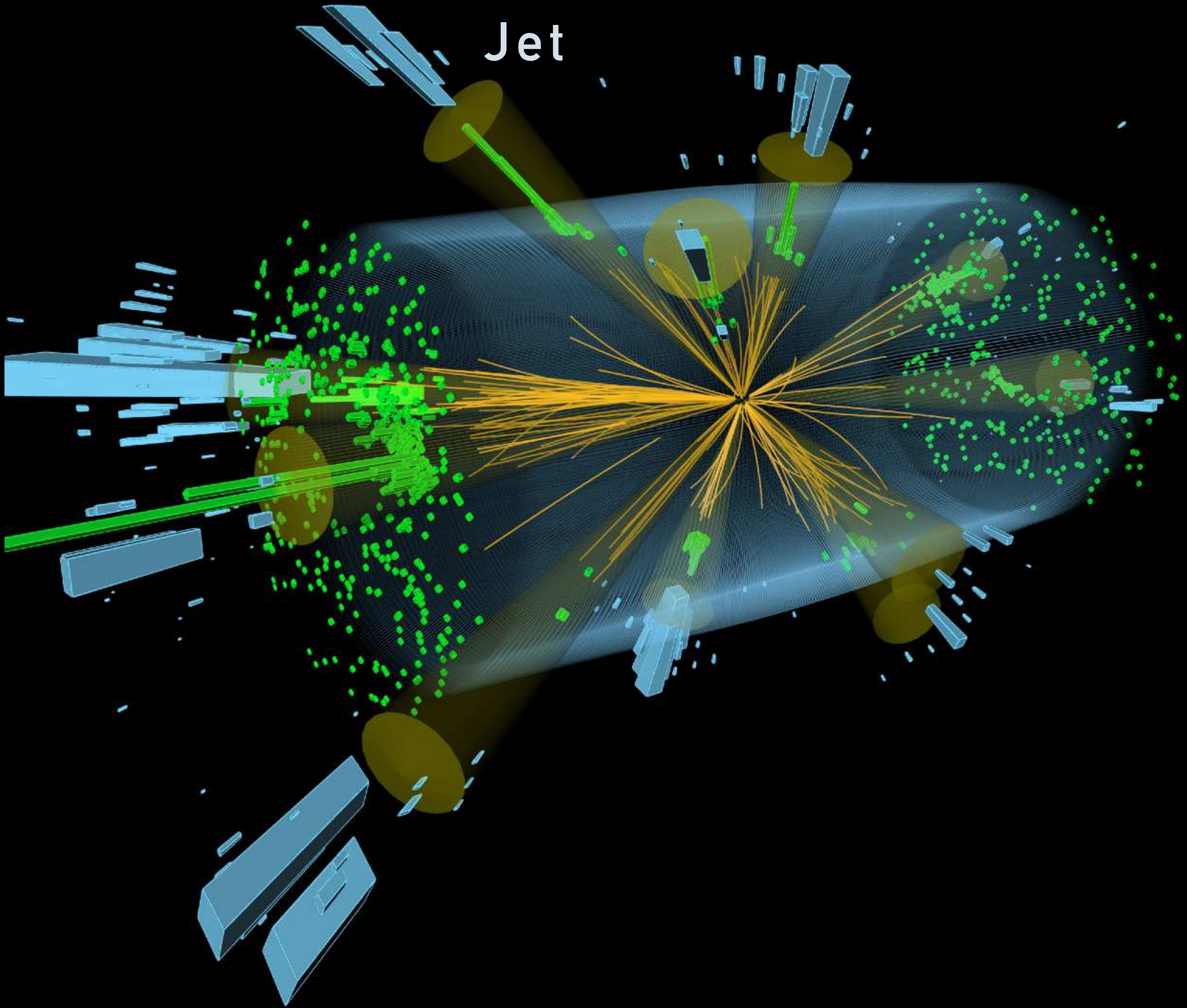
We are also very keen on using this!



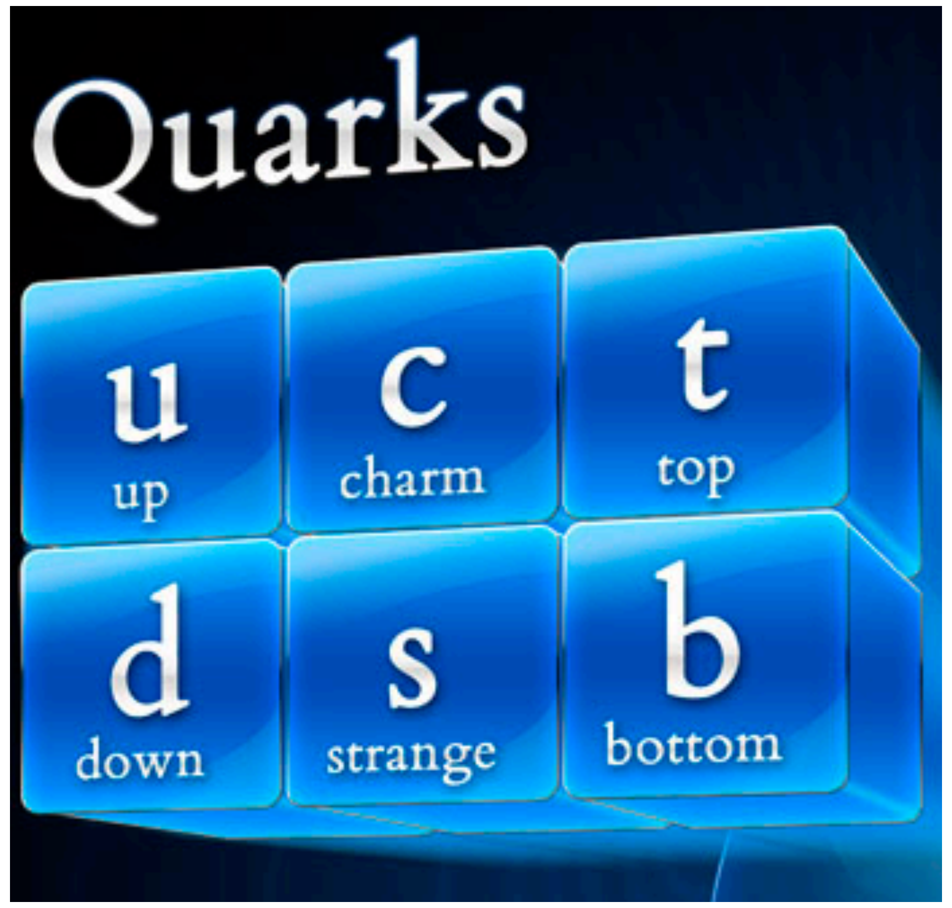
Fully supervised

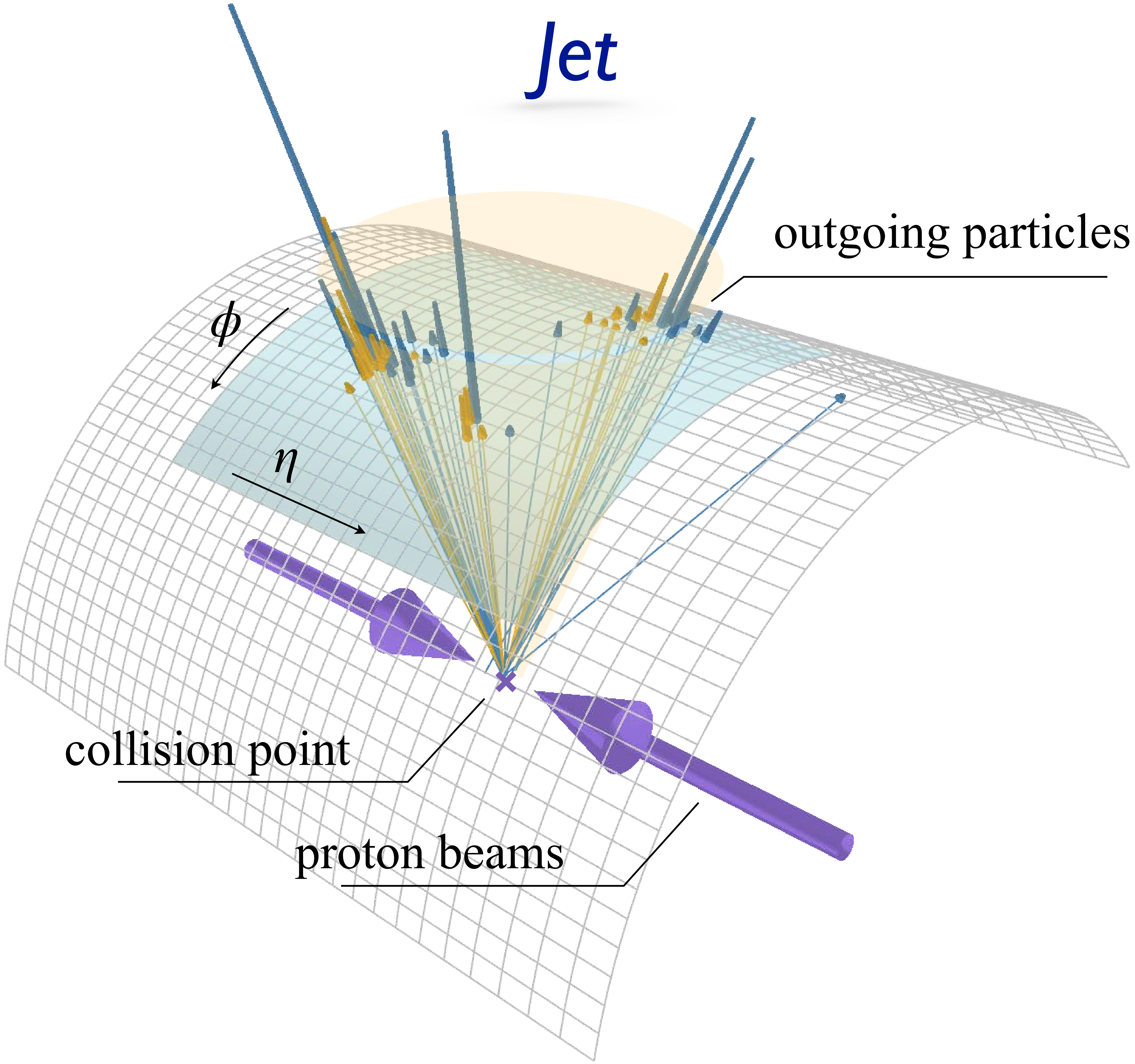
- Requires truth labels
- Only possible using simulation

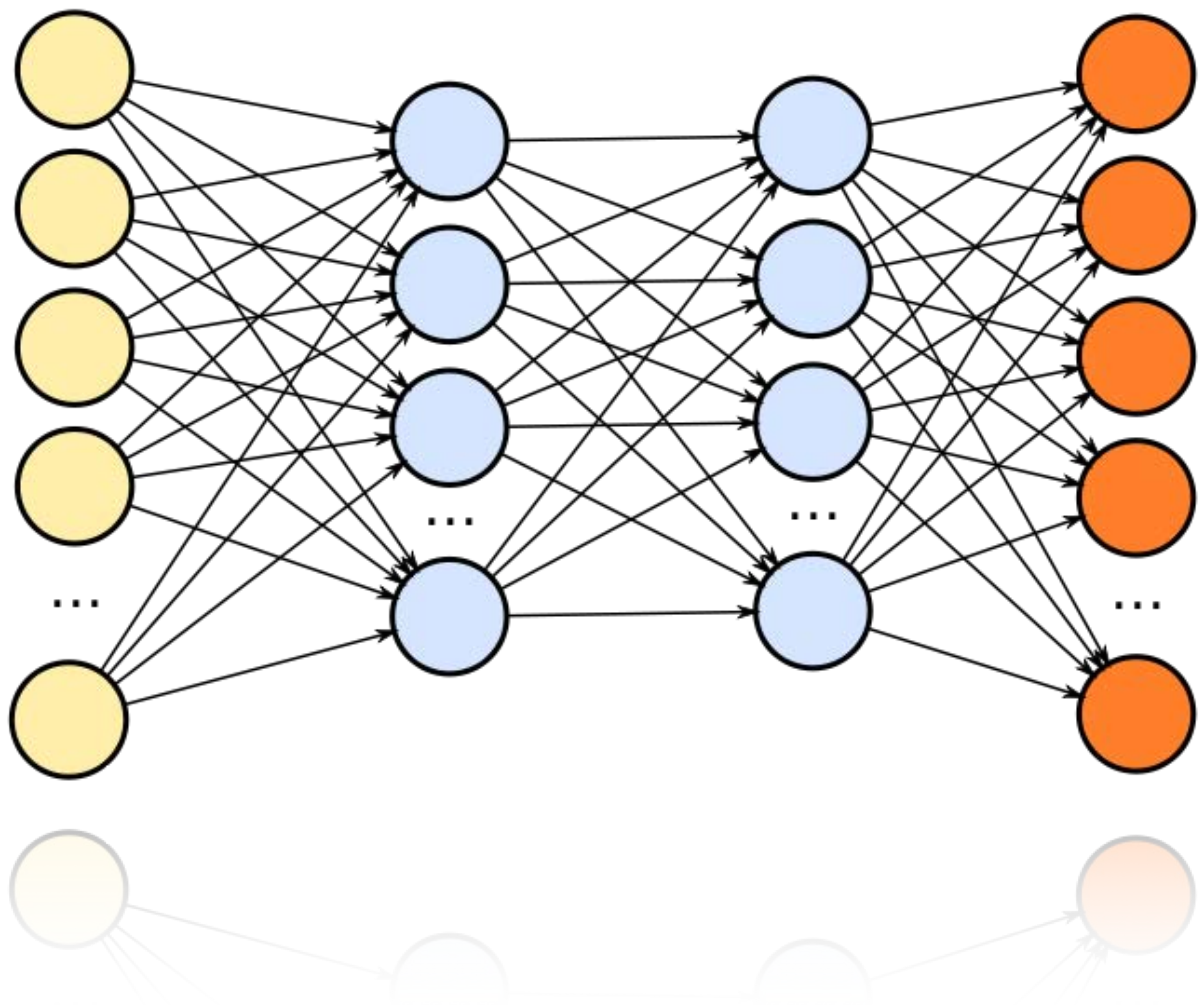
We have a lot of high quality simulated data that we want to use


Jet

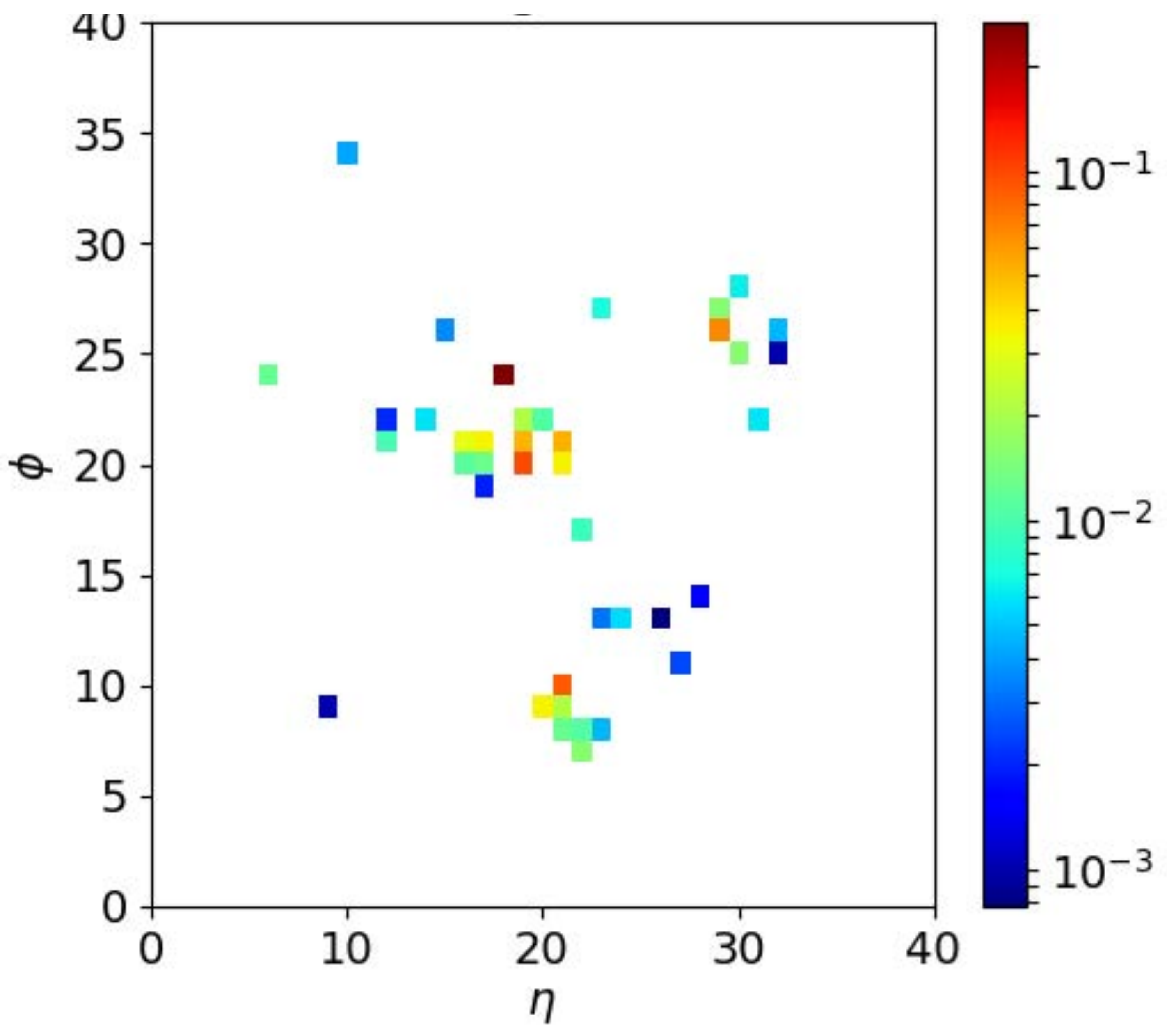
?


Quarks
u
up
c
charm
t
top
d
down
s
strange
b
bottom


Jet
outgoing particles
ϕ
η
collision point
proton beams

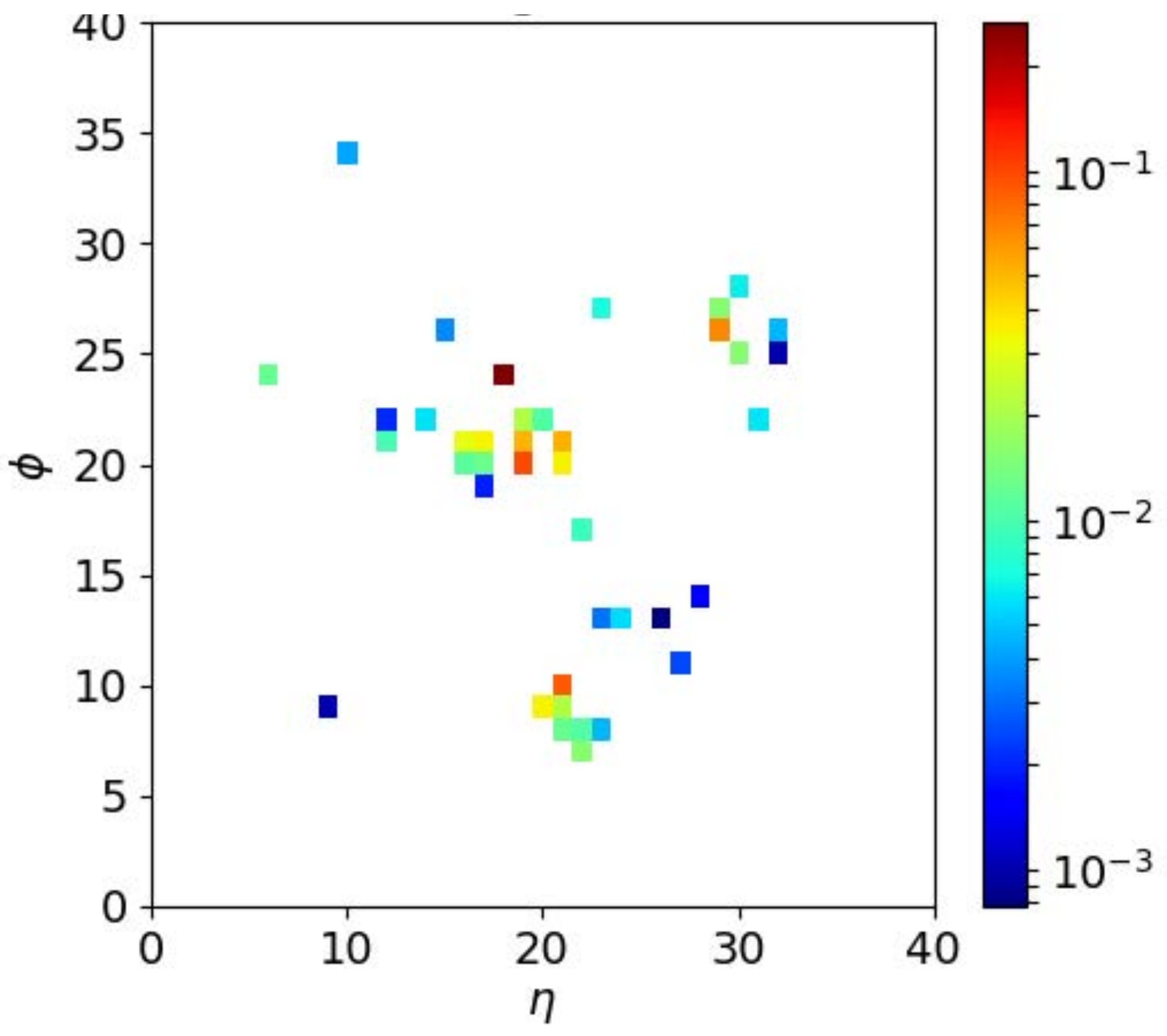


Image



arXiv:1511.05190

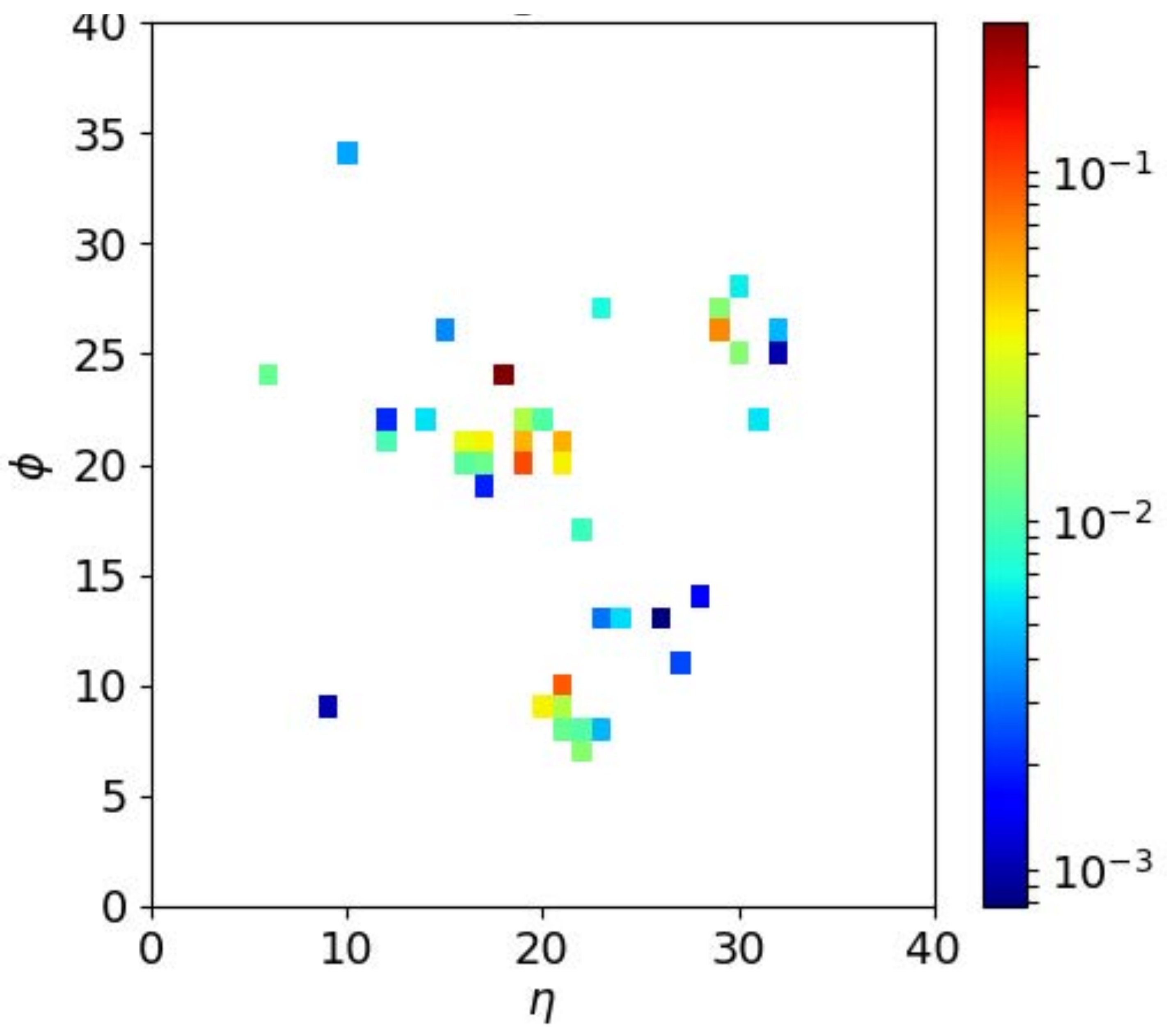
Image



arXiv:1511.05190

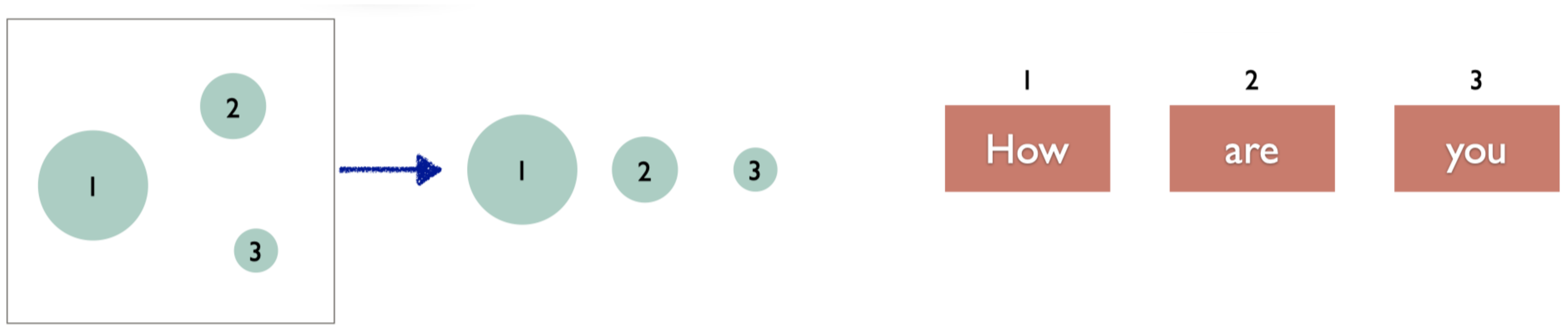
But... inhomogeneous geometry, high sparsity

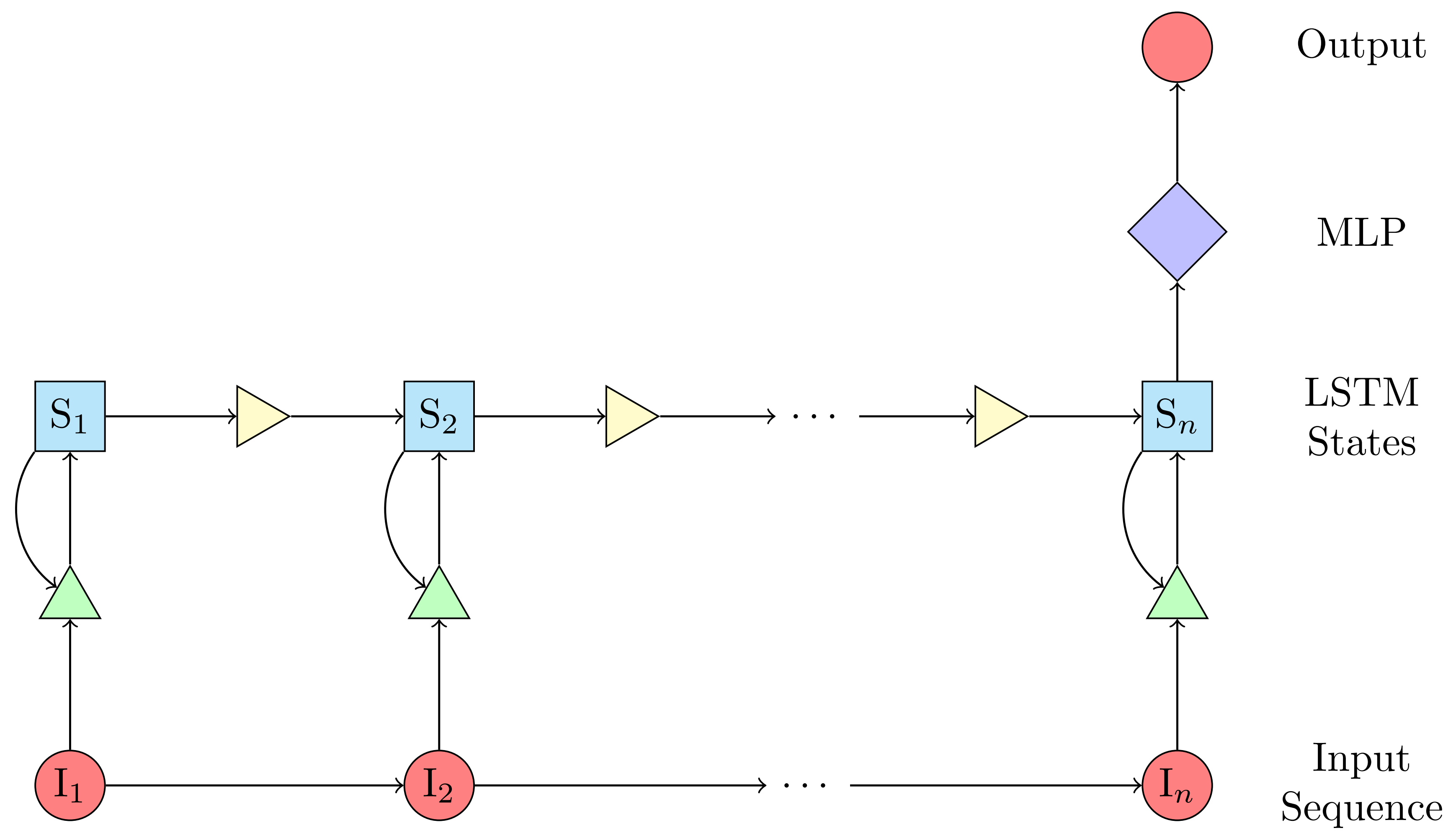
Image



arXiv:1511.05190

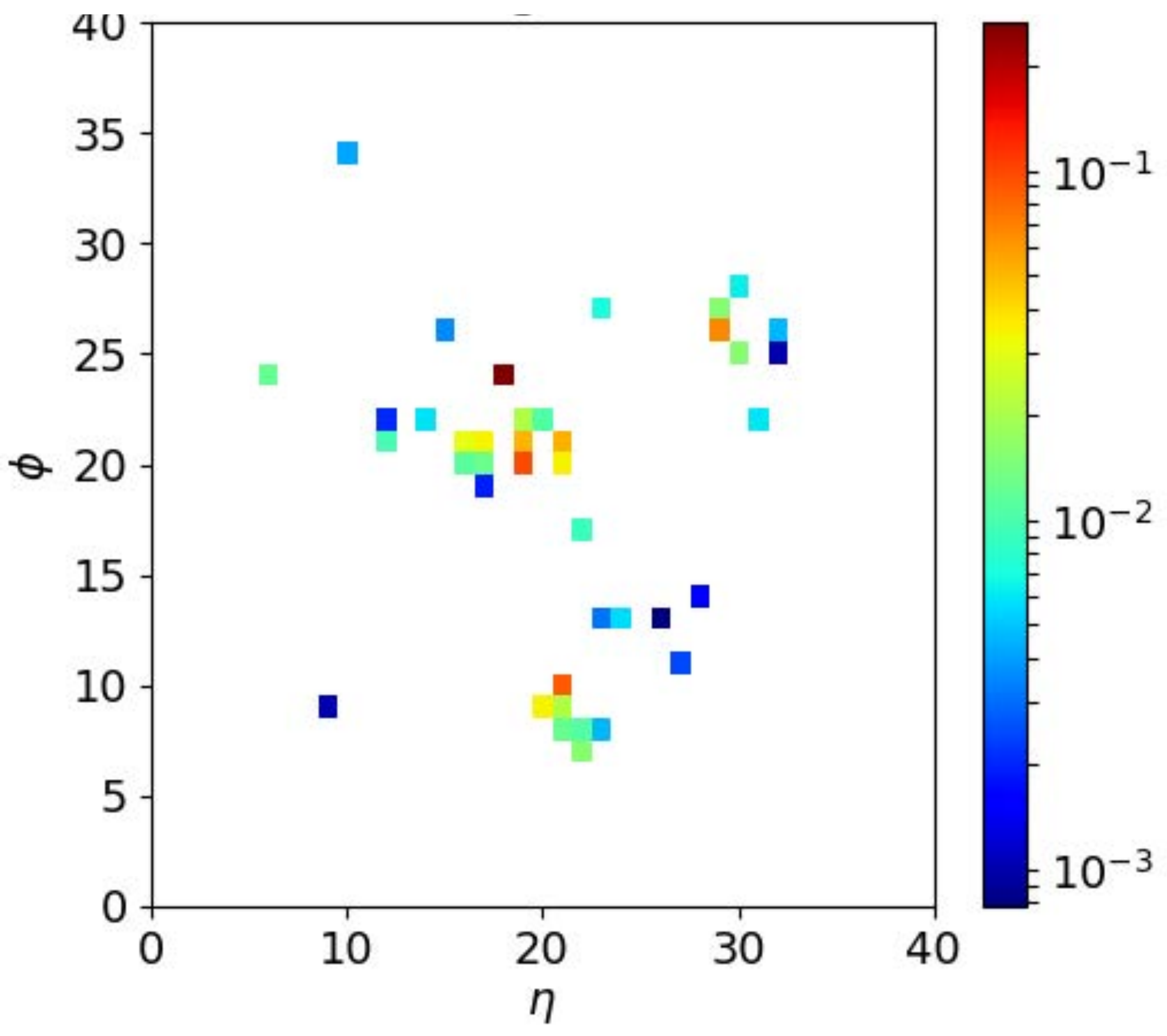
Sequence





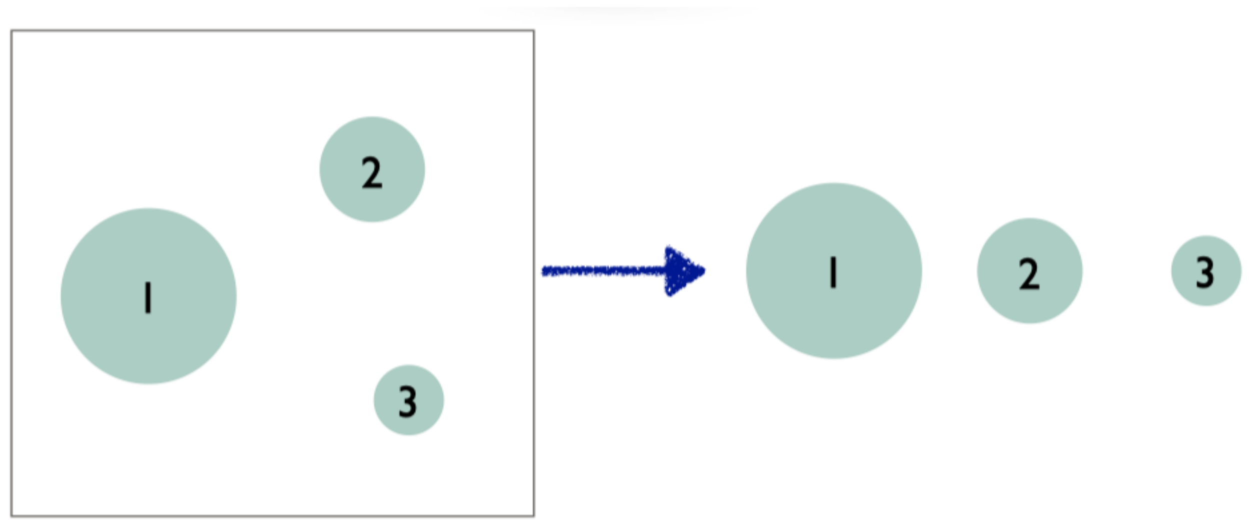
arXiv:1607.08633

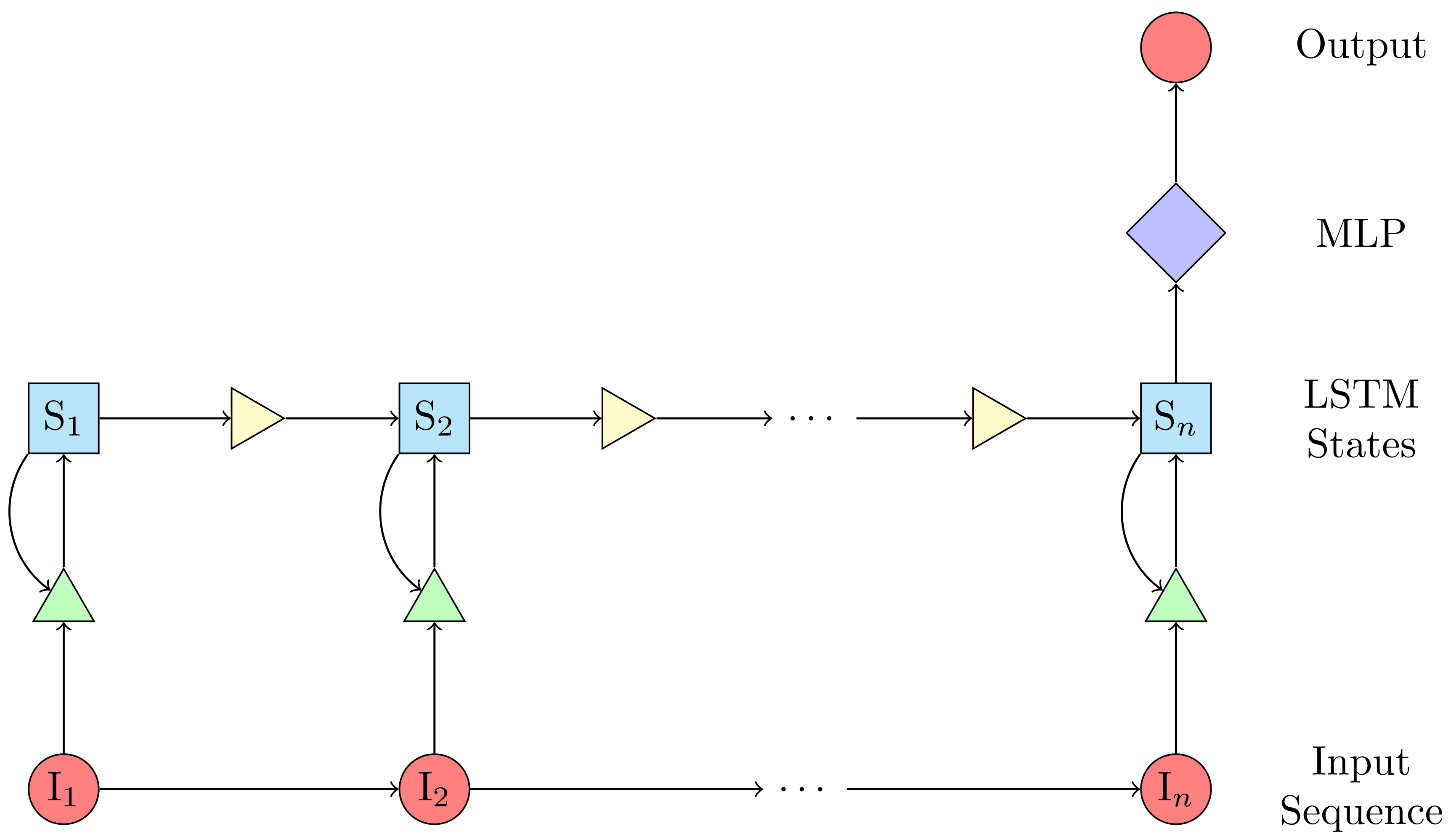
Image



arXiv:1511.05190

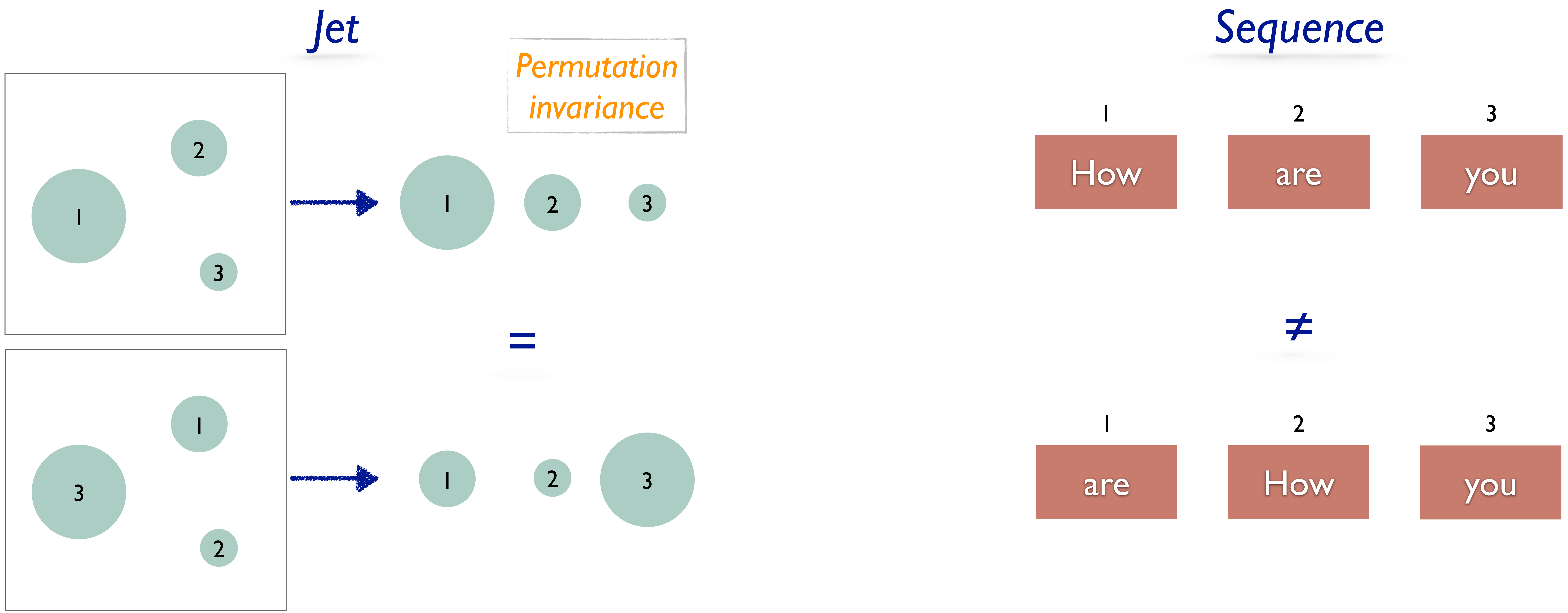
Sequence



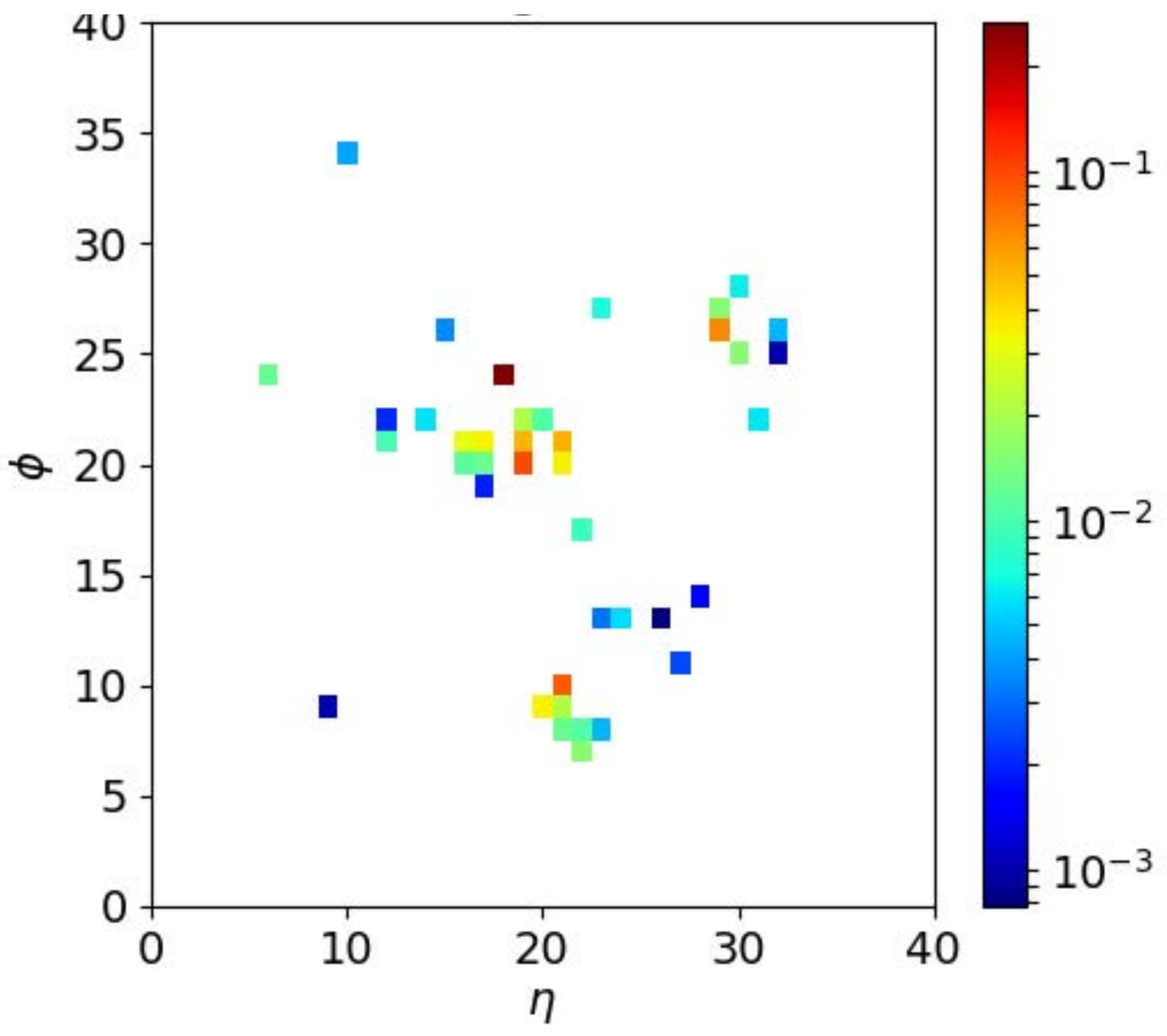


arXiv:1607.08633

But… permutation-invariance

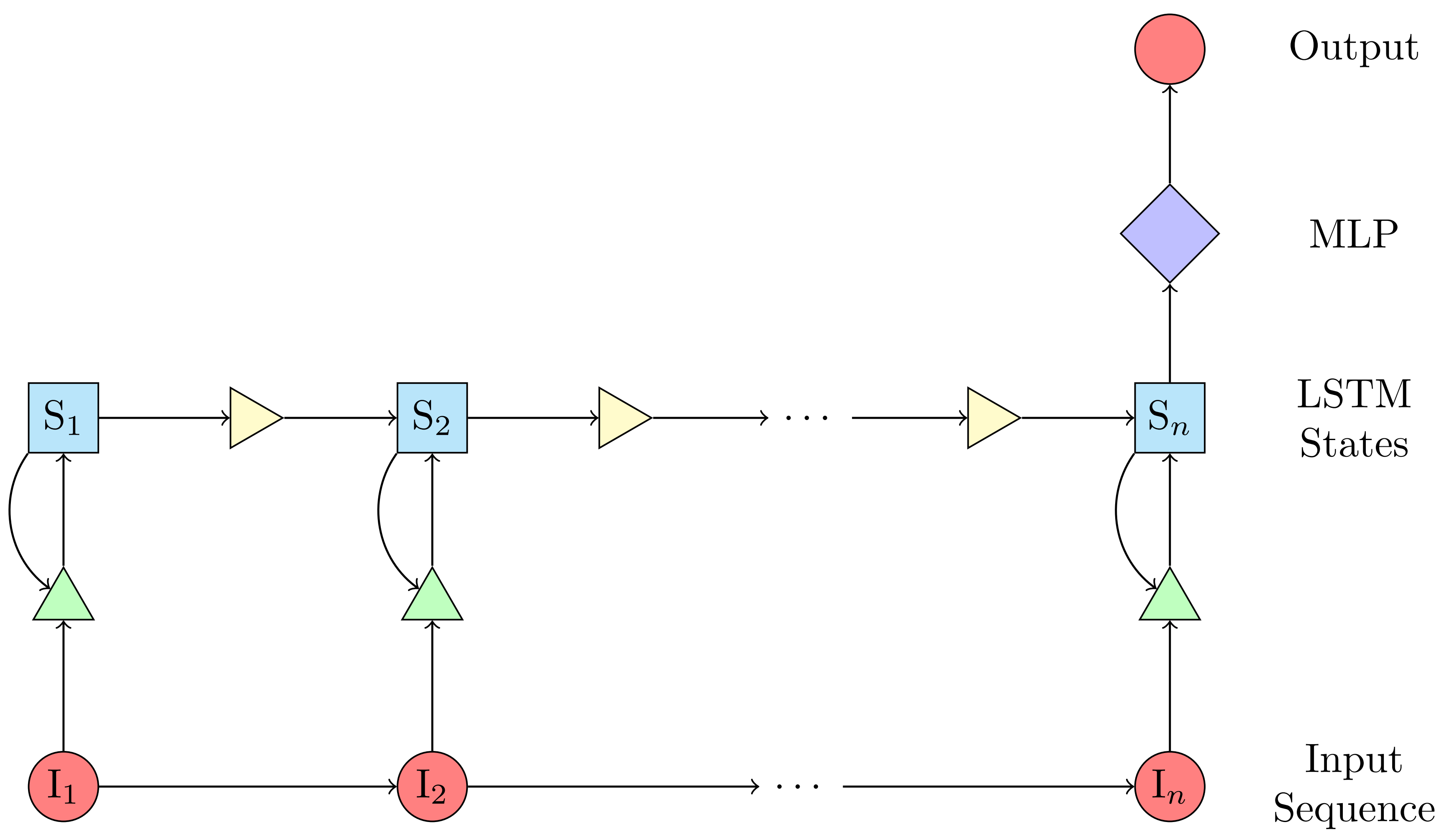

Jet
Permutation invariance
2
1
3
1
2
3
=
1
3
2
1
2
3
Sequence
1
2
3
How
are
you
≠
1
2
3
are
How
you

Image



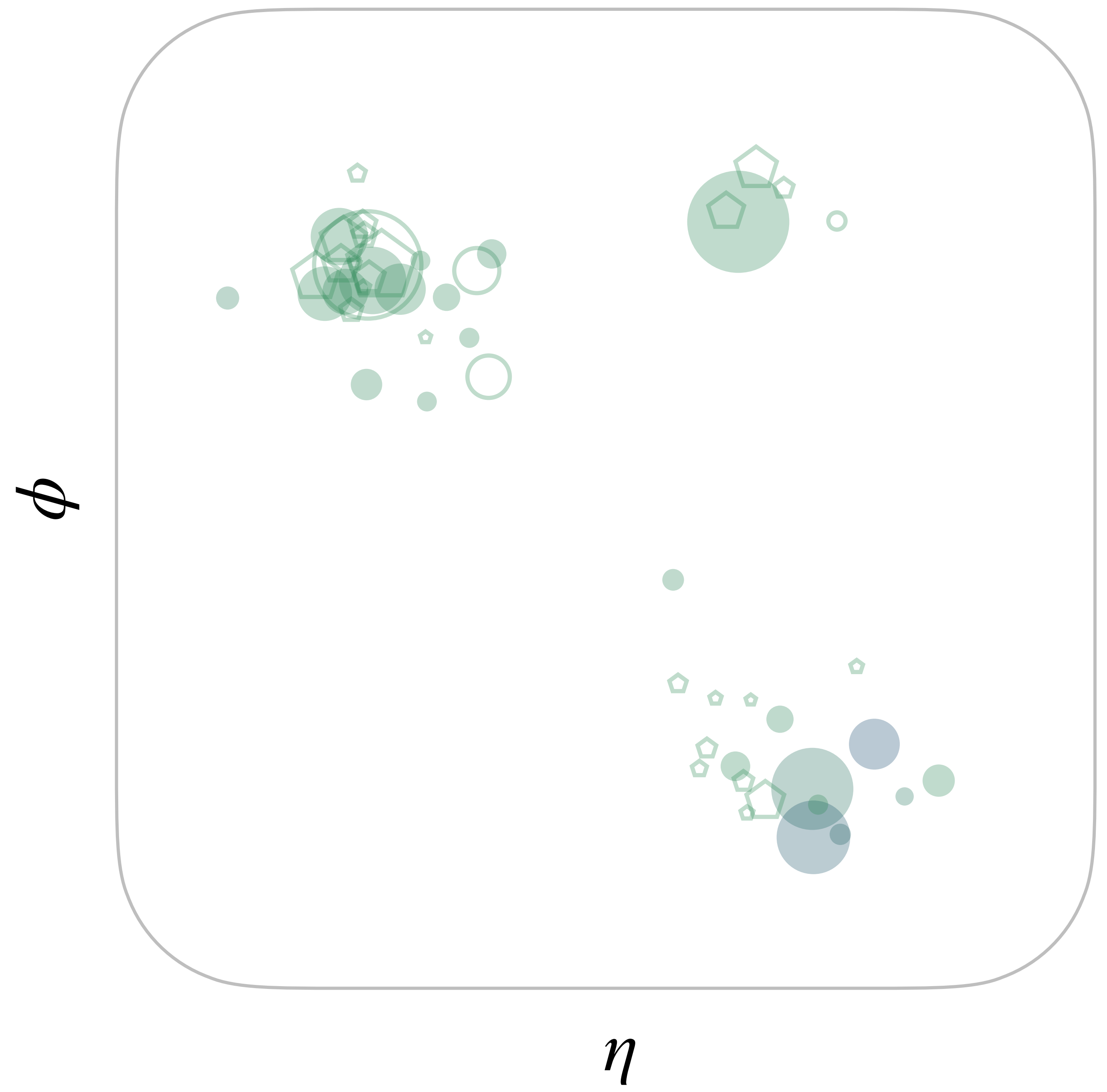
arXiv:1511.05190

Sequence

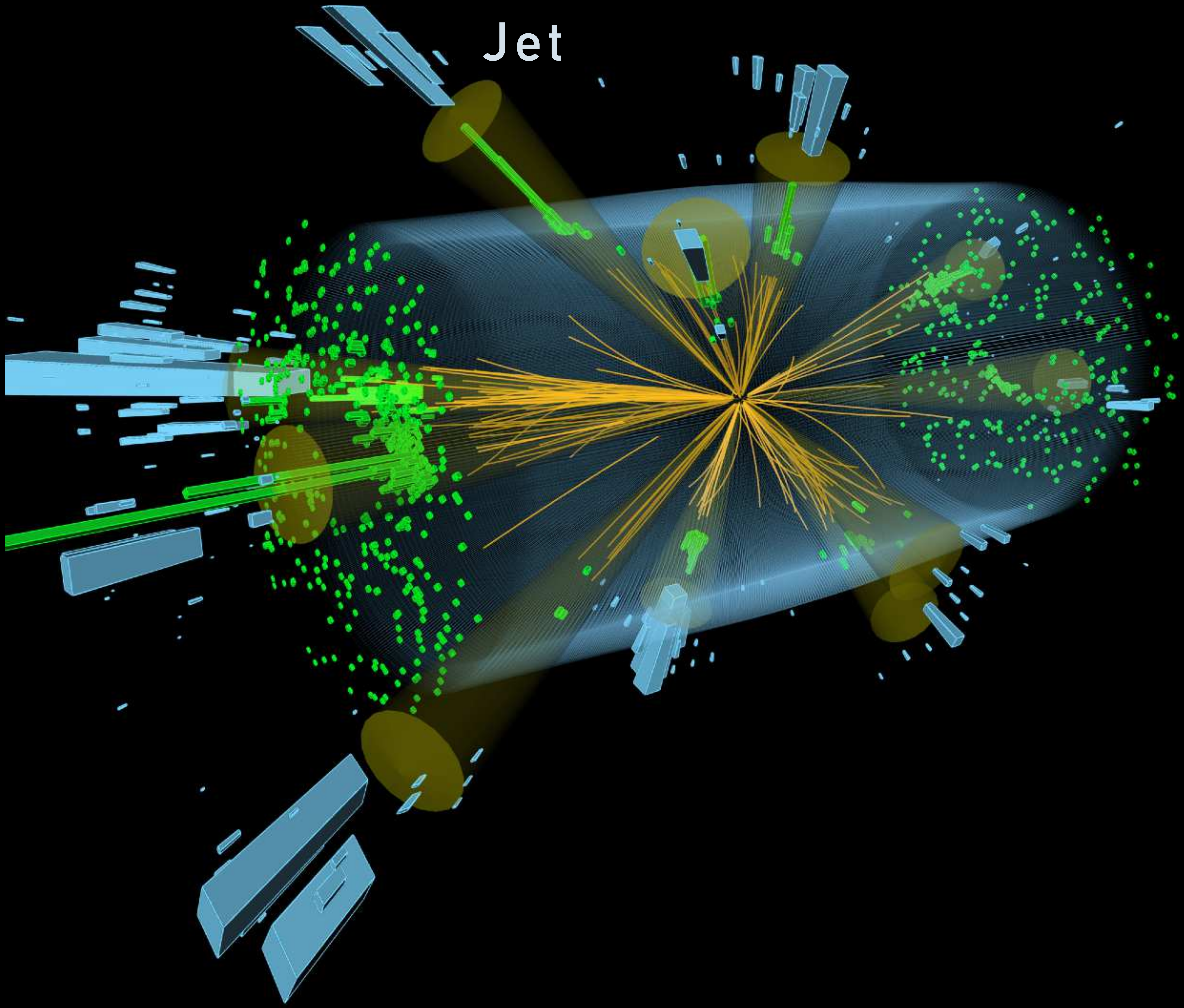


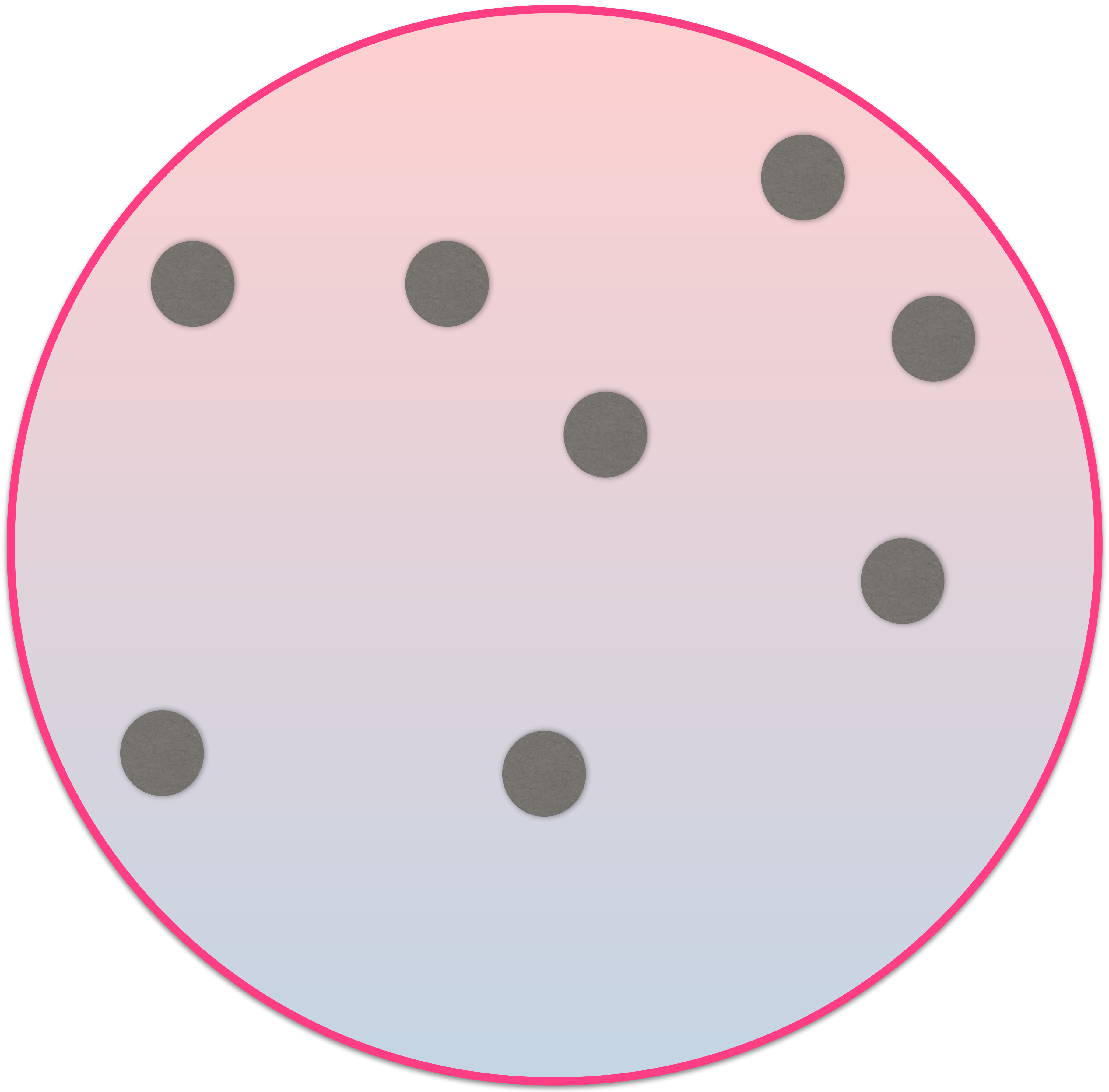
arXiv:1607.08633

Point Cloud



PRD:101.056019


Jet

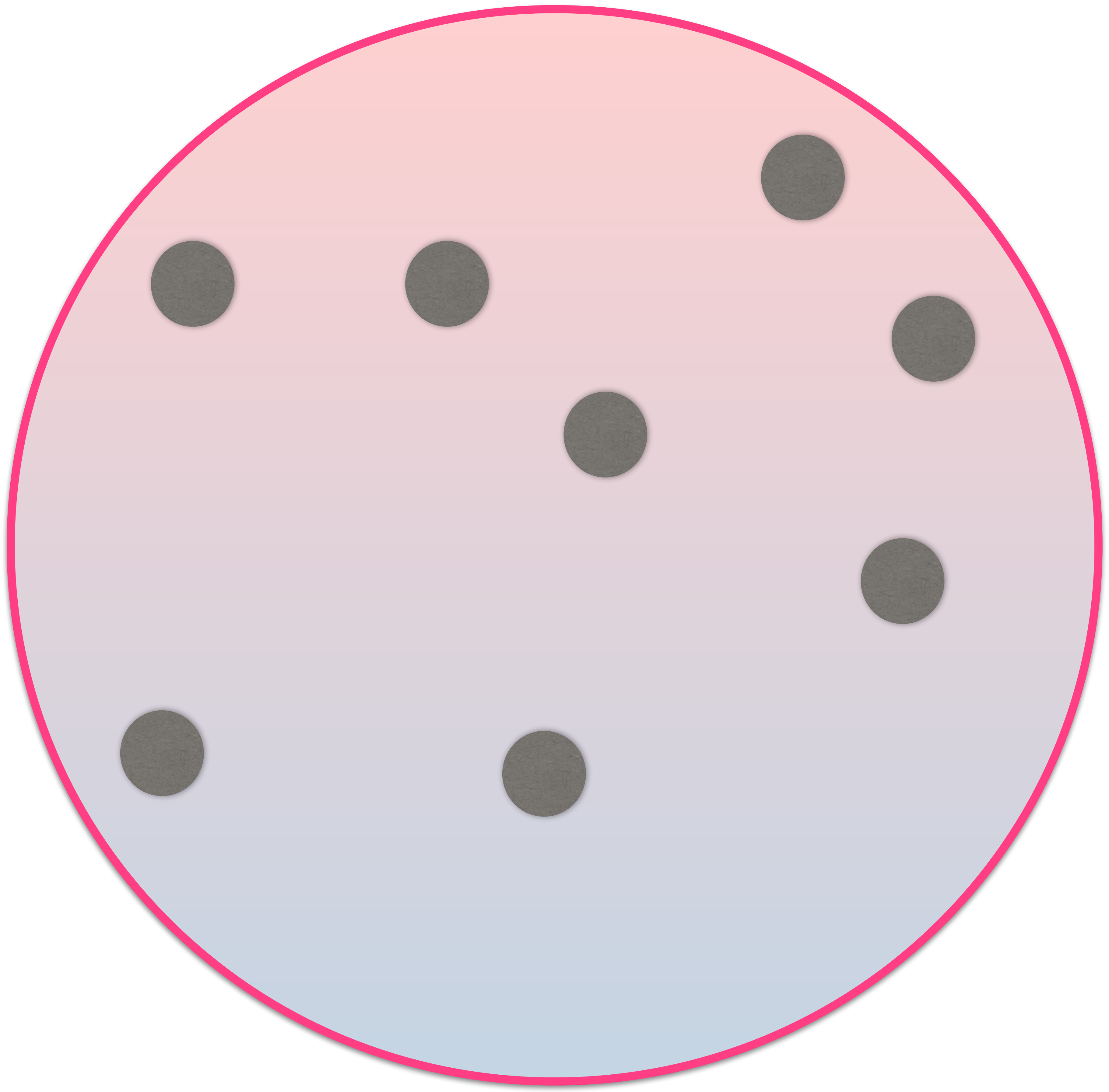


Graph Neural Networks

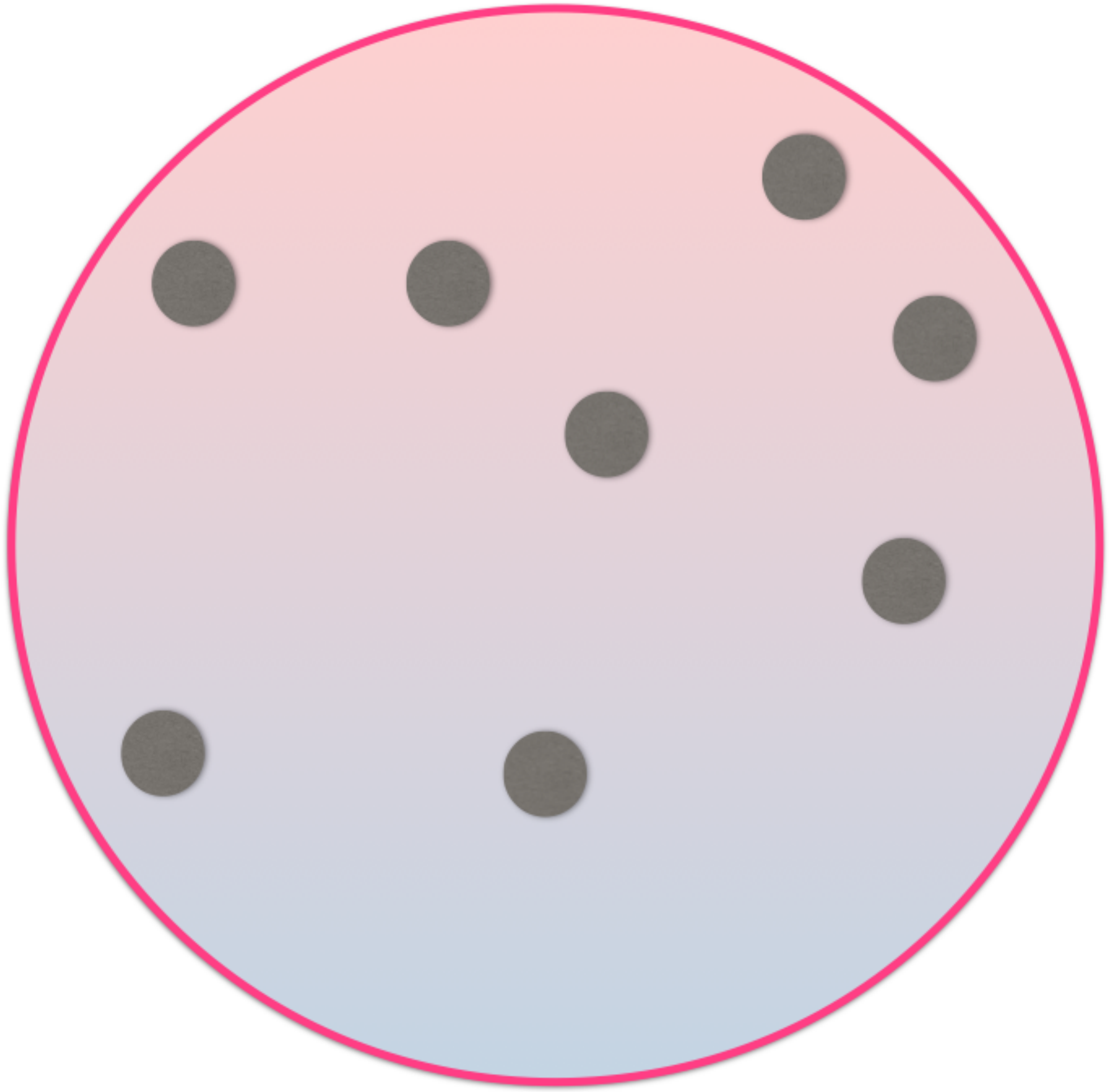
Properties of physics data:

- Measurements distributed in space (and time) **irregularly**
- **Sparse** (most detector channels are empty), but pockets of density
- Complex **interdependencies** between measurements
- Physics "objects" composed of **multiple measurements**
- Inherent **symmetries** (Lorentz boosts, rotational)

Graph (or point cloud) embedding of the data can handle these properties!



Graph Neural Networks



Graph (global) features u: jet mass

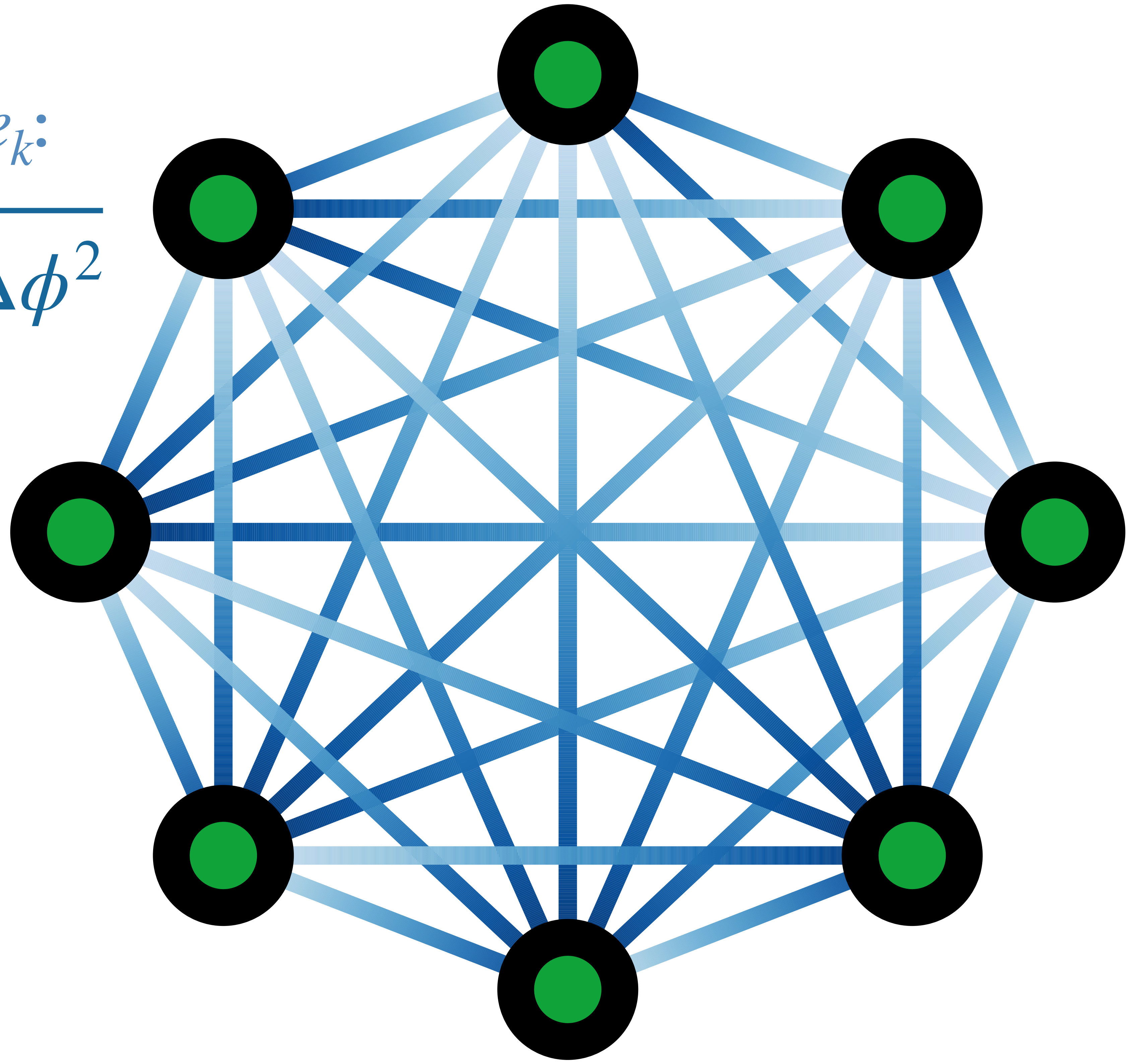
$$m = \sqrt{\sum_{i \in \text{jet}} E_i^2 - p_{x,i}^2 - p_{y,i}^2 - p_{z,i}^2}$$

Node features v_i:

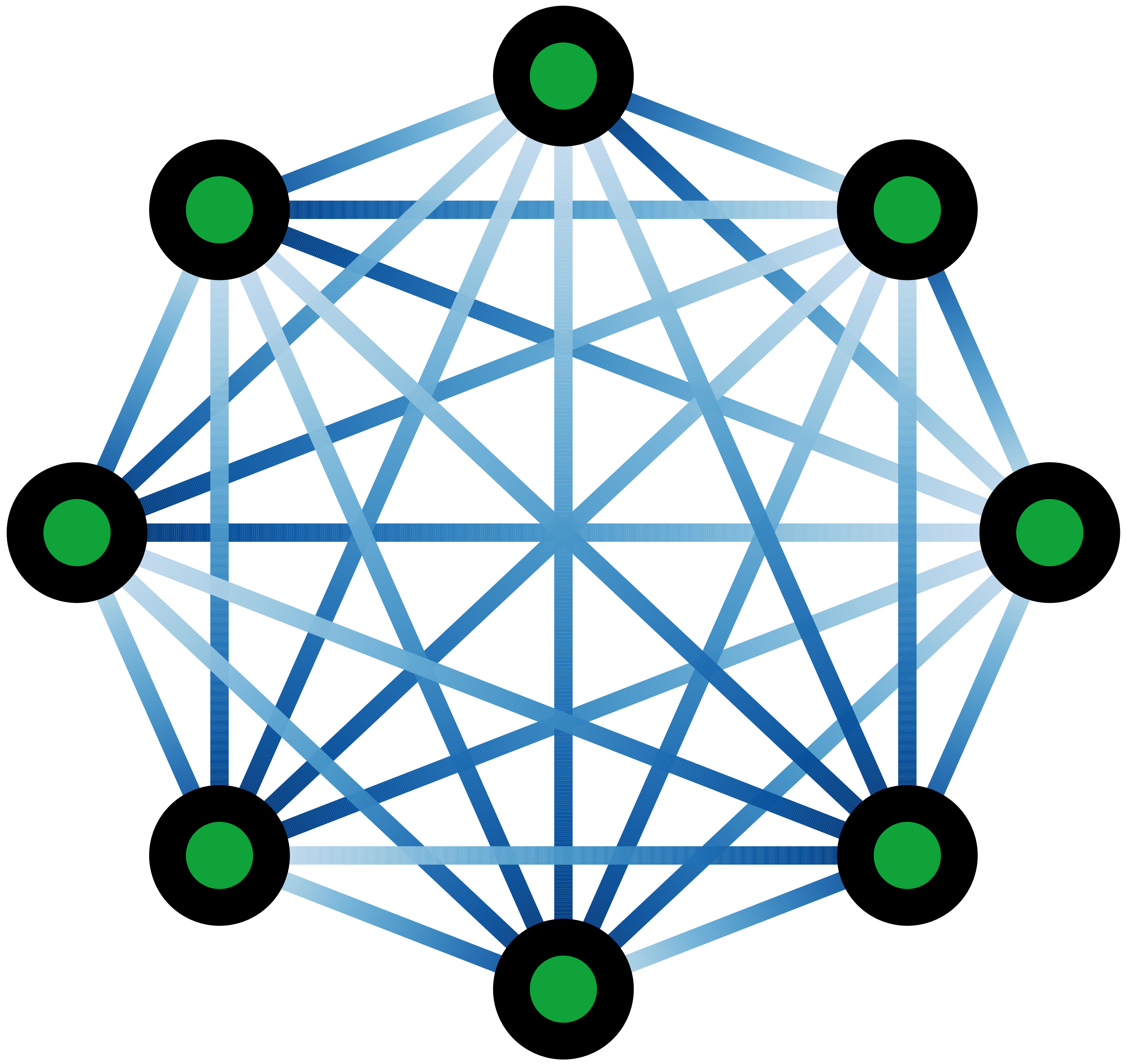
$$p = [E, p_x, p_y, p_z] \equiv [p_\mathrm{T}, \eta, \phi, m]$$

Edge features e_k:

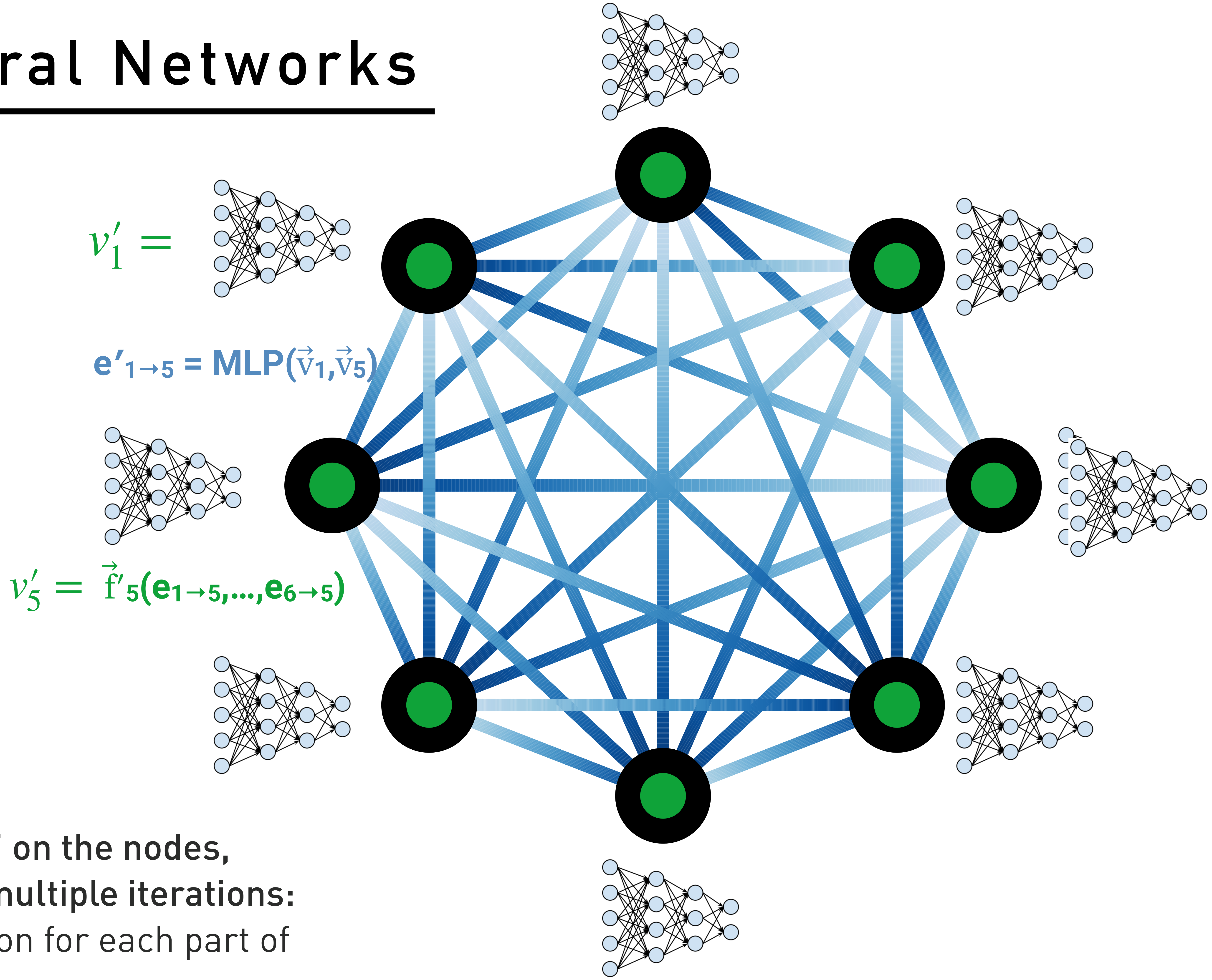
$$\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2}$$



Graph Neural Networks



Graph Neural Networks

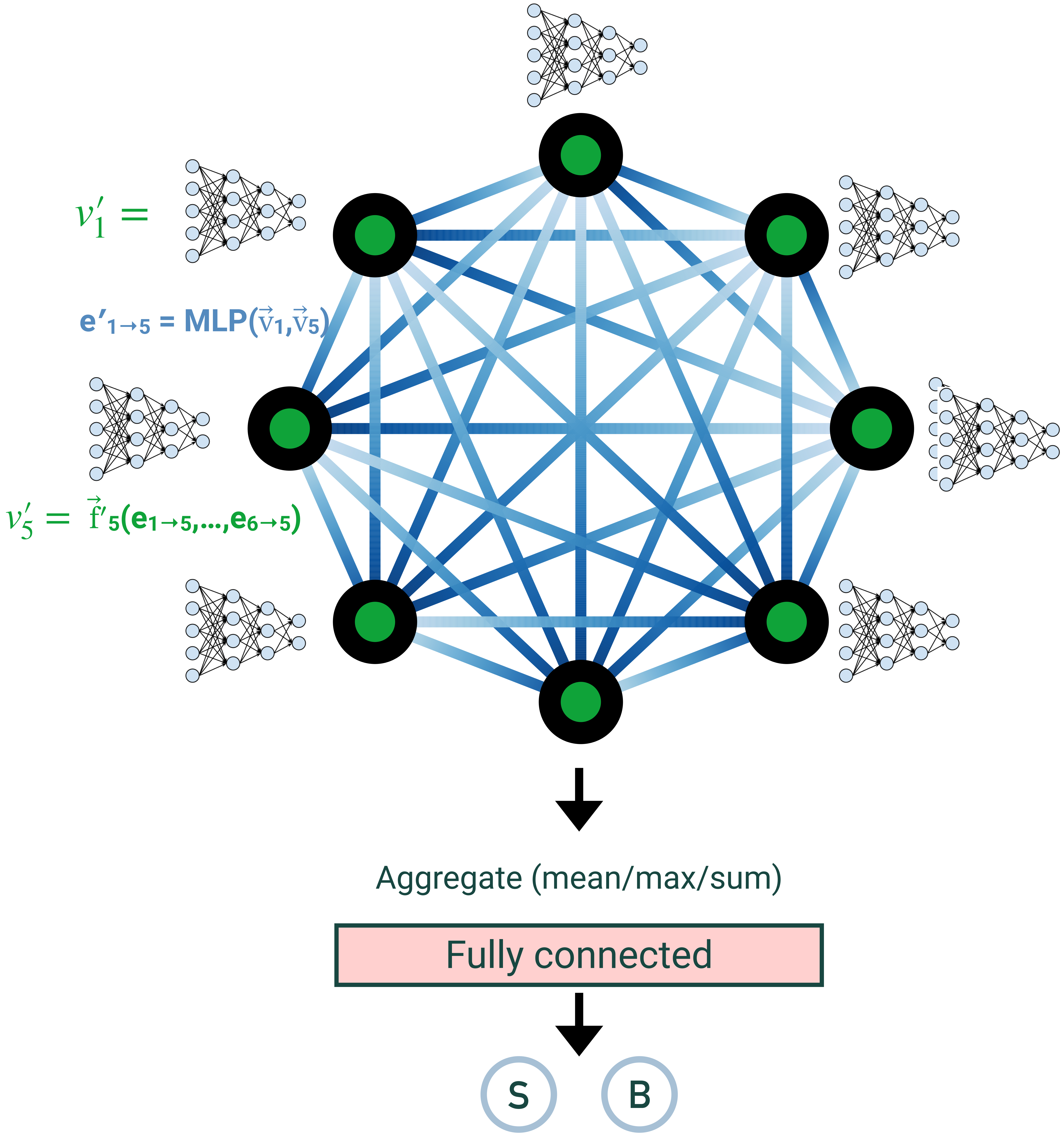


Want to create "new features" on the nodes, edges, or the full graph with multiple iterations:

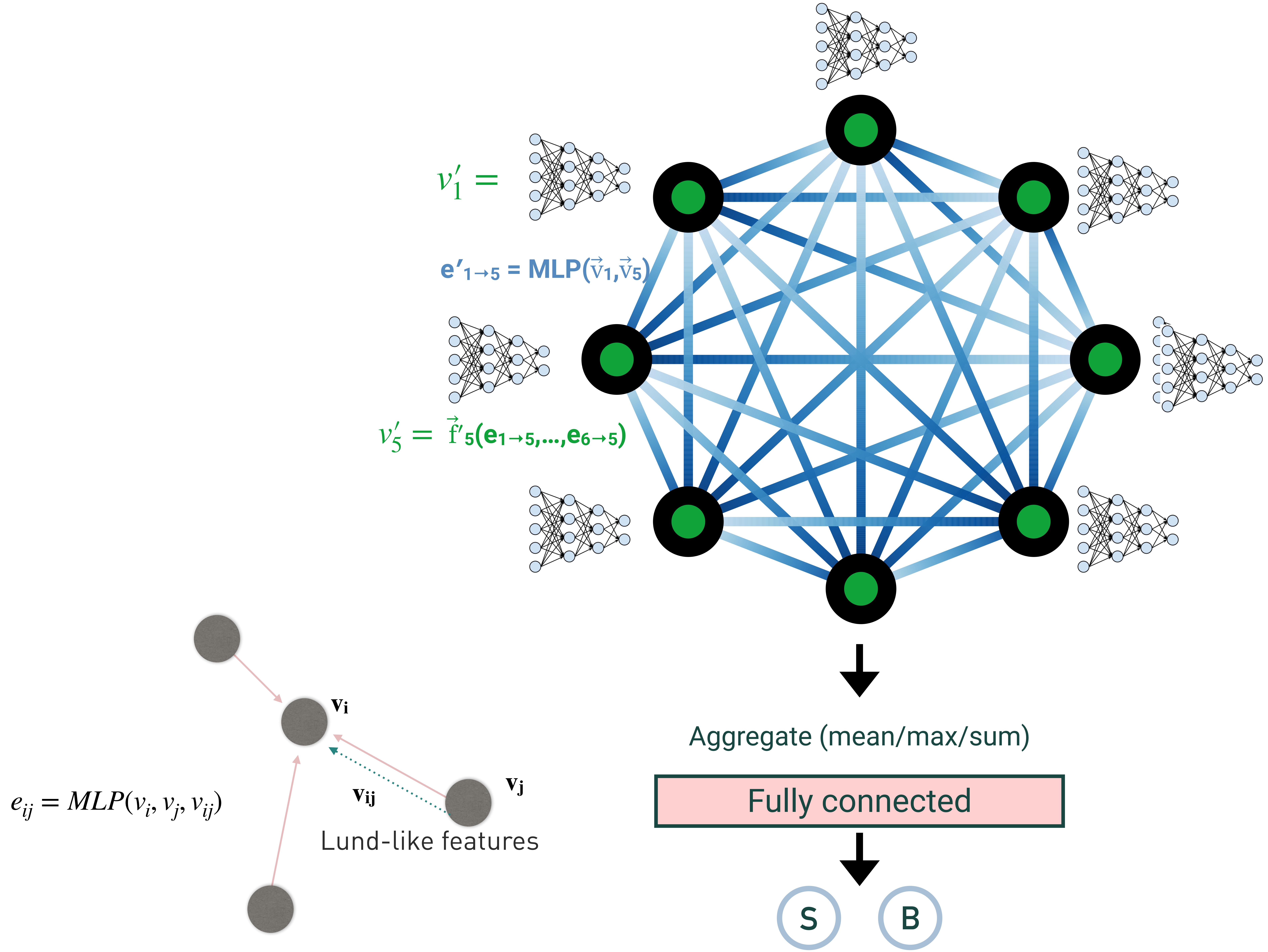
- Create a new representation for each part of the graph
- These"updates" are usually DNNs!

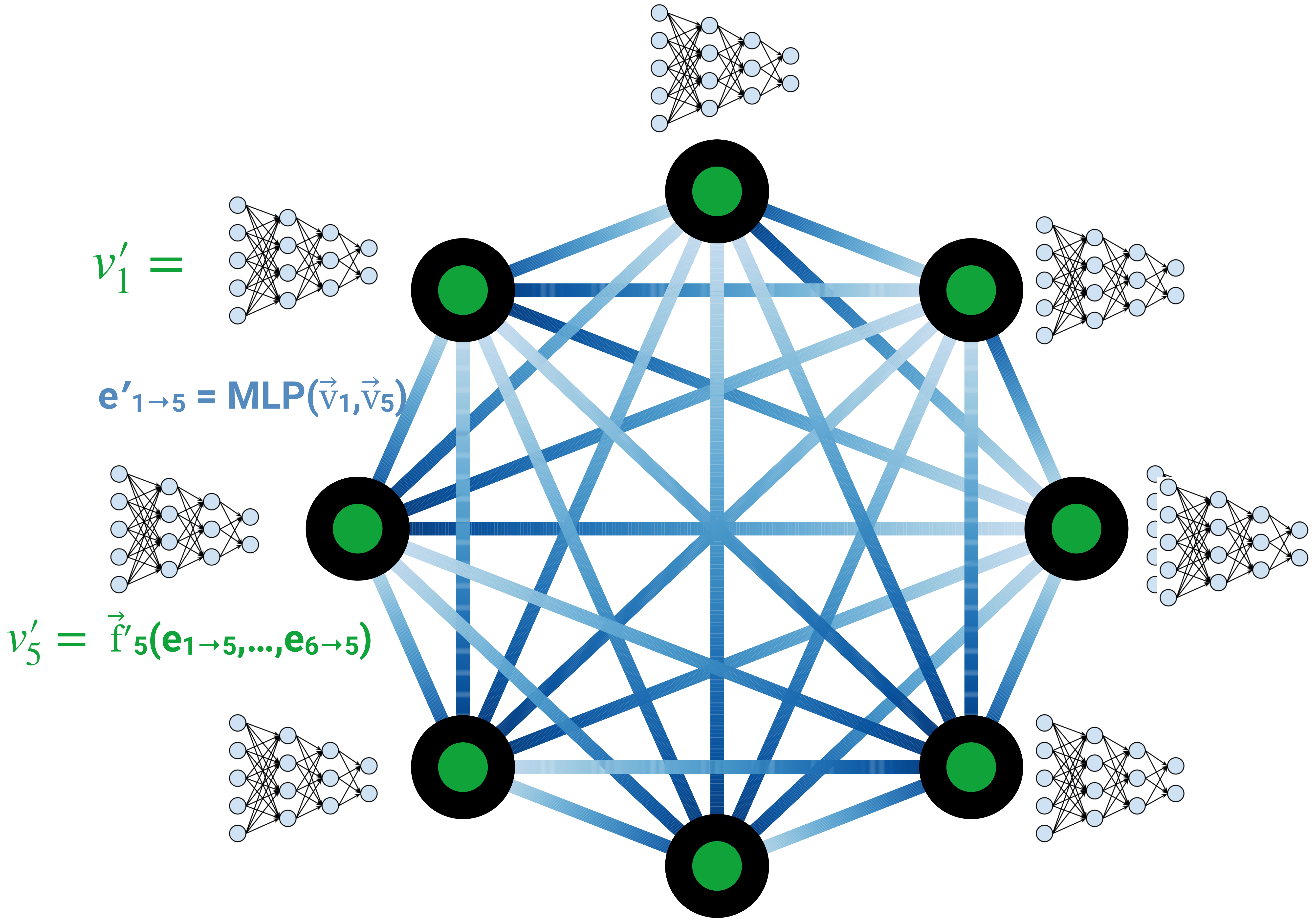
$$\begin{aligned}\mathbf{e}'_k &= \boxed{\phi^e\left(\mathbf{e}_k, \mathbf{v}_{r_k}, \mathbf{v}_{s_k}, \mathbf{u}\right)} \\ \mathbf{v}'_i &= \boxed{\phi^v\left(\bar{\mathbf{e}}'_i, \mathbf{v}_i, \mathbf{u}\right)} \\ \mathbf{u}' &= \boxed{\phi^u\left(\bar{\mathbf{e}}', \bar{\mathbf{v}}', \mathbf{u}\right)}\end{aligned}$$

DNNs to be trained!



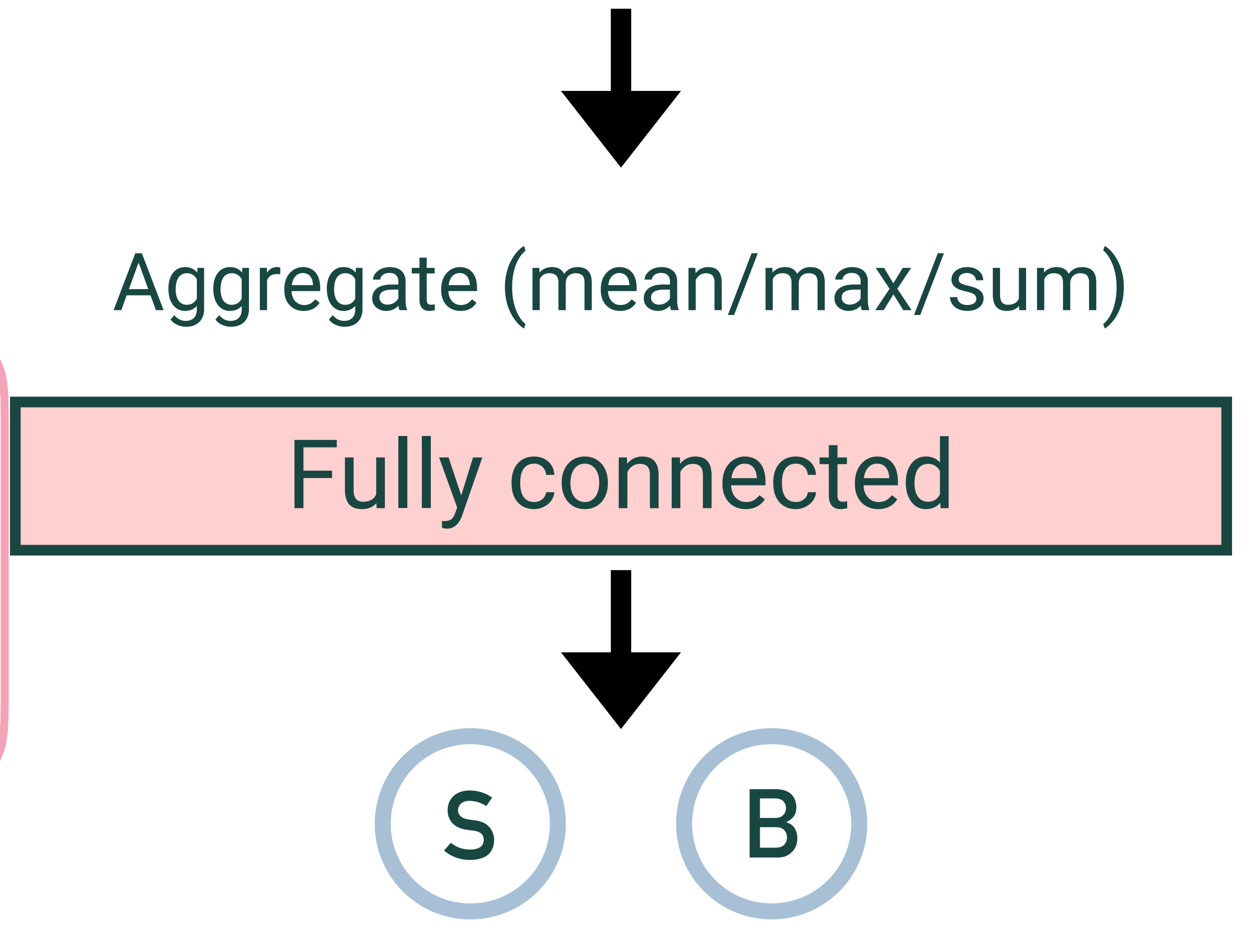
$v'_1 =$
$e'_{1\to5} = \mathrm{MLP}(\vec{v}_1,\vec{v}_5)$
$v'_5 = \vec{f}'_5(e_{1\to5},\ldots,e_{6\to5})$
Aggregate (mean/max/sum)
Fully connected
S
B


$v'_1 =$
$e'_{1\to5} = MLP(\vec{v}_1, \vec{v}_5)$
$v'_5 = \vec{f}'_5(e_{1\to5}, \dots, e_{6\to5})$
Aggregate (mean/max/sum)
Fully connected
S
B
v_i
v_j
v_{ij}
$e_{ij} = MLP(v_i, v_j, v_{ij})$
Lund-like features



SOTA: GNNs acting on point cloud data

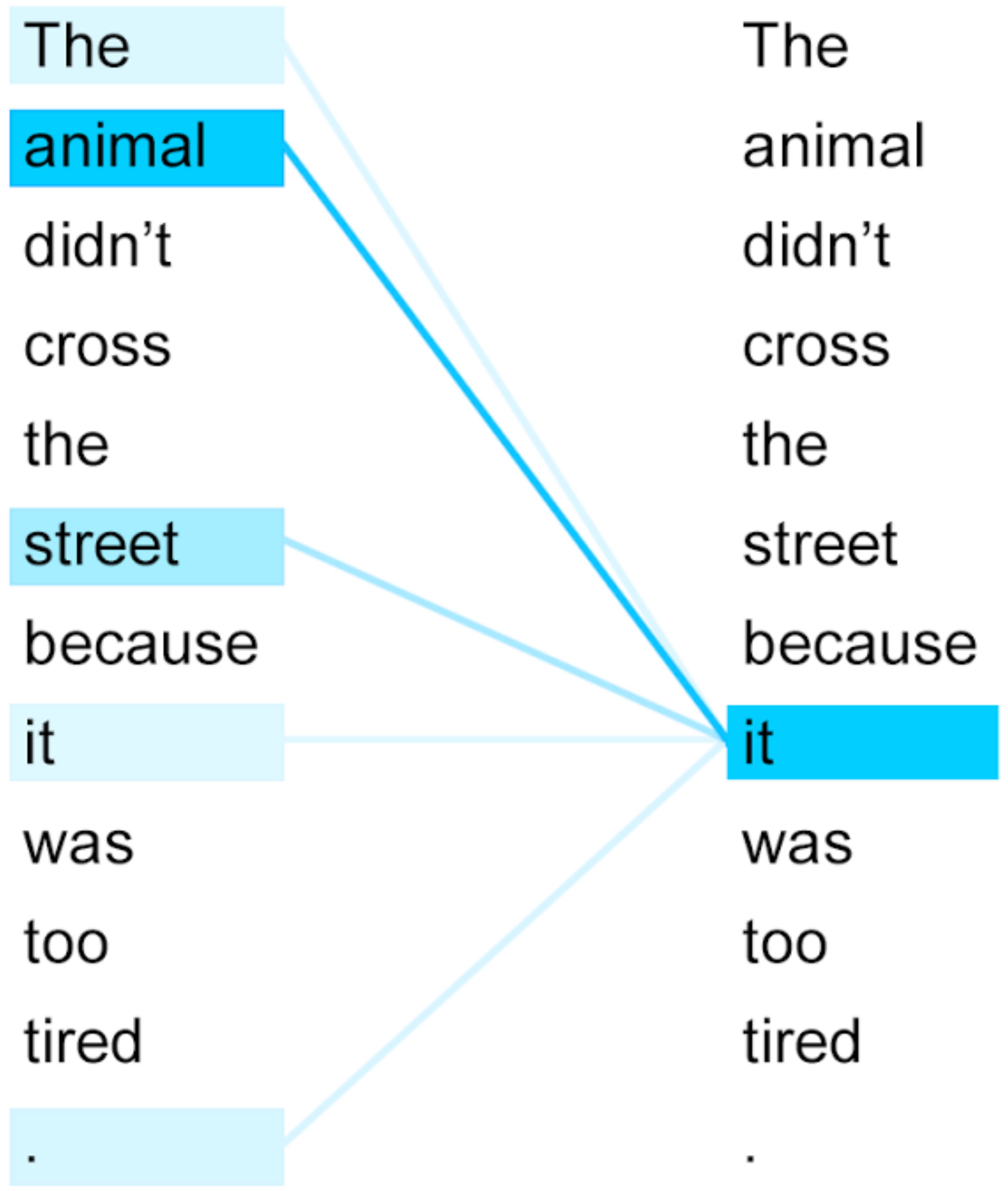
- ParticleNet (GNN on point cloud)
  LundNet (GNN,Lund plane)
  ABCNet (GNN, attention)
  Point Cloud Transformers (transformer, attention)
  ParticleNeXt (GNN, attention, Lund)
  ParT (transformer, attention)

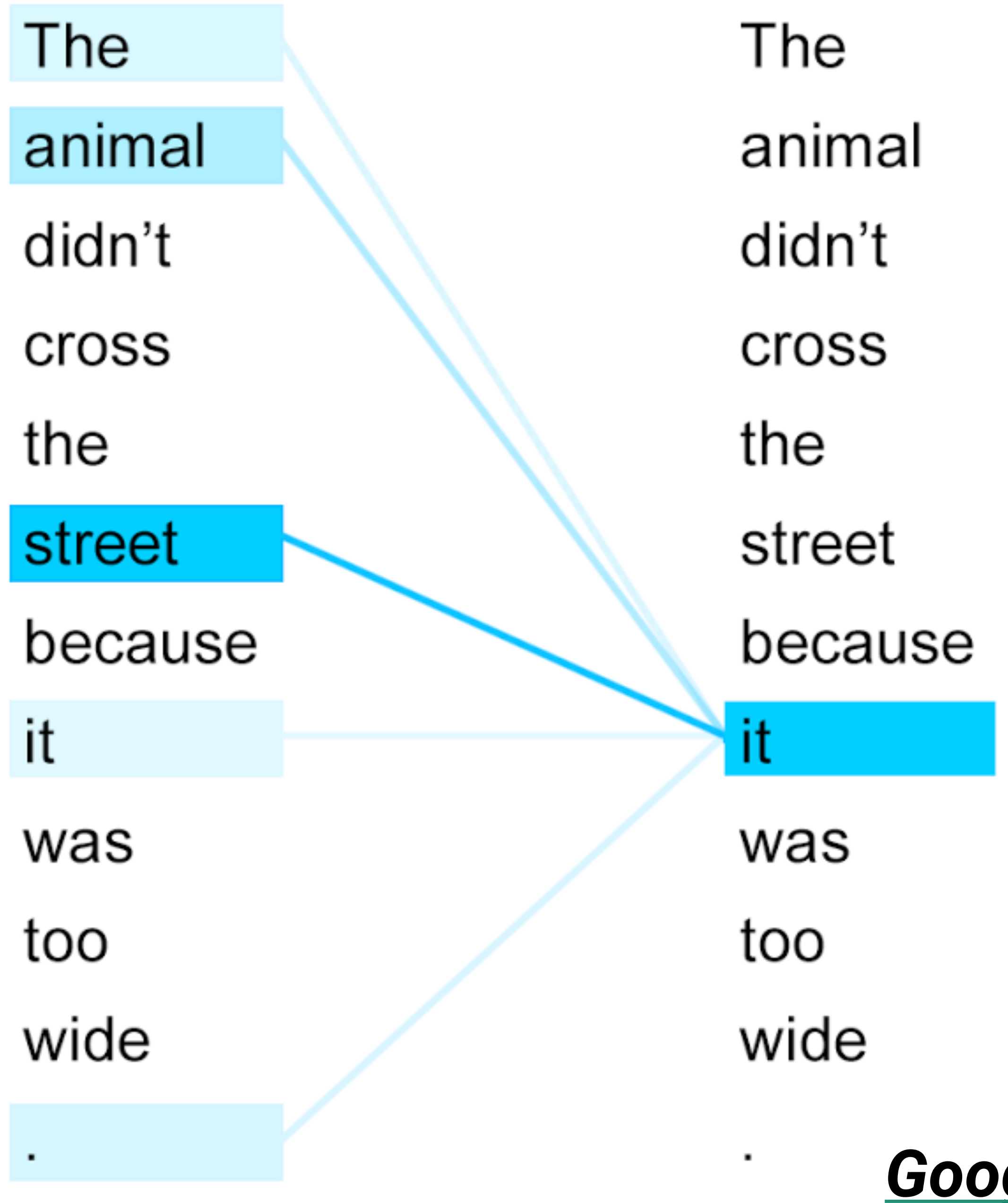


Transformers and (self-)attention

(Self-)Attention

- Allows inputs to interact with each other ("self") and find out who they should pay more attention to ("attention").
- Outputs: aggregates of interactions and attention scores



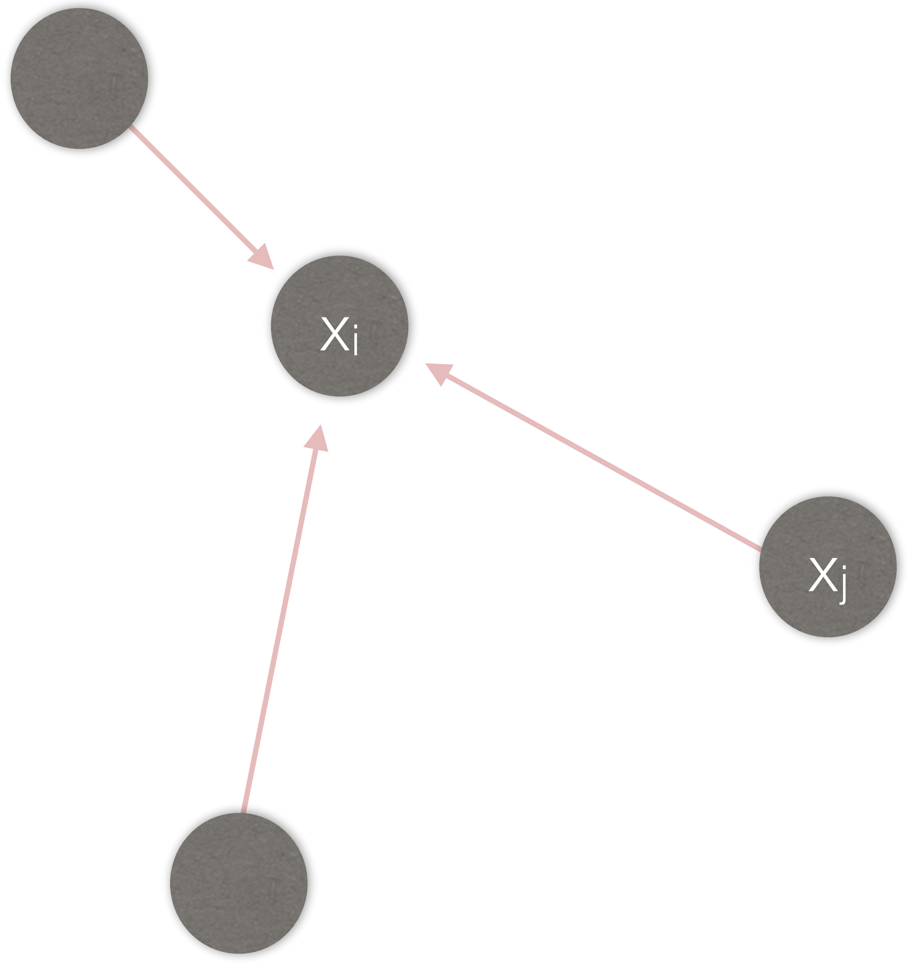


Google AI blog

Transformers and (self-)attention

(Self-)Attention

- Allows inputs to interact with each other ("self") and find out who they should pay more attention to ("attention").
- Outputs: aggregates of interactions and attention scores



Weighted sum over all input vectors:

$$\boldsymbol{y}_i = \sum_j w_{ij} \boldsymbol{x}_j$$

Weight (how related inputs are):

$$w'_{ij} = \boldsymbol{x}_i{}^{\mathsf{T}} \boldsymbol{x}_j$$

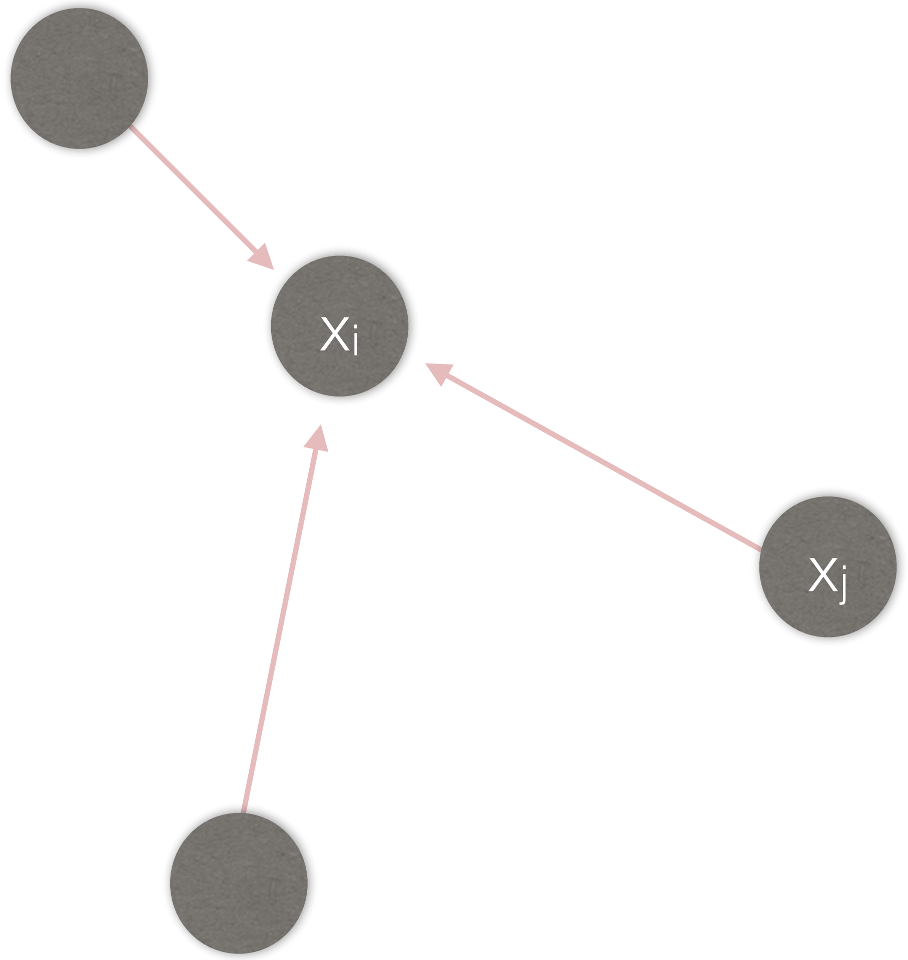
Map to [0,1]:

$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$$

Transformers and (self-)attention

(Self-)Attention

- Allows inputs to interact with each other ("self") and find out who they should pay more attention to ("attention").
- Outputs: aggregates of interactions and attention scores



Weighted sum over all input vectors:

$$\boldsymbol{y}_i = \sum_j w_{ij} \boldsymbol{x}_j$$

Weight (how related inputs are):

$$w'_{ij} = \boldsymbol{x}_i{}^{\mathsf{T}} \boldsymbol{x}_j$$

$x_j \rightarrow \mathrm{MLP}(x_J)$
$x_i \rightarrow \mathrm{MLP}(x_i)$

Map to [0,1]:

$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$$

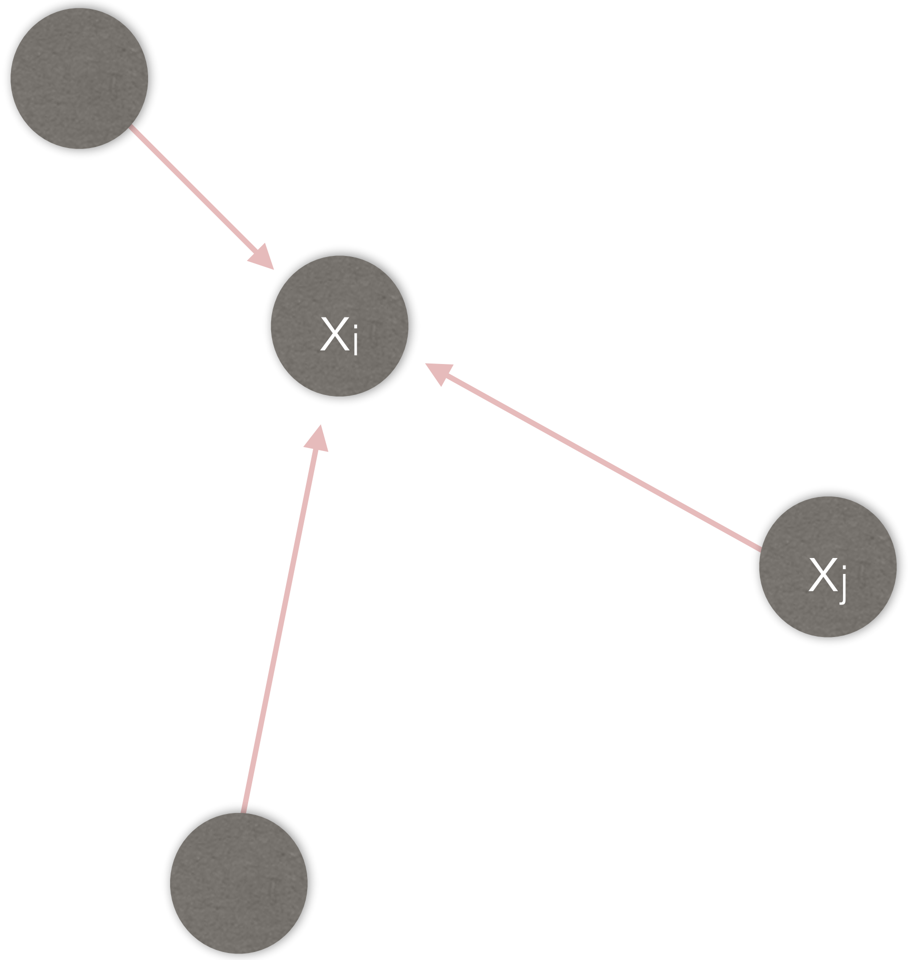
Transformers and (self-)attention

(Self-)Attention

- Allows inputs to interact with each other ("self") and find out who they should pay more attention to ("attention").
- Outputs: aggregates of interactions and attention scores

Attention weights: weighted importance between each pair of particles

- Determine relationship between all particles of point cloud
- Jet features become parameters of the model
- Several attention layers → different important features (multi-head attention)



Weighted sum over all input vectors:

$$\boldsymbol{y}_i = \sum_j w_{ij} \boldsymbol{x}_j$$

Weight (how related inputs are):

$x_j \rightarrow MLP(x_J)$
$x_i \rightarrow MLP(x_i)$

$$w'_{ij} = \boldsymbol{x}_i^{\mathsf{T}} \boldsymbol{x}_j$$

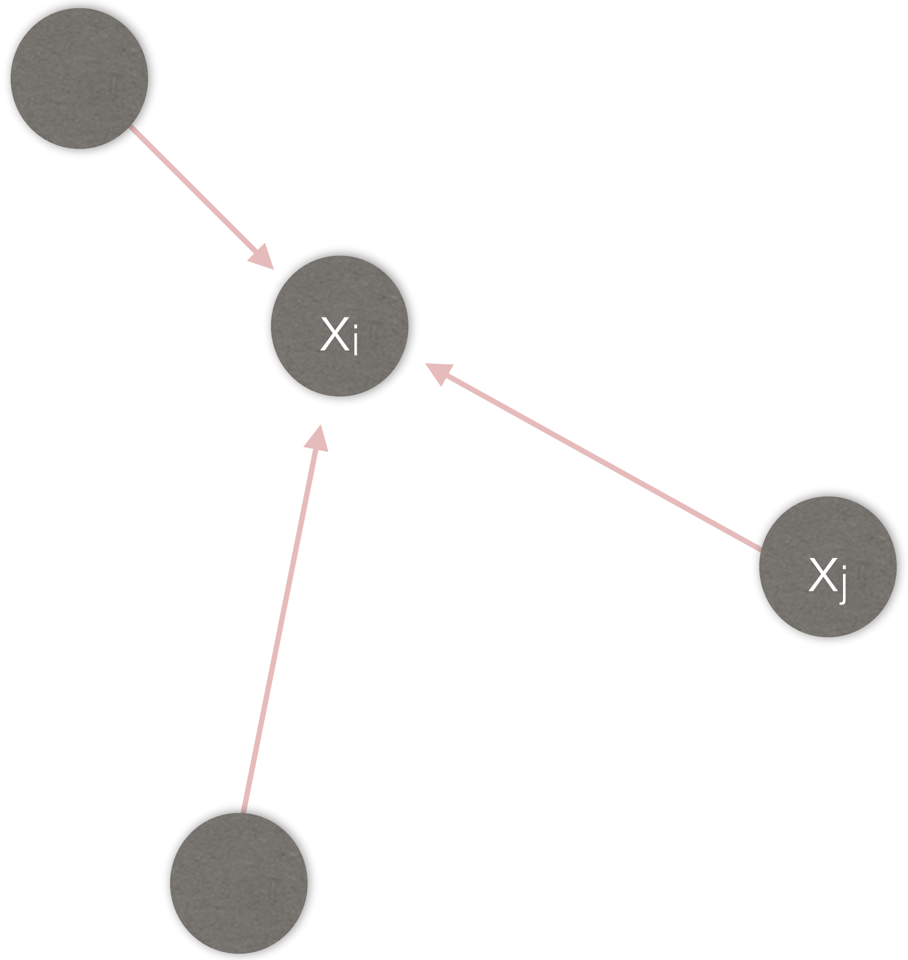
Map to [0,1]:

$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$$

Transformers and (self-)attention

(Self-)Attention

- Allows inputs to interact with each other ("self") and find out who they should pay more attention to ("attention").
- Outputs: aggregates of interactions and attention scores



Attention weights: weighted importance between each pair of particles

- Determine relationship between all particles of point cloud
- Jet features become parameters of the model
- Several attention layers → different important features (multi-head attention)

Transformer:

- Only set of interaction between units is self-attention!

Weighted sum over all input vectors:

$$\mathbf{y}_i = \sum_j w_{ij} \mathbf{x}_j$$

Weight (how related inputs are):

$x_j \rightarrow \mathrm{MLP}(x_J)$
$x_i \rightarrow \mathrm{MLP}(x_i)$

$$w'_{ij} = \mathbf{x}_i{}^{\mathsf{T}} \mathbf{x}_j$$

Map to [0,1]:

$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$$

Transformers and (self-)attention

(Self-)Attention

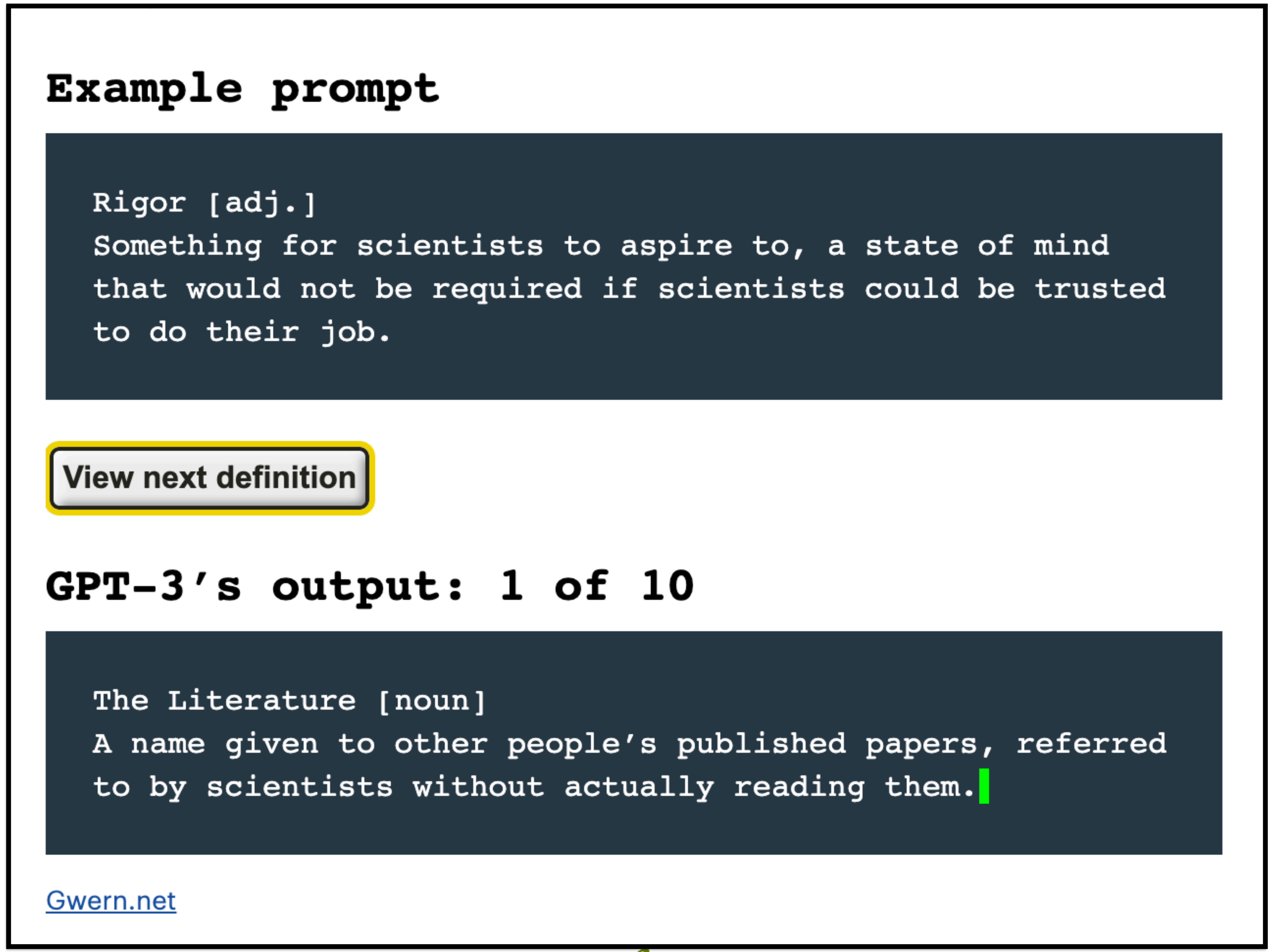
- Allows inputs to interact with each other ("self") and find out who they should pay more attention to ("attention").
- Outputs: aggregates of interactions and attention scores

Attention weights: weighted importance between each pair of particles

- Determine relationship between all particles of point cloud
- Jet features become parameters of the model
- Several attention layers → different important features (multi-head attention)

Transformer:

- Only set of interaction between units is self-attention!



Weight (how related inputs are):

$x_j \rightarrow \mathrm{MLP}(x_J)$
$x_i \rightarrow \mathrm{MLP}(x_i)$

$$w'_{ij} = {x_i}^{\mathsf{T}} x_j$$

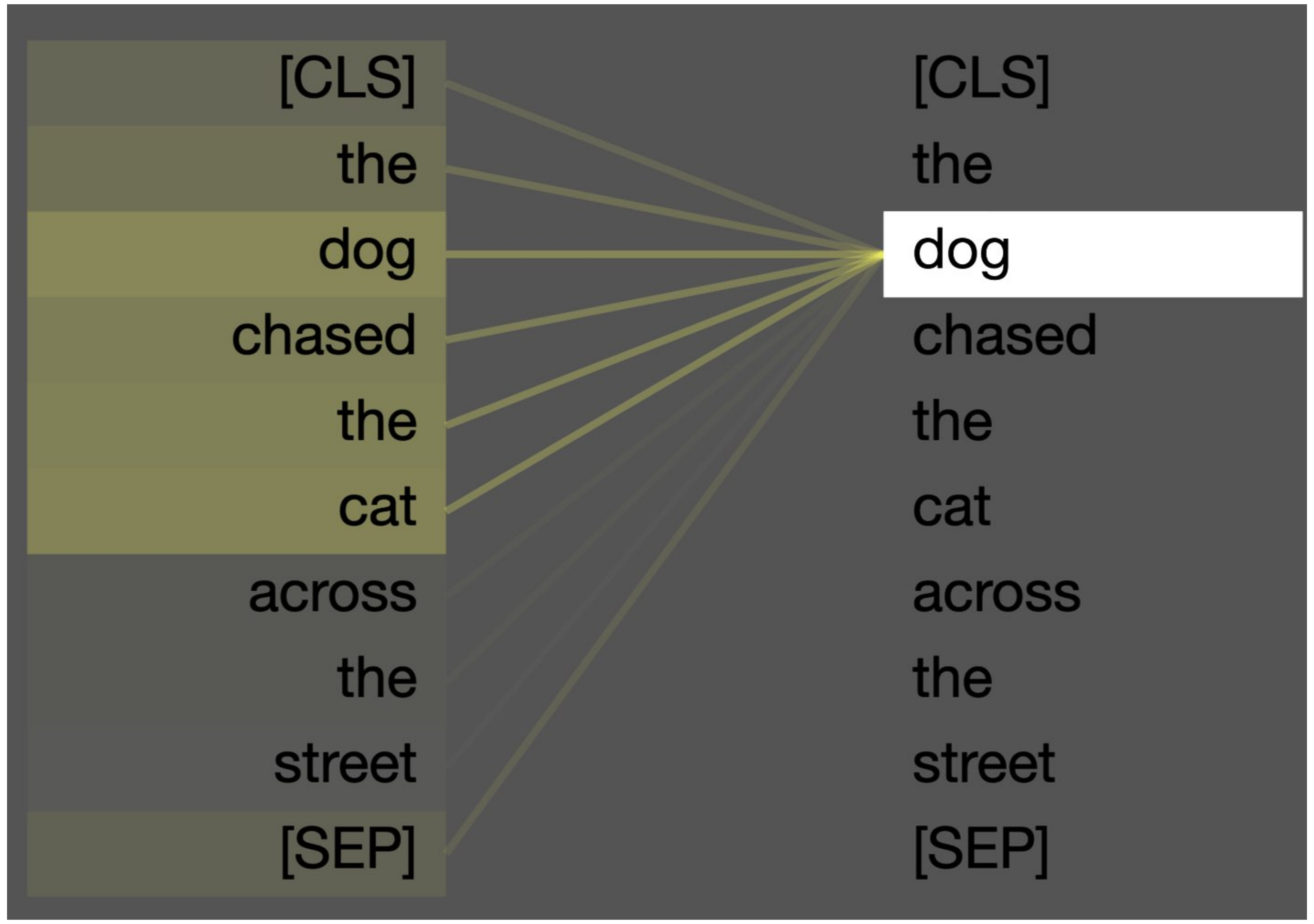
Map to [0,1]:

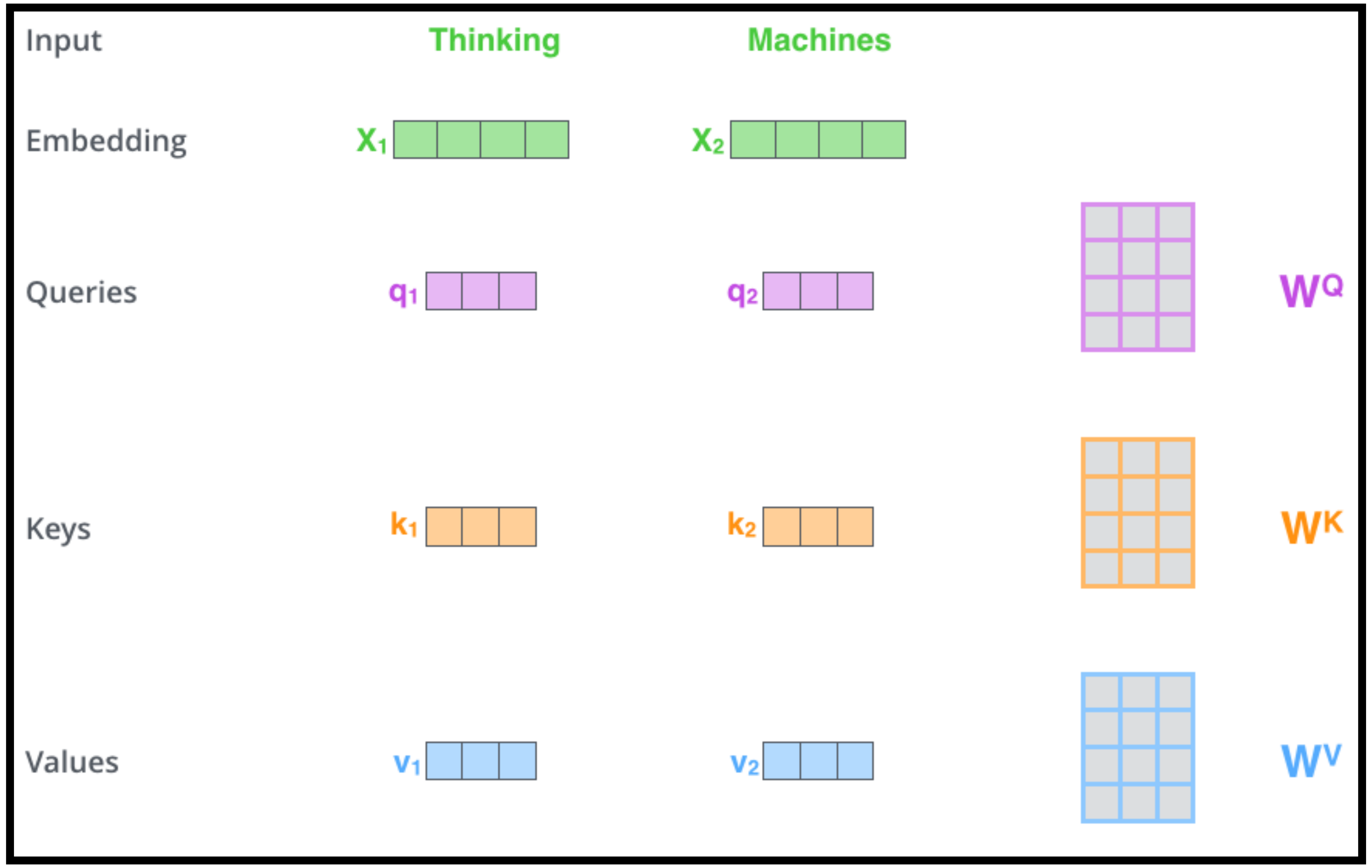
$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$$

Transformers

Query, Key and Value: How self-attention is implemented

- **The query**: dog
  I'm looking for verbs, adjectives related to me
- **The key**: every word in the sentence!
  I'm a noun, an adjective or a verb
  (What am I? What features do I posses in relation to the sentence?)
- **The value:** the meaning of this word in general not specifically for this sentence (What are my embeddings? What's the semantic information I posses?)





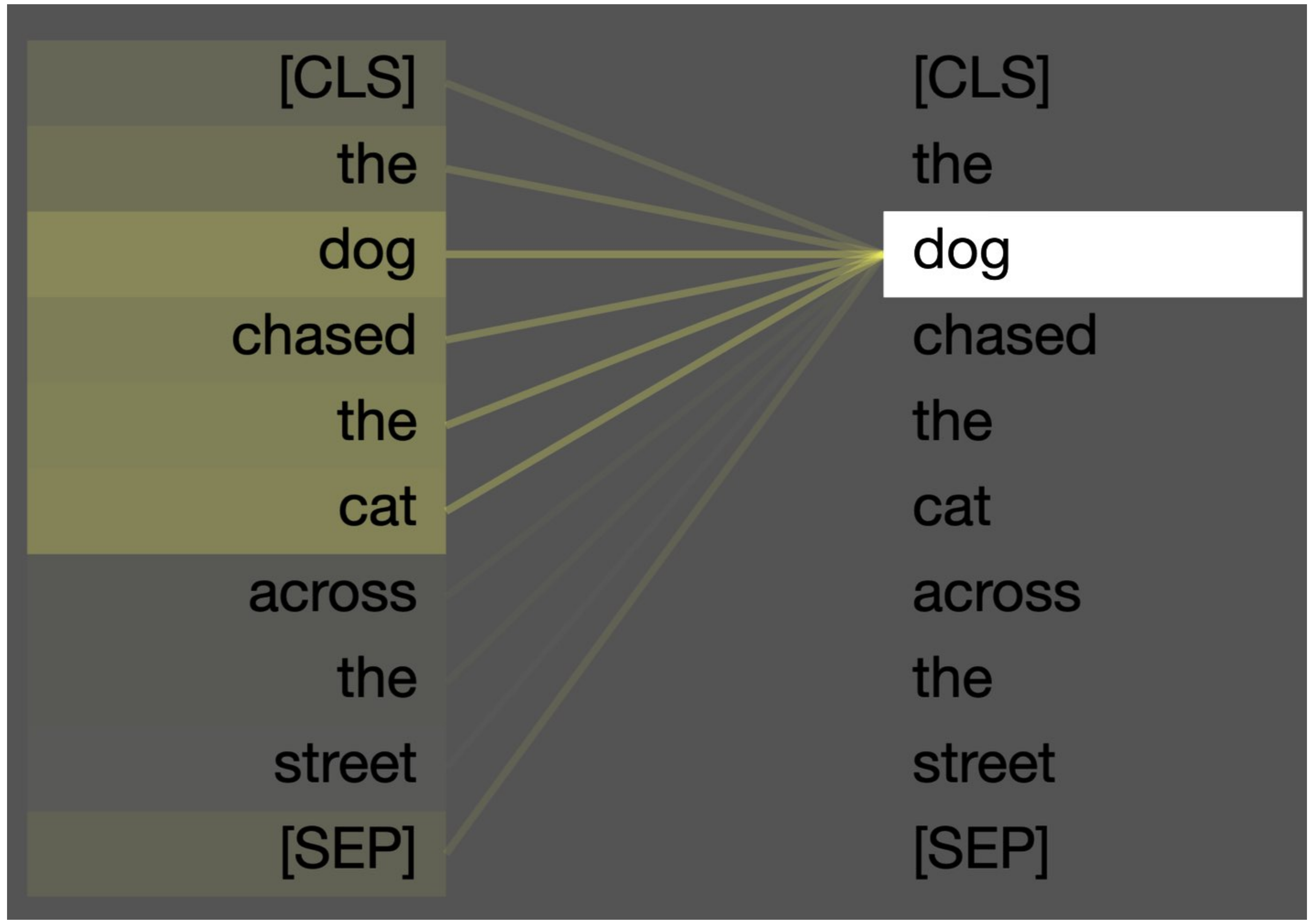
Transformers

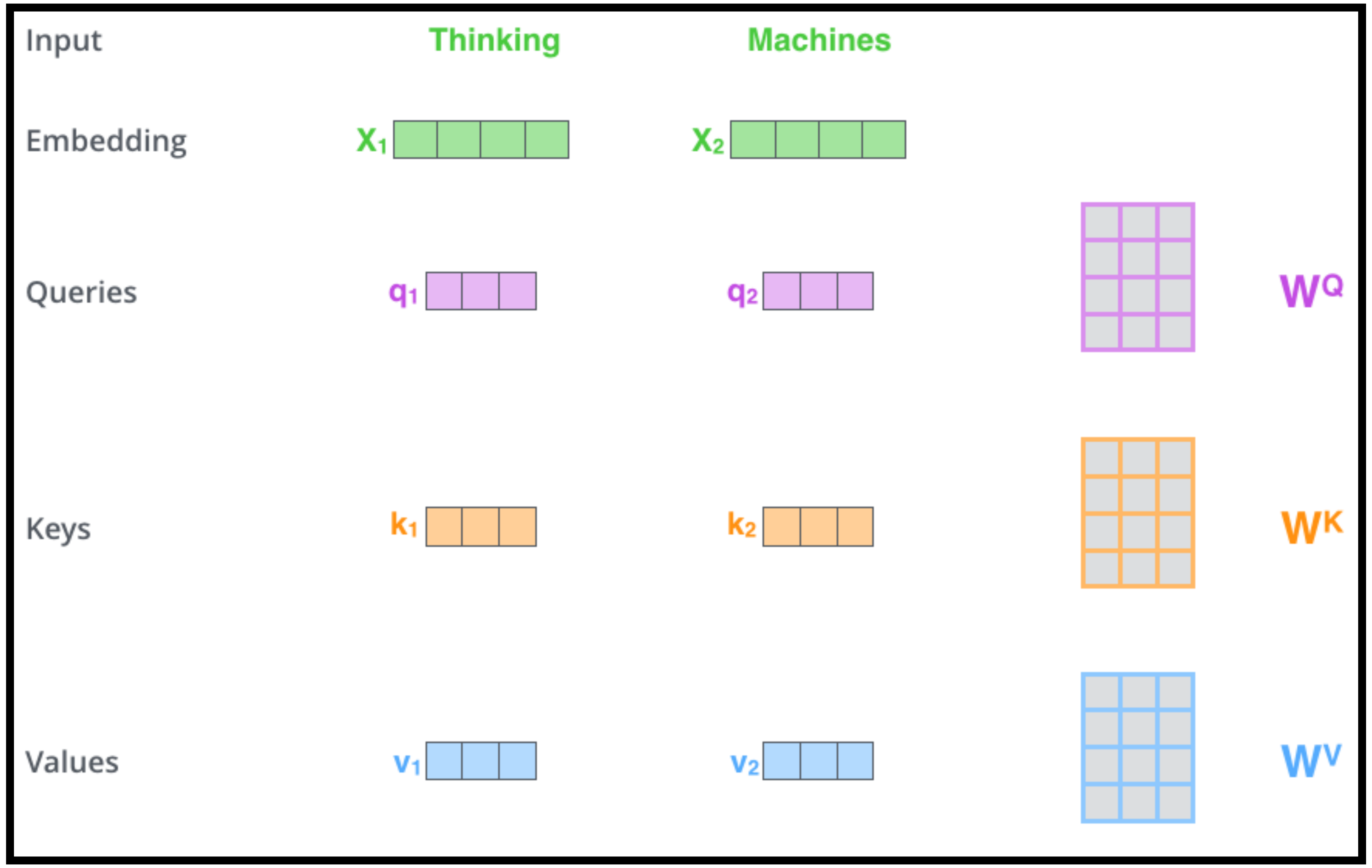
Query, Key and Value: How self-attention is implemented

- **The query**: dog
  I'm looking for verbs, adjectives related to me
- **The key**: every word in the sentence!
  I'm a noun, an adjective or a verb
  (What am I? What features do I posses in relation to the sentence?)
- **The value:** the meaning of this word in general not specifically for this sentence (What're my embeddings? What's the semantic information I posses?)

Self-Attention for word dog:

- **Dog (Query Vector):** Multiplied with all other words (Keys) to get Attention Map
- **Attention Map:** represent importance of every other word related to Dog
- This attention map will be multiplied by the Embeddings of the Sentence words (Values), and produce a weighted sum of the embeddings based on the relevancy of the words





ABCNet:

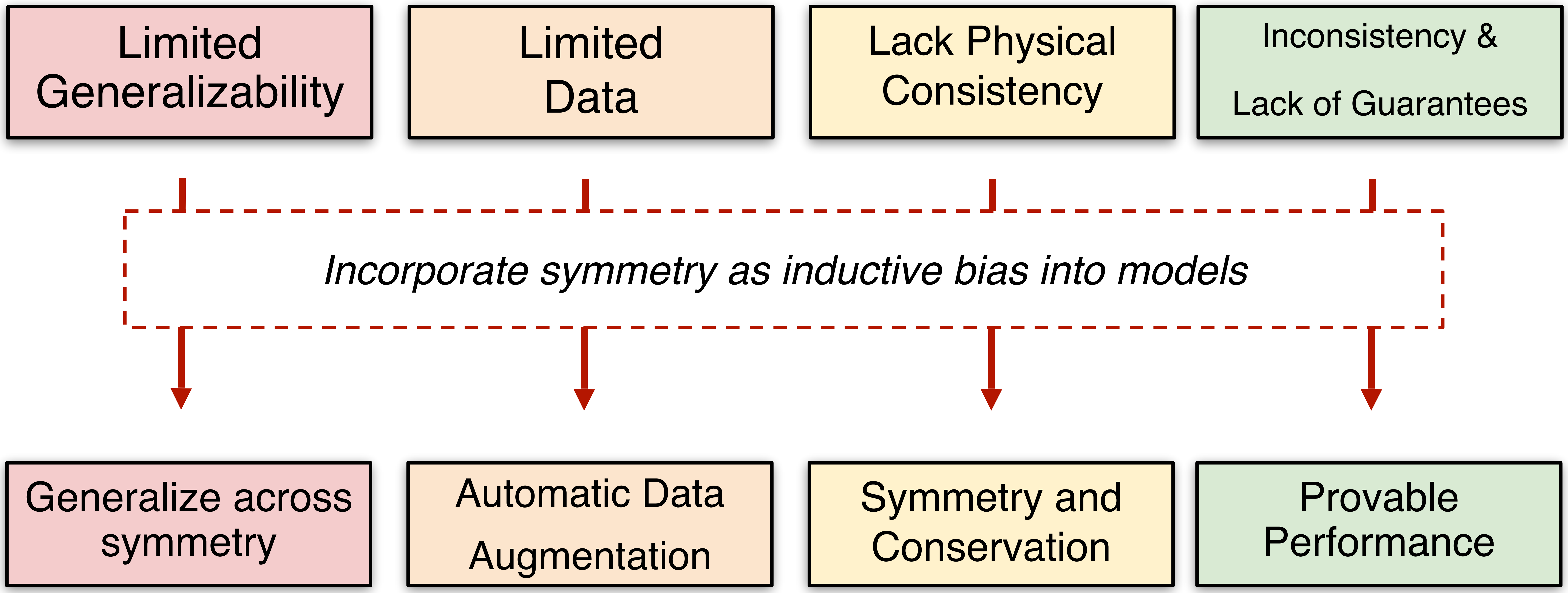
Pixel intensity = particle importance w.r.t most energetic particle in jet, from attention weights

No substructure information given, learned through attention layers!

Symmetries

Symmetries is an extremely important concept, also in Machine Learning.

- If there is a symmetry in your system, integrate it into your model, and it can do more with less!

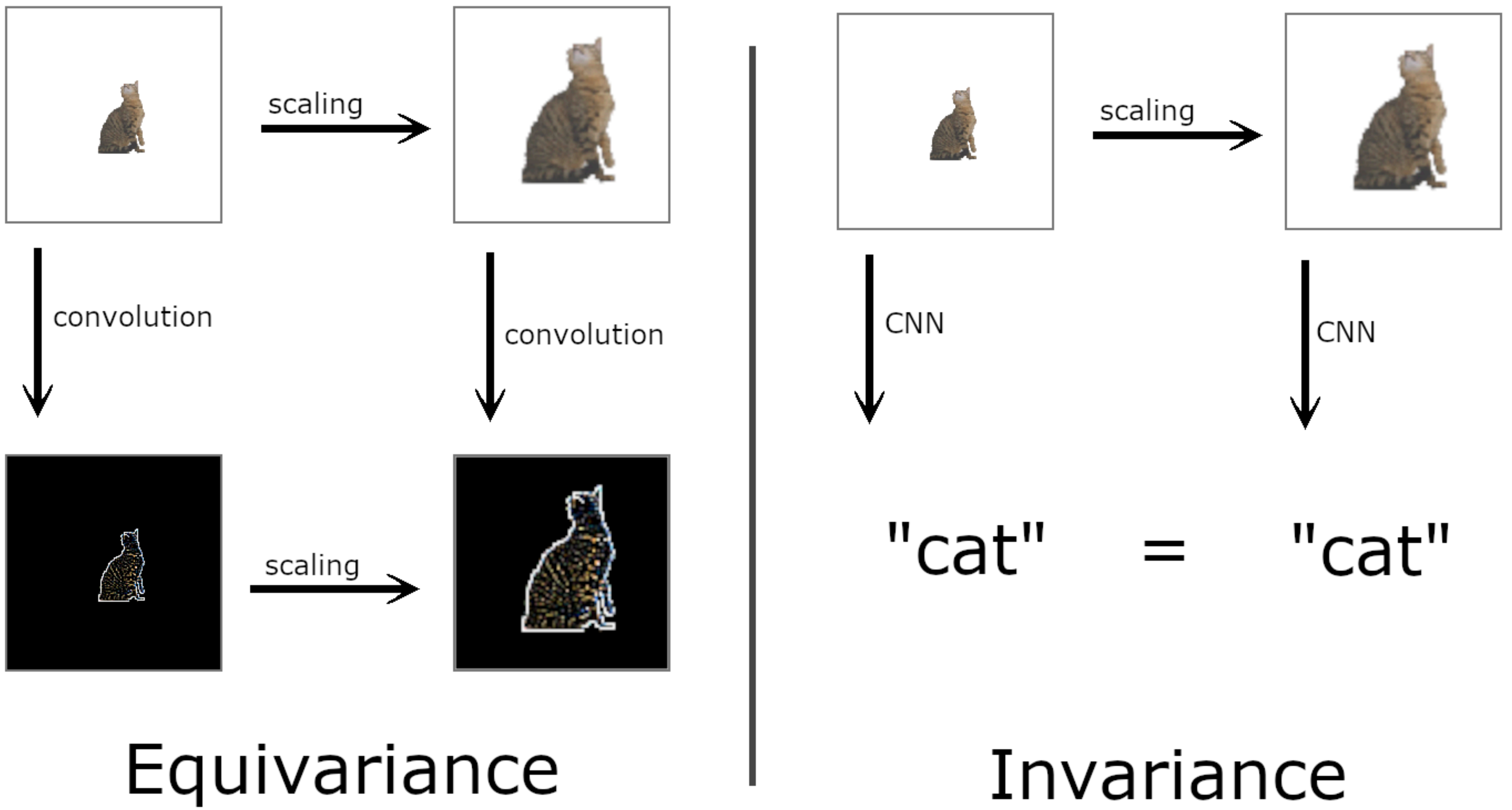


Walters, IAIFI School 2023

Symmetries

Result changes in "the same way" as the input

Result doesn't change when you change the input

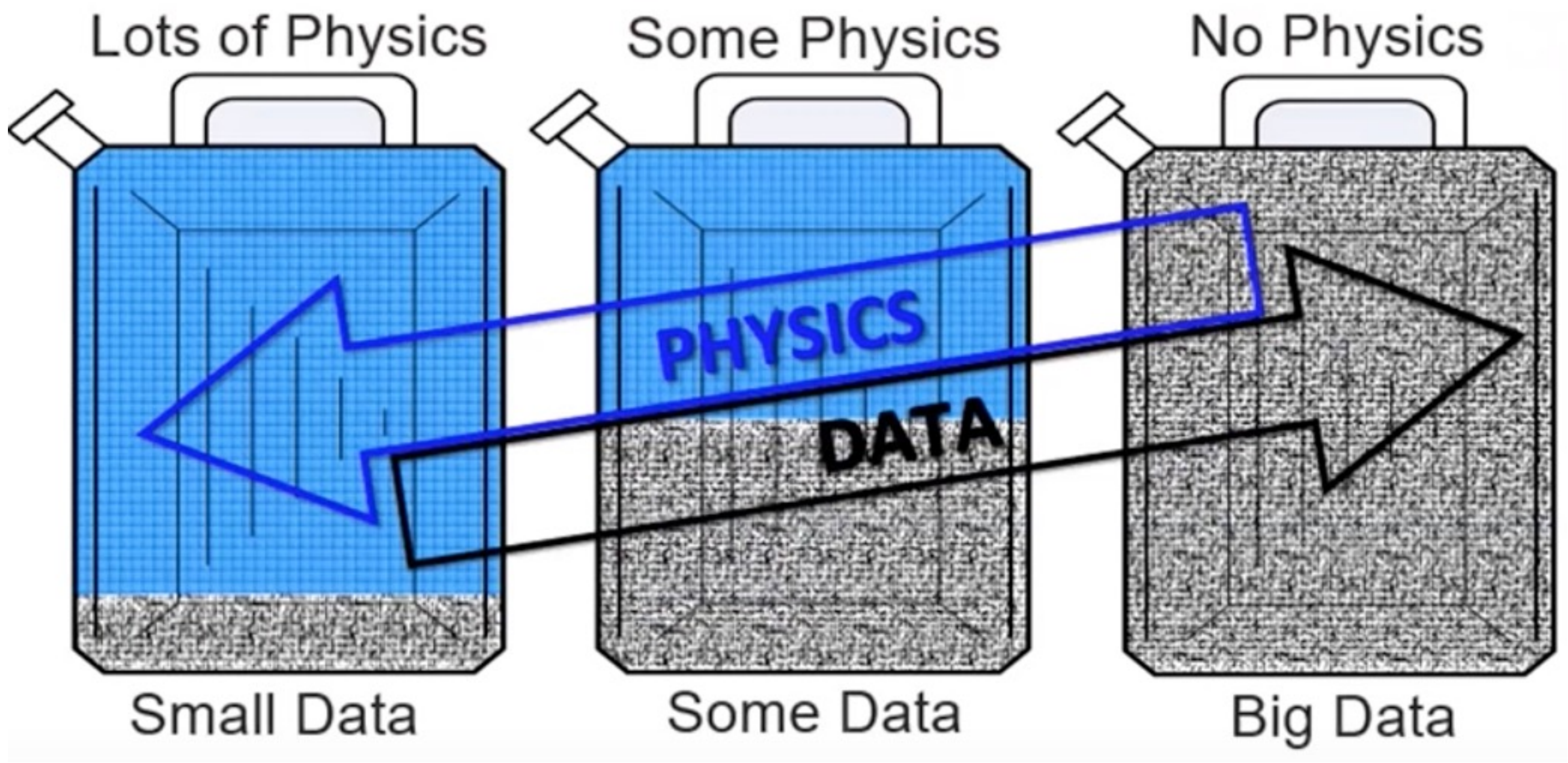


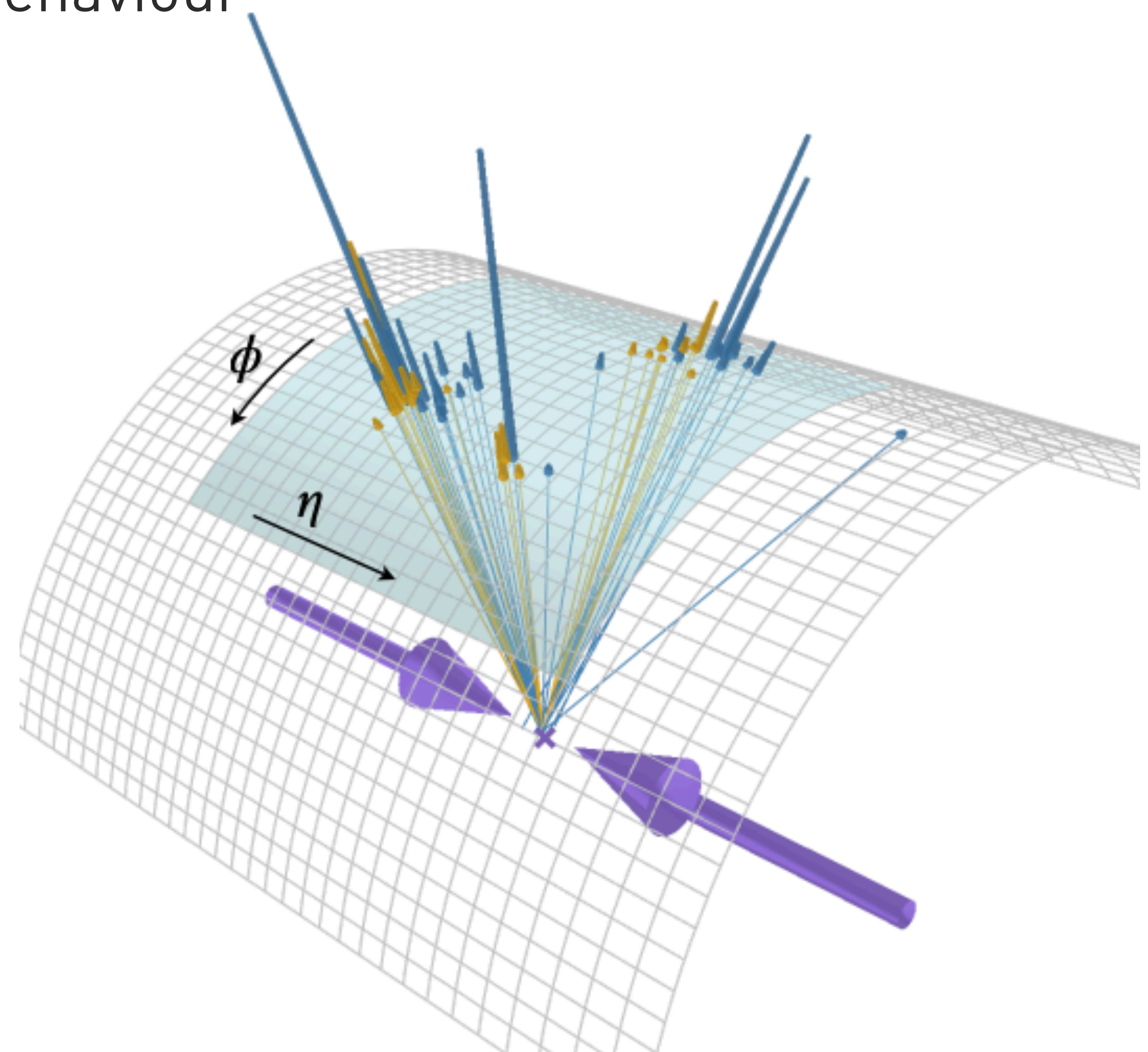
E.g CNNs & GNNs (see later)! Invariances are usually obtained through weight sharing

Symmetries

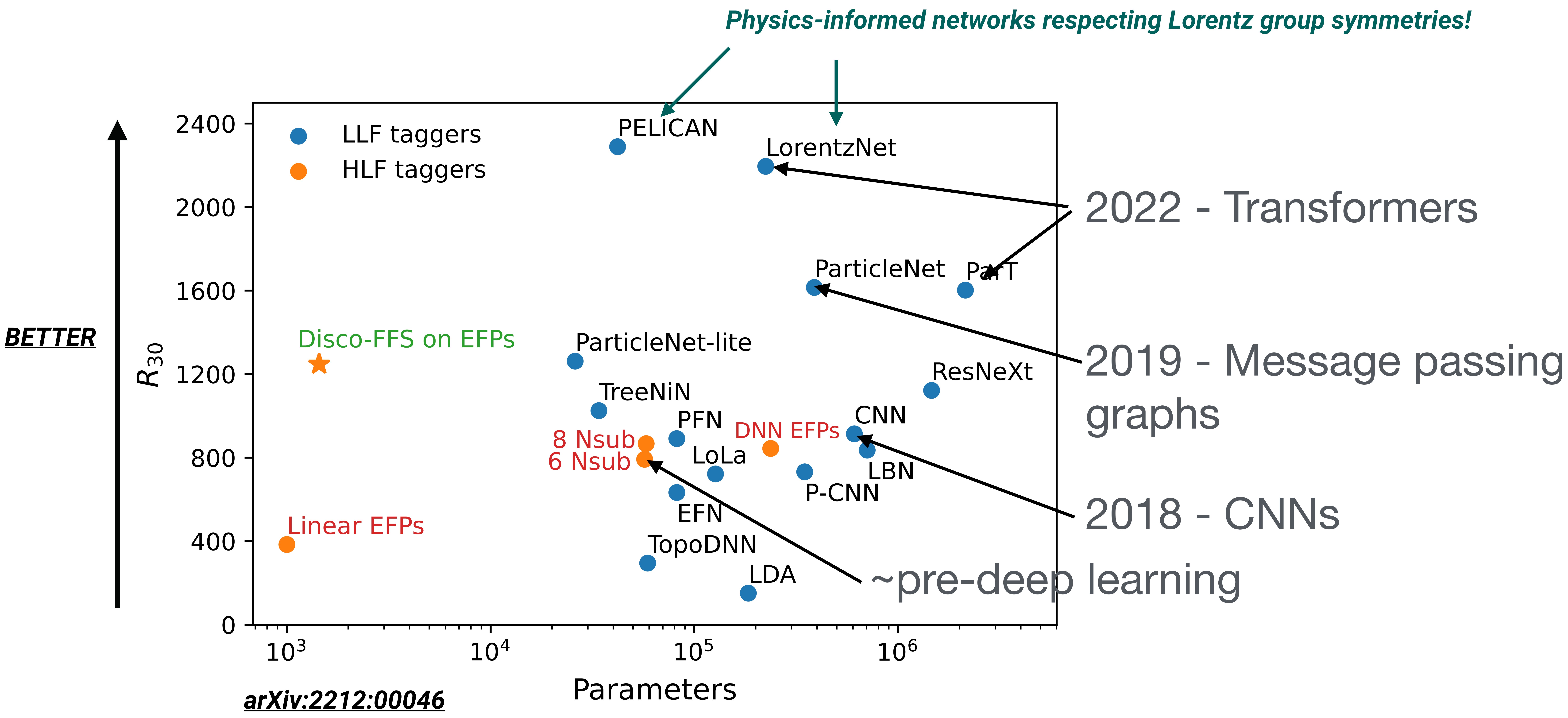
More and more work in HPE try to utilise symmetries when designing DNNs, e .g invariant under Lorentz symmetries!

- Other than observed data, we know the invariances that govern physical phenomena
- Conservation of mass, momentum, energy
- In many cases, we also have approximate models that can predict the system behaviour

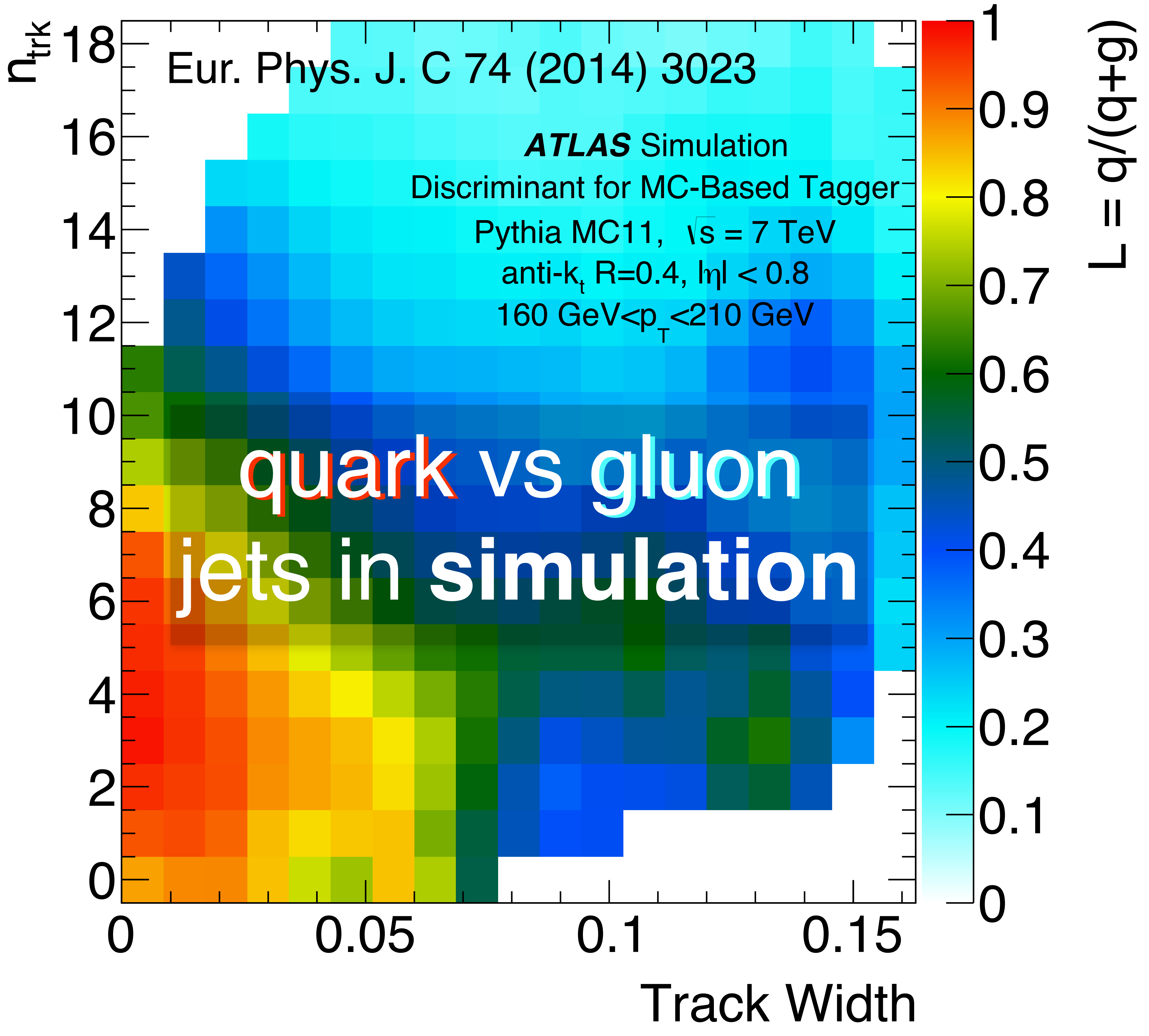


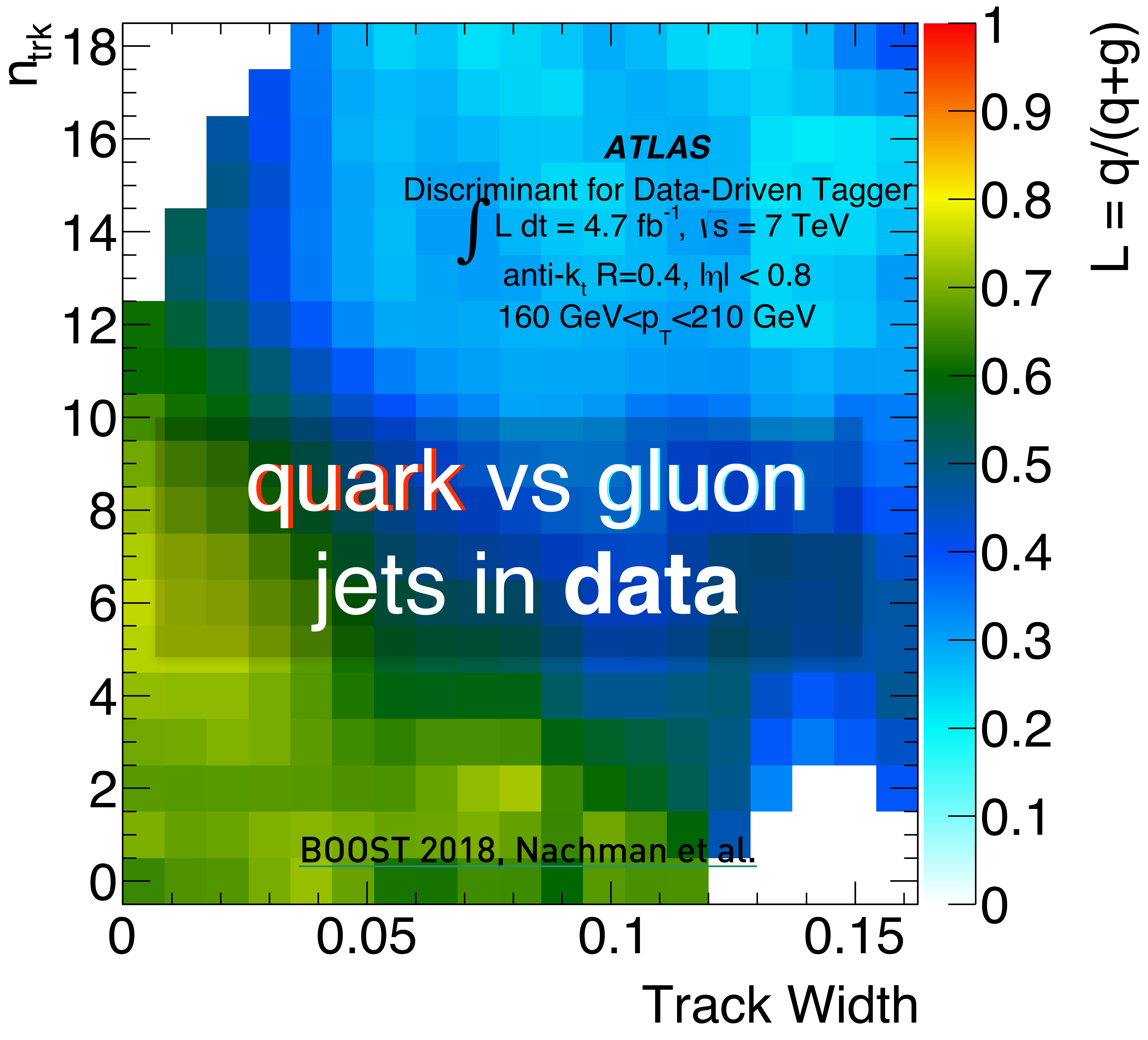


<u>The label of a jet should be invariant under any transformation of the input jet, right?</u>

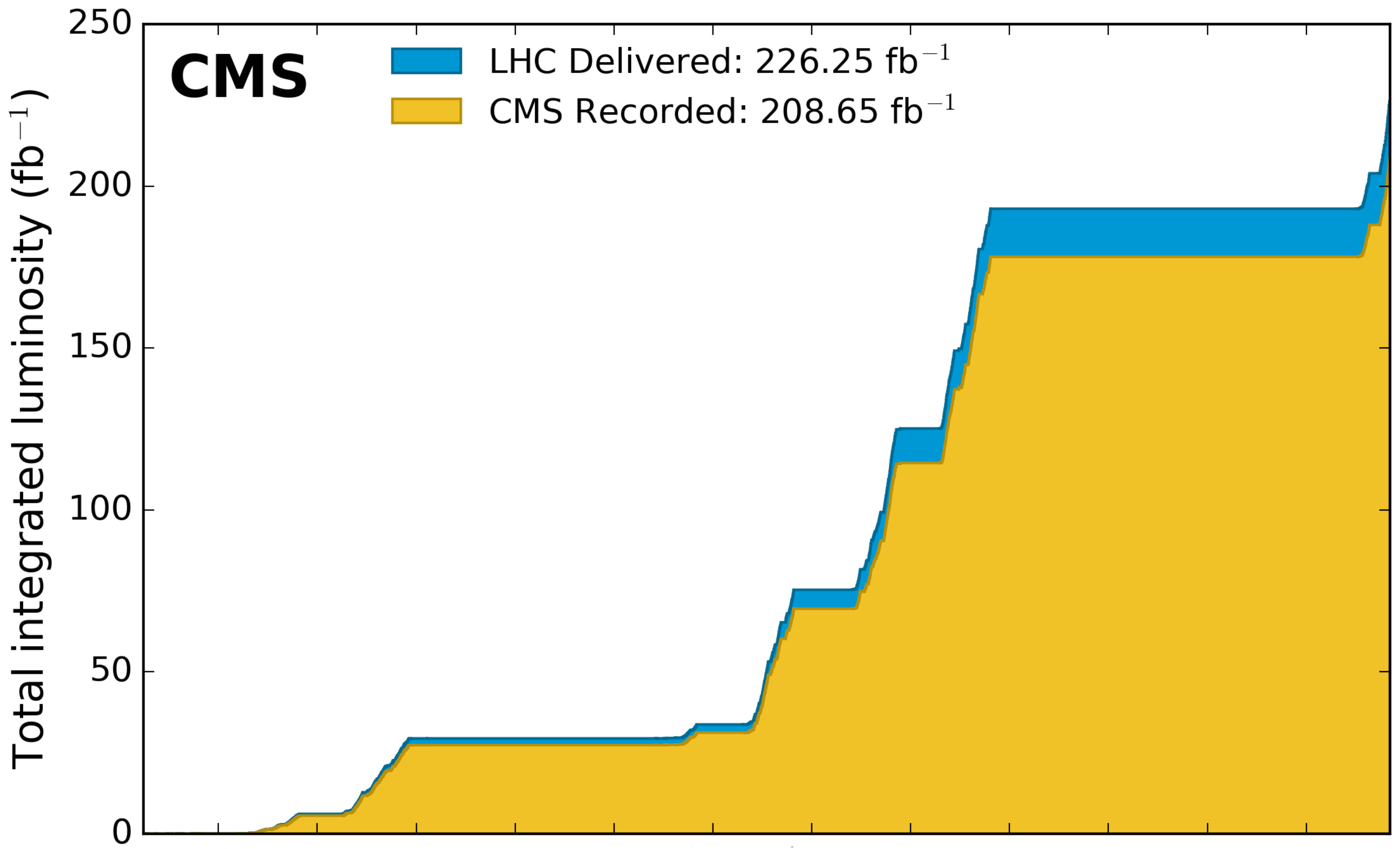

Physics-informed networks respecting Lorentz group symmetries!
BETTER
R_{30}
LLF taggers
HLF taggers
PELICAN
LorentzNet
ParticleNet
ParT
Disco-FFS on EFPs
ParticleNet-lite
ResNeXt
TreeNiN
PFN
CNN
DNN EFPs
8 Nsub
6 Nsub
LoLa
LBN
P-CNN
EFN
Linear EFPs
TopoDNN
LDA
Parameters
2022 - Transformers
2019 - Message passing graphs
2018 - CNNs
~pre-deep learning
arXiv:2212:00046

Train on simulation, test on data





If data and simulation differ, this is sub-optimal!



Simulation != test data

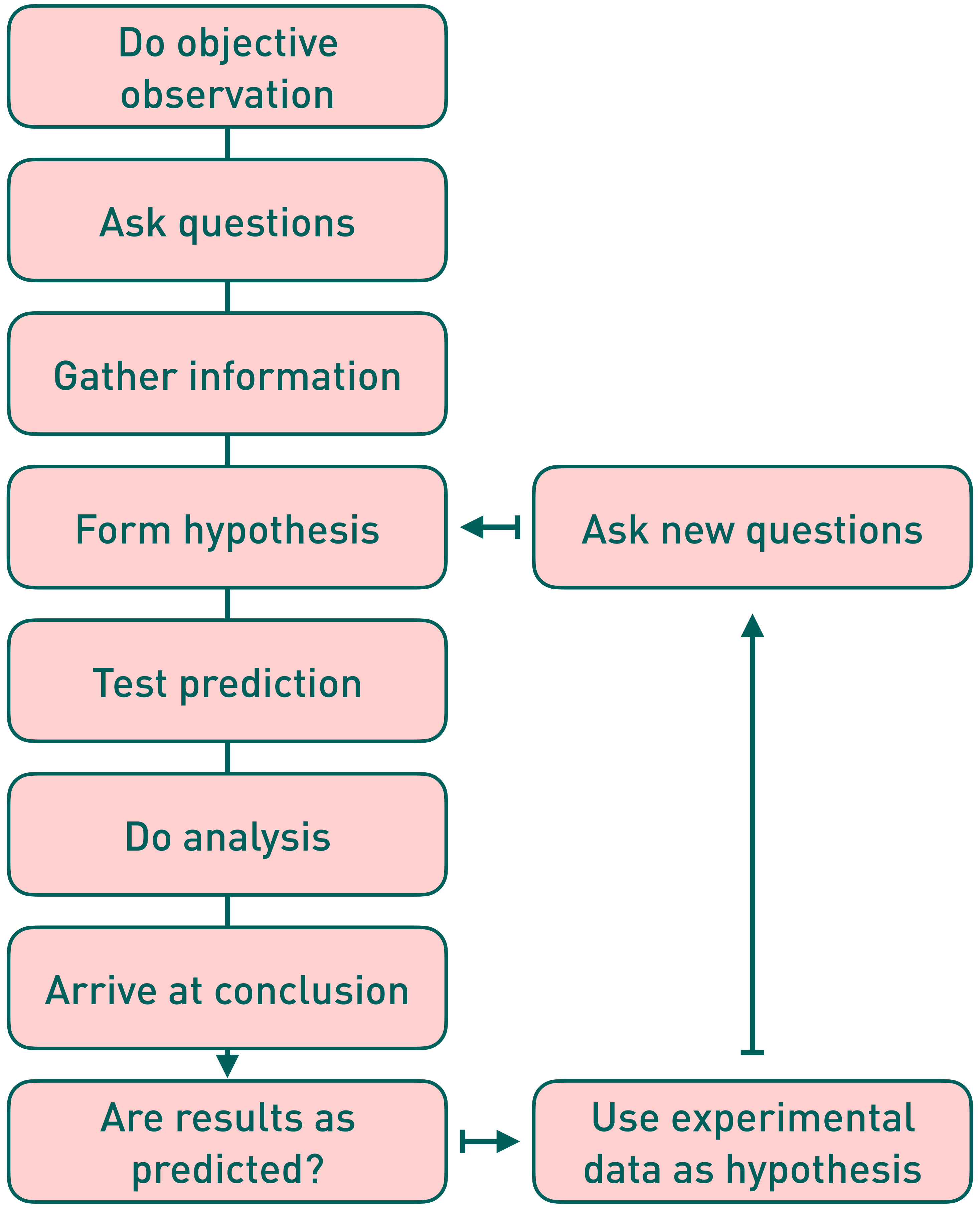
Mostly (SM)background samples, small signal datasets

Unsupervised/SSL

- No labels, completely data driven

We are also very keen on using this!

The scientific method

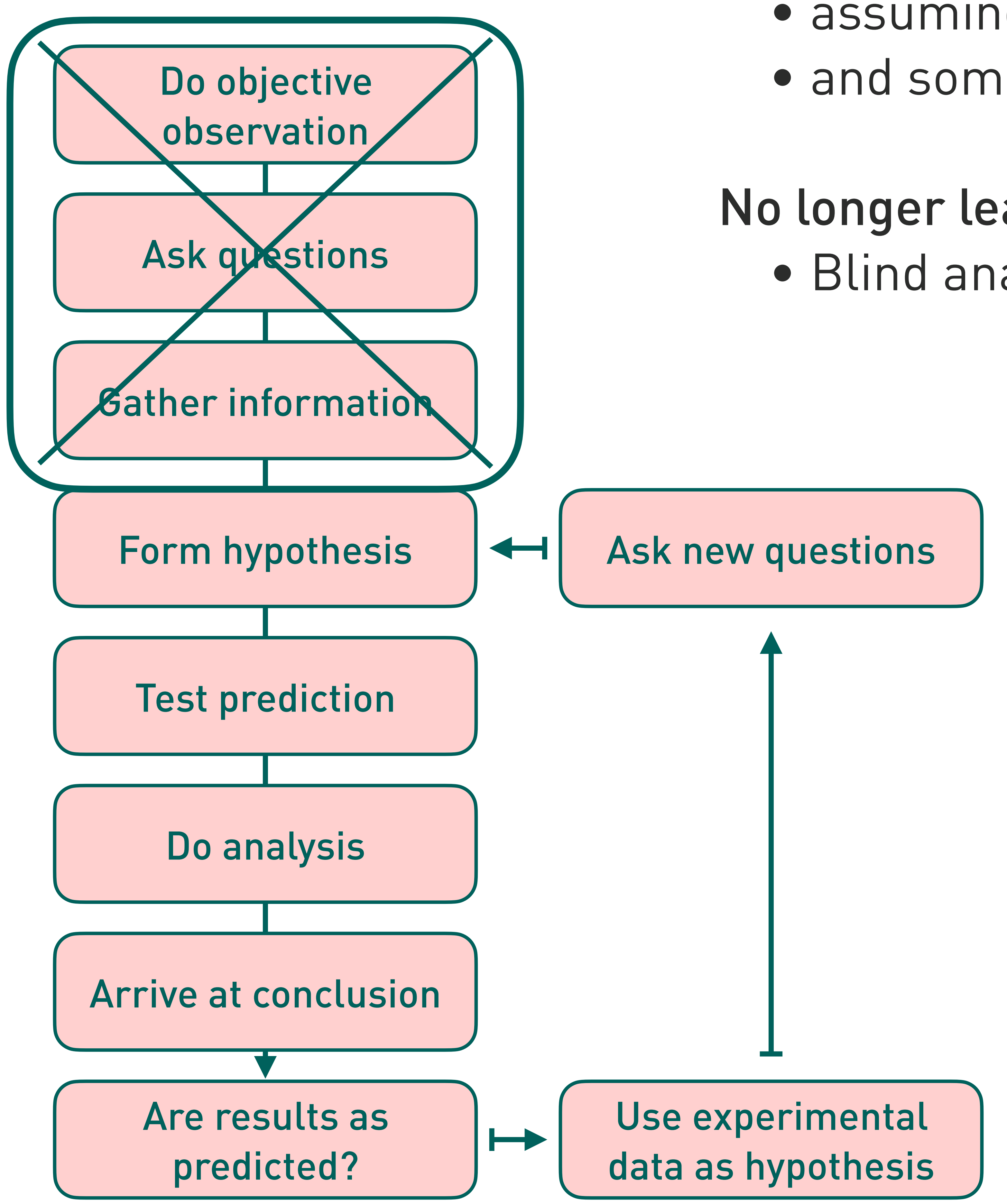


Searches at LHC

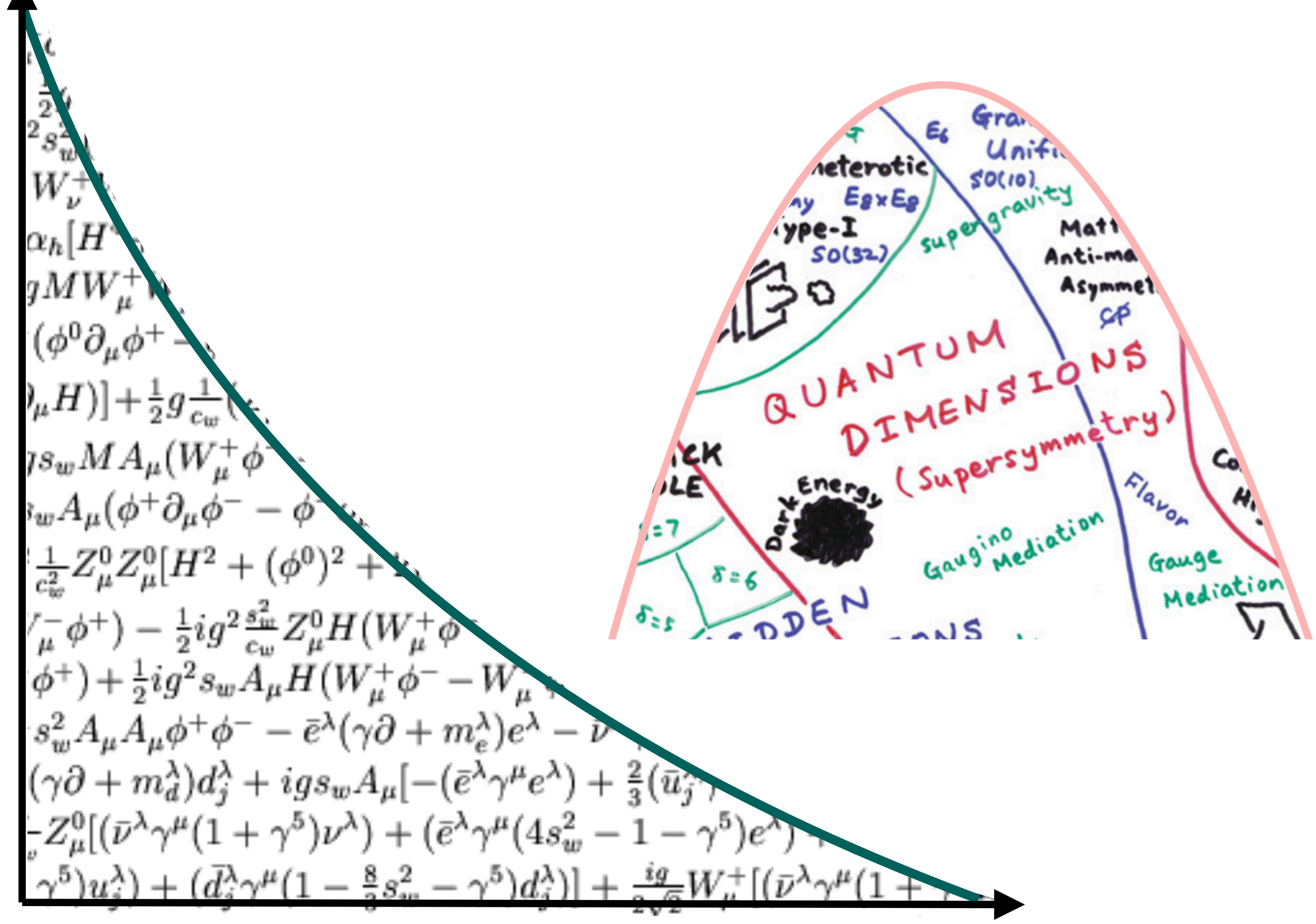
Searches at LHC (almost) always start with by
- assuming Standard Model
- and some signal hypothesis

No longer learn from observation
- Blind analysis only way we perform searches

Replaced by:



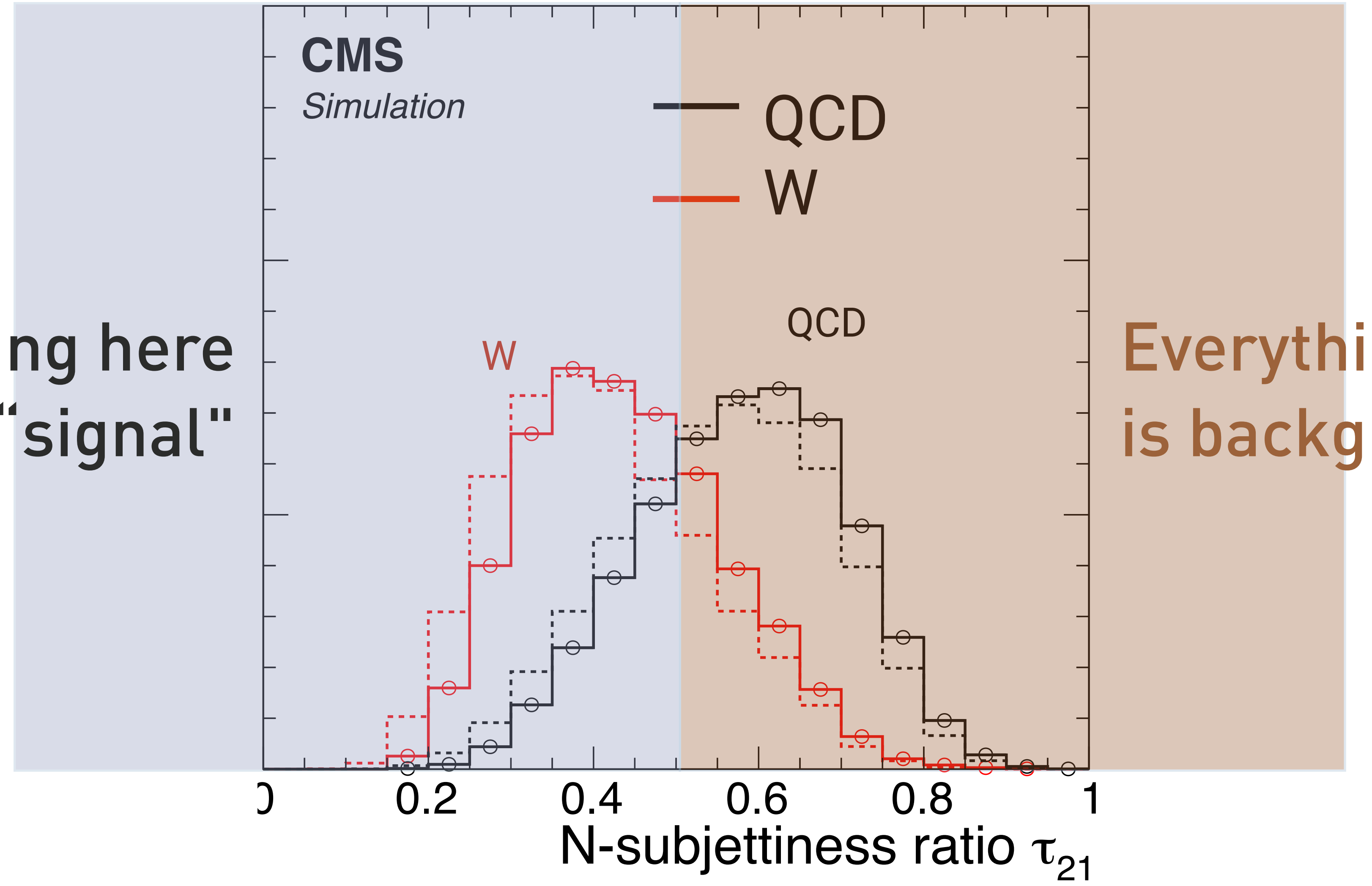
Standard Model (MC) **Signal hypothesis (MC)**



Searches at LHC

This is fine when you know what you are looking for

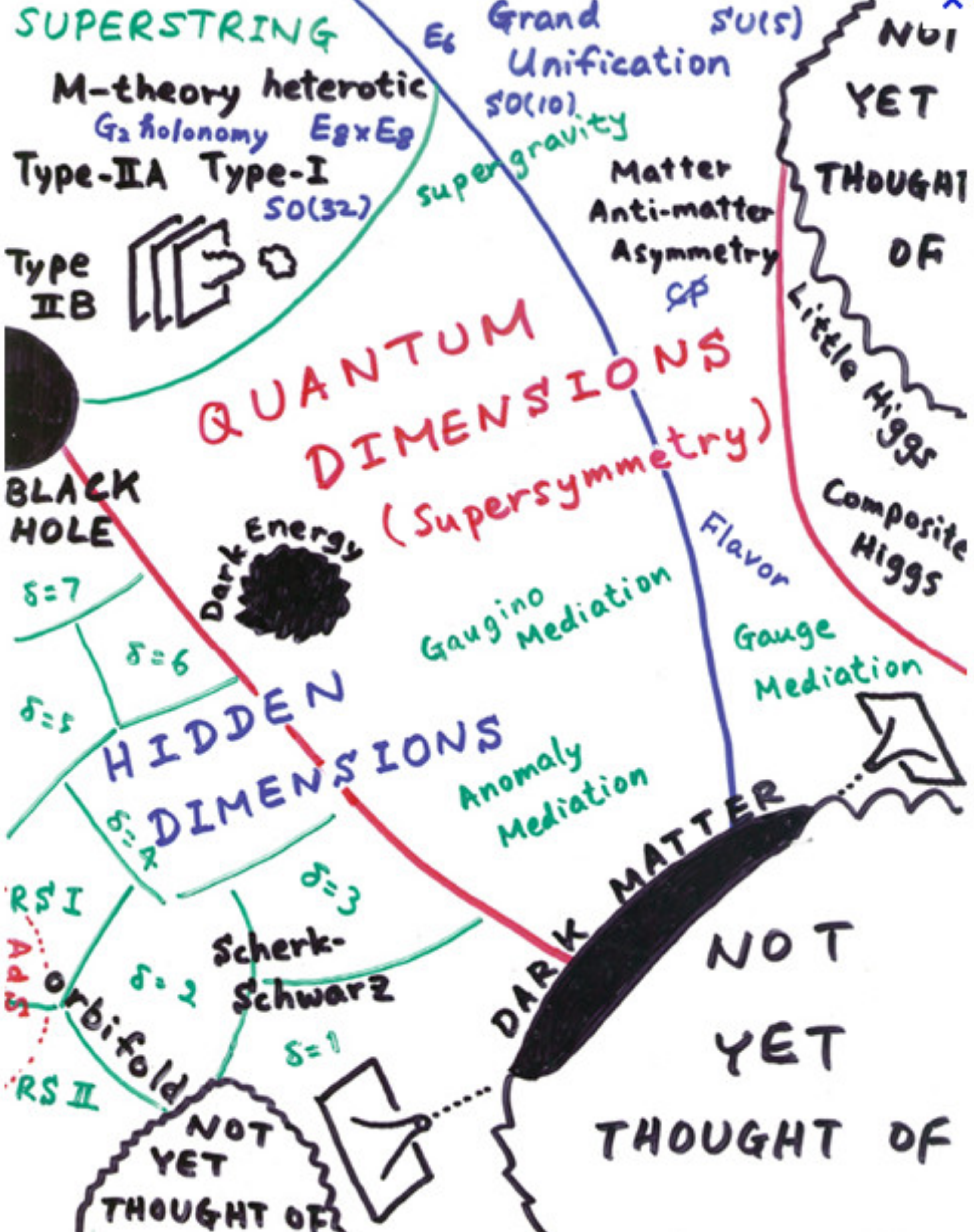
- Tailor search to a given theory
- Motivated by belief/disbelief
- Powerful, but **limited to model of choice**



Everything here is “signal"

Everything here is background


CMS Preliminary
LHCP 2017
Vector mediator
EXCLUDED
(a) CMS
DM Simplified Model Exclusions
ATLAS Preliminary March 2017
EXCLUDED
EXCLUDED
(a) CMS
EXCLUDED
(b) ATLAS
EXCLUDED
EXCLUDED
EXCLUDED
EXCLUDED
(a) CMS
EXCLUDED
EXCLUDED
(b) ATLAS
EXCLUDED
EXCLUDED
(a) CMS
EXCLUDED
EXCLUDED
CMS Preliminary
LHCP 2020
Axial-vector mediator
EXCLUDED
2016 diboson combination
CMS
EXCLUDED
Fall 2020
CMS Preliminary
EXCLUDED
EXCLUDED
EXCLUDED
EXCLUDED
EXCLUDED
(a) CMS
EXCLUDED
(b) ATLAS
CMS Preliminary
LHCP 2020
EXCLUDED

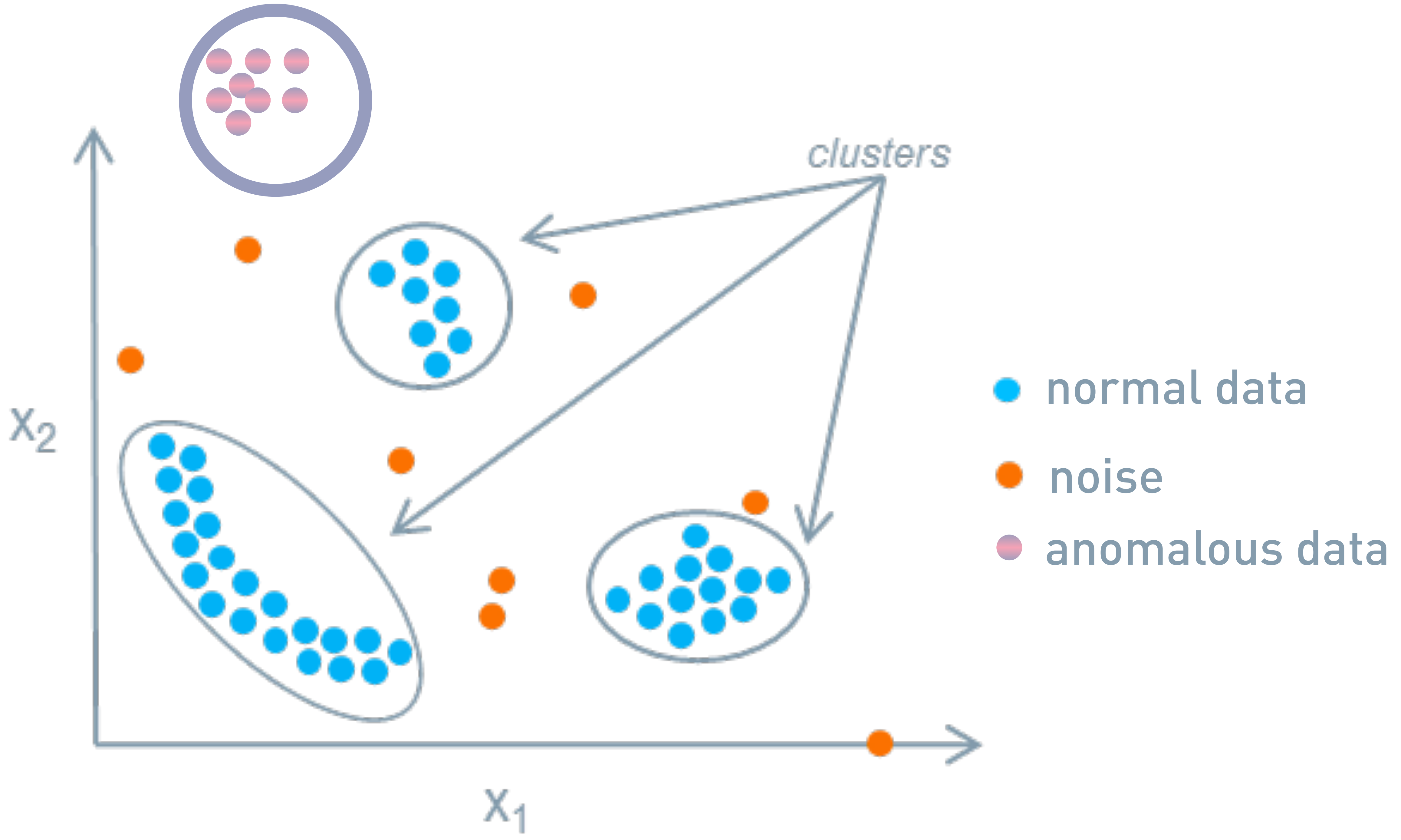

SUPERSTRING
M-theory
heterotic
G_2 holonomy
$E_8 \times E_8$
Type-IIA
Type-I
SO(32)
Type IIB
BLACK HOLE
E_6
Grand Unification
SU(5)
SO(10)
supergravity
Matter Anti-matter Asymmetry
YET
OF
QUANTUM DIMENSIONS
(Supersymmetry)
Little Higgs
Composite Higgs
Dark Energy
Flavor
Gaugino Mediation
Gauge Mediation
$\delta=7$
$\delta=6$
$\delta=5$
HIDDEN DIMENSIONS
Anomaly Mediation
$\delta=3$
RS I
AdS
orbifold
$\delta=2$
Scherk-Schwarz
$\delta=1$
RS II
DARK MATTER
NOT YET THOUGHT OF
NOT YET THOUGHT OF


ARE WE LOOKING IN THE WRONG WAY?
CMS Preliminary
LHCP 2017
Vector mediator
Dirac DM
EXCLUDED
Mediator mass M_{med} [GeV]
Dark matter mass m_{DM} [GeV]
(a) CMS
DM Simplified Model Exclusions
ATLAS Preliminary March 2017
(b) ATLAS
2016 diboson combination
CMS
Combination
95% CL upper limits
HVT model B
CMS Preliminary
LHCP 2020
Axial-vector mediator
Pseudoscalar, Dirac, $g_q=1$, $g_\chi=1$
Baryonic Z'
Z' mass (GeV)
$c\tau_0$ (mm)
$Z' \to q\bar{q}$

Learning from data

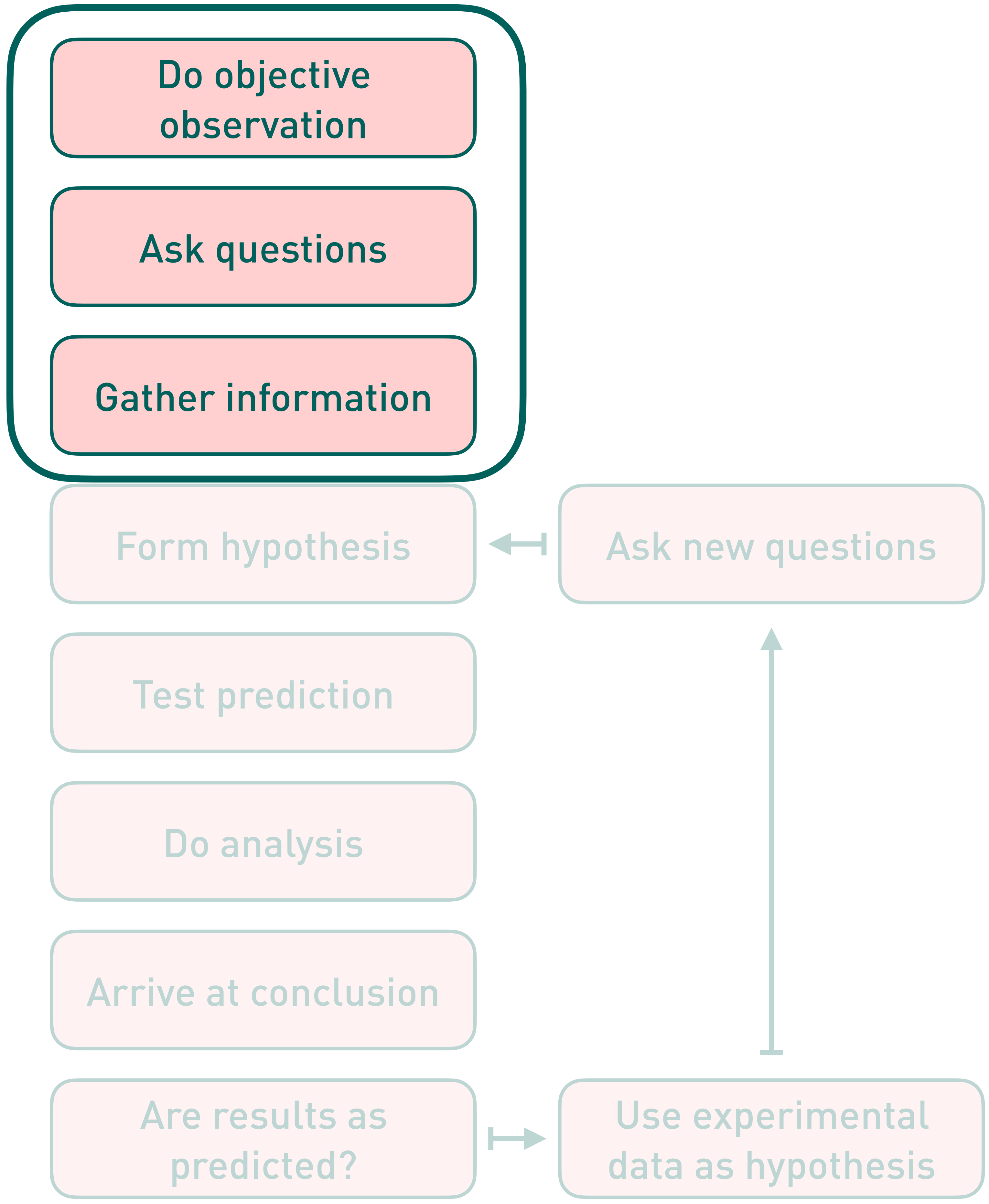
Look at **data** rather than defining signal hypothesis a priori

- Can we “classify” objects/events?



What are “normal” data and what are “outliers” (and what is noise)?

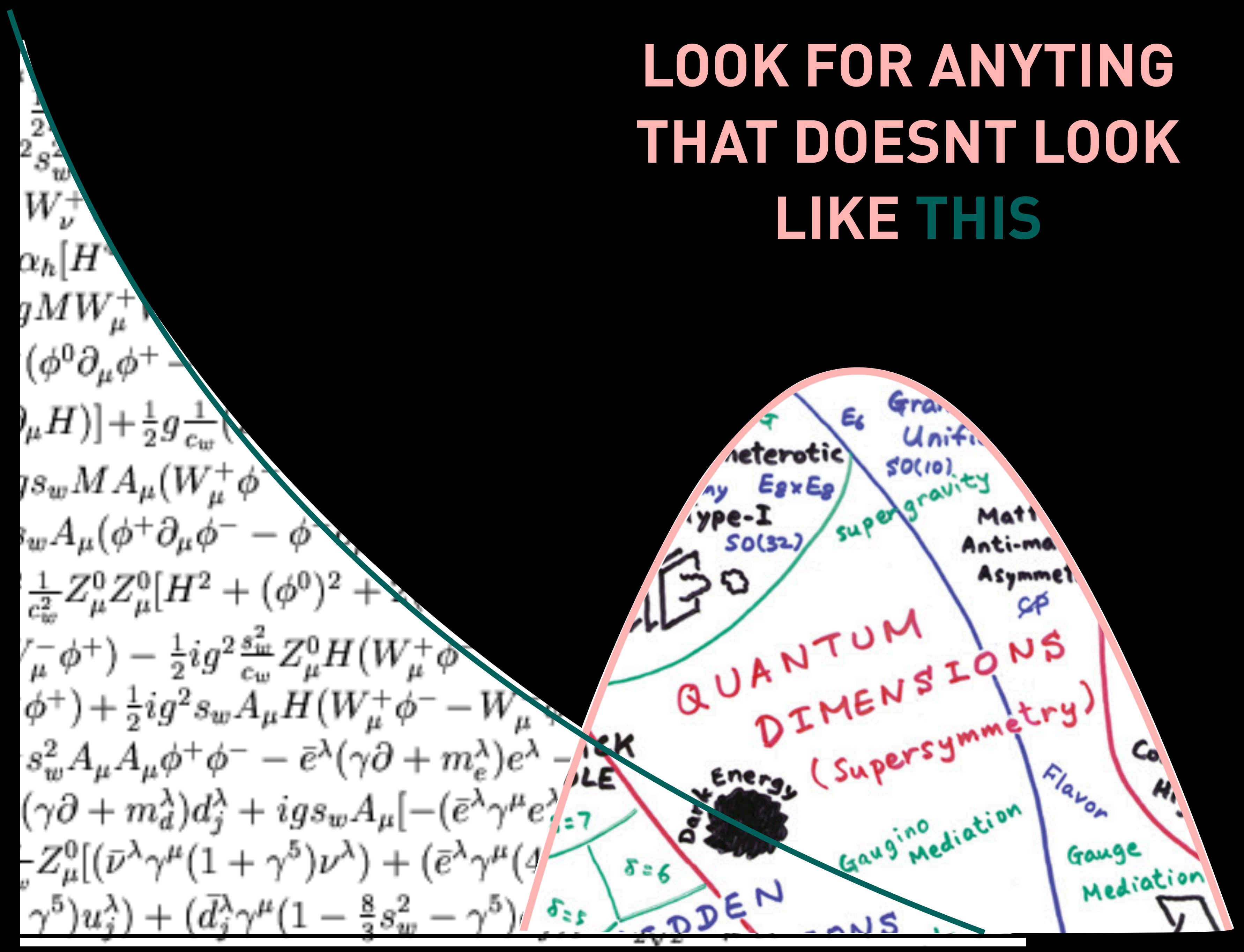
Let’s get back here!



Anomaly detection for New Physics searches

LEARN THIS FROM DATA

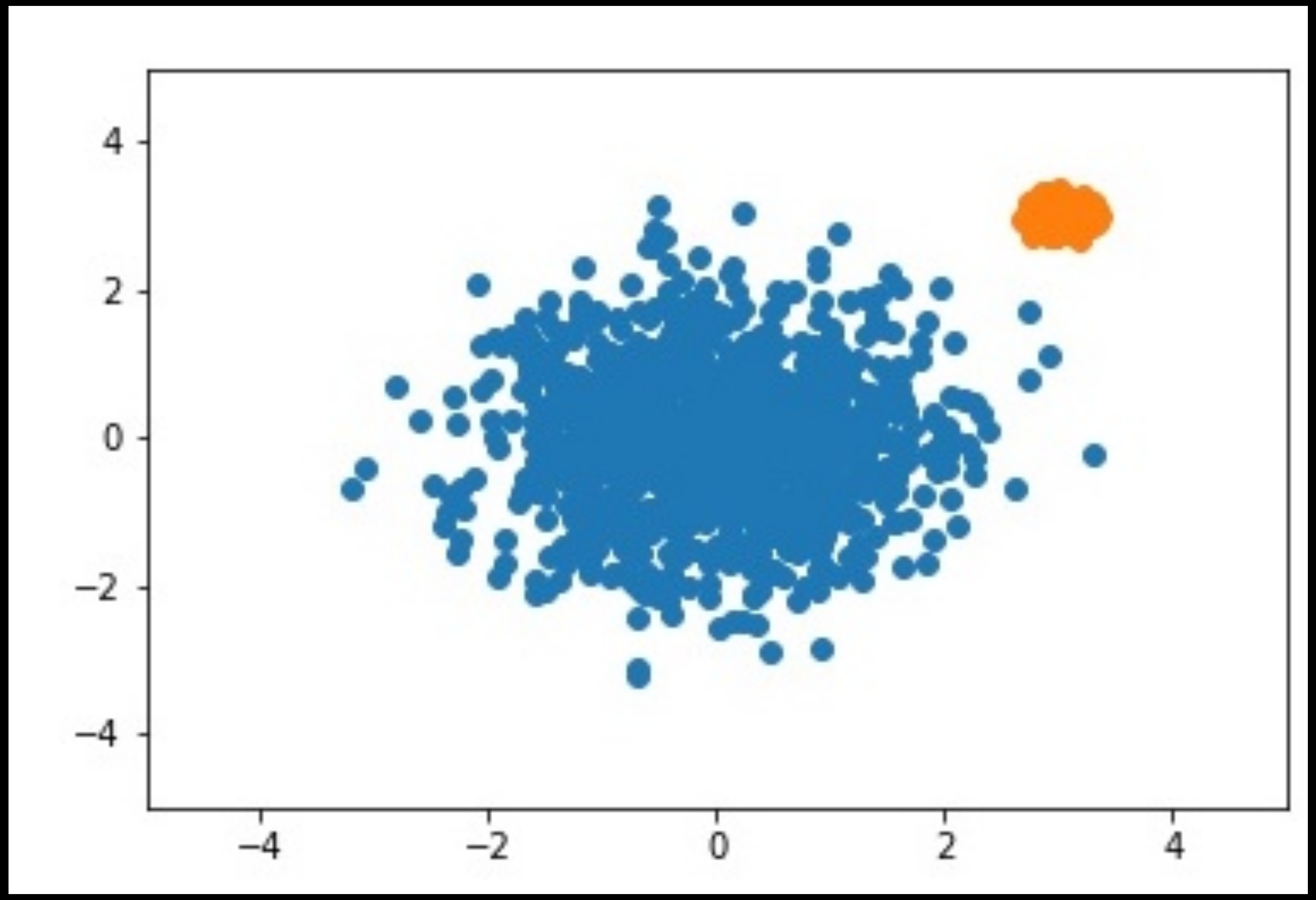
LOOK FOR ANYTING THAT DOESNT LOOK LIKE THIS



Types of anomaly detection

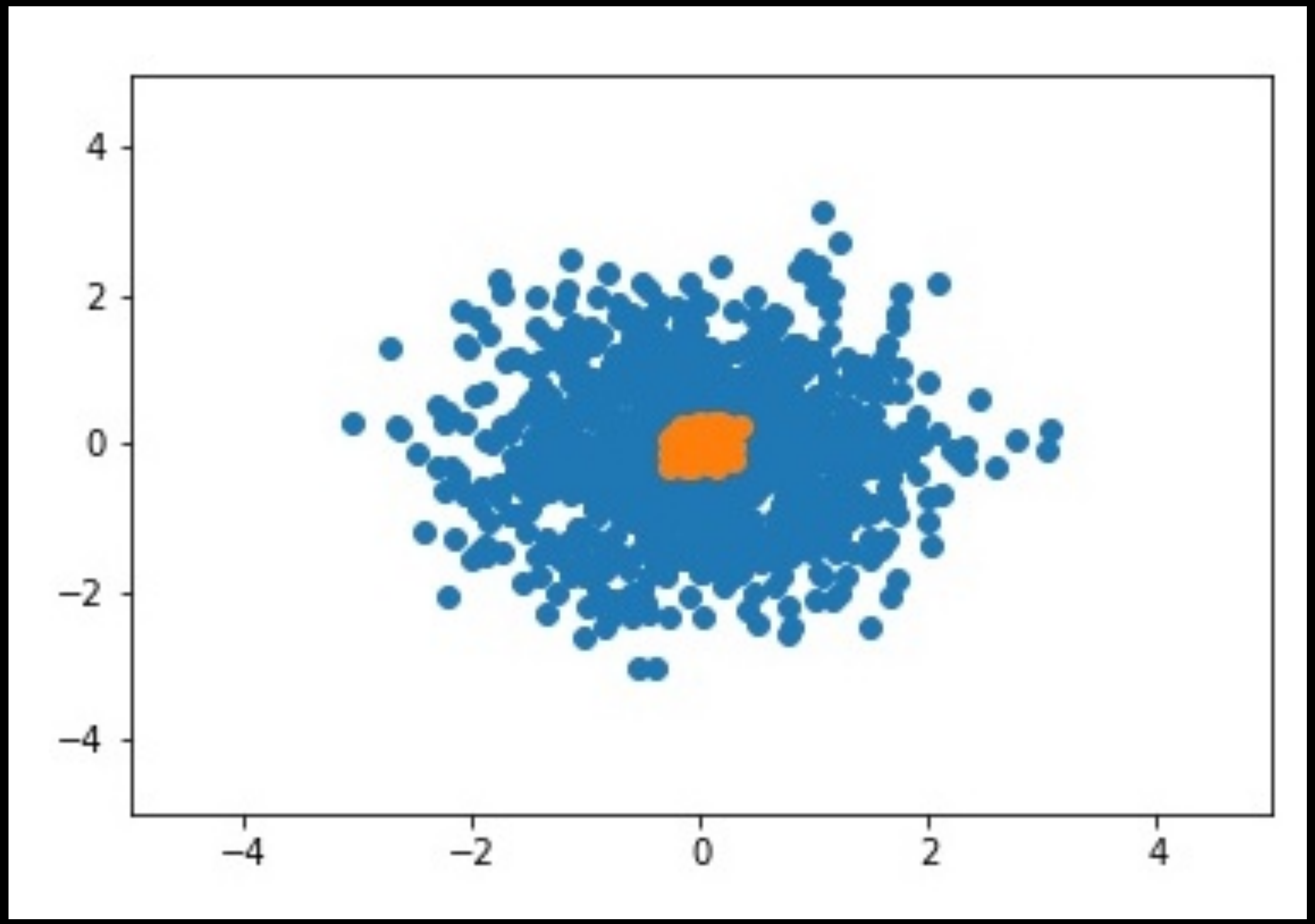
Outlier detection

Find (non-resonant) out-of-distribution datapoints



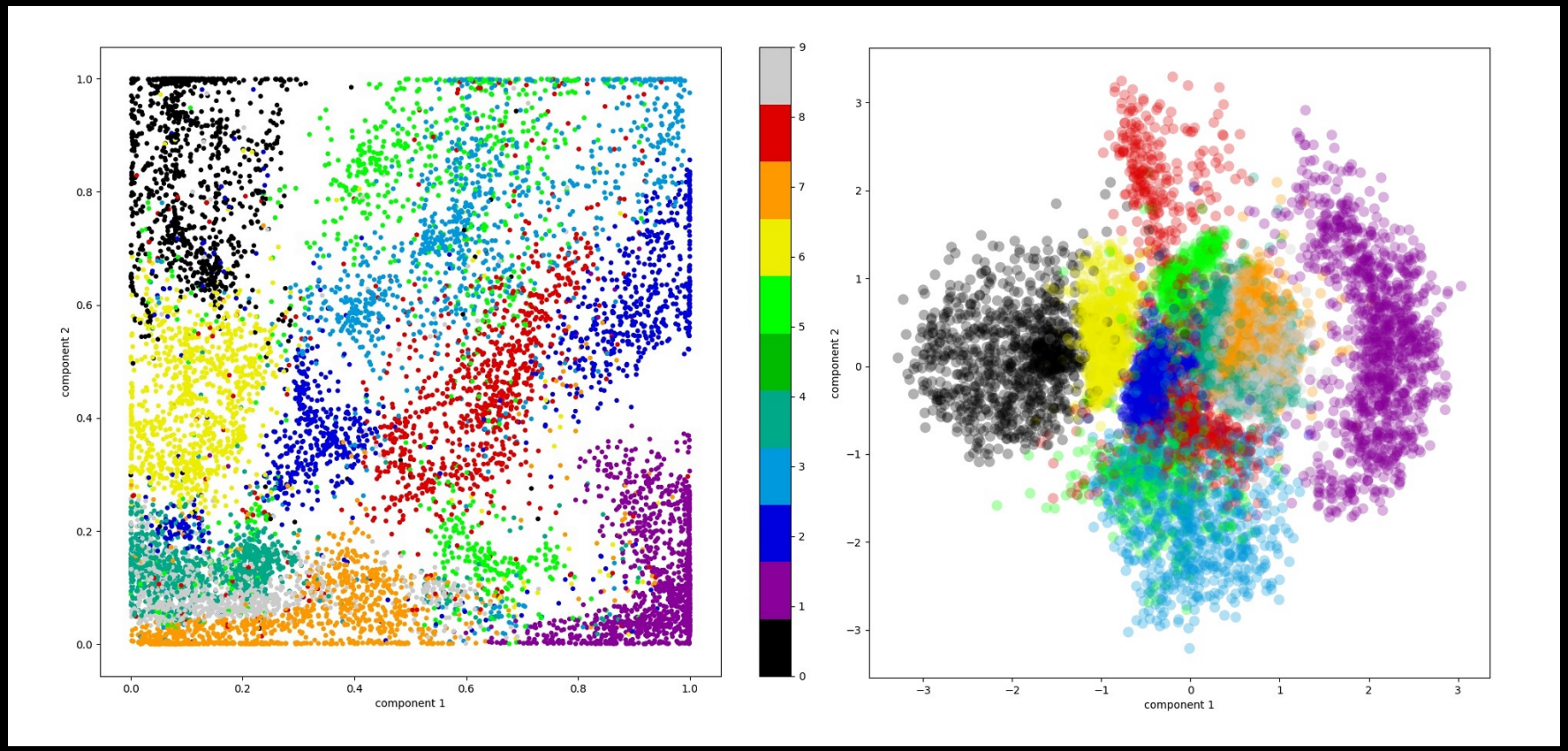
Detecting overdensities

Find (resonant) overdensities in distributions



Types of anomaly detection

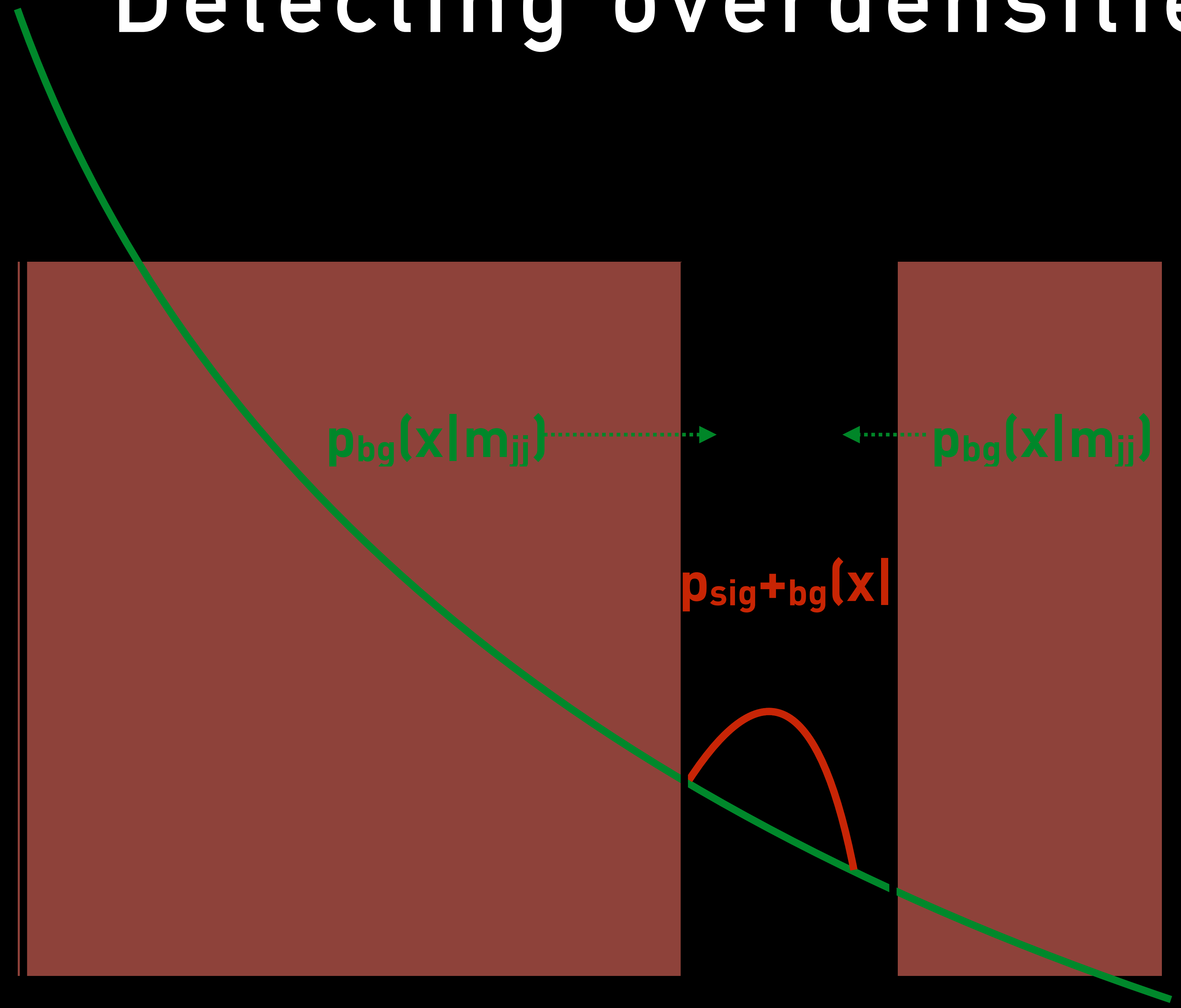
Outlier detection



Non-resonant, tail of distributions

- Often (variational) auto-encoders
- Useful for triggering/“selecting”!

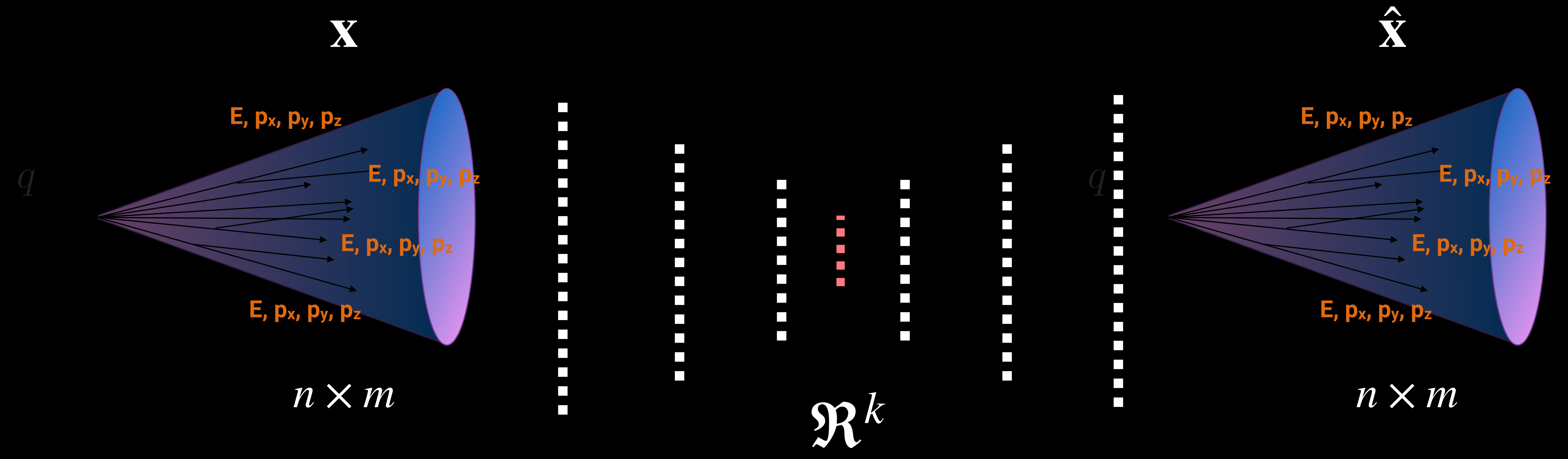
Detecting overdensities



Resonant, similar to a bump hunt

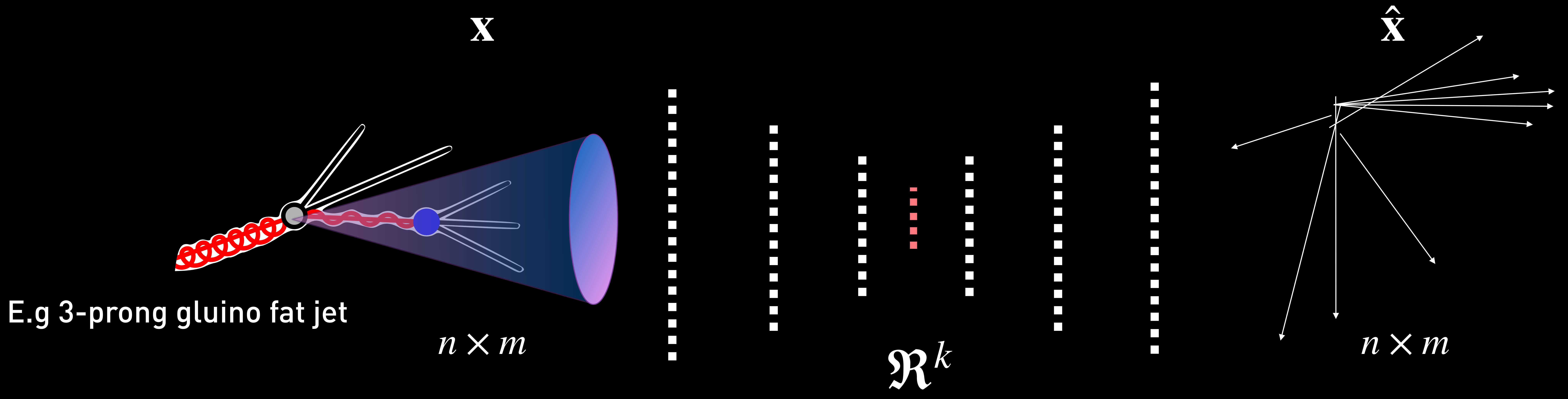
- Density estimation methods
- Useful for offline analysis

Outlier detection



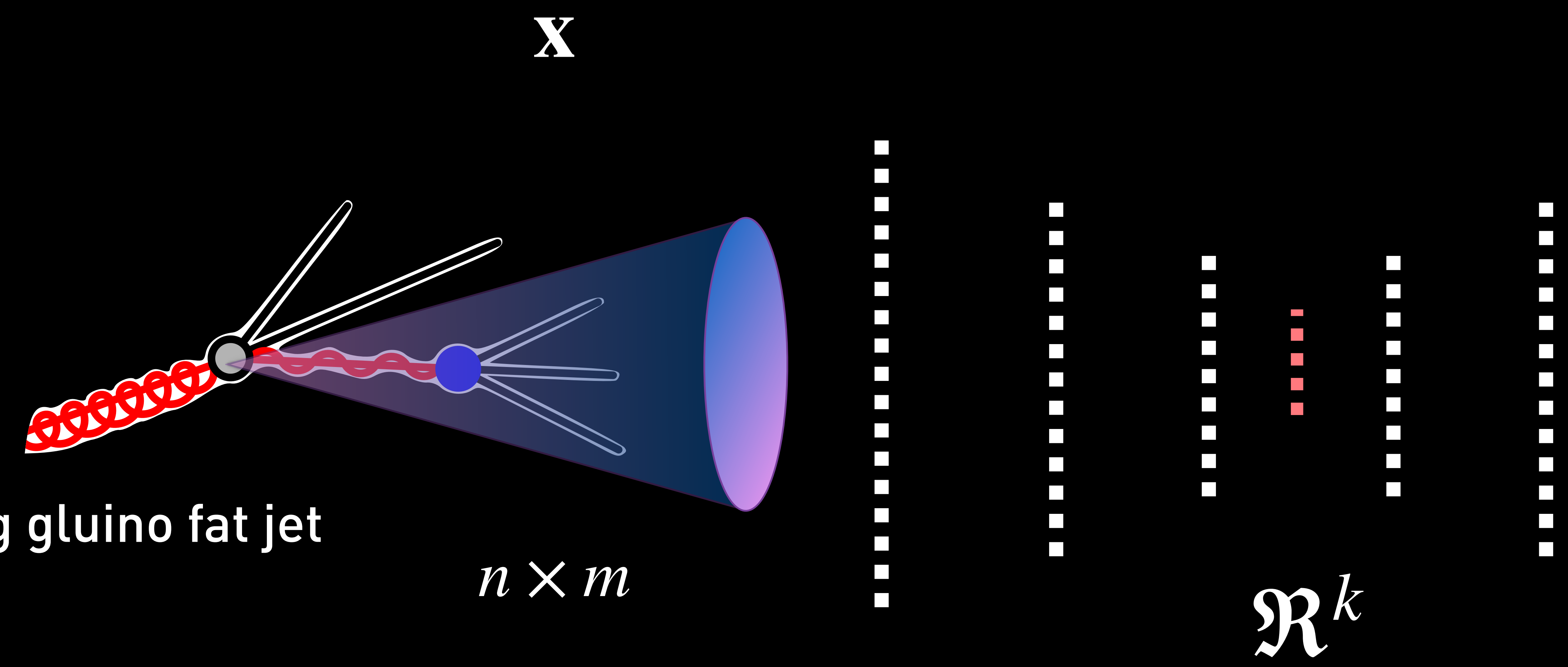
Compressed representation of x.
Latent space $\Re^k$, k < m×n
prevents memorisation of input, must learn

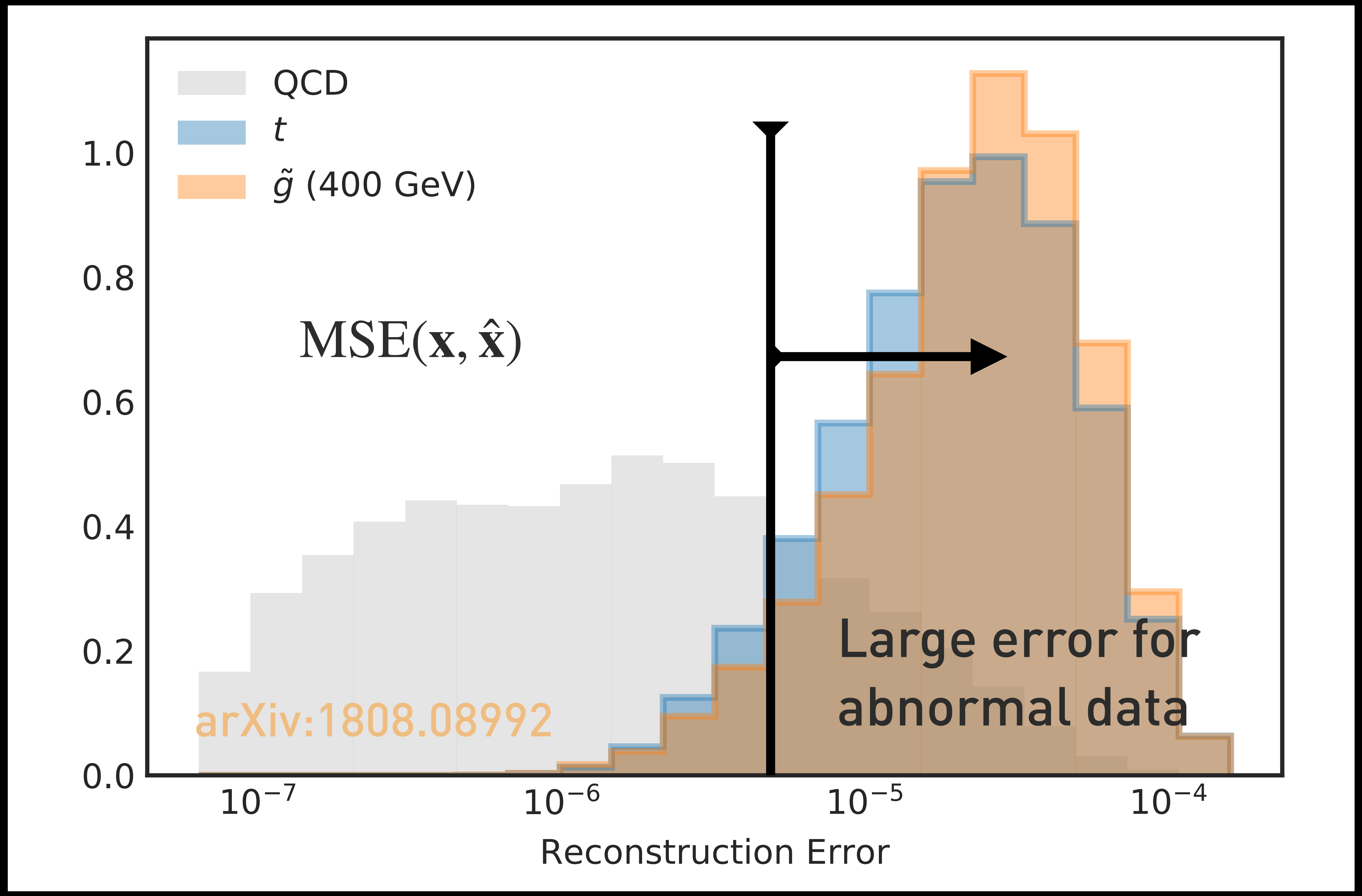
Outlier detection



$\mathscr{L}(\mathbf{x}, \hat{\mathbf{x}})$ is Mean Squared Error$(\mathbf{x}, \hat{\mathbf{x}})$, "high error events" proxy for "degree of abnormality"

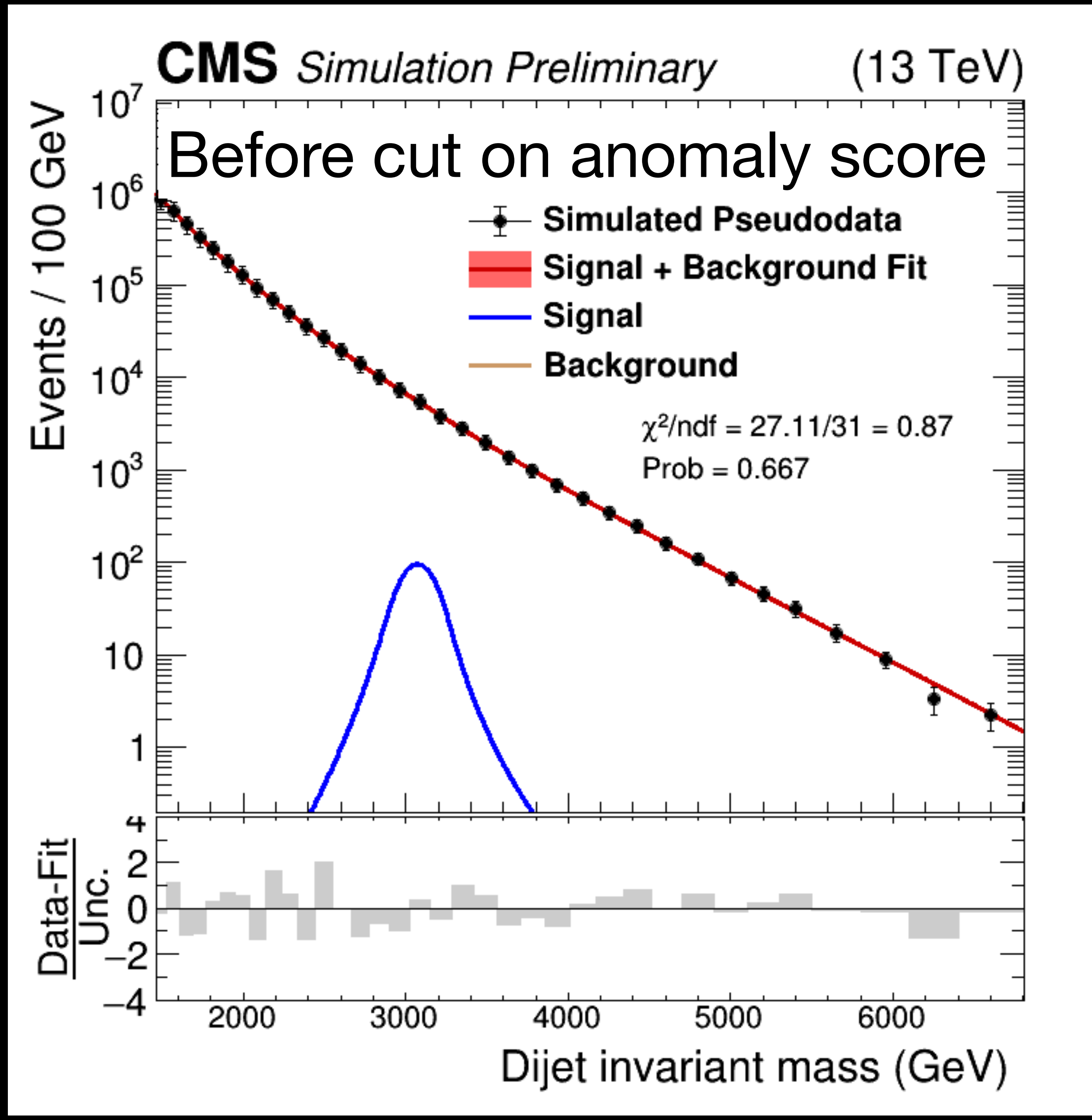
Outlier detection

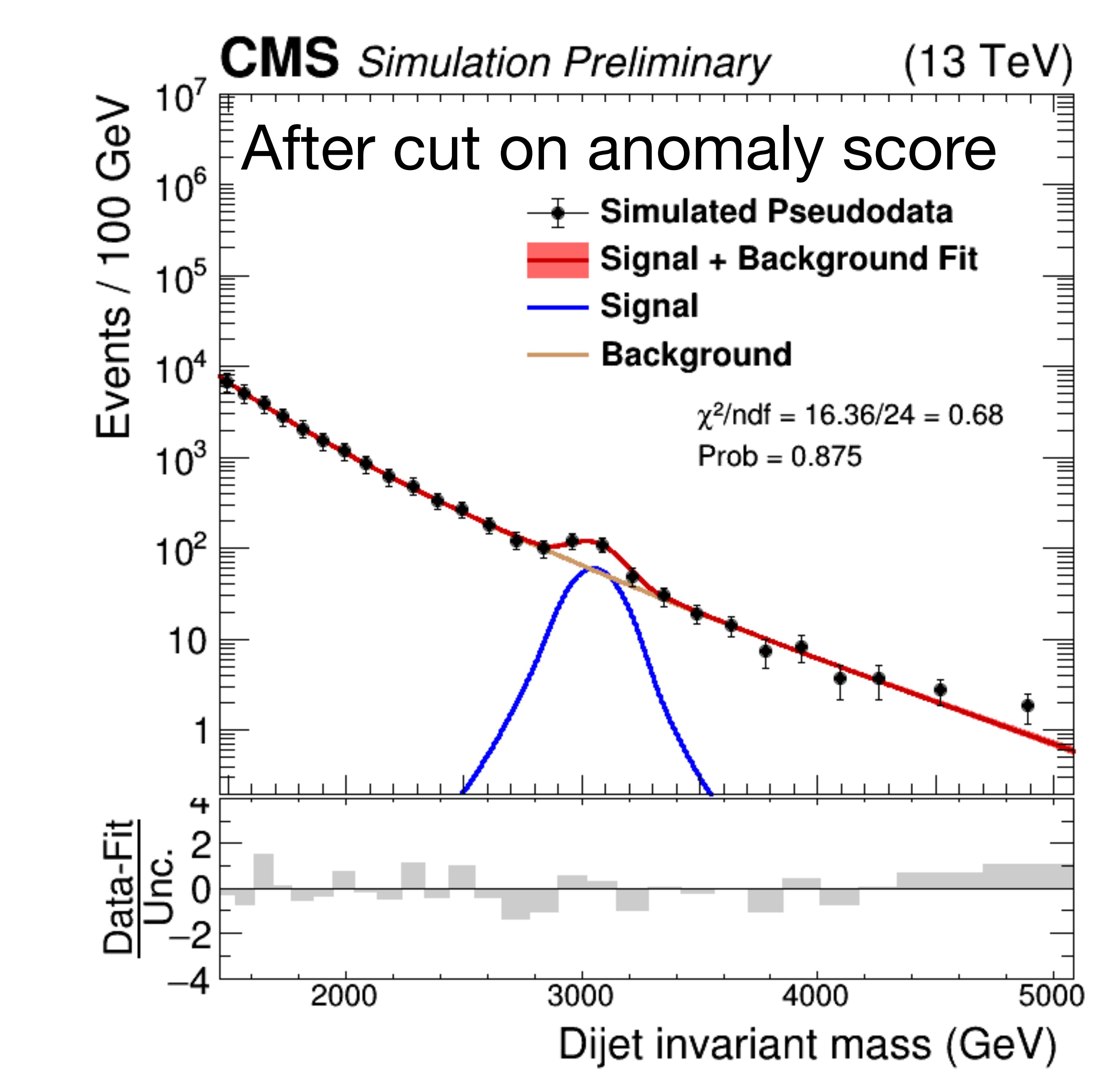




Outlier detection in analysis

E.g

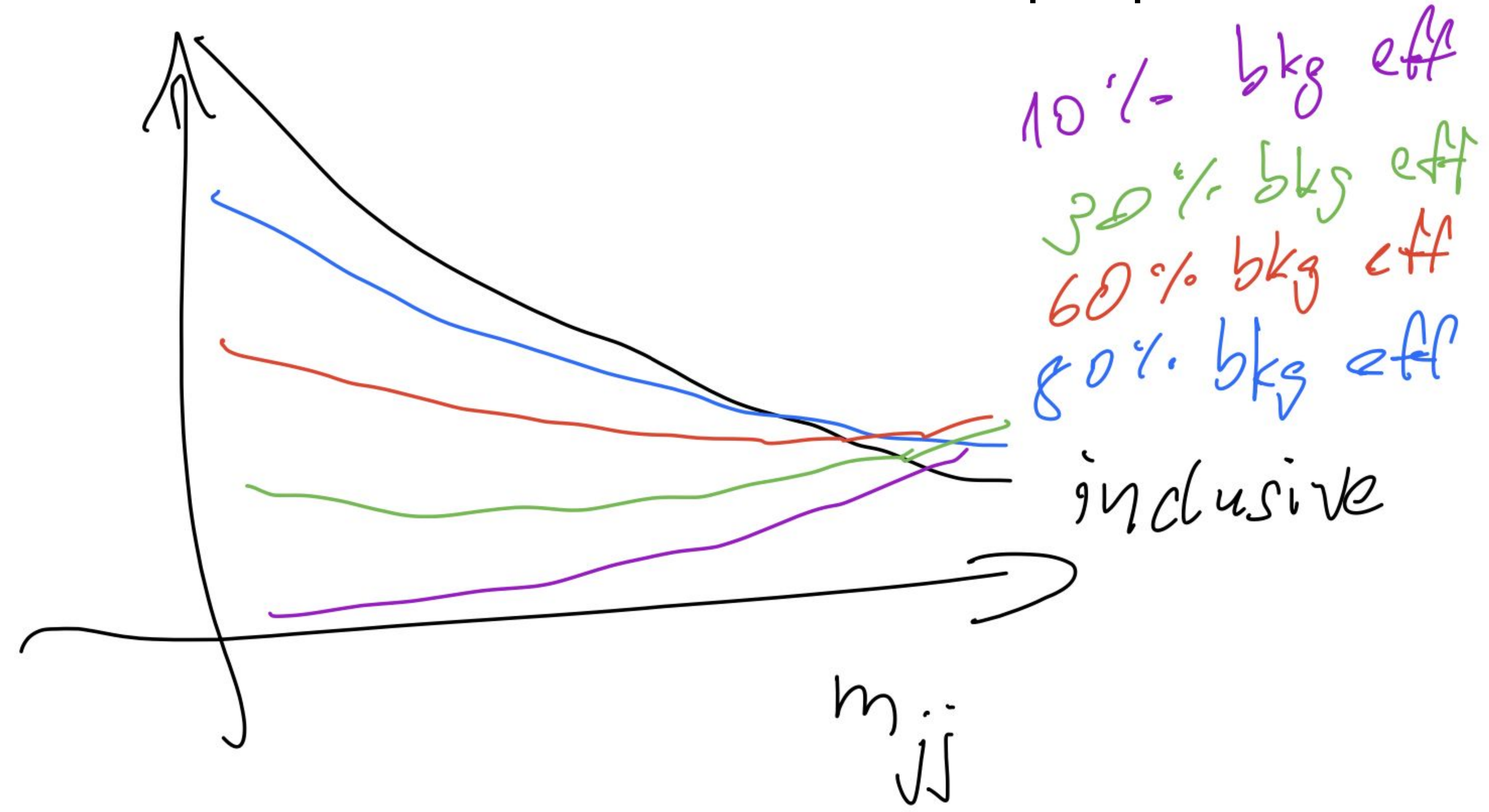




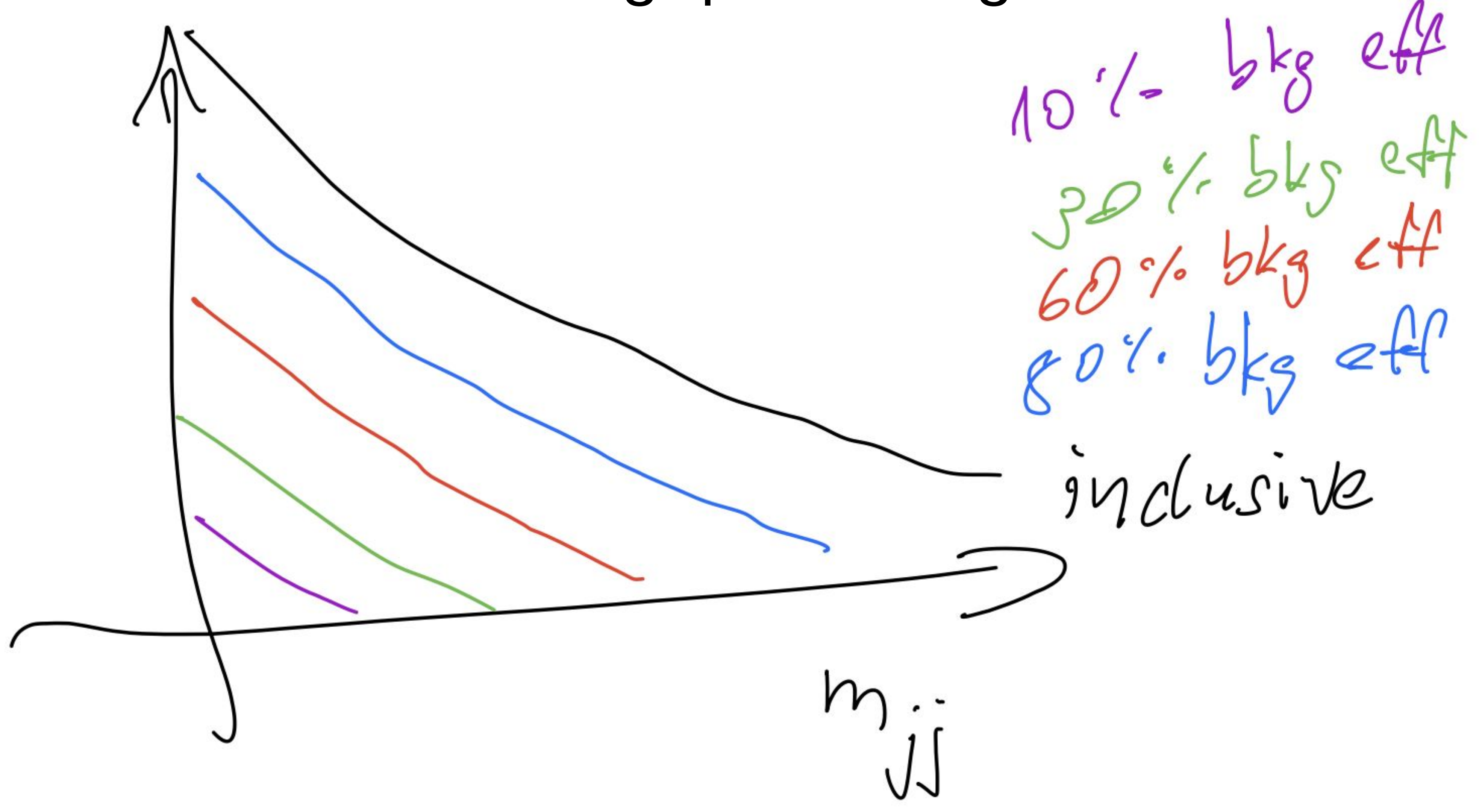
Outlier detection in analysis

E.g

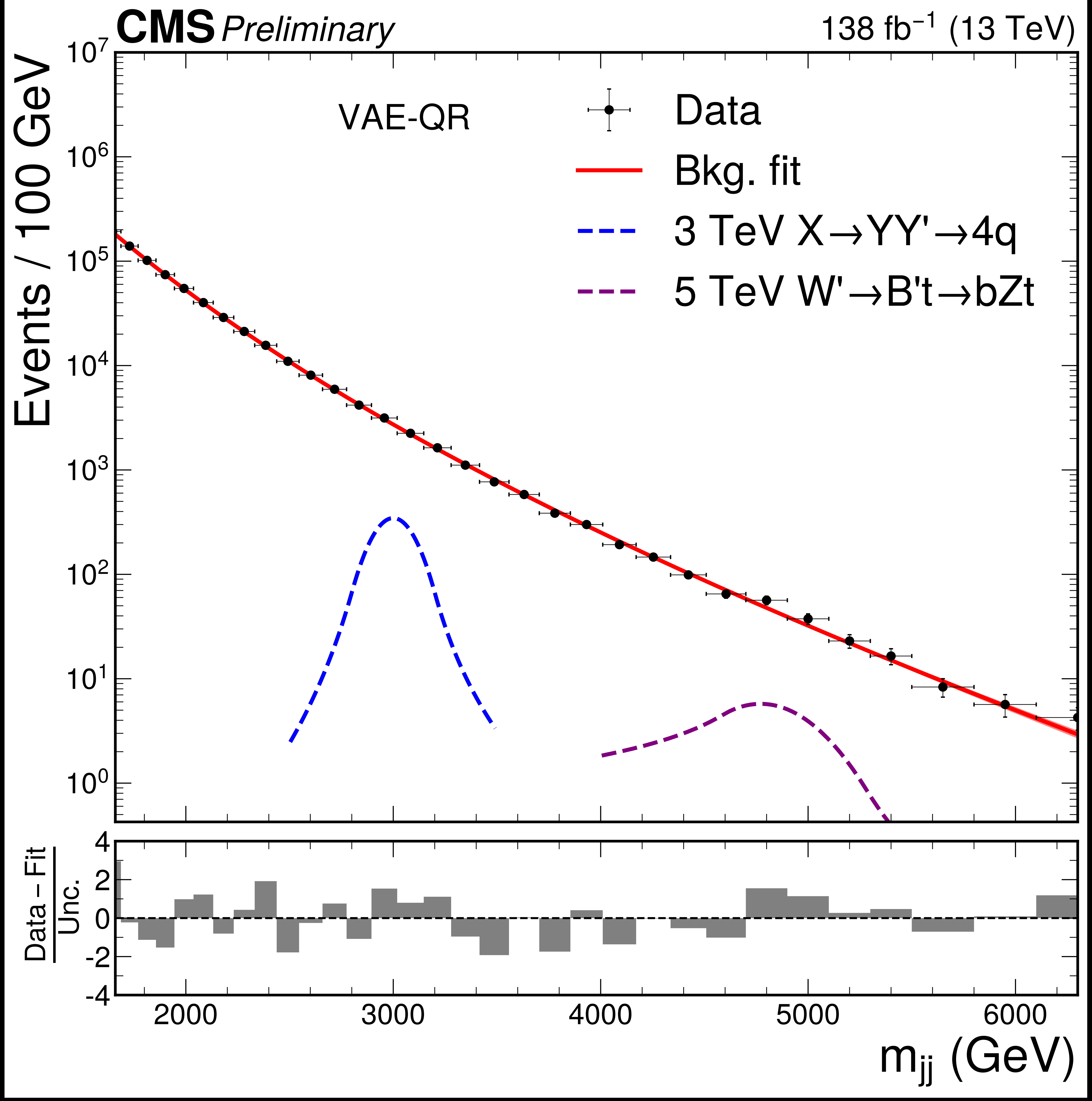
Careful! Cut on score can sculpt spectrum



Can fix using quantile regression

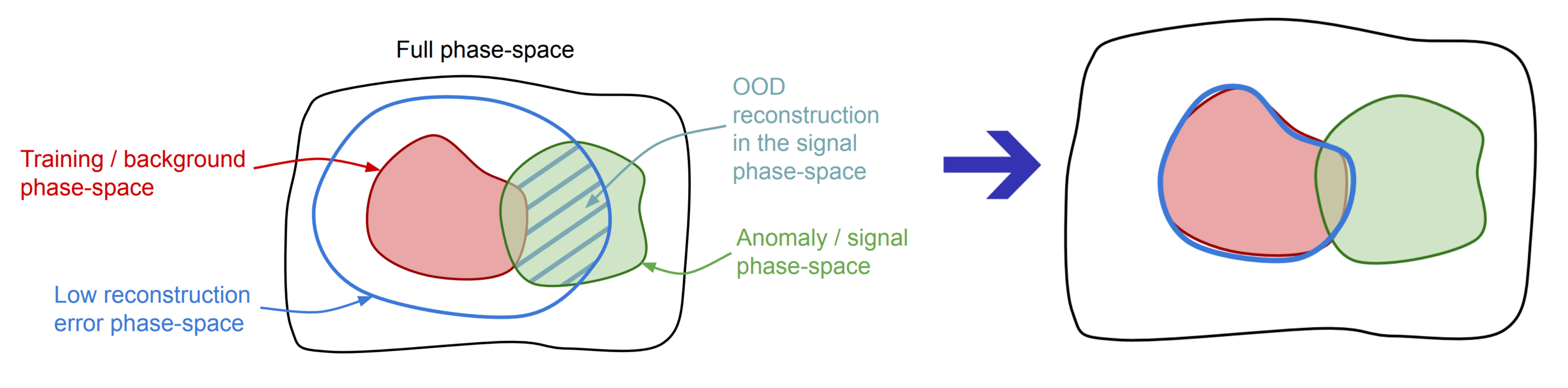


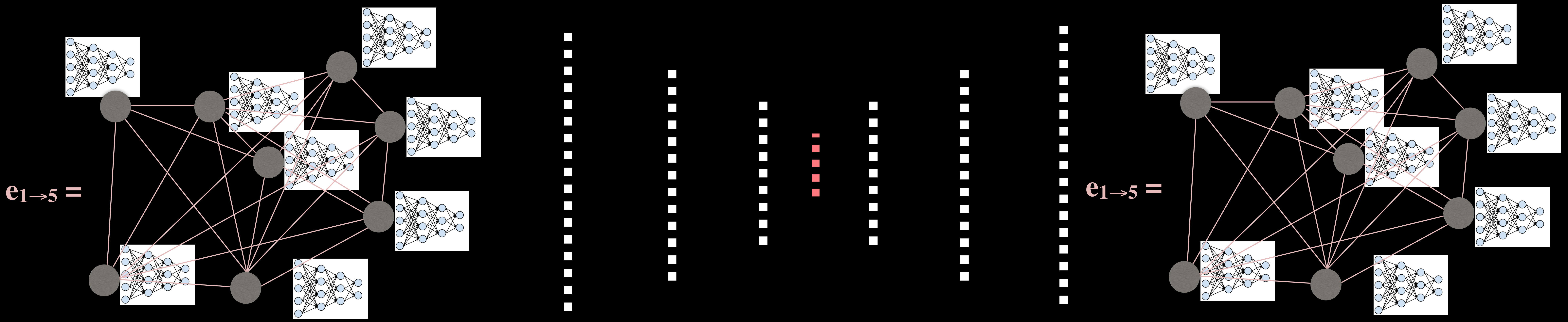
Outlier detection in analysis



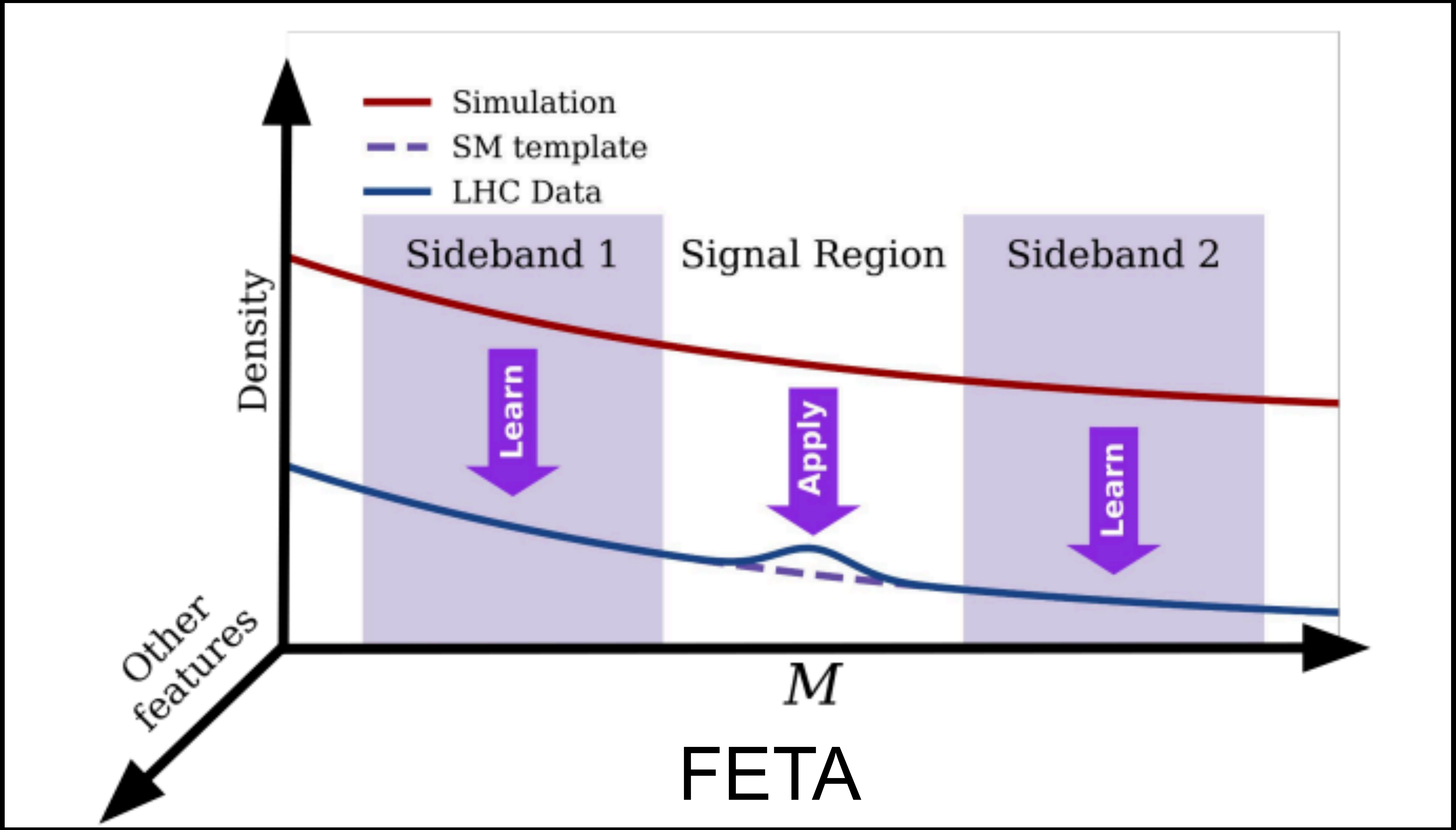
Example for semi-visible jets

Normalized autoencoders : Lund Graph autoencoders

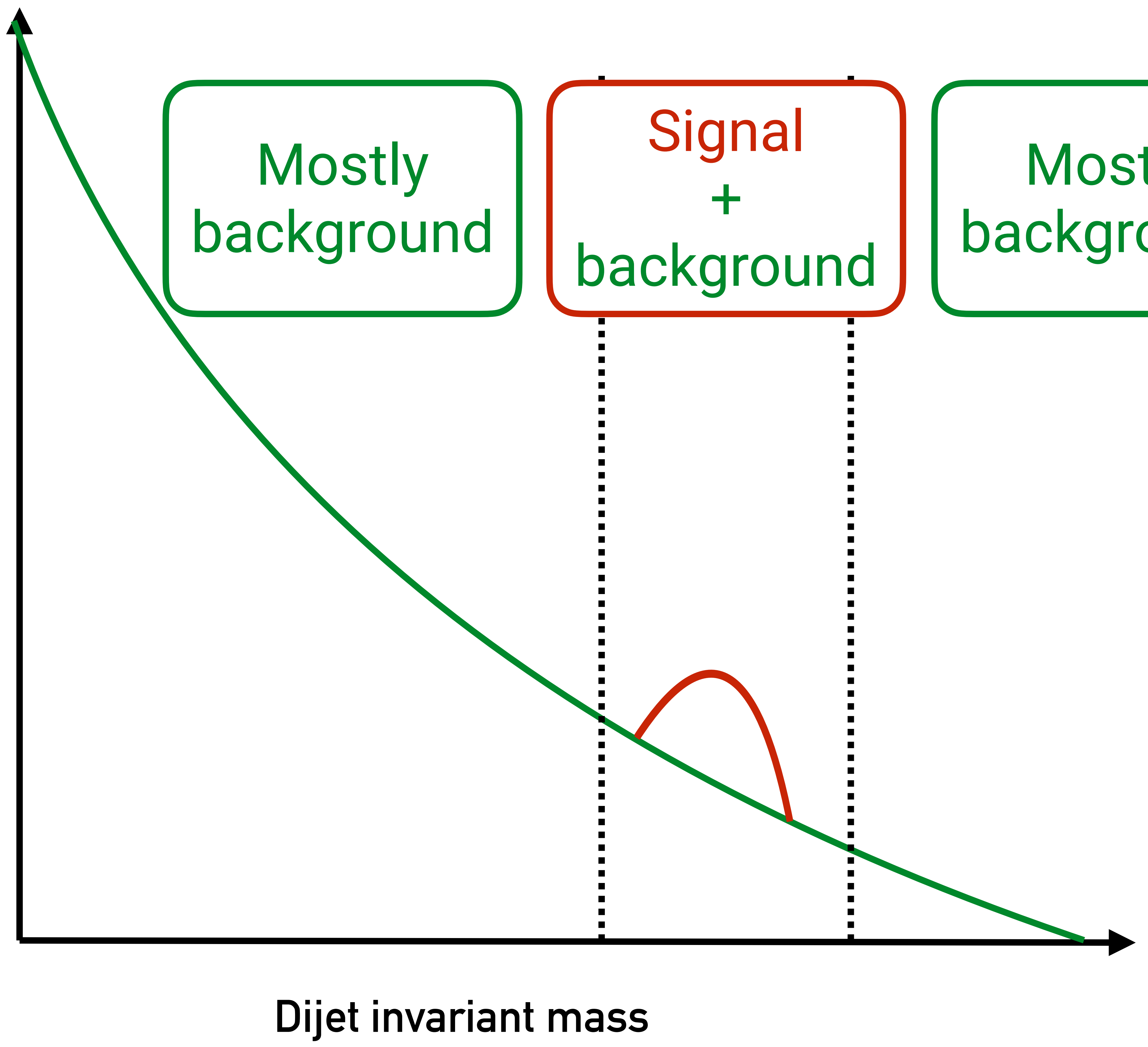




Finding overdensities



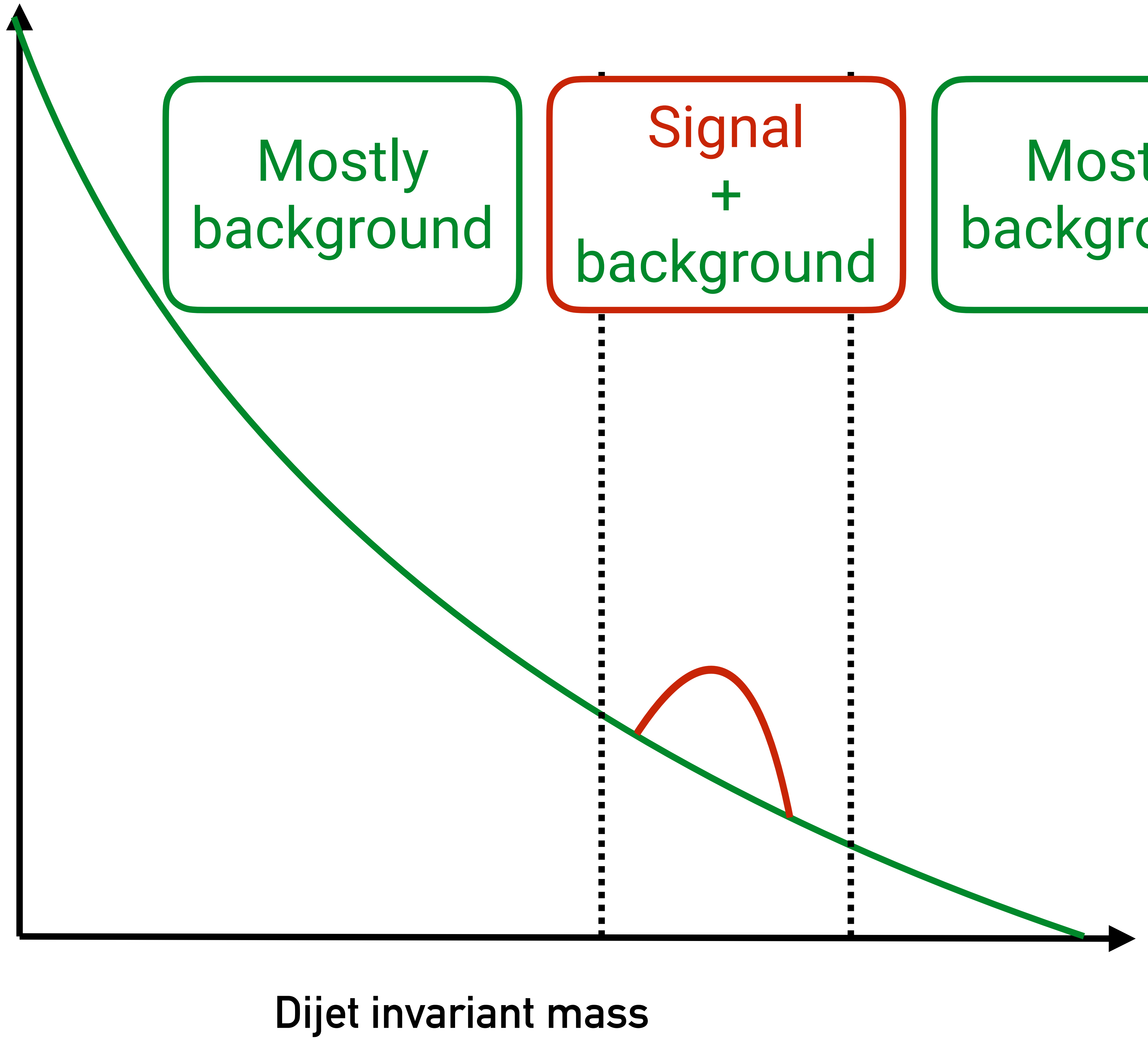
Weak classification without labels (CWoLa)

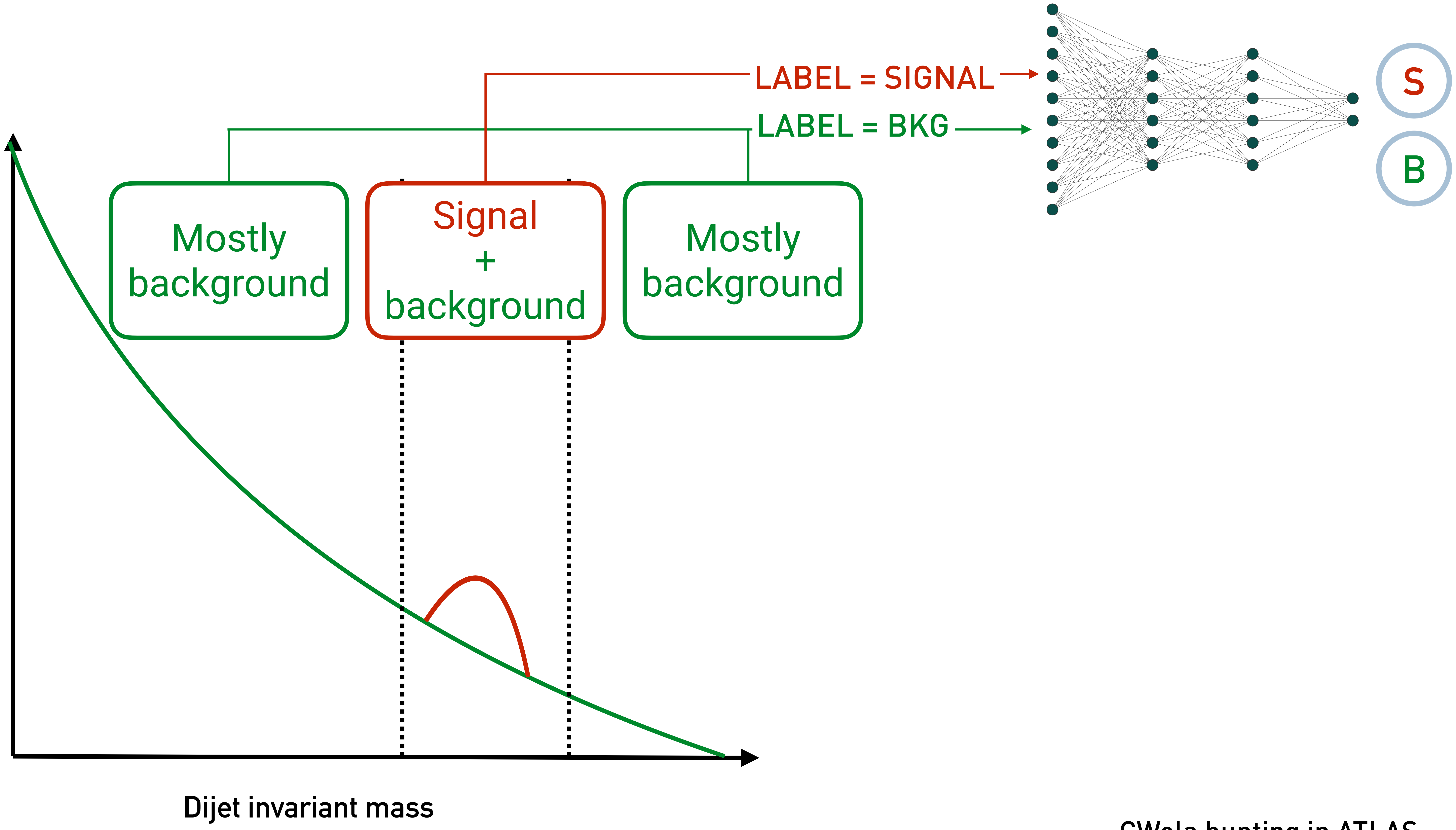


Weak classification without labels (CWoLa)

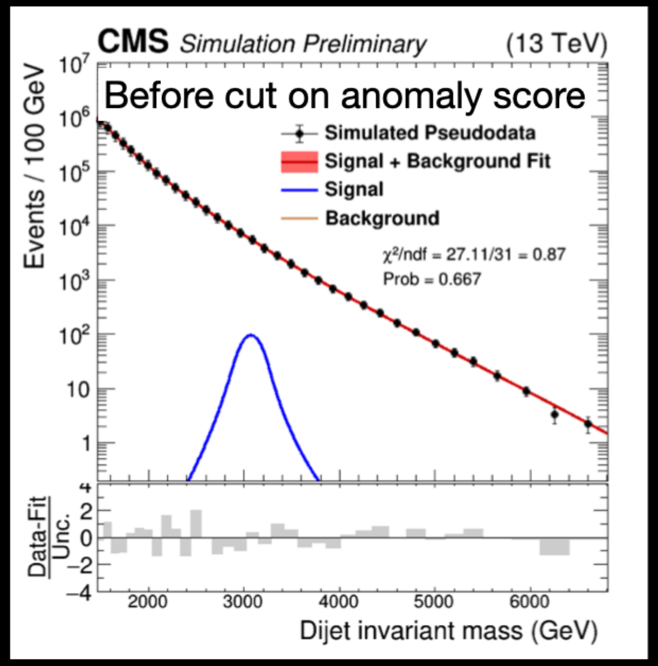
Classification Without Labels

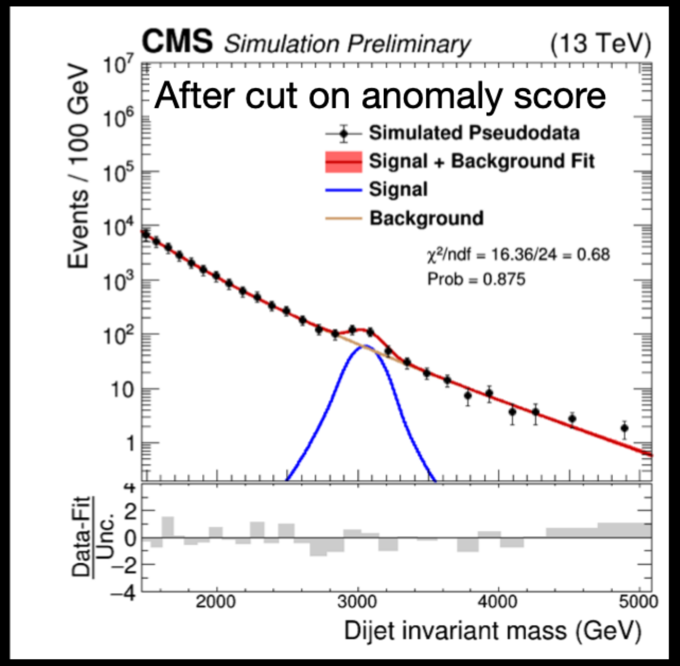
- Lemma: "Given mixed S+B samples SB and SR, optimal classifier trained to distinguish SB and SR is also optimal for distinguishing S from B"



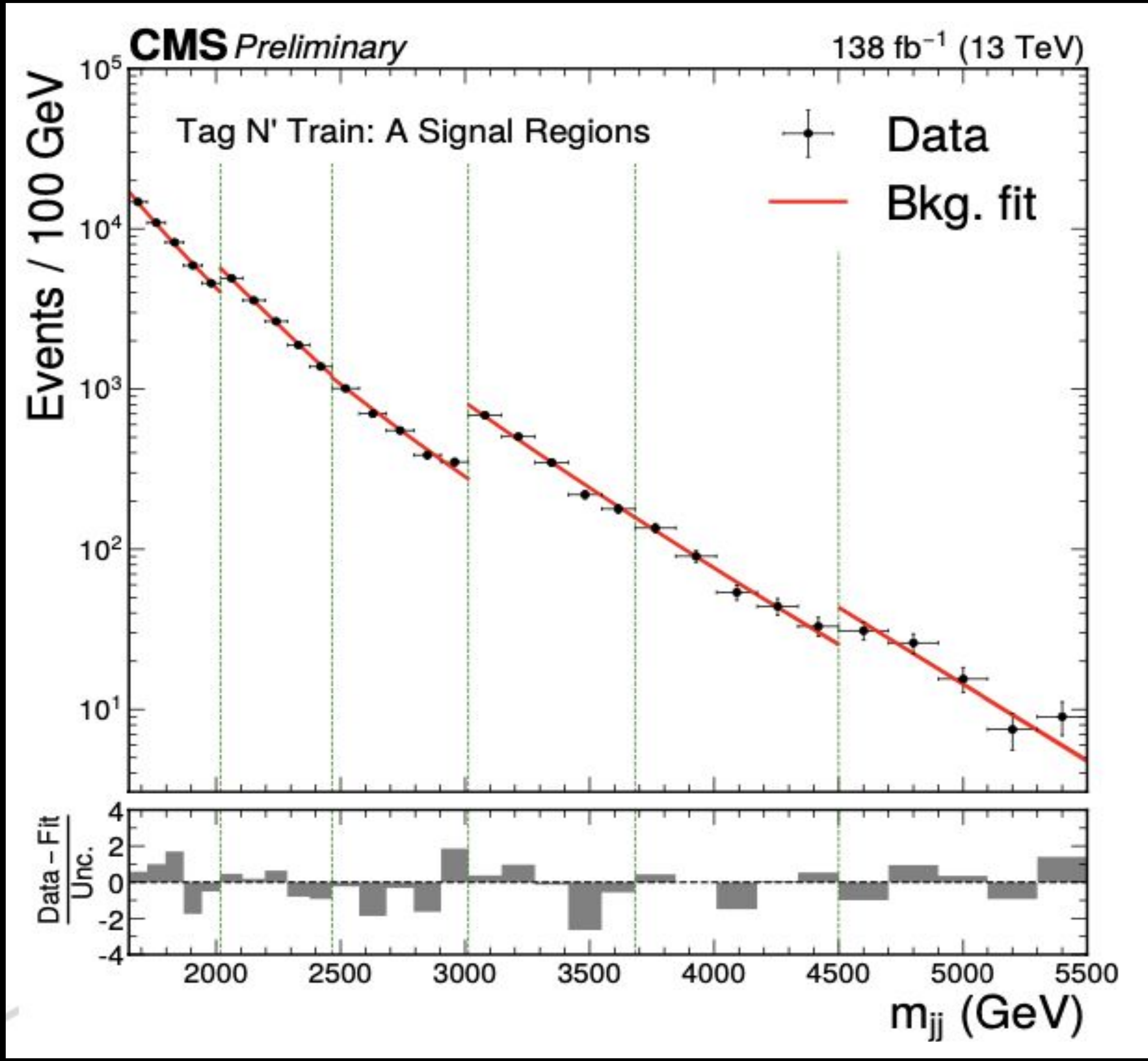


CWola hunting in ATLAS


CMS Simulation Preliminary
(13 TeV)
Before cut on anomaly score
Simulated Pseudodata
Signal + Background Fit
Signal
Background
Events / 100 GeV
Data-Fit / Unc.
Dijet invariant mass (GeV)


CMS Simulation Preliminary
(13 TeV)
After cut on anomaly score
Simulated Pseudodata
Signal + Background Fit
Signal
Background
Events / 100 GeV
Data-Fit / Unc.
Dijet invariant mass (GeV)

E.g


CMS Preliminary
138 fb^{-1} (13 TeV)
Tag N' Train: A Signal Regions
Data
Bkg. fit
Events / 100 GeV
Data – Fit / Unc.
m_{jj} (GeV)

DNN likelihood

Alternative approach: End-to-end DNN search

- How do we get around defining a signal hypothesis?
- What is alternate hypothesis to test reference?

Idea: Assume alternate model n(x|w) can be parametrised in terms of reference model n(x|R)

$$n(x\,|\,\vec{w}) = n(x\,|\,R)e^{f(x;\vec{w})}$$ ← Set of real functions

- Let DNN parametrise alternative model

$$f(x;\vec{w}) = NN$$

DNN likelihood

Alternative approach: End-to-end DNN search

- How do we get around defining a signal hypothesis?
- What is alternate hypothesis to test reference?

Idea: Assume alternate model n(x|w) can be parametrised in terms of reference model n(x|R)

$$n(x \,|\, \vec{w}) = n(x \,|\, R) e^{f(x;\vec{w})}$$ ← Set of real functions

- Let DNN parametrise alternative model

$$f(x; \vec{w}) = NN$$

- Formulate loss as log likelihood.
  → Trained DNN **is** the maximum likelihood fit to data and reference log-ratio
  → best approximate of true data distribution

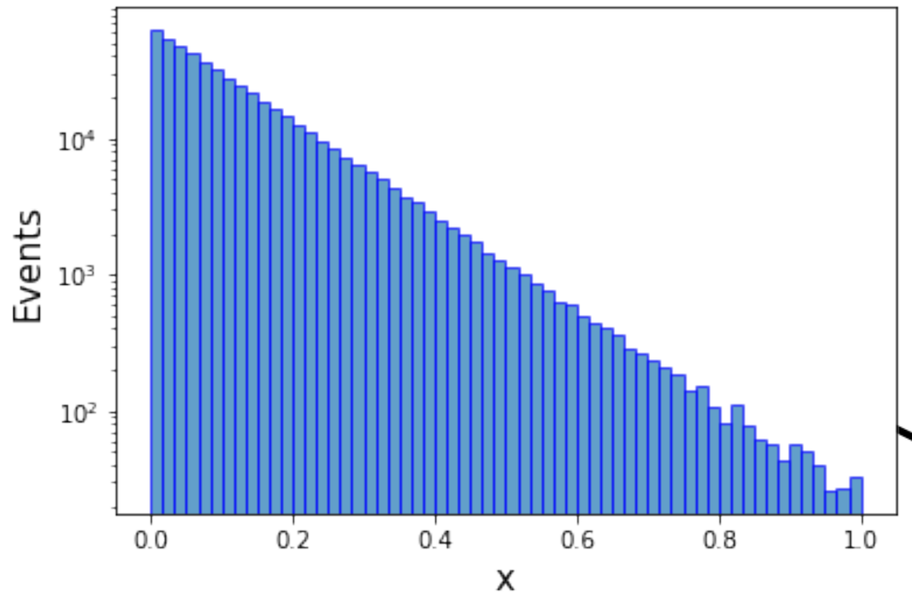
$$f(x, \widehat{\mathbf{w}}) \simeq \log\left[\frac{n(x|\mathrm{T})}{n(x|\mathrm{R})}\right]$$

← True underlying data distribution
← MC distribution

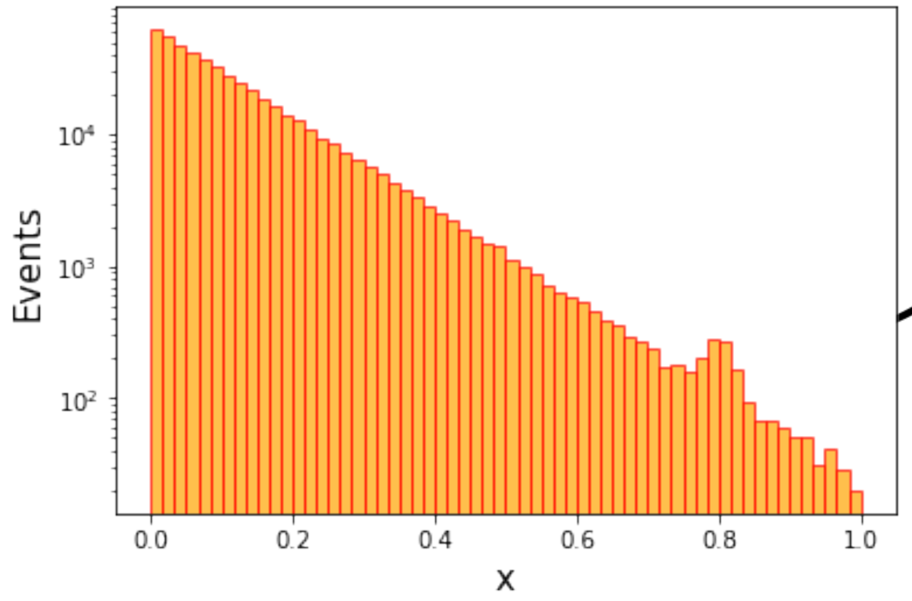
INPUTS

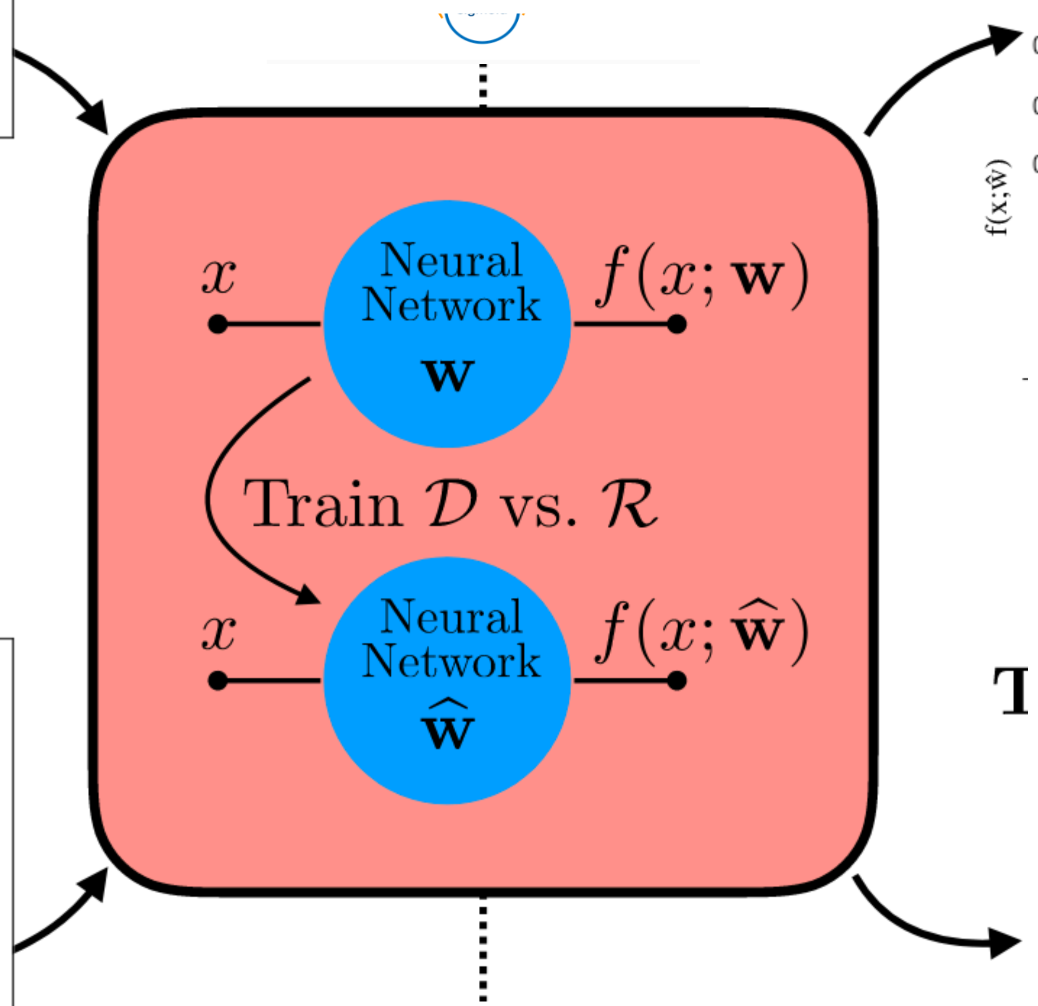
- any high level features

QCD MC R



CMS DATA D





$$f(x,\widehat{\mathbf{w}}) \simeq \log\left[\frac{n(x|\mathrm{T})}{n(x|\mathrm{R})}\right]$$

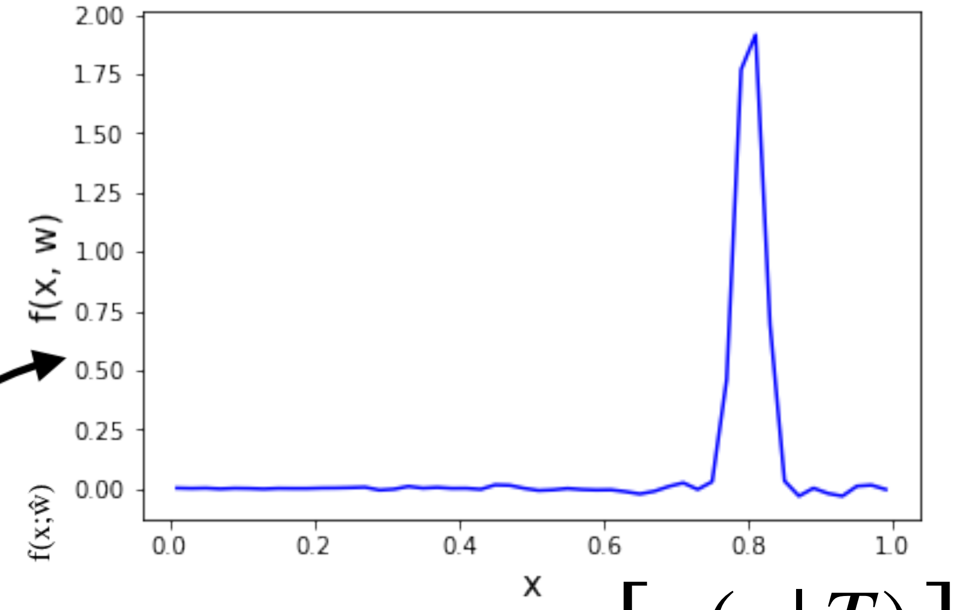
← True underlying data distribution

← MC distribution

OUTPUTS

-tobs and f(x; ŵ)

1) Best fit log ratio of data and MC PDFs



$$f(x,\hat{w}) \simeq \log\left[\frac{n(x\,|\,T)}{n(x\,|\,R)}\right]$$

2) test-statistic on data sample t_{obs}

$$t(\mathcal{D}) = -2\,\underset{\{\mathbf{w}\}}{\mathrm{Min}}\, L[f]$$

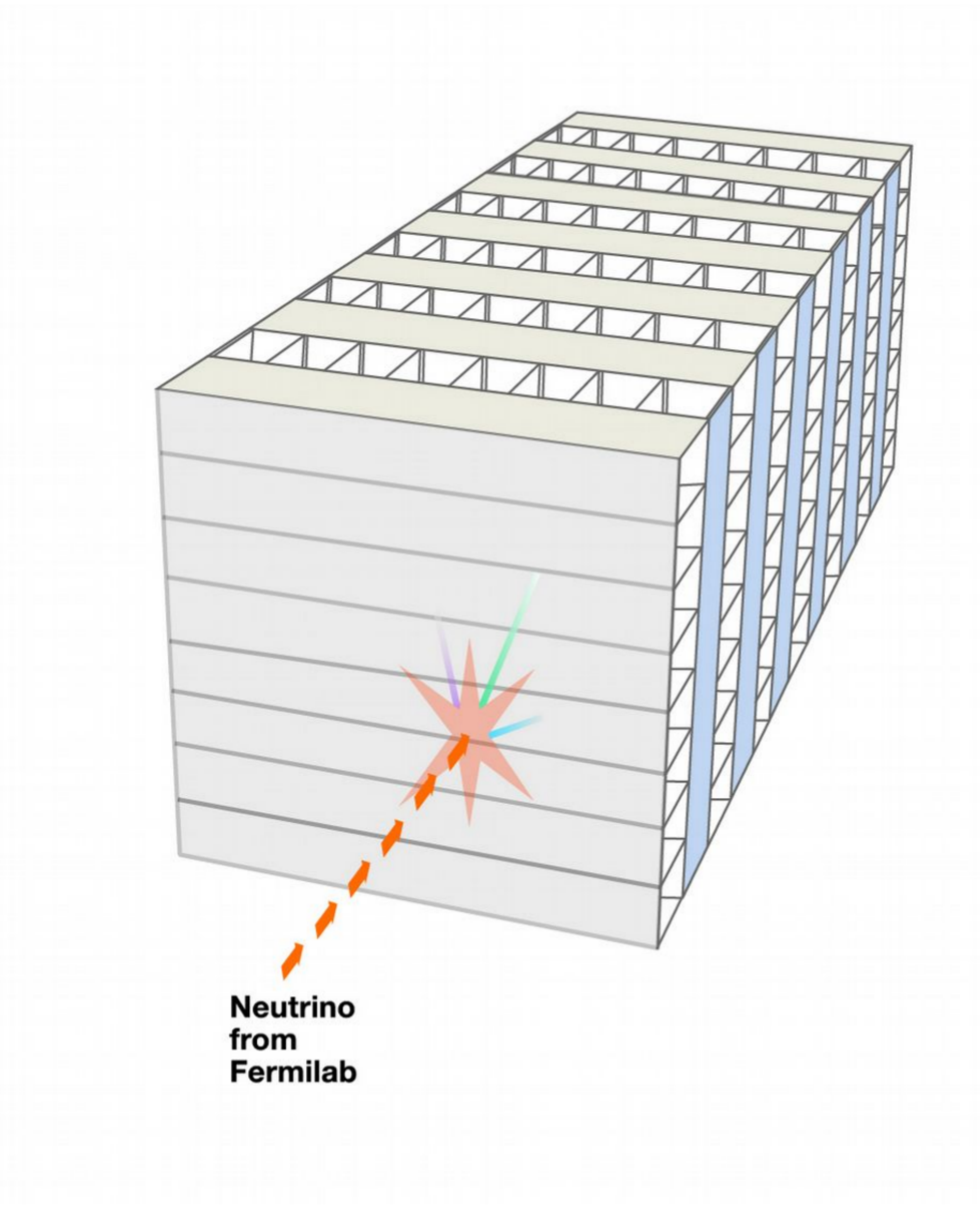
← **DNN loss function!**

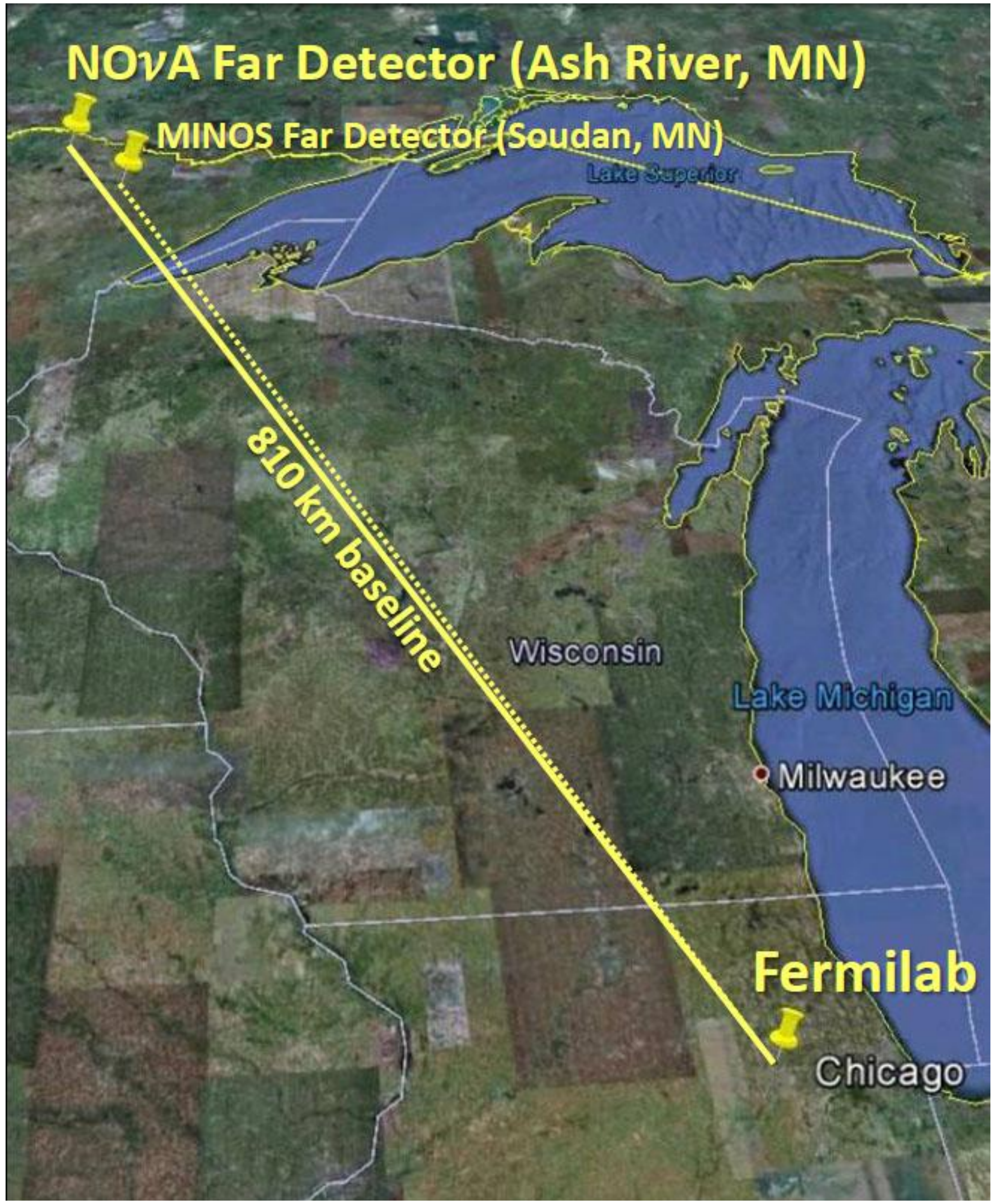
Can be used to build hypothesis test + p-value

Data → toys under R, repeat

Hybrid approaches - NoVa

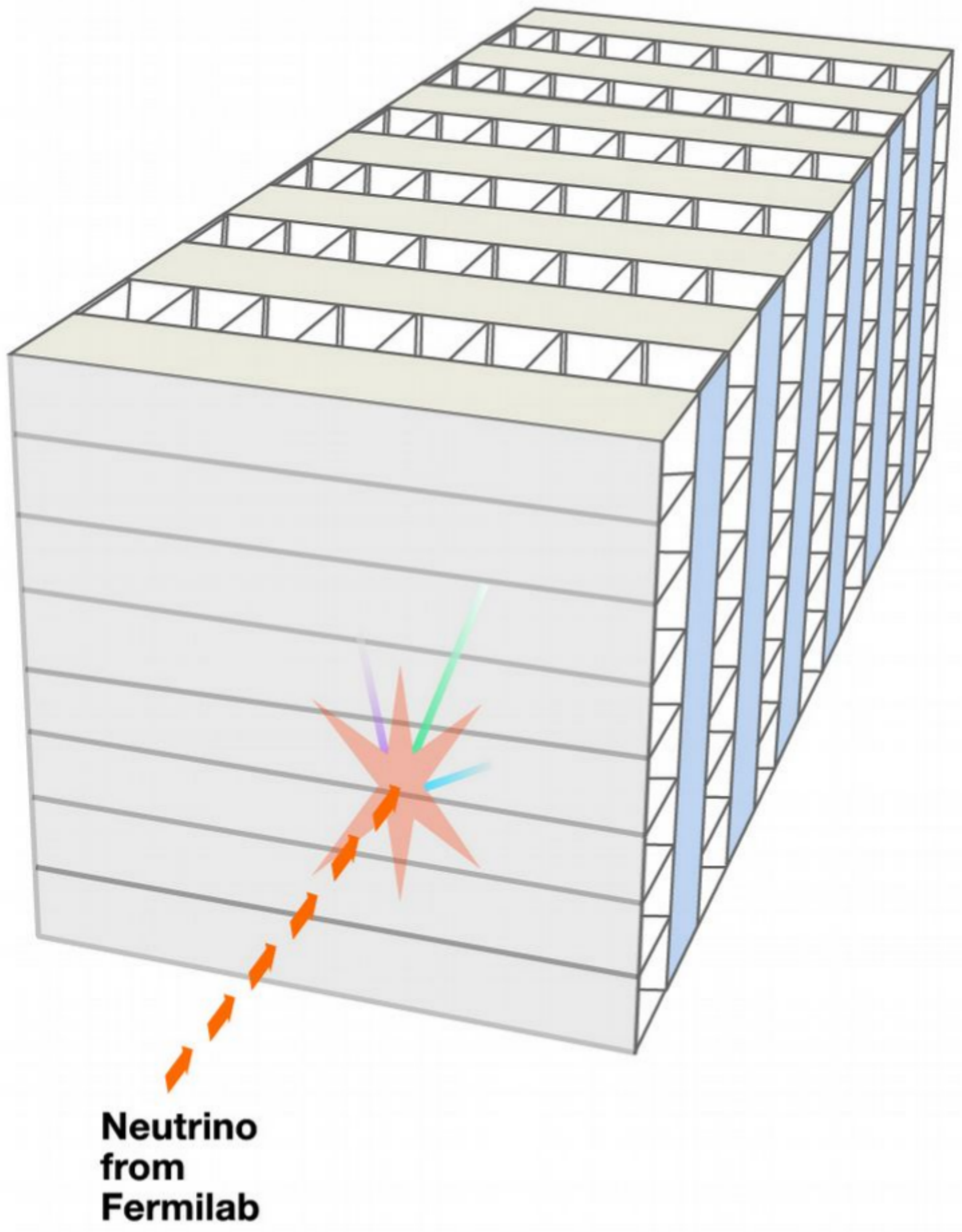
Aurisano et al
K. Sachdev

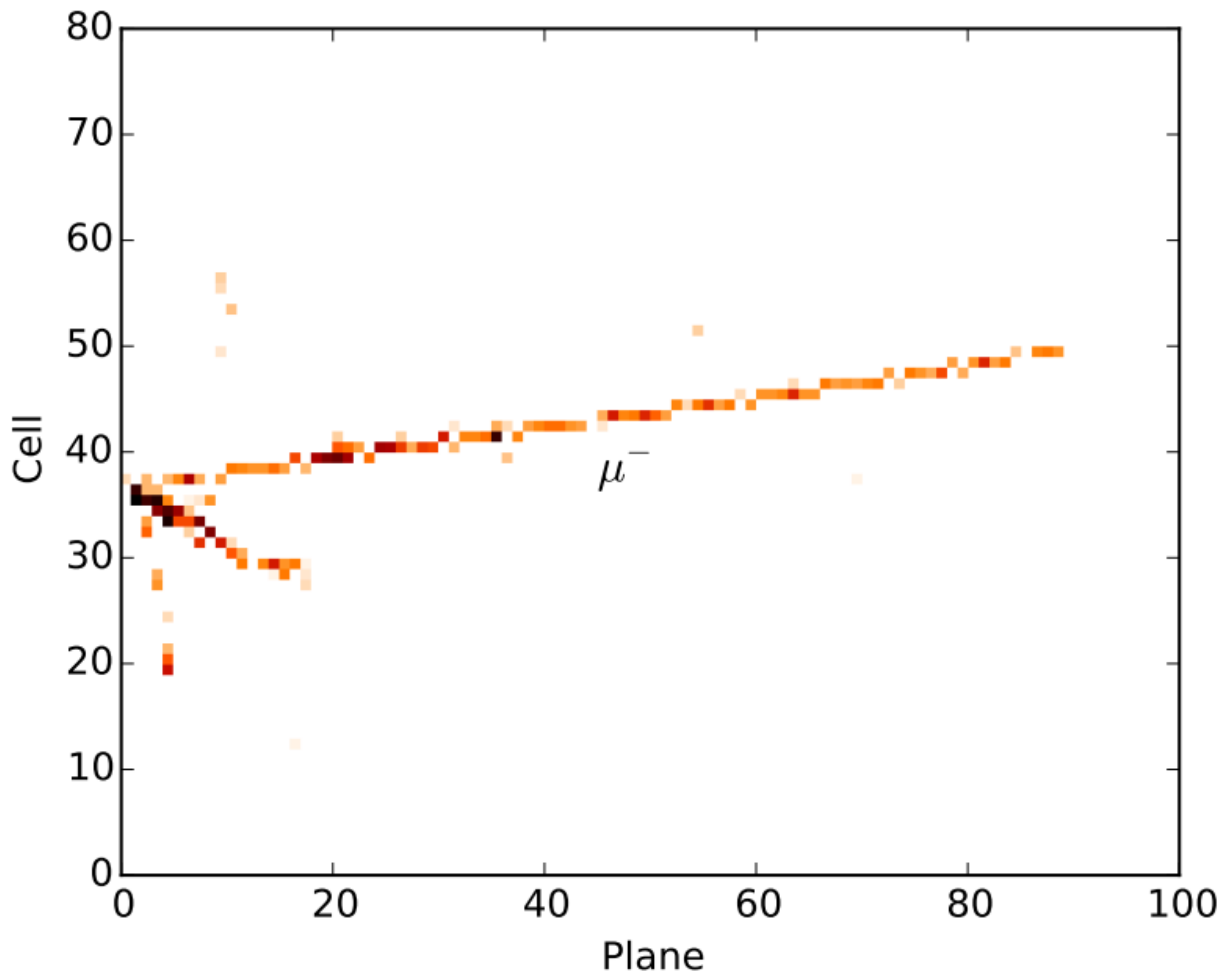


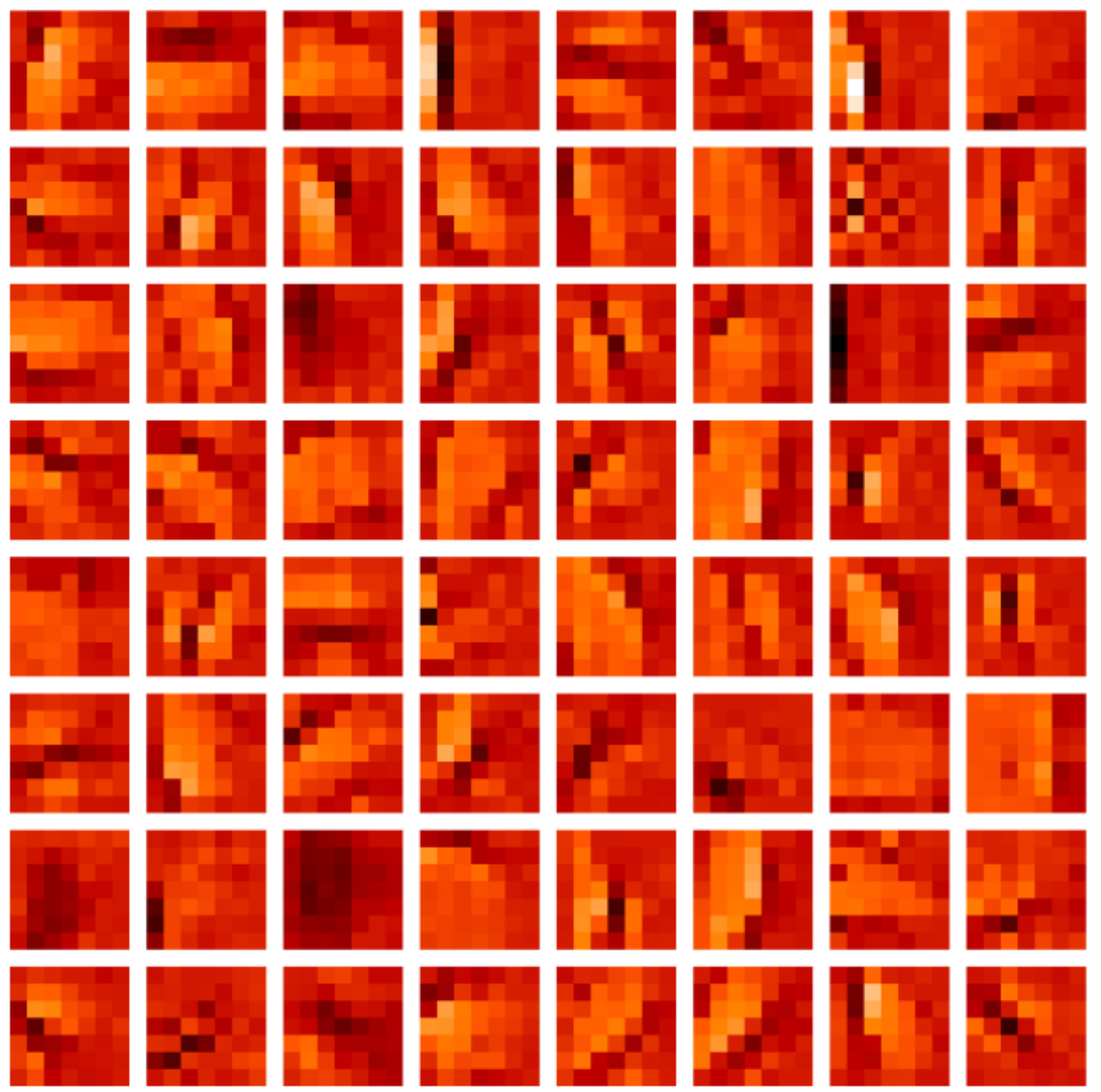


Hybrid approaches - NoVa

Aurisano et al
K. Sachdev



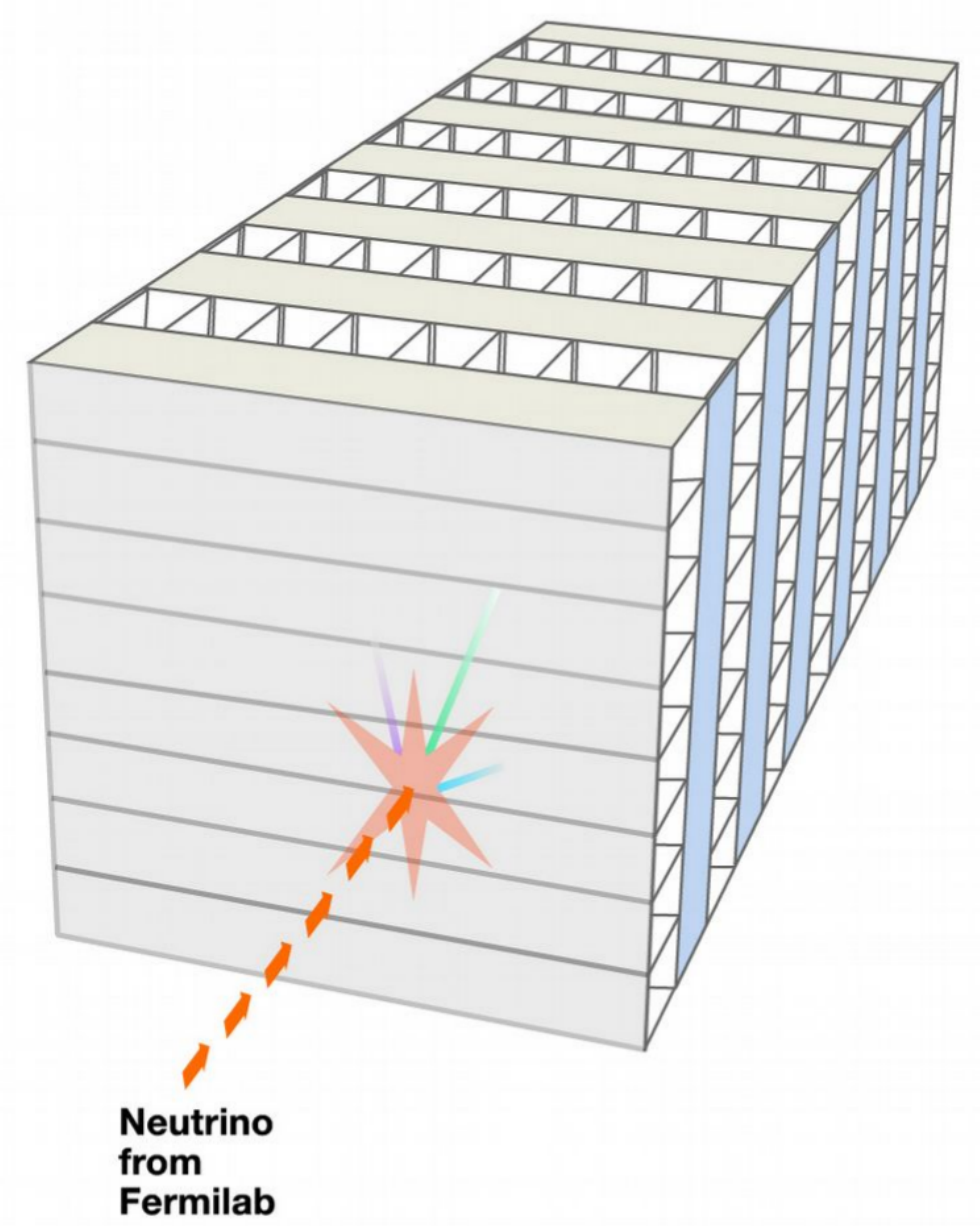




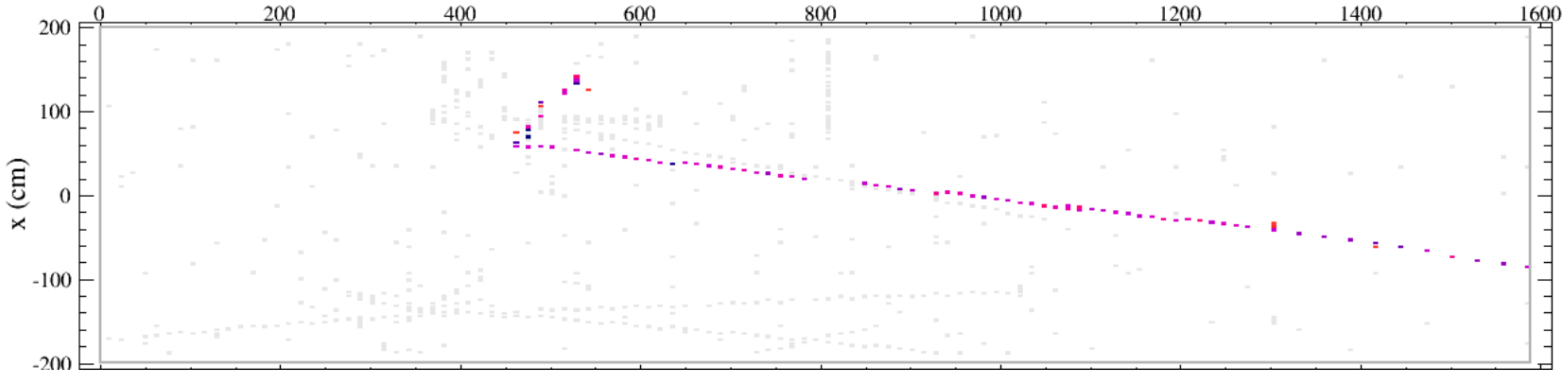
Efficiency of selecting electron neutrinos improved by 40%

Hybrid approaches - NoVa

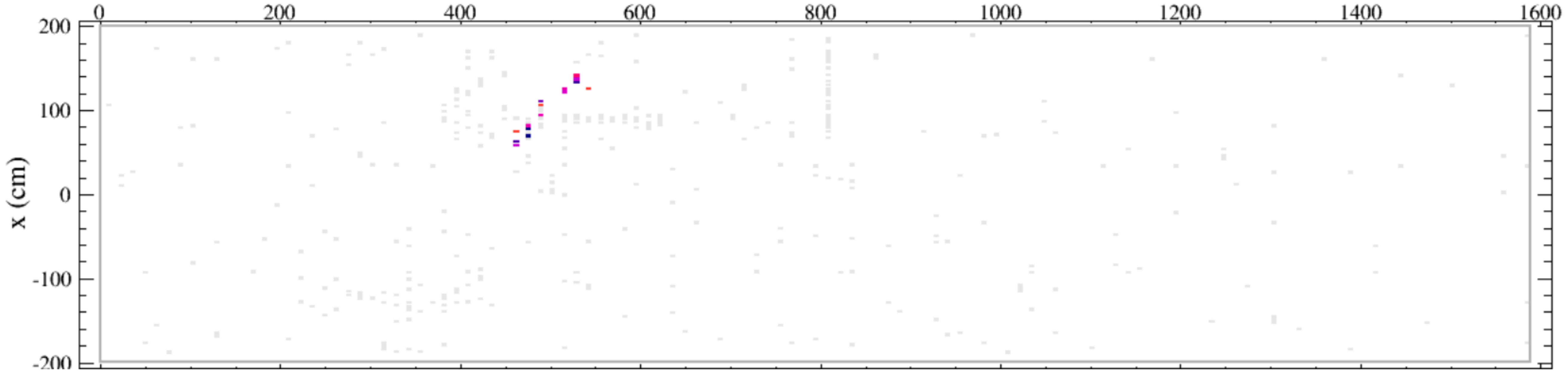
Aurisano et al
K. Sachdev



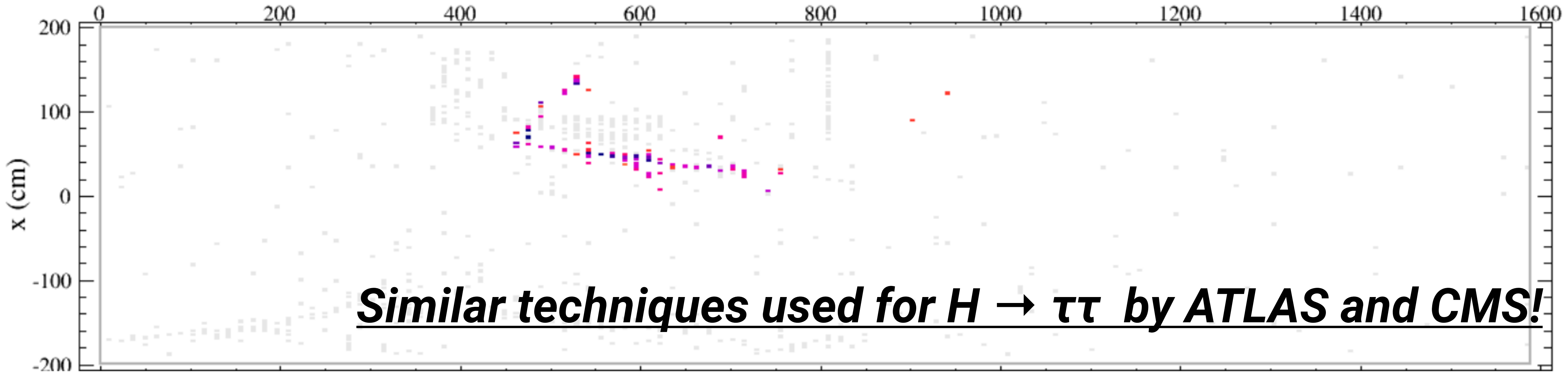
Efficiency of selecting electron neutrinos improved by 40%



(a) A candidate ν_μ CC interaction in ND data

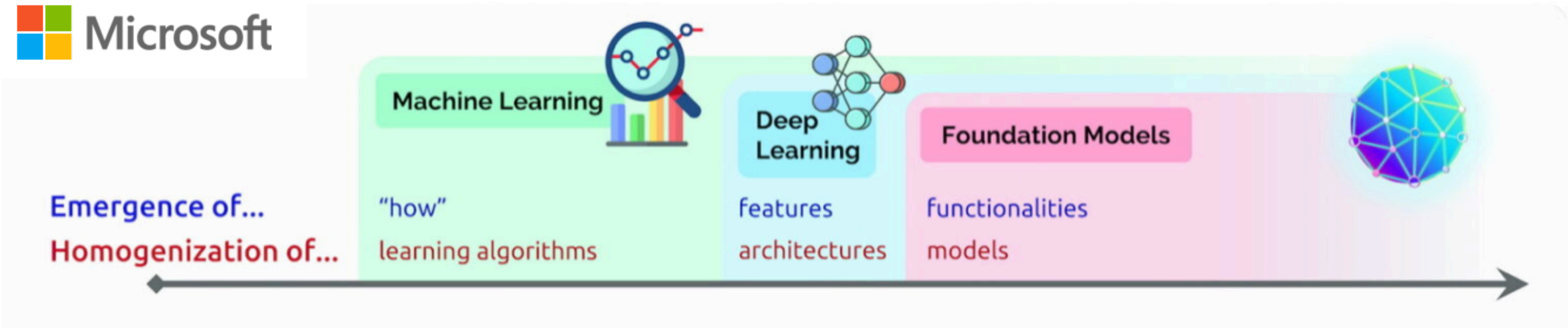


(b) The muon removed or MRCC version of the event



Similar techniques used for $H \rightarrow \tau\tau$ by ATLAS and CMS!

(c) A simulated electron is inserted in place of the muon to make an MRE event.



BigScience	BLOOM	176B	July 2022
	T0pp	11B	October 2021
EleutherAI	GPT-J	6B	July 2021
	GPT-NeoX	20B	February 2022
清華大學 Tsinghua University	GLM	130B	August 2022
Google Research	UL2	20B	October 2022
	T5	11B	February 2020
	OPT	175B	June 2022

The New York Times

A.I. and Chatbots › | How the A.I. Race Began | One Year of ChatGPT | Key Figures in A.I. | How A.I. Could Be Regulated

THE SHIFT

Maybe We Will Finally Learn More About How A.I. Works

Stanford researchers have ranked 10 major A.I. models on how openly they operate.

AI Explainer: Foundation models and the next era of AI

Published March 23, 2023

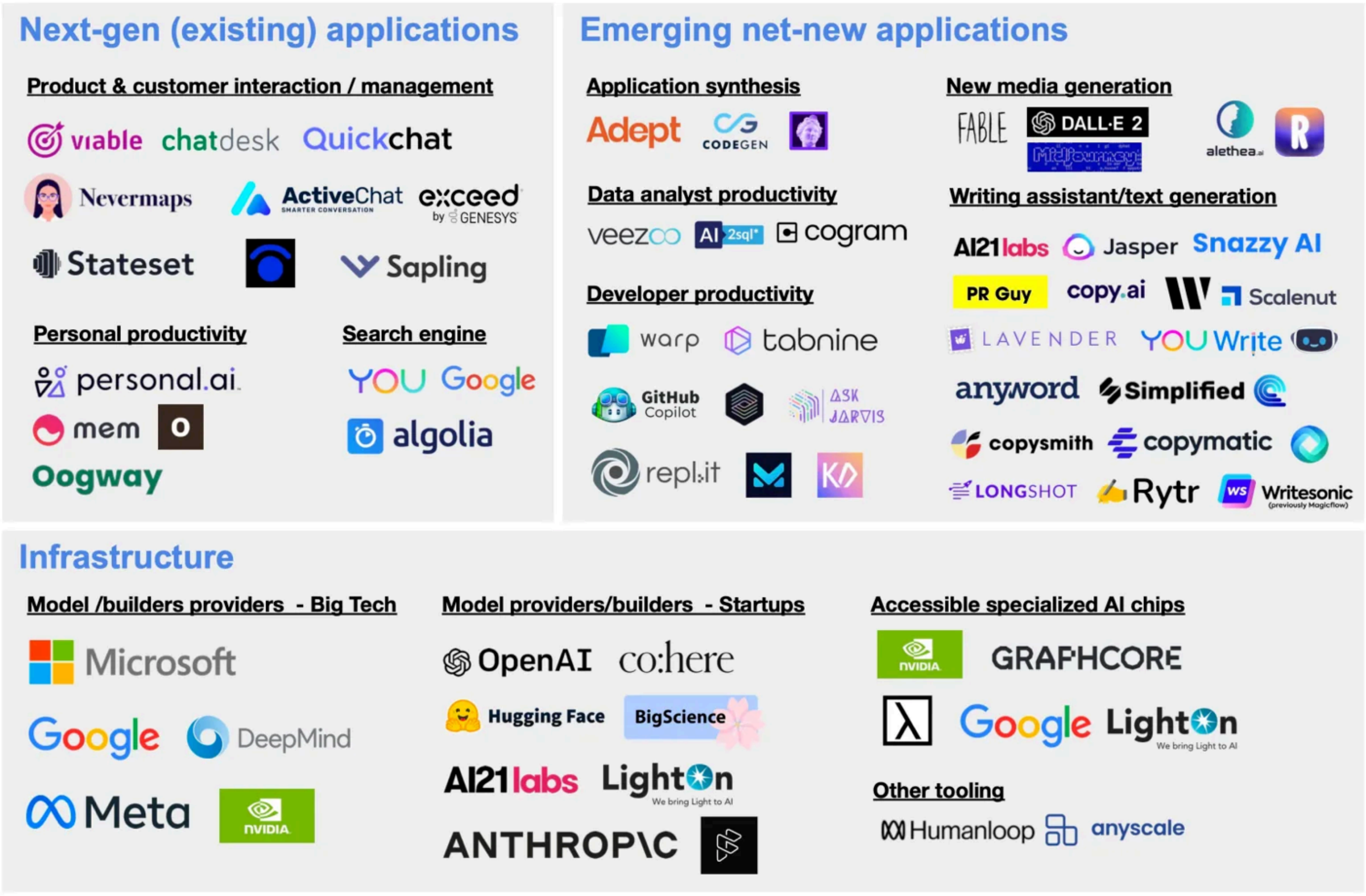
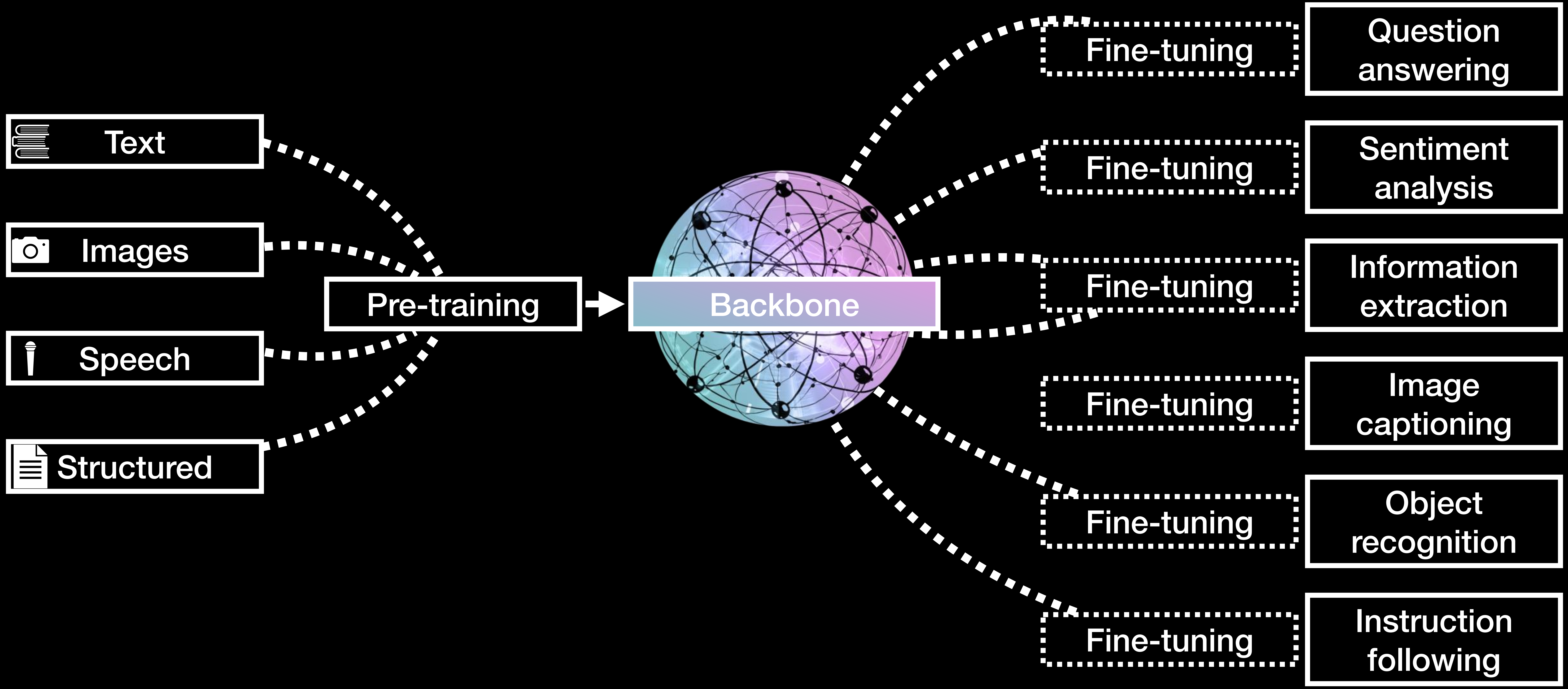


Figure 5: Representative sample of companies that have publicly stated that they are using, building, or enabling

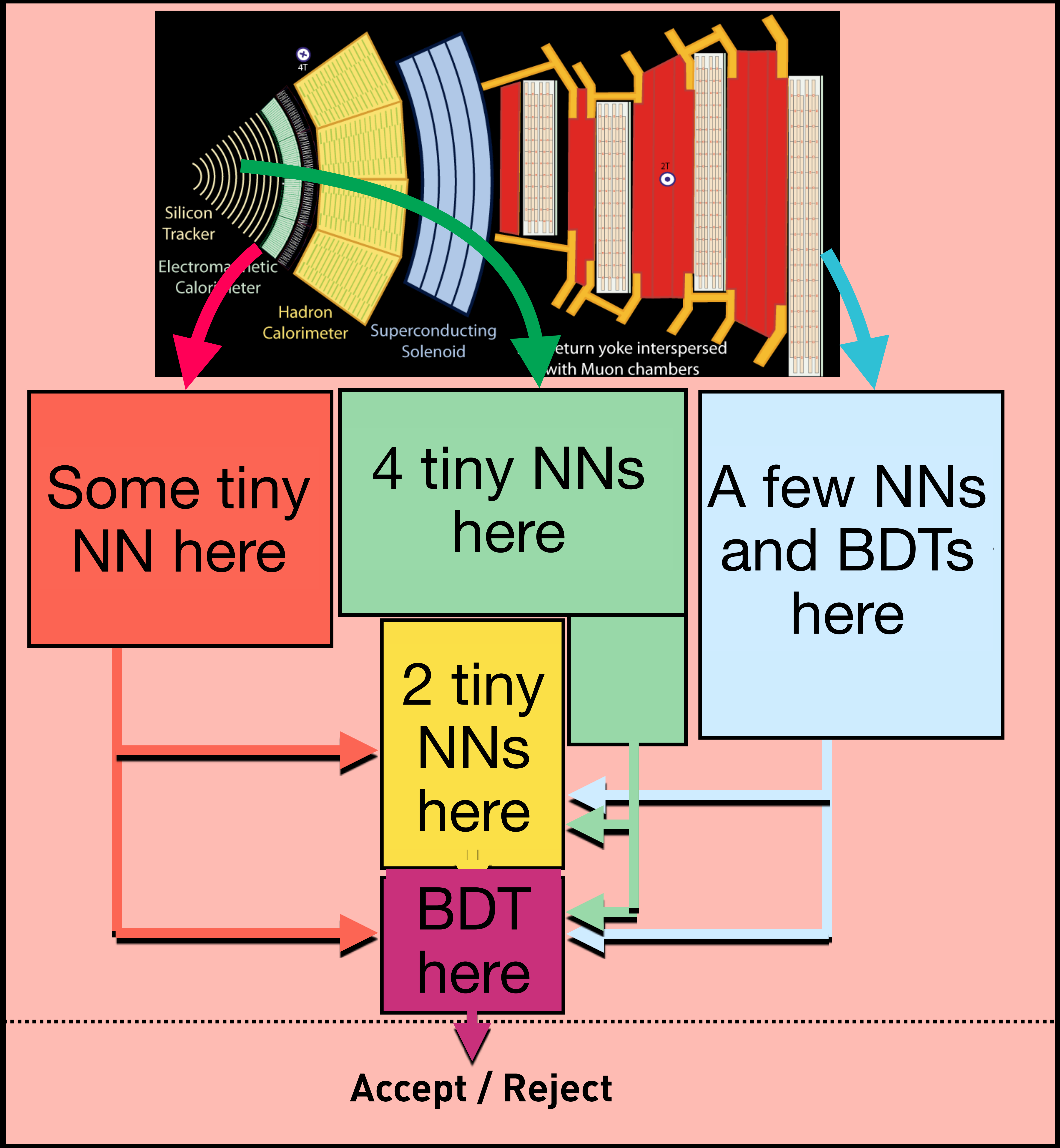

Live

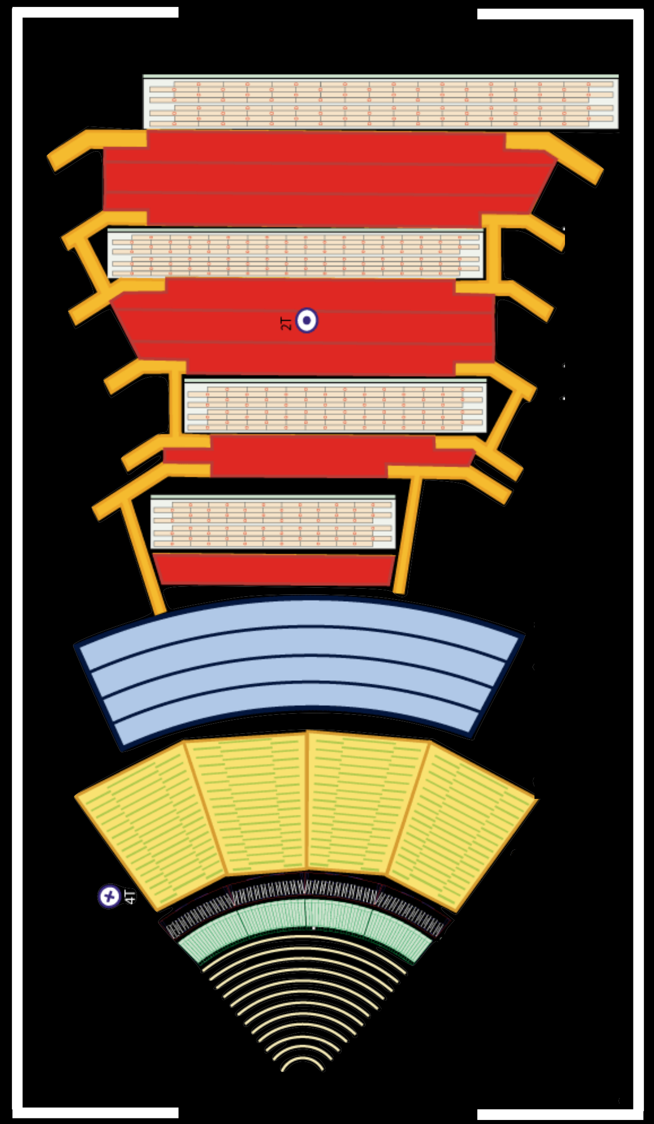

Live

Foundation Models




Heterogeneous detector
Multi-modal input!
Pre-training
Backbone
Fine-tuning
Jet reconstruction
Fine-tuning
Electron reconstruction
Fine-tuning
Pile-up removal
Fine-tuning
Missing energy computation
Fine-tuning
Anomaly Detection
Fine-tuning
0/1?
Generate simulation?

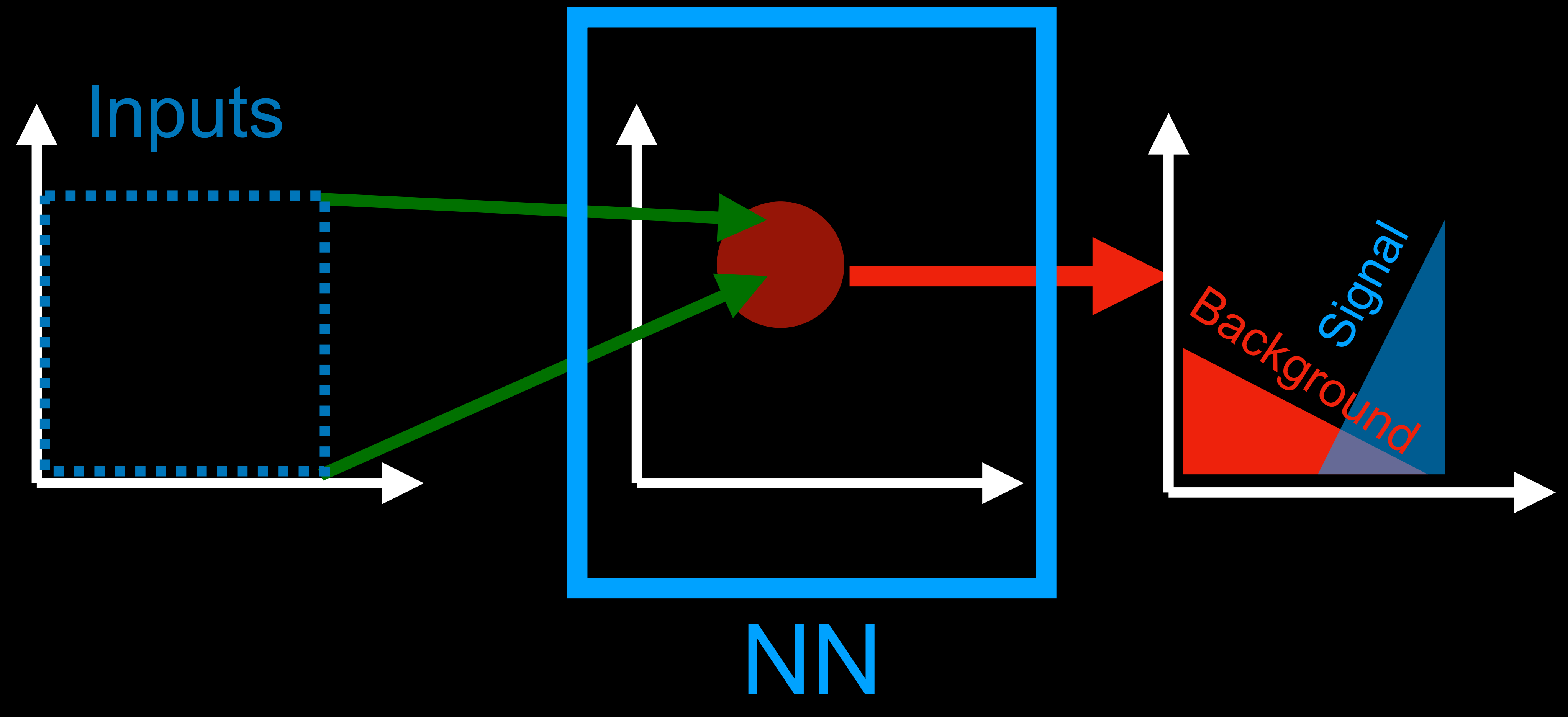




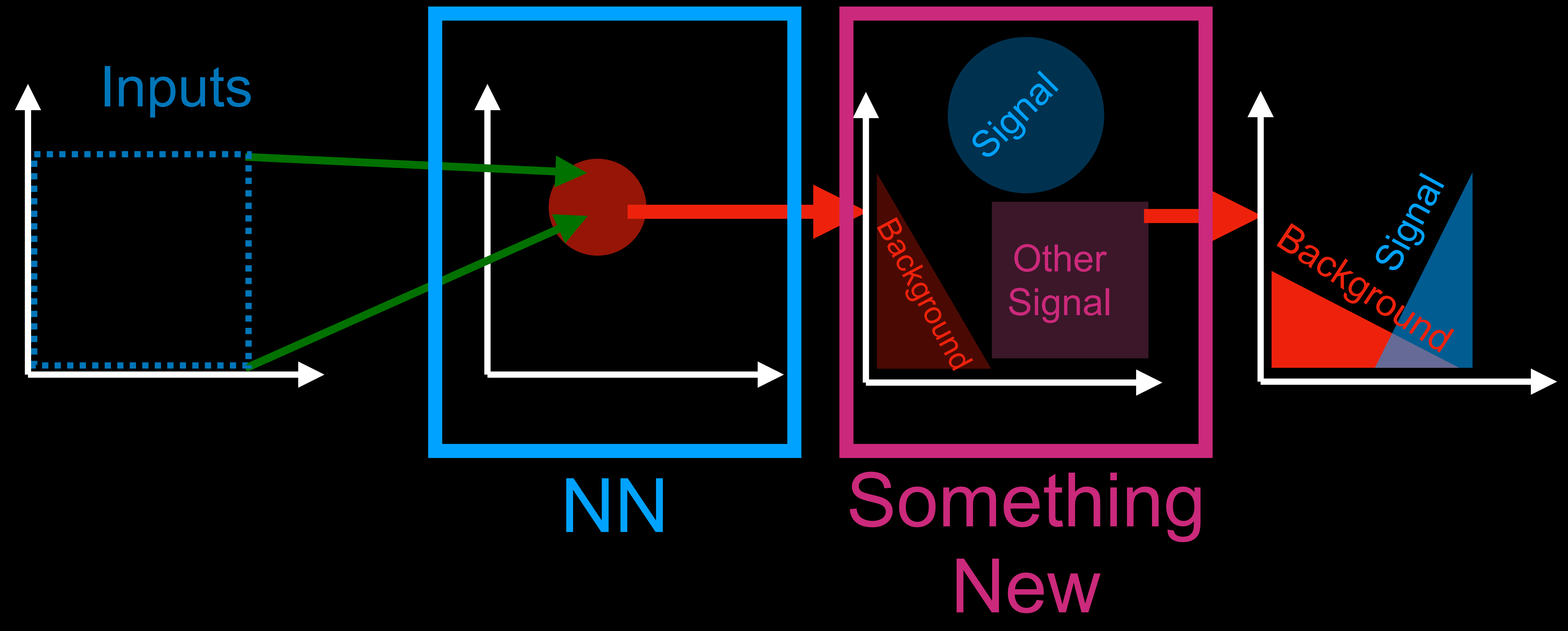
$$x = (x_1, x_2, \ldots,)$$

$$\Rightarrow f(x; w^*) \Rightarrow \hat{y}$$

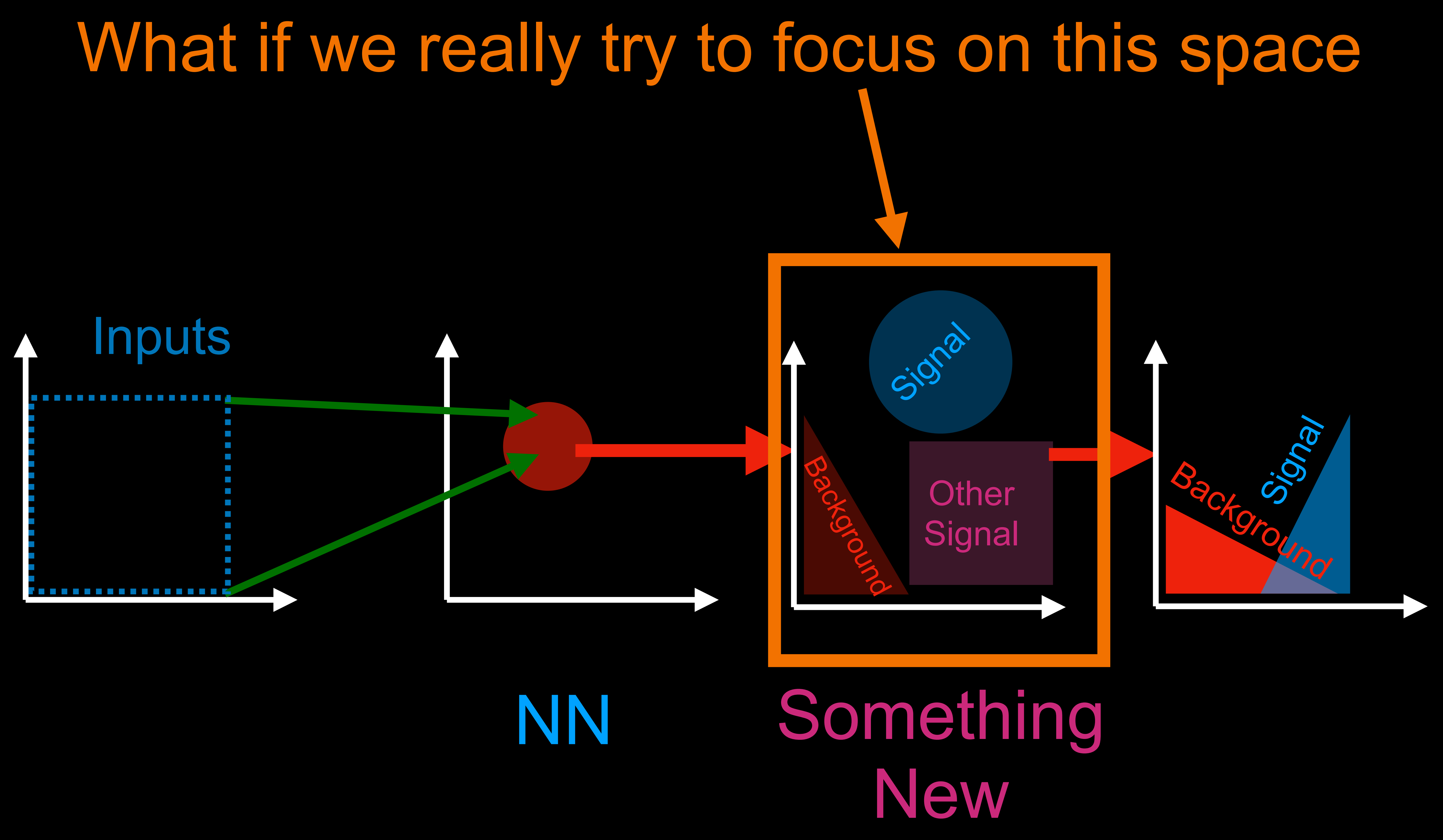
Too many models, too little learning?



Discrimination

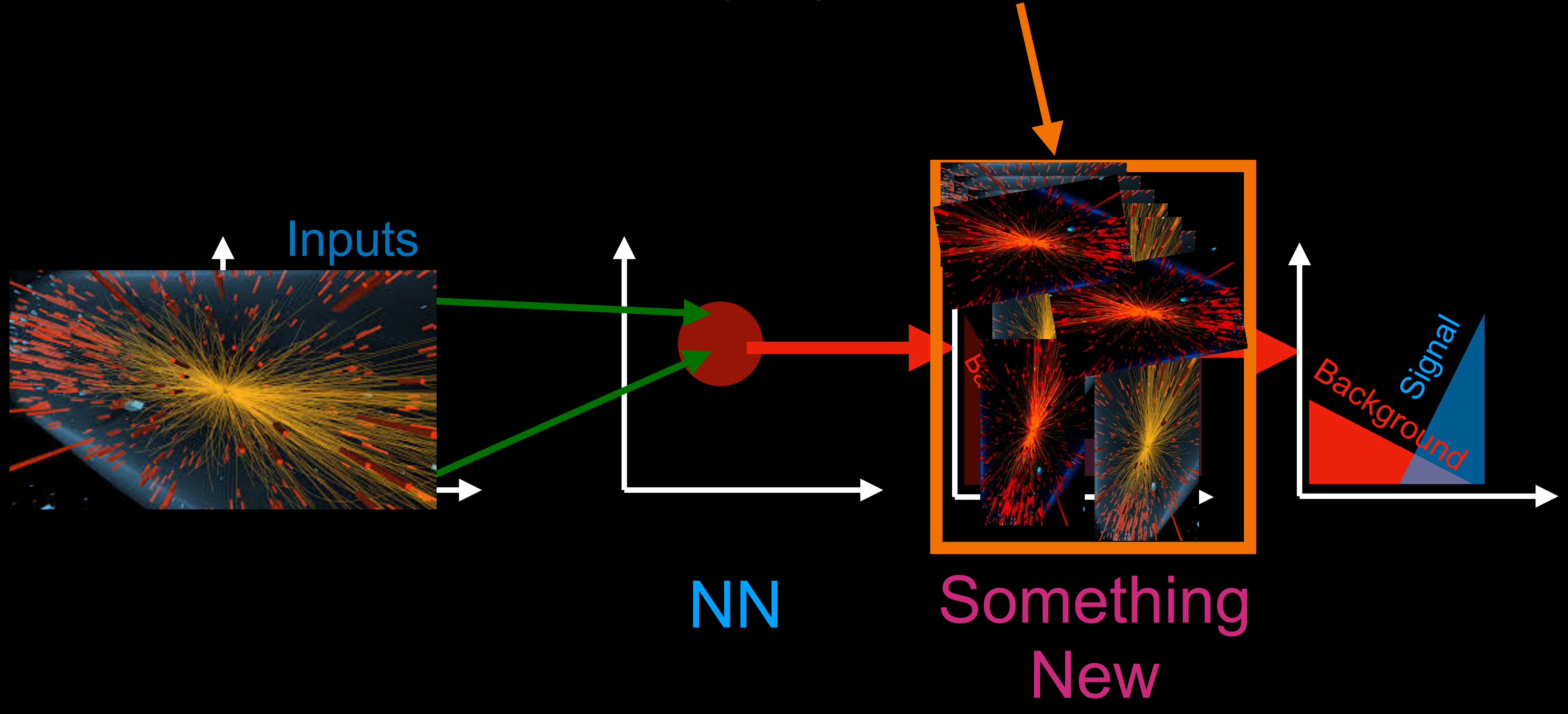


Metric Learning



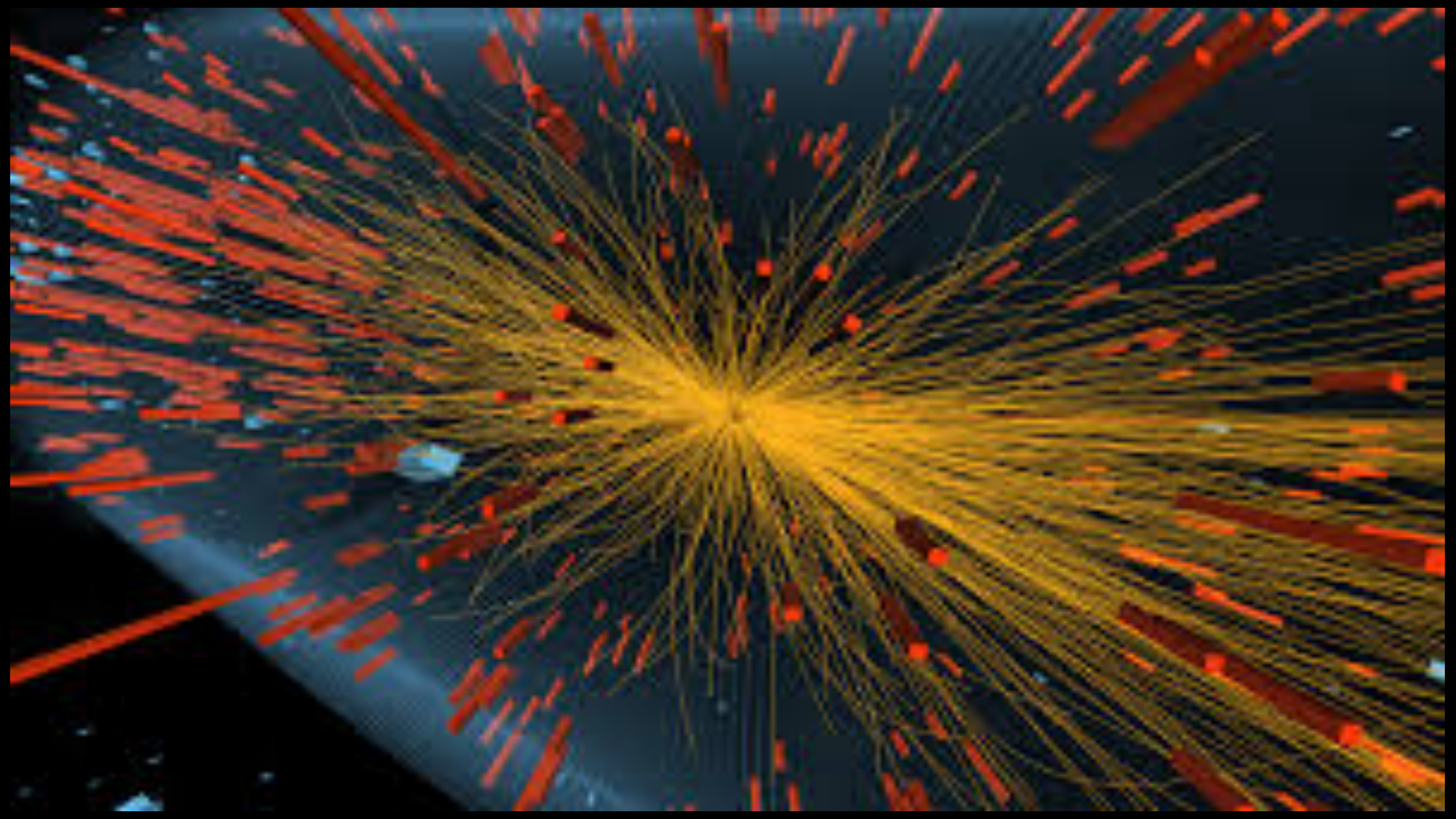
Neural embedding

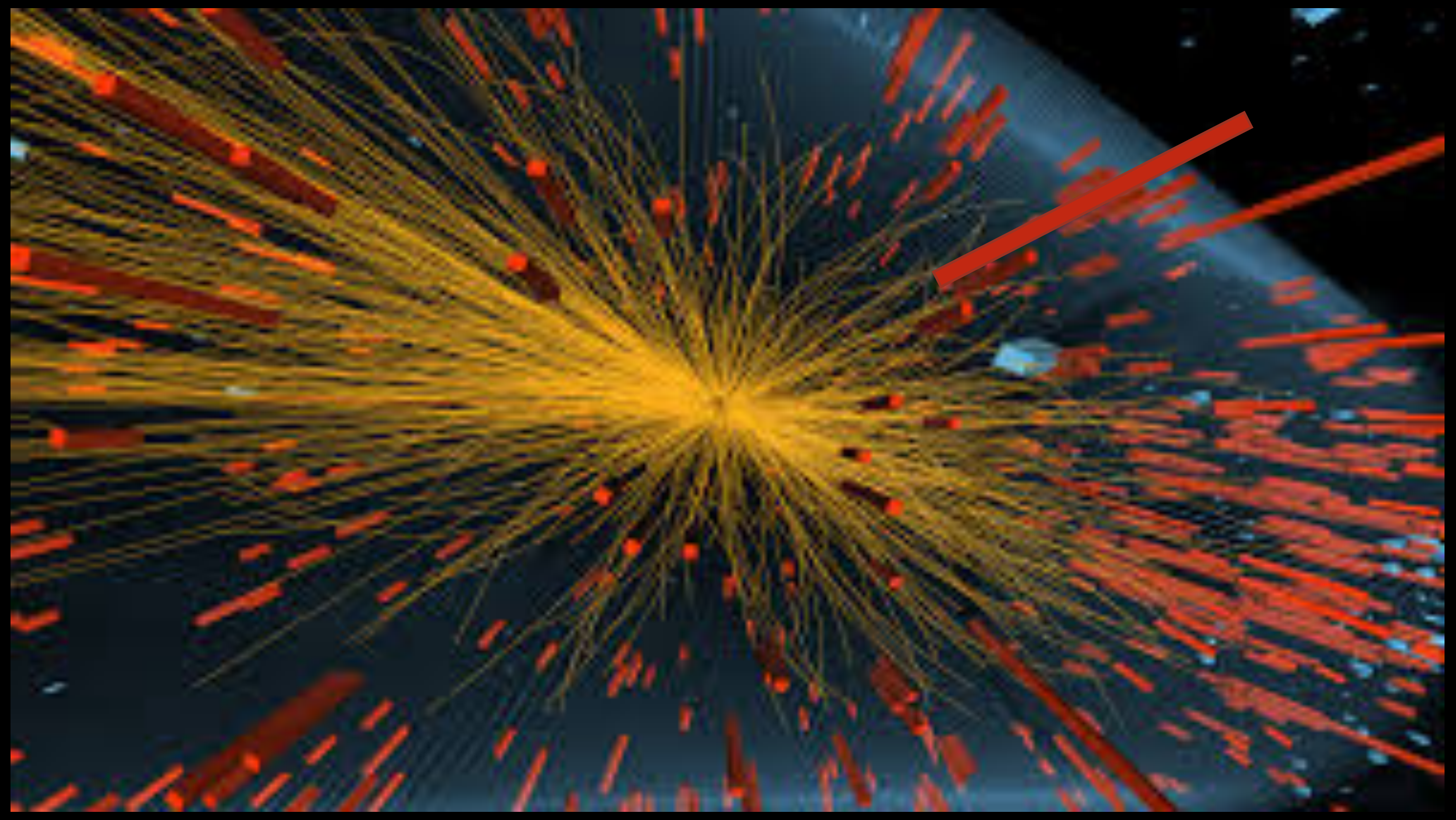
What if we really try to focus on this space



Neural embedding

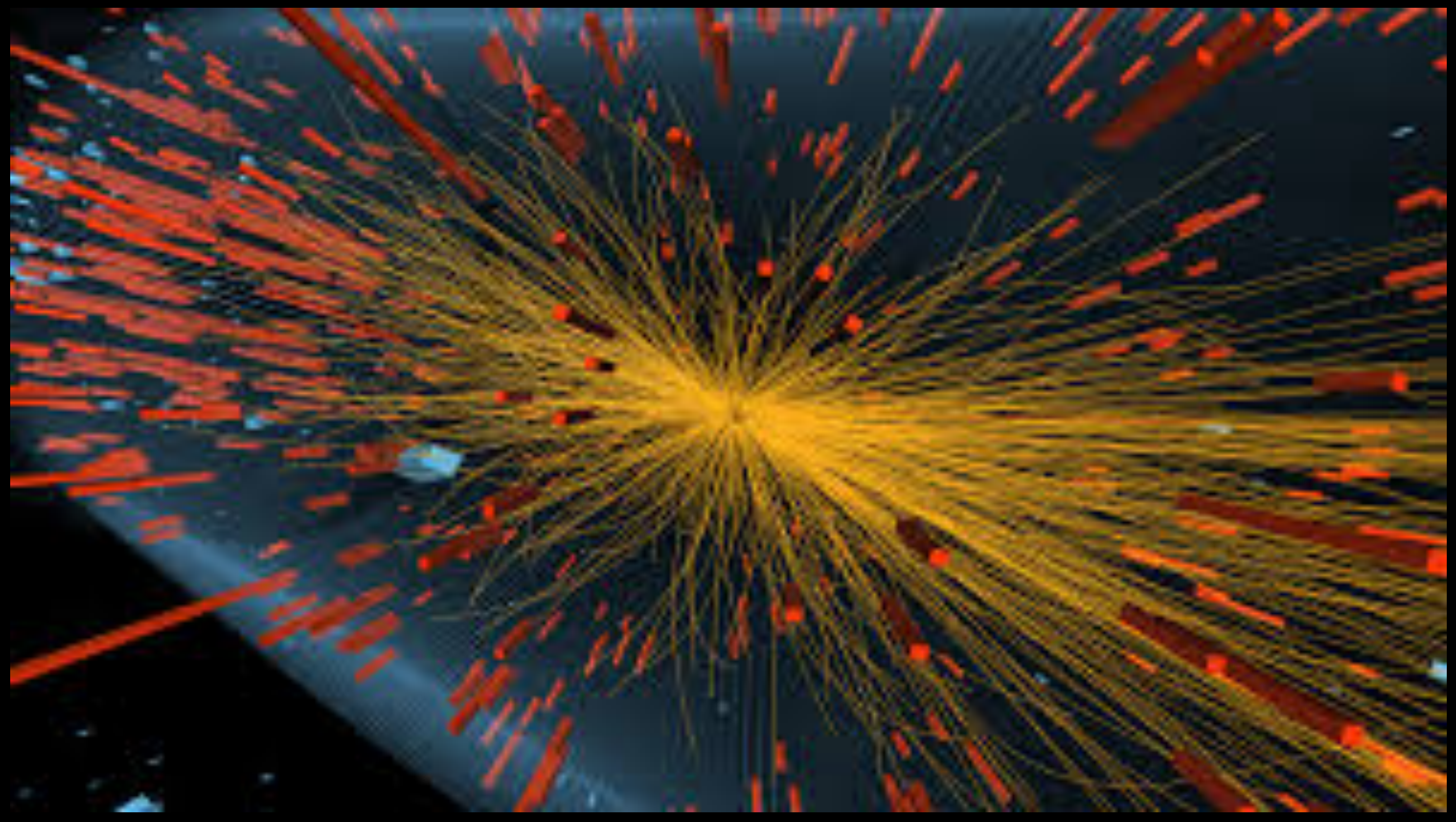
Learning the space

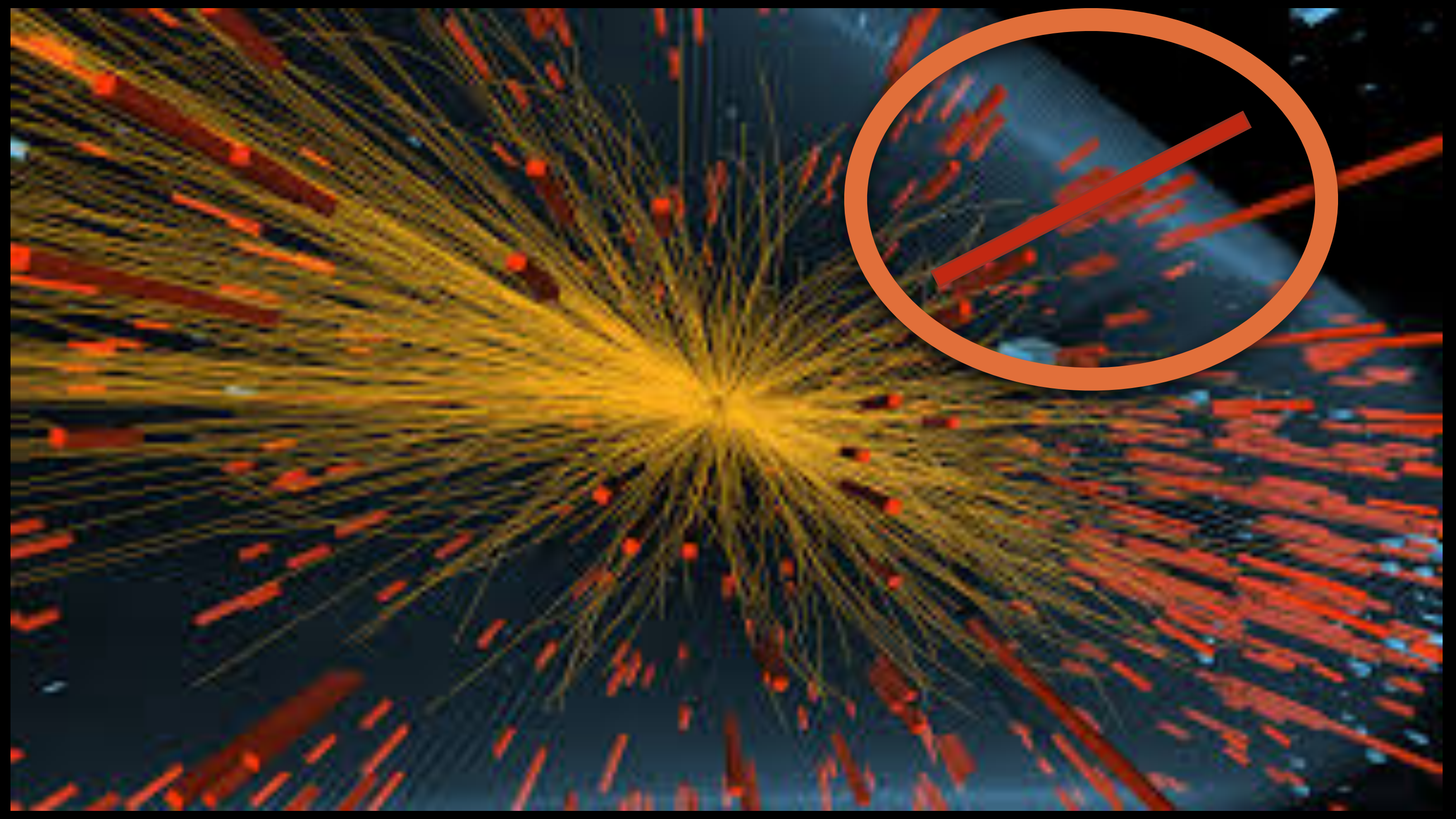




Learning the space

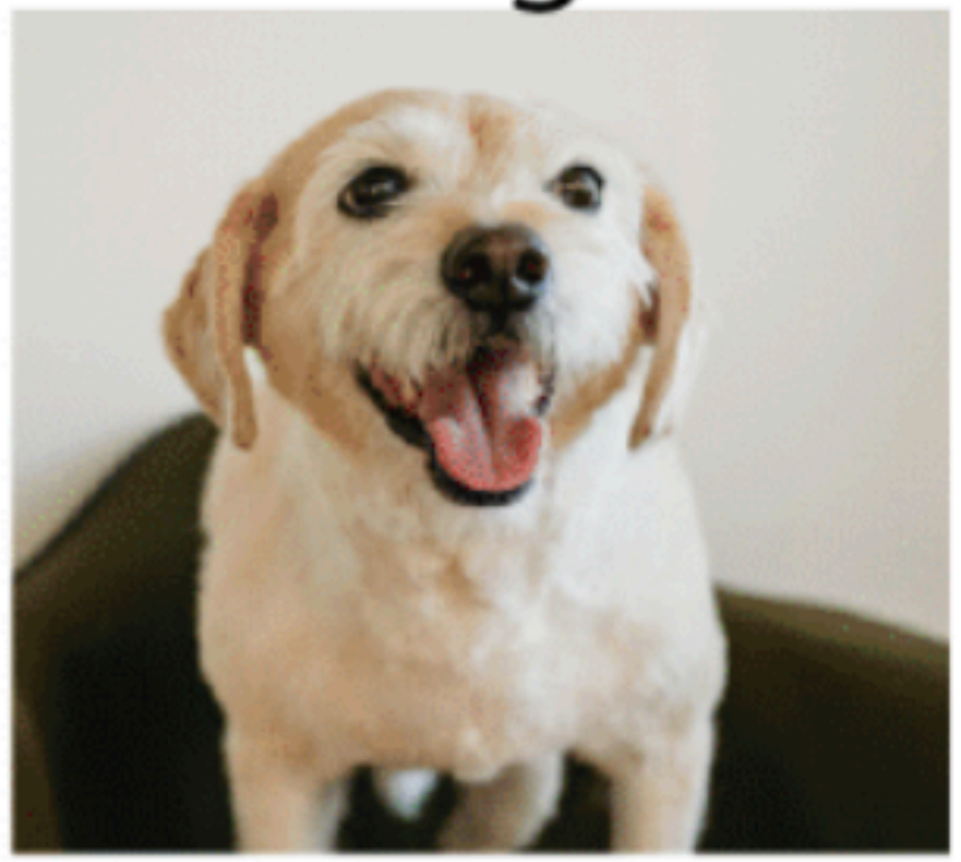
- By looking at data, we can learn a lot
  - Go over input piece by piece
  - Analyze every aspect
  - Compare every feature
- Find distinctive style of the input
  - can be done e.g by looking for a deviation

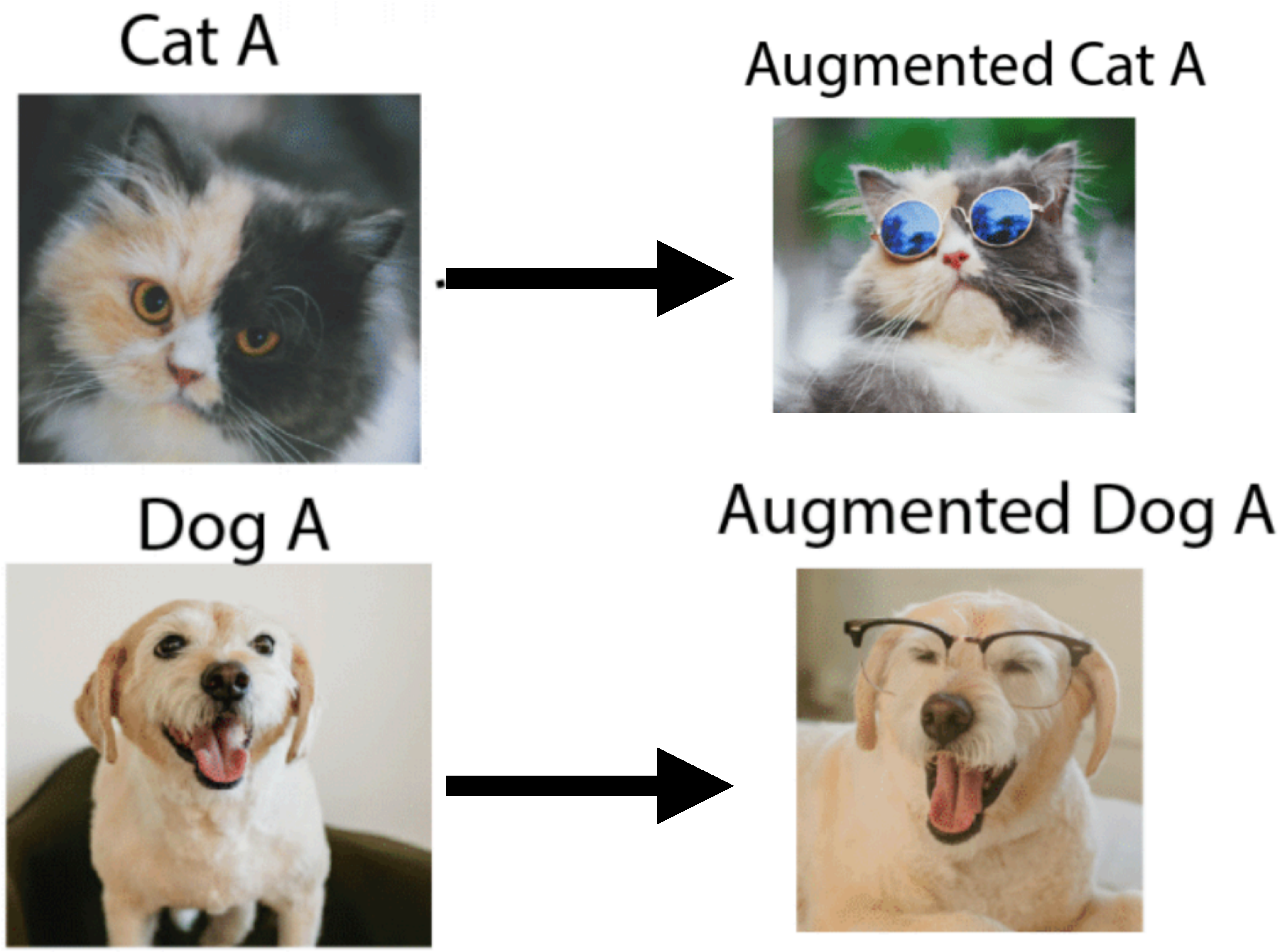




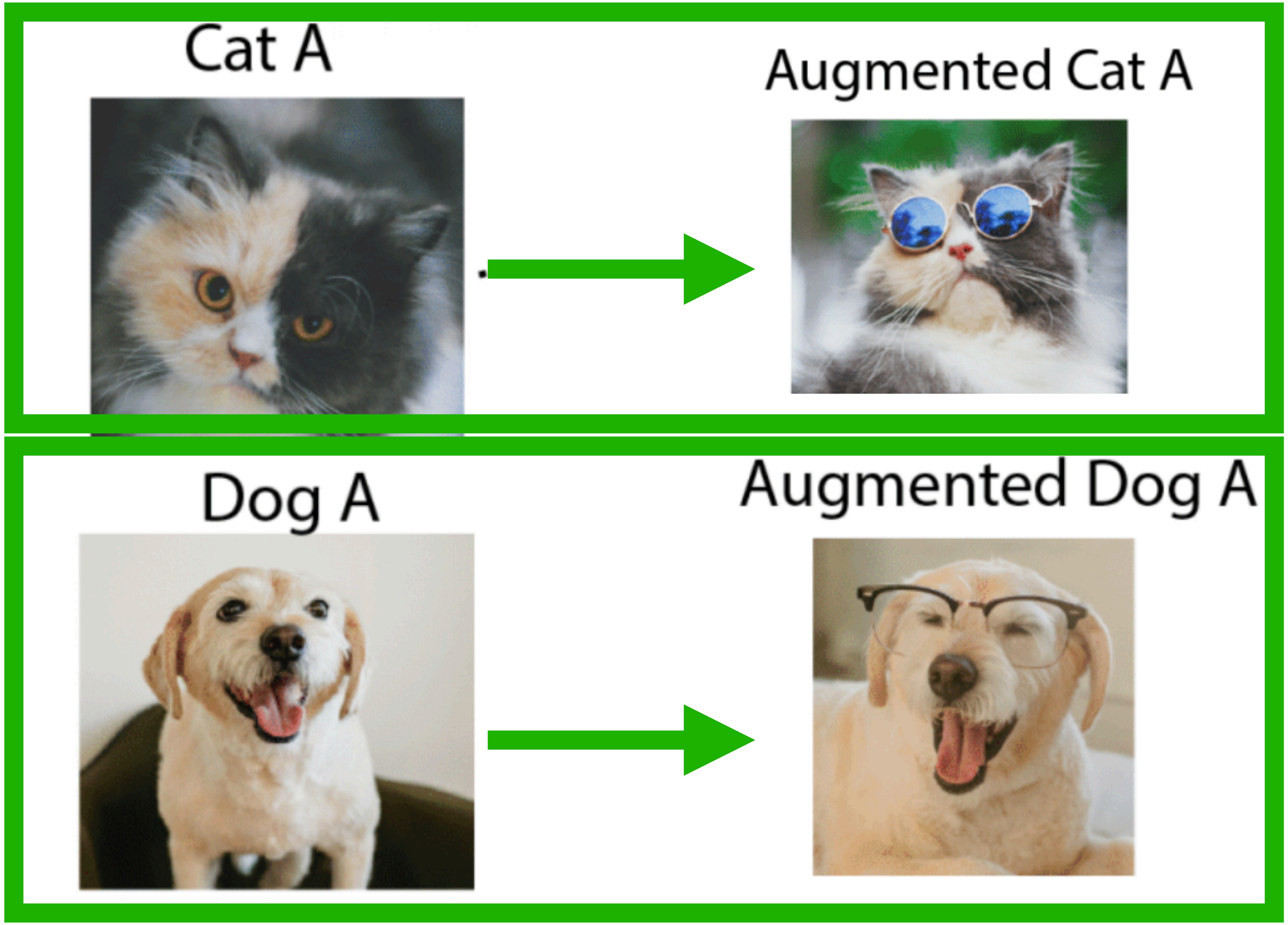
Cat A

Dog A

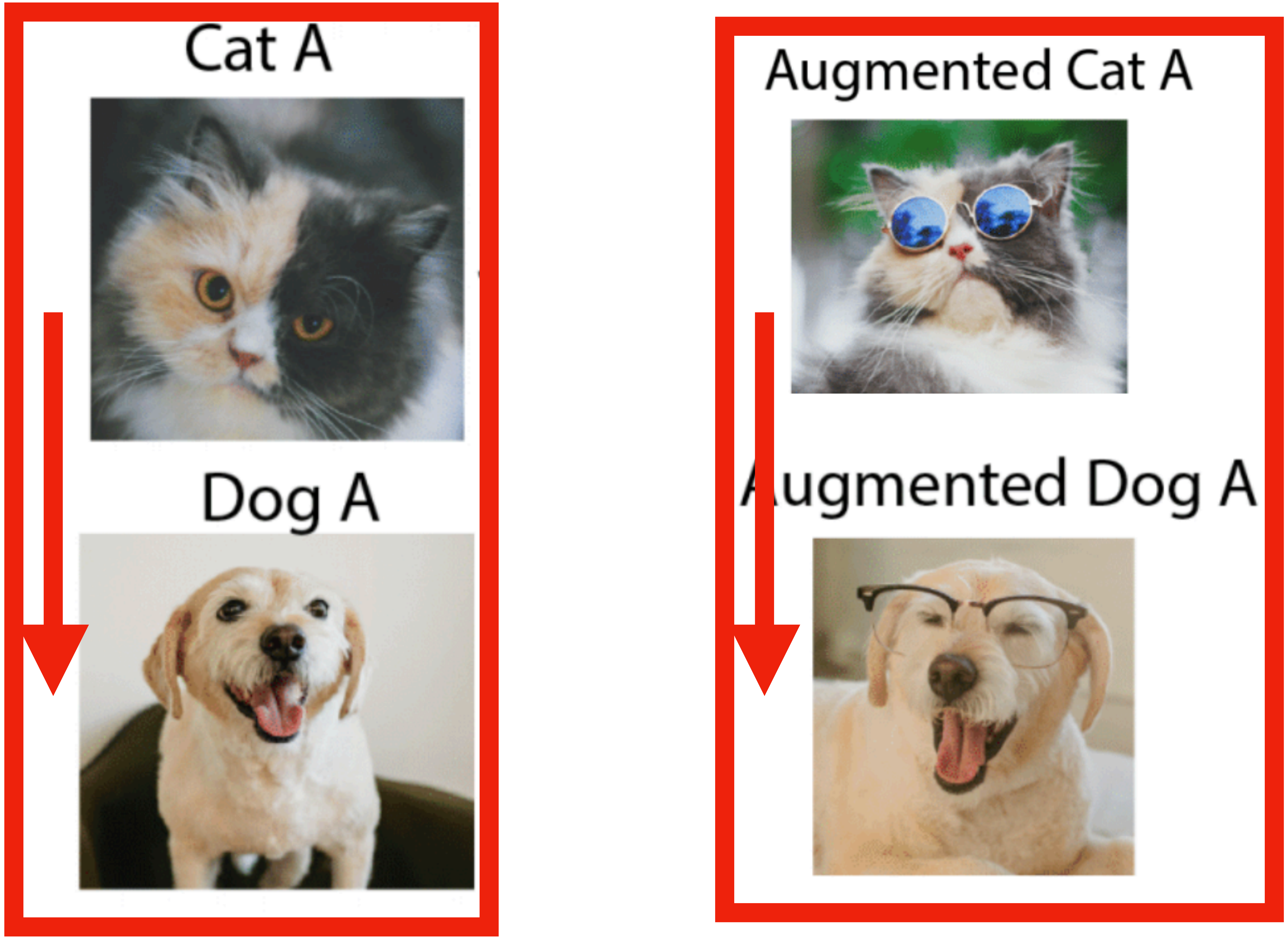



Cat A
Augmented Cat A
Dog A
Augmented Dog A

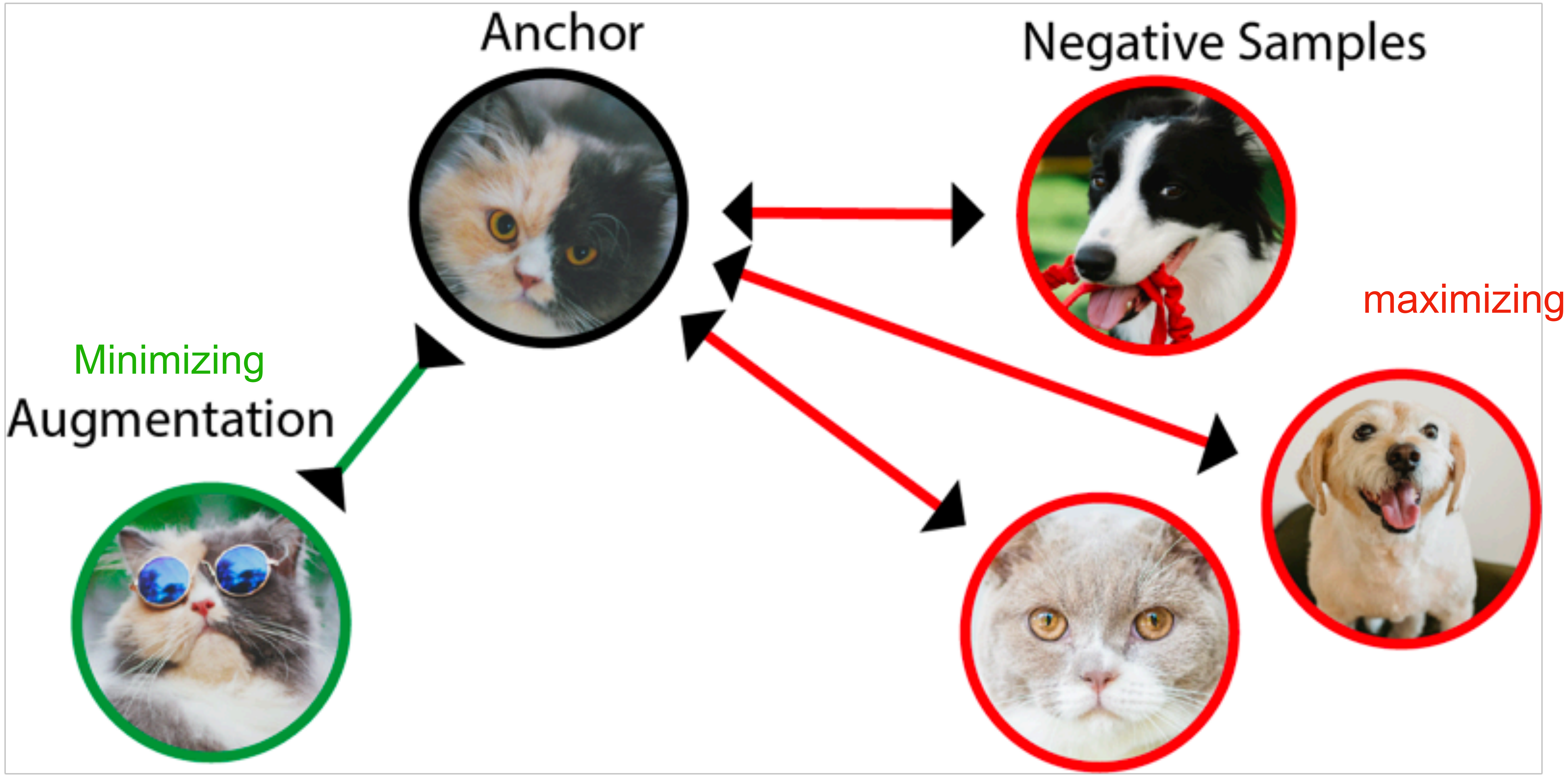
Minimize



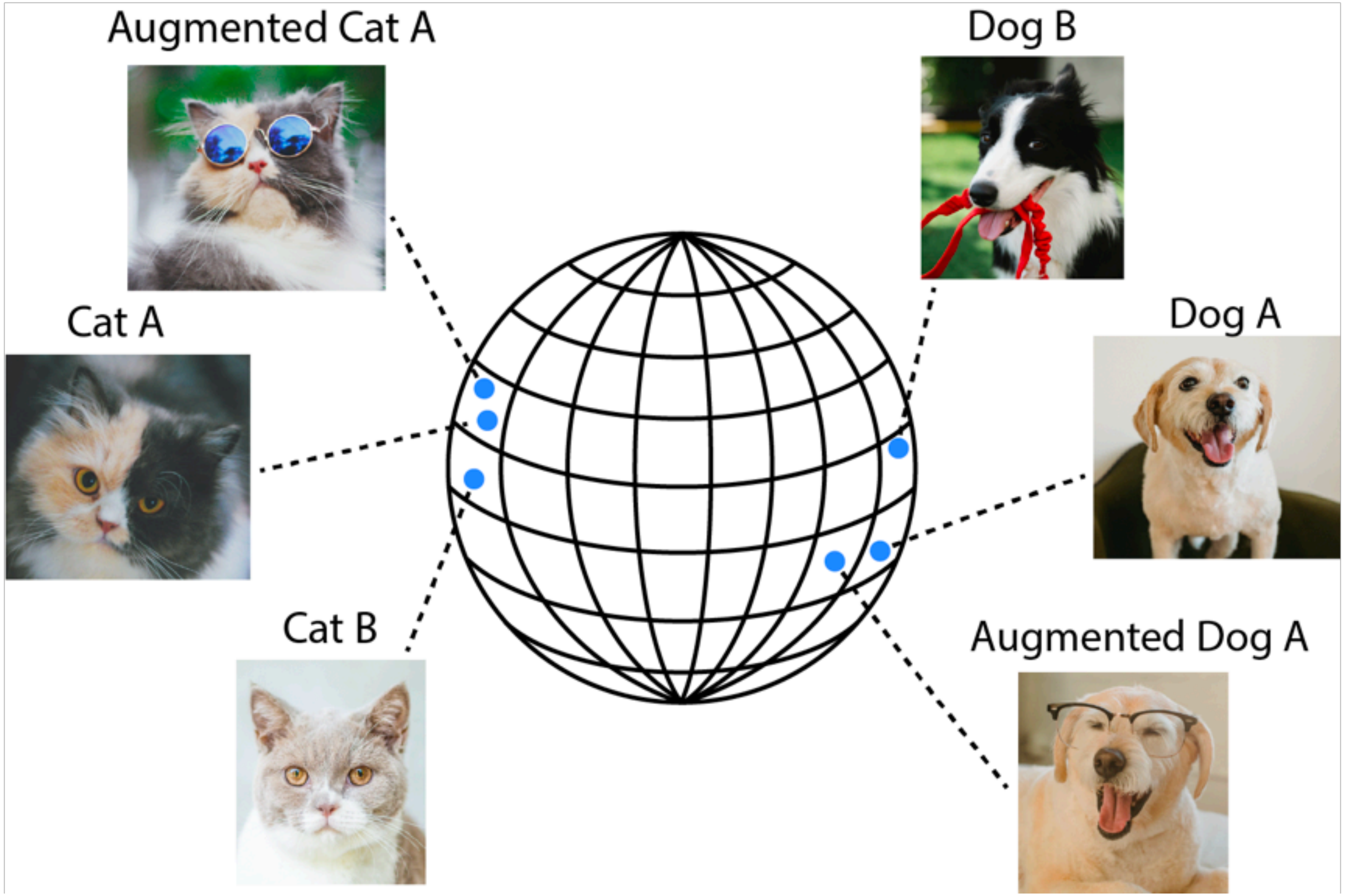
Maximize



Physically motivated augmentations?

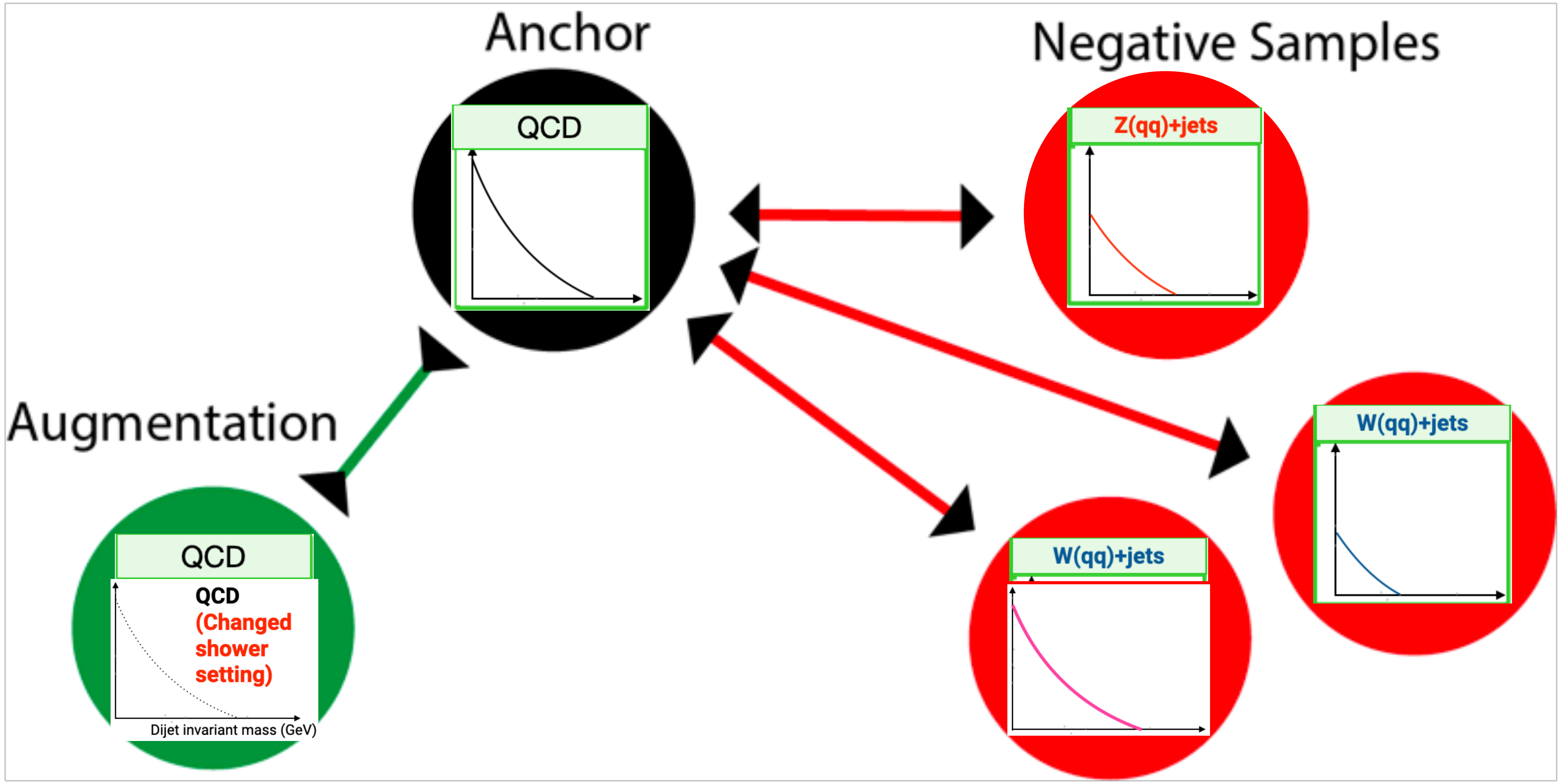


- Minimizing and maximizing distances learns a space



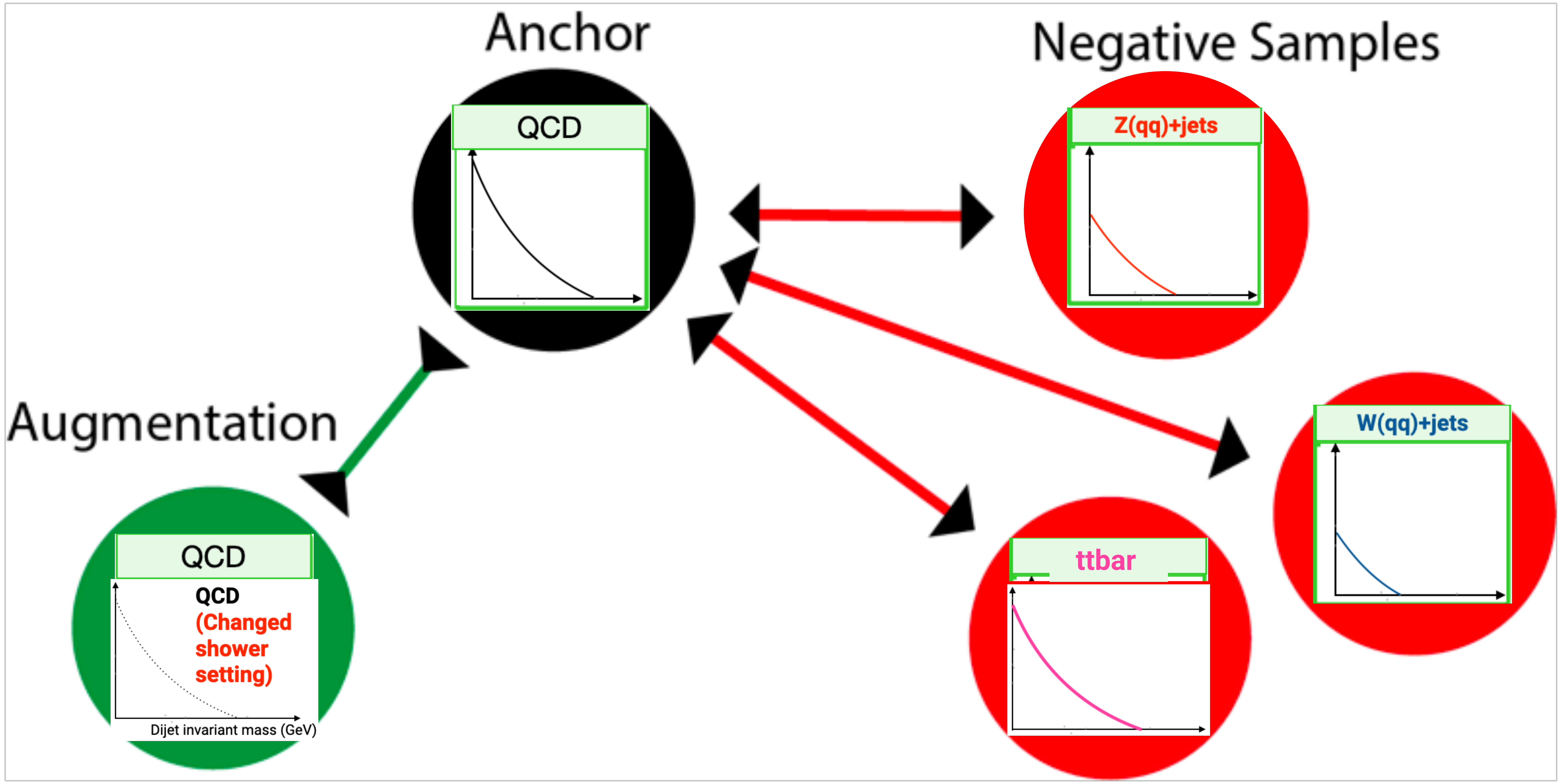
Augmented Cat A
Dog B
Cat A
Dog A
Cat B
Augmented Dog A

Physically motivated augmentations?

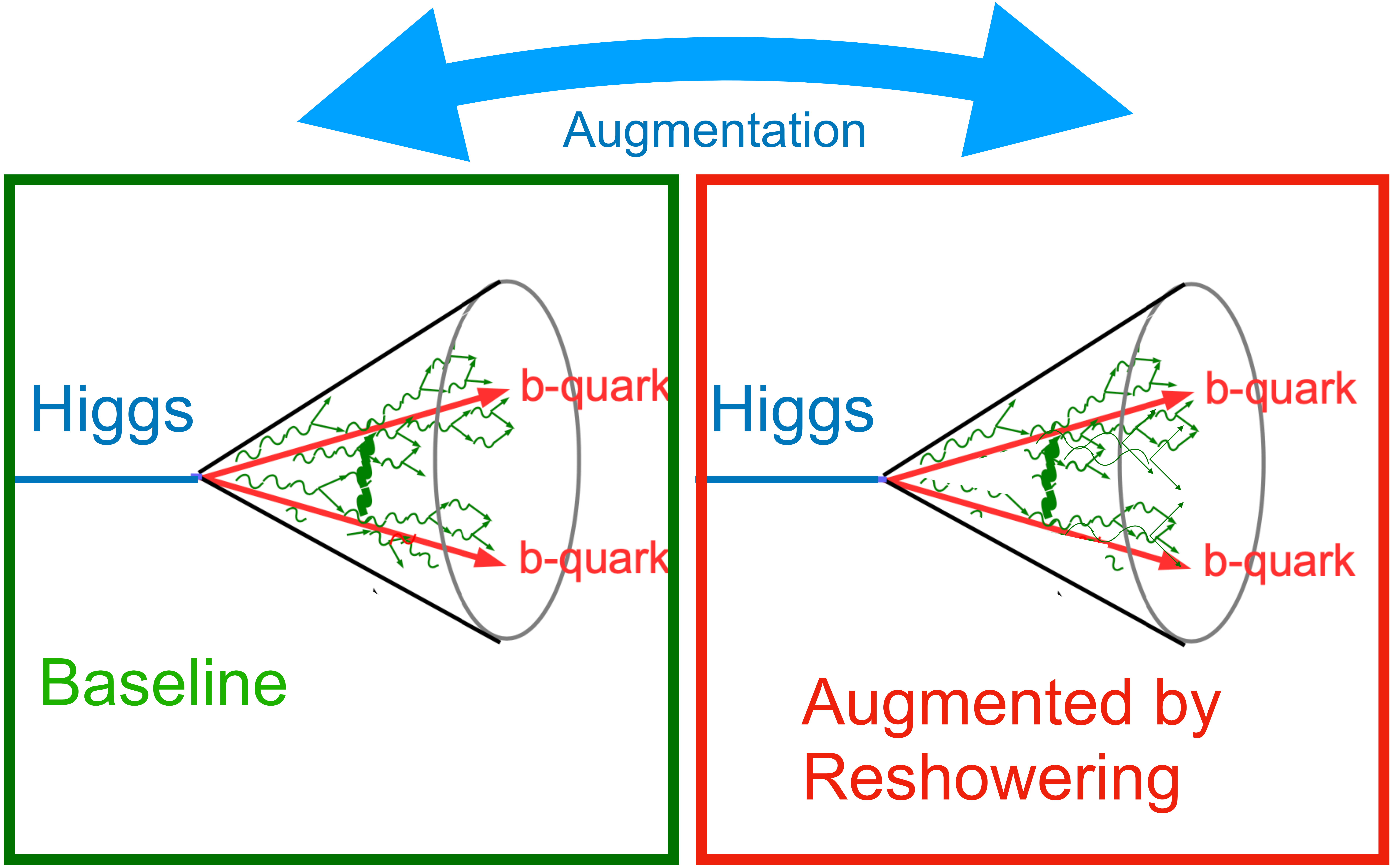


No class labels used in training! How do we augment detector data?

Physically motivated augmentations?

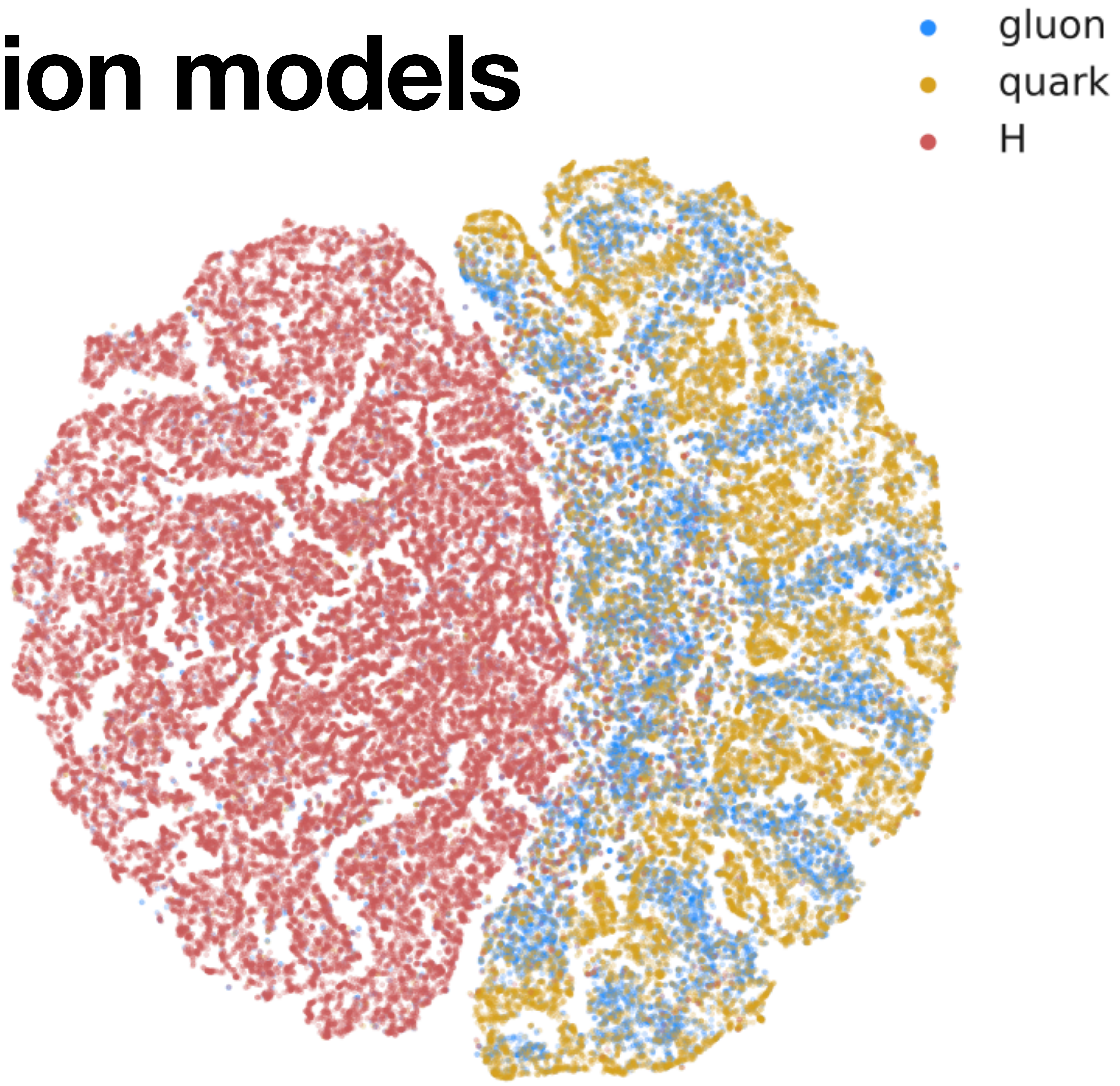


No class labels used in training! How do we augment detector data?

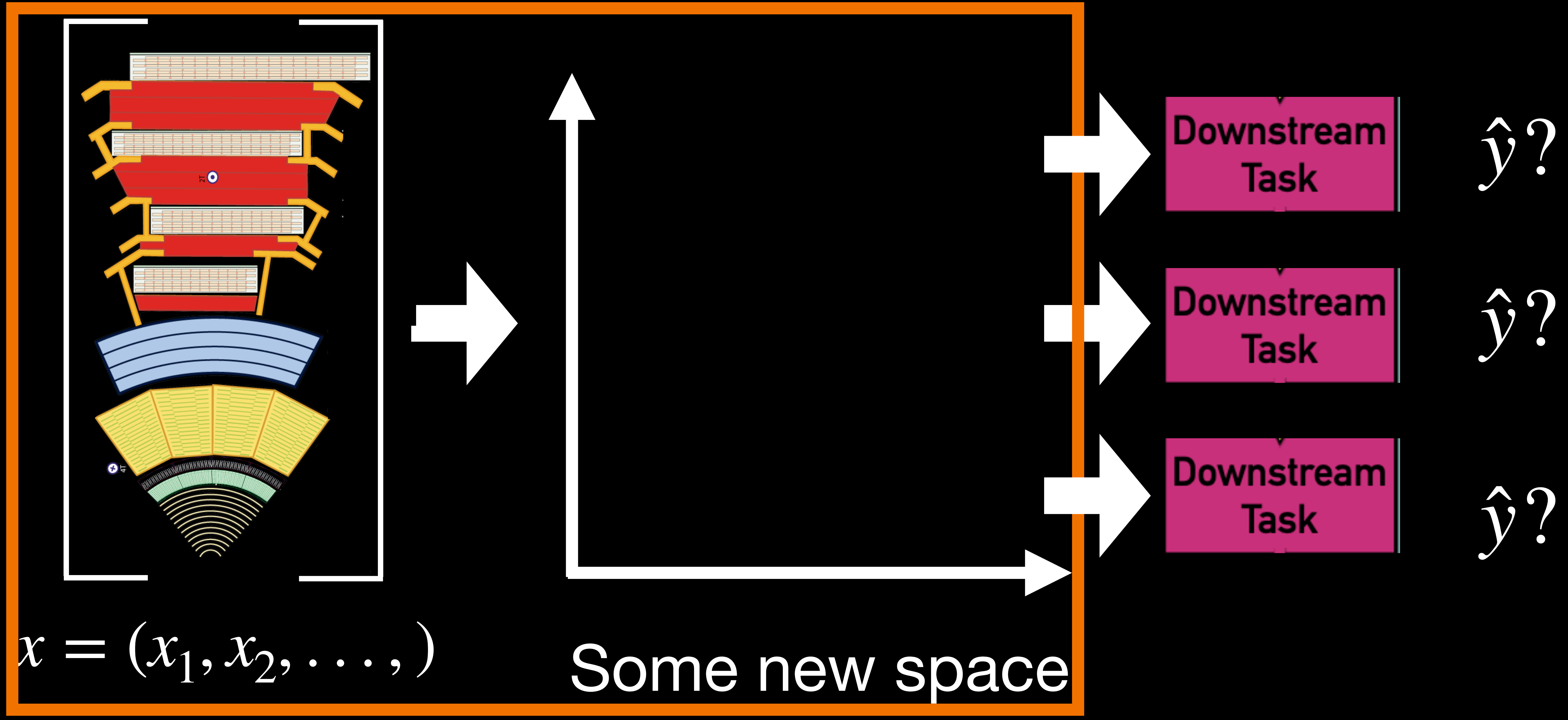


Embedded Space can use any NN to embed

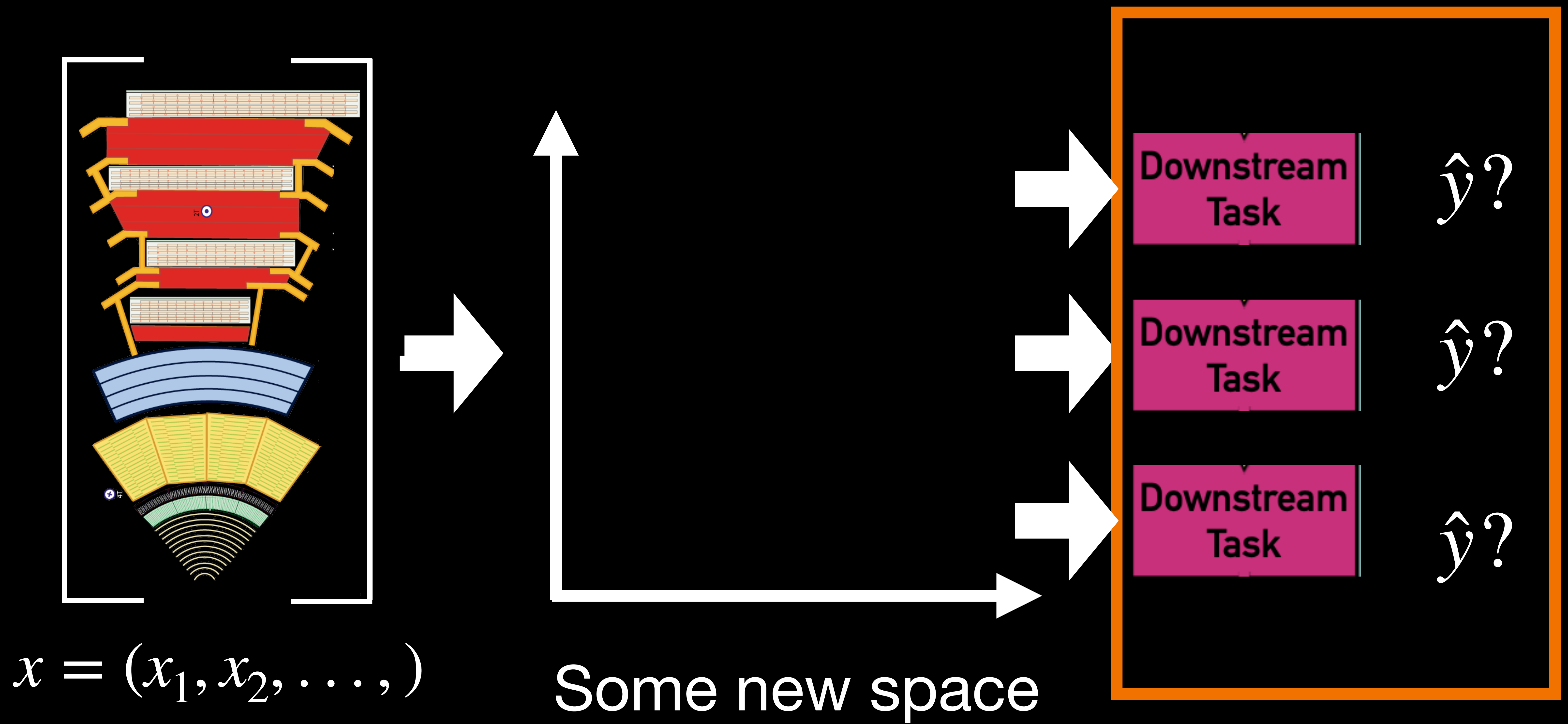
QM foundation models



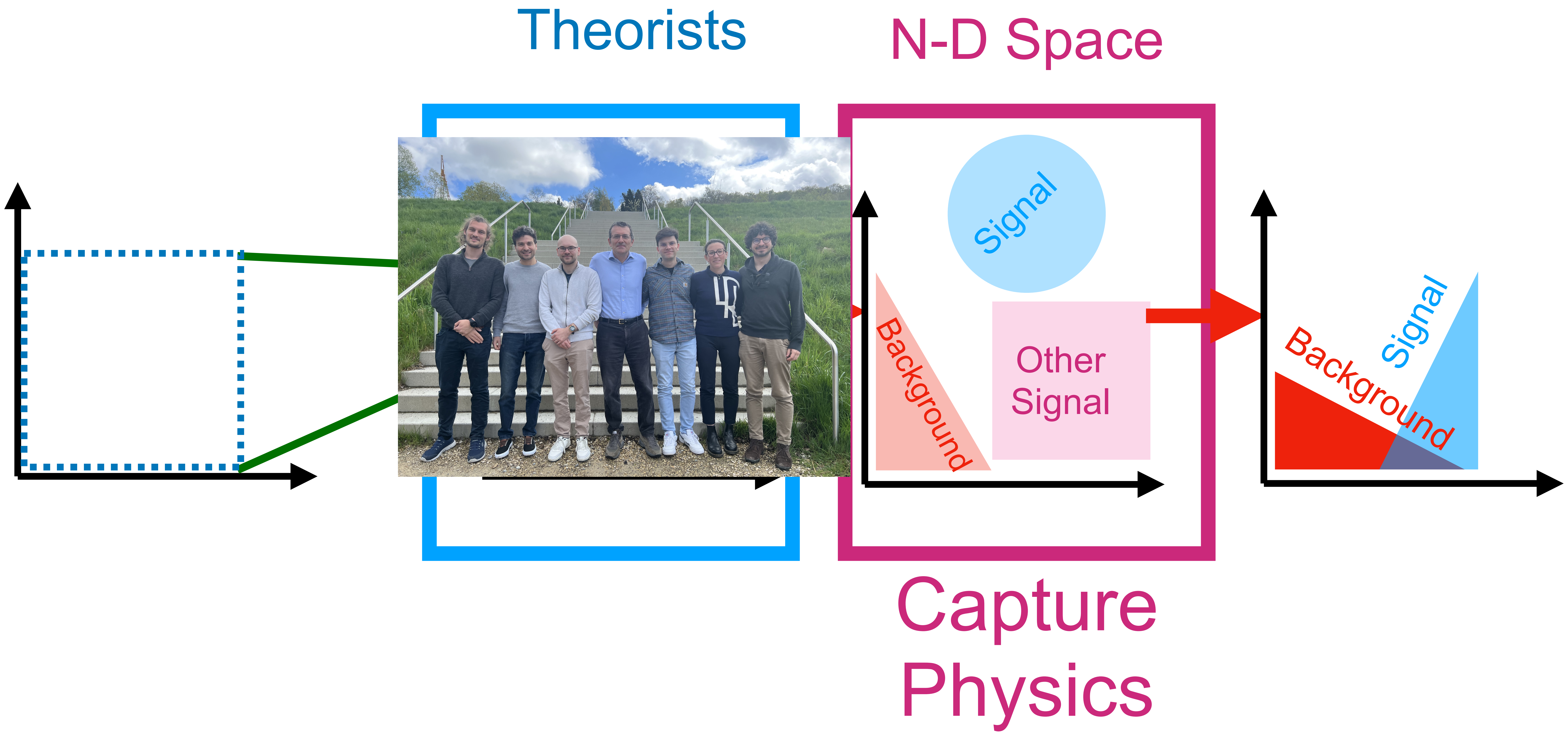
→ embedding quantum mechanics into AI algorithm



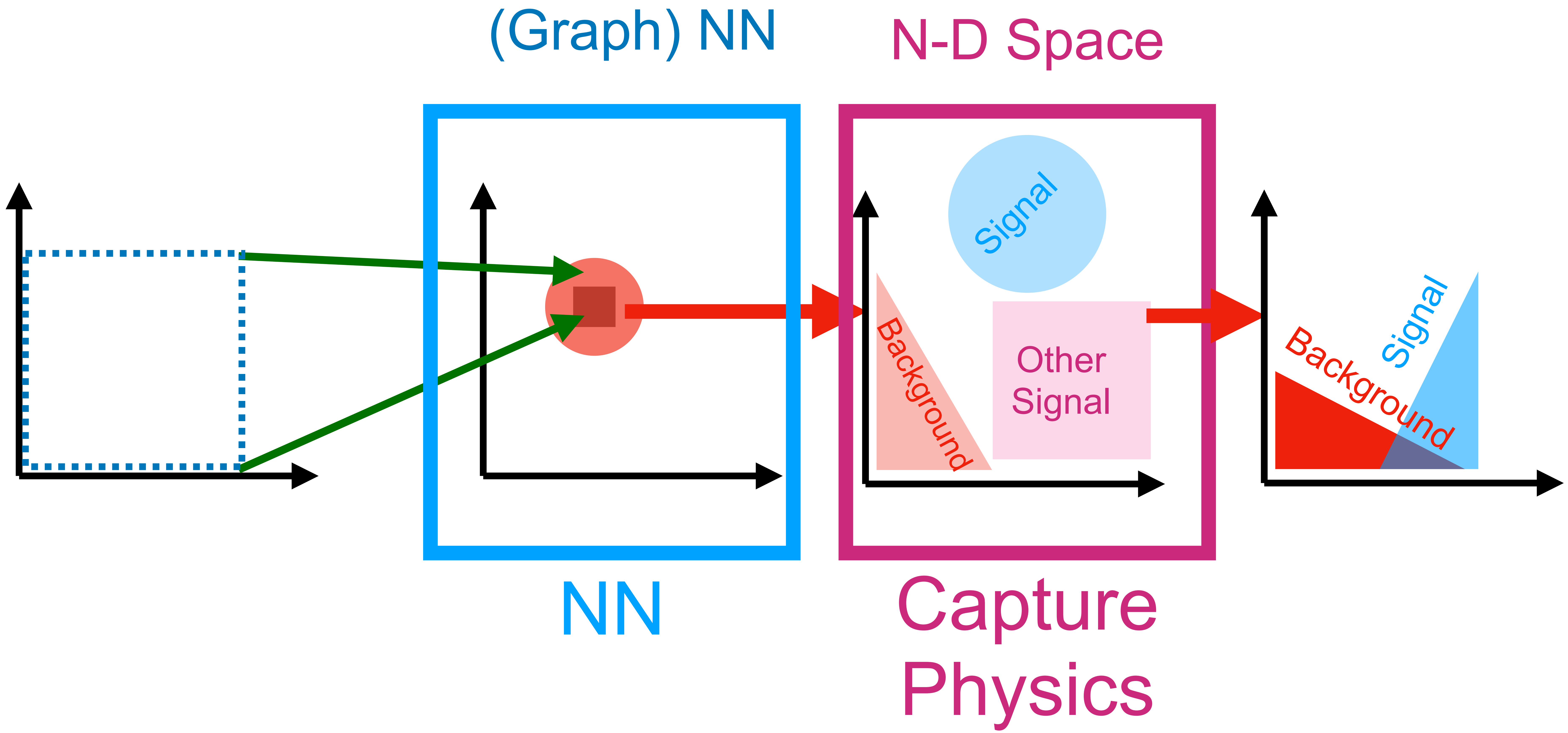
**Training 1: Learn neural embedding
(on a lot of data, for a long time)
On simulation? On data?**



Training 2: Fine tune for specific task (fast, small dataset, simulation)


Theorists
N-D Space
Signal
Background
Other
Signal
Signal
Background
Capture
Physics

We can replace the QCD theorist with a NN
(And it works better)



Masked language modelling

Next-token-prediction

The model is given a sequence of words with the goal of predicting the next word.

Example:
Hannah is a ____

Hannah is a *sister*
Hannah is a *friend*
Hannah is a *marketer*
Hannah is a *comedian*

Masked-language-modeling

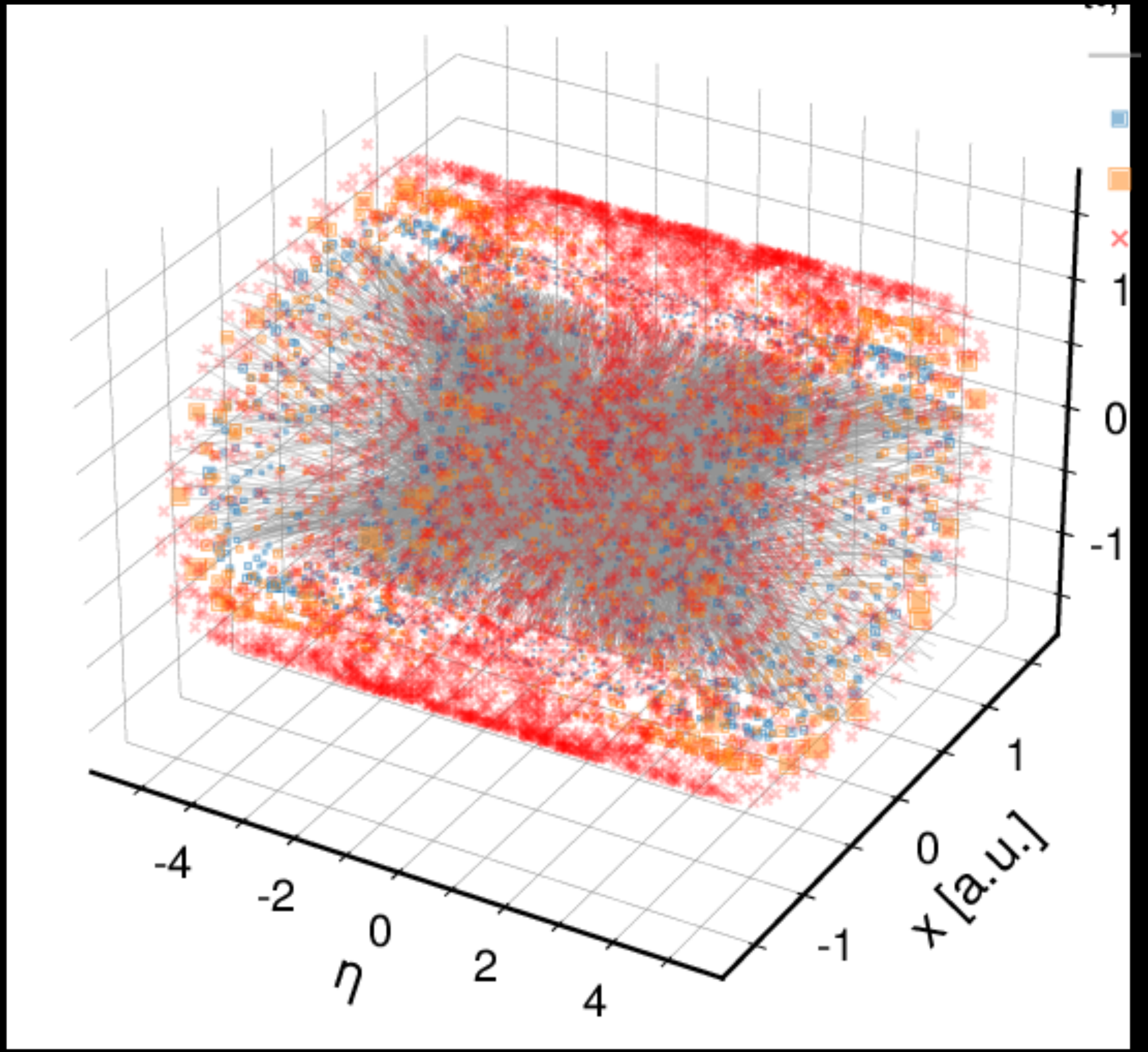
The model is given a sequence of words with the goal of predicting a 'masked' word in the middle.

Example
Jacob [mask] reading

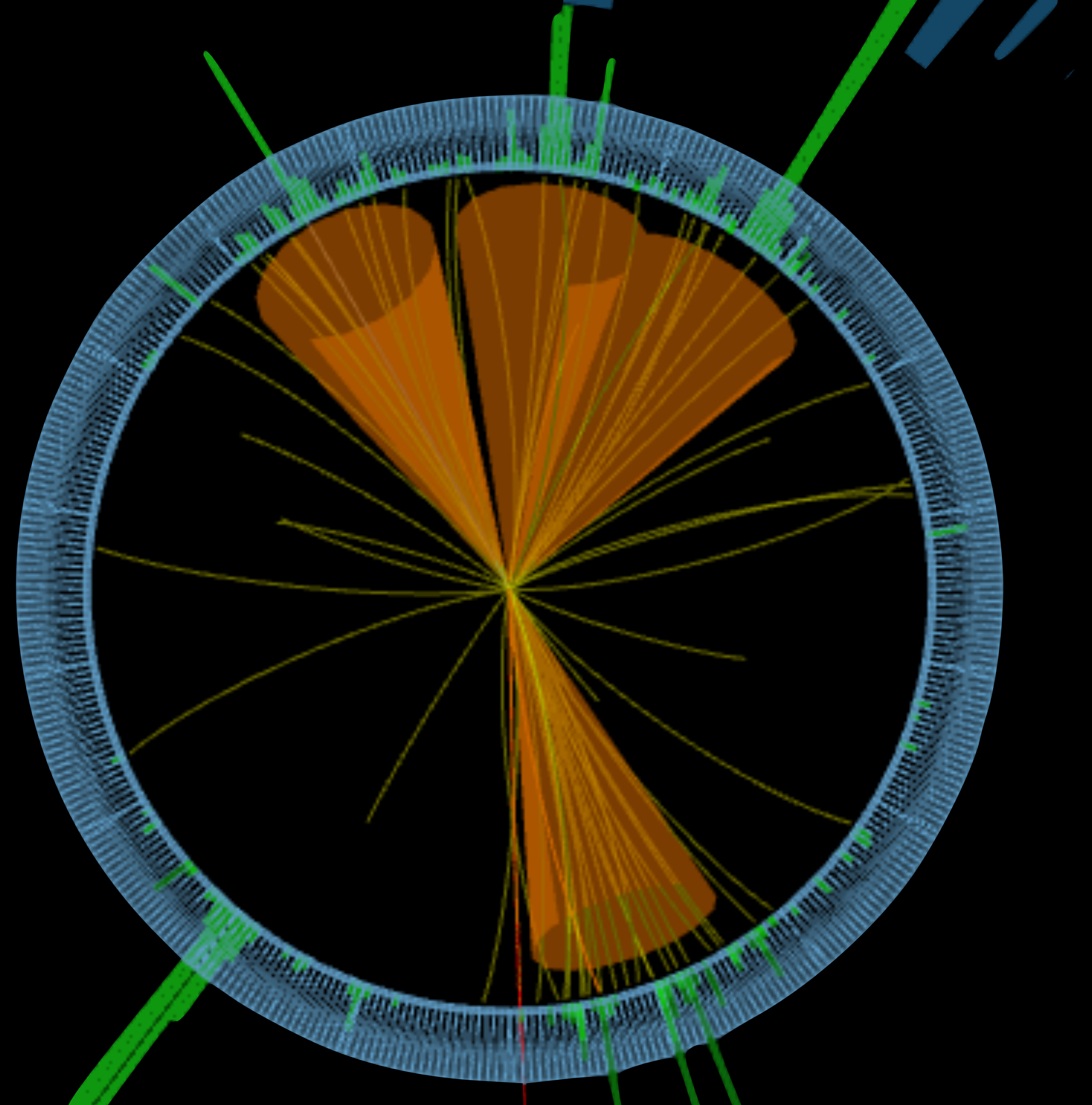
Jacob *fears* reading
Jacob *loves* reading
Jacob *enjoys* reading
Jacob *hates* reading

Self-supervised pre-training

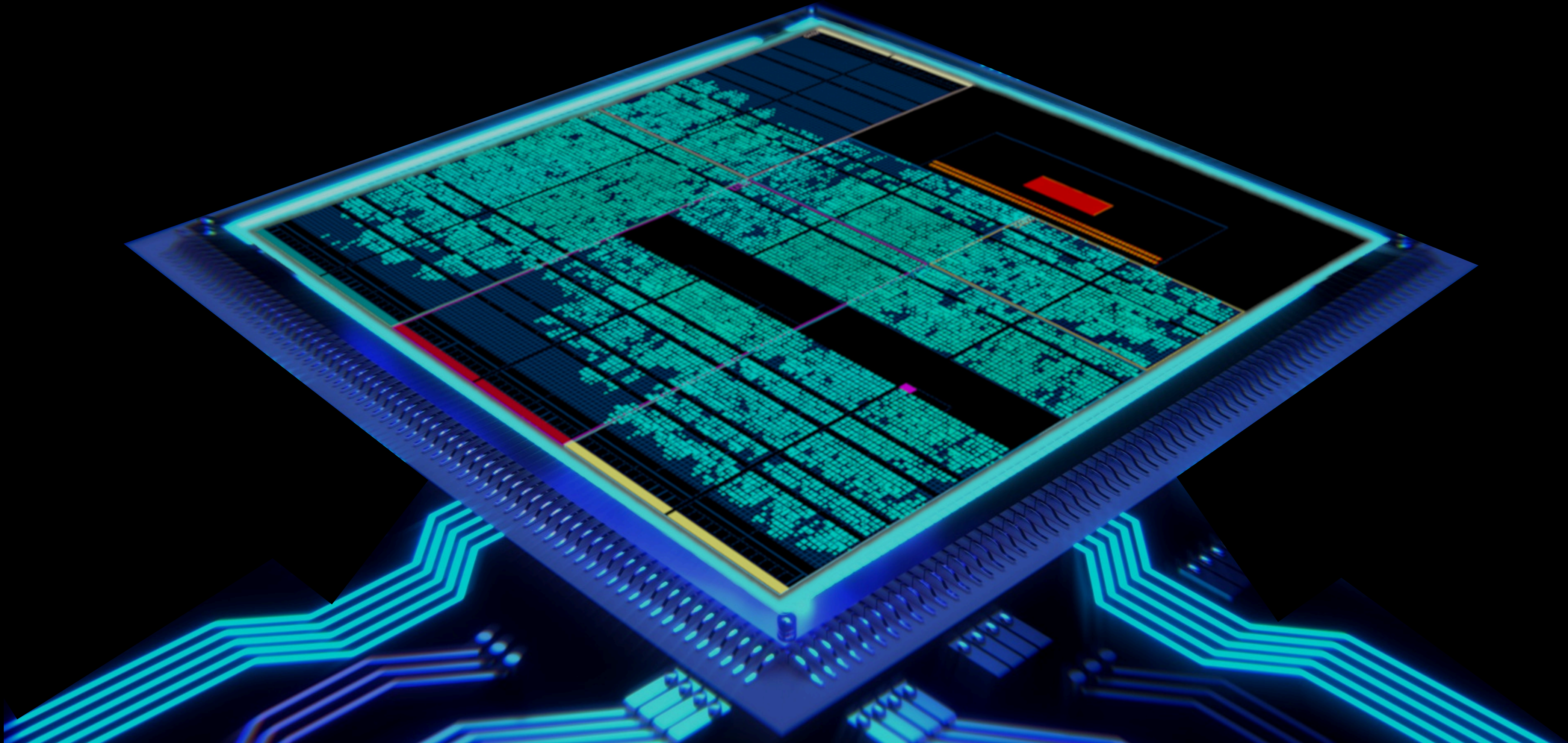
Masked particle modelling



Masked calorimeter pre-training?



Part2:
ML in HEP

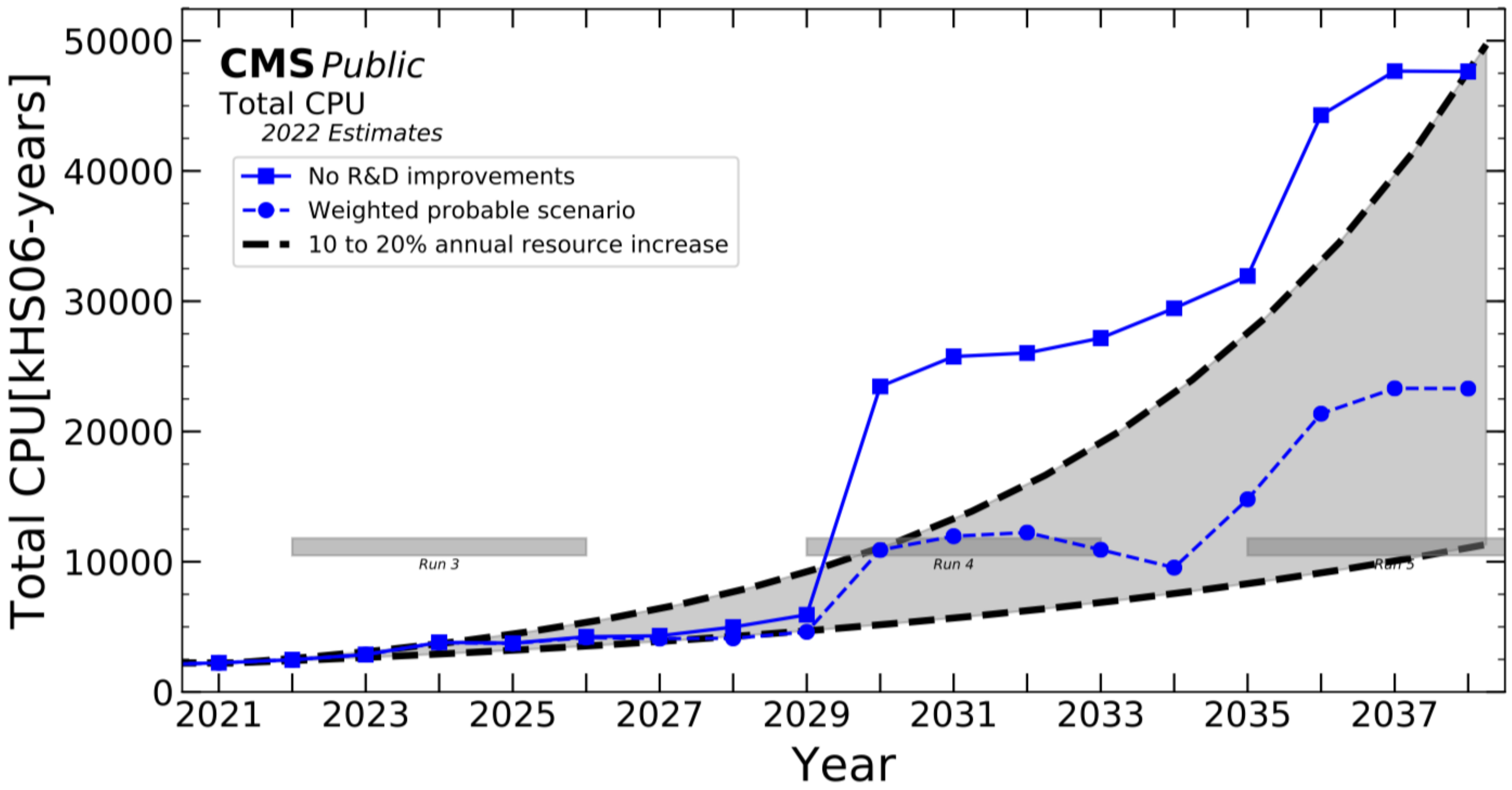


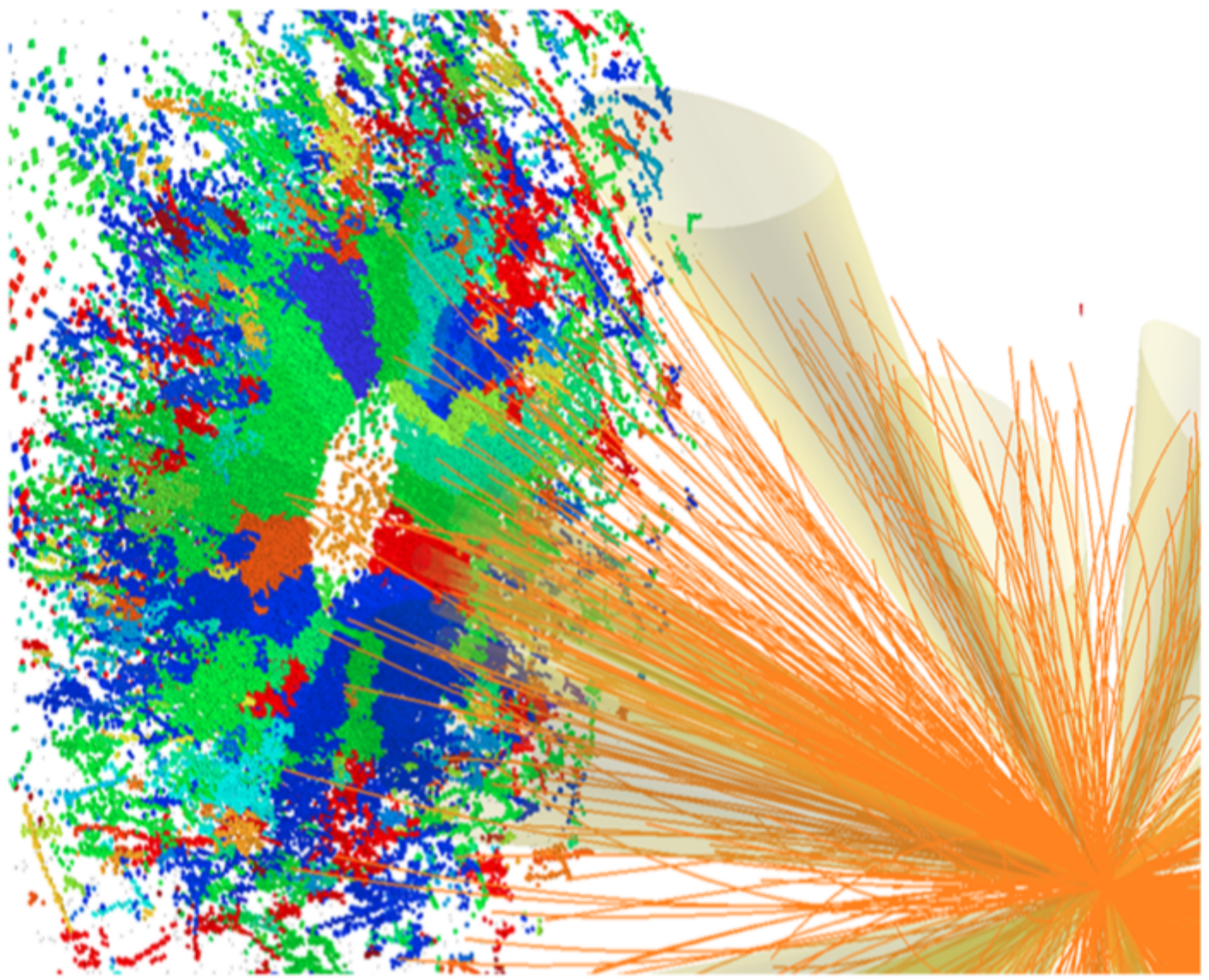
Thea K. Årrestad (ETH Zürich)
thea.aarrestad@cern.ch
thaarres.github.io

QCD School 2024

__CMS Offline Computing Results__

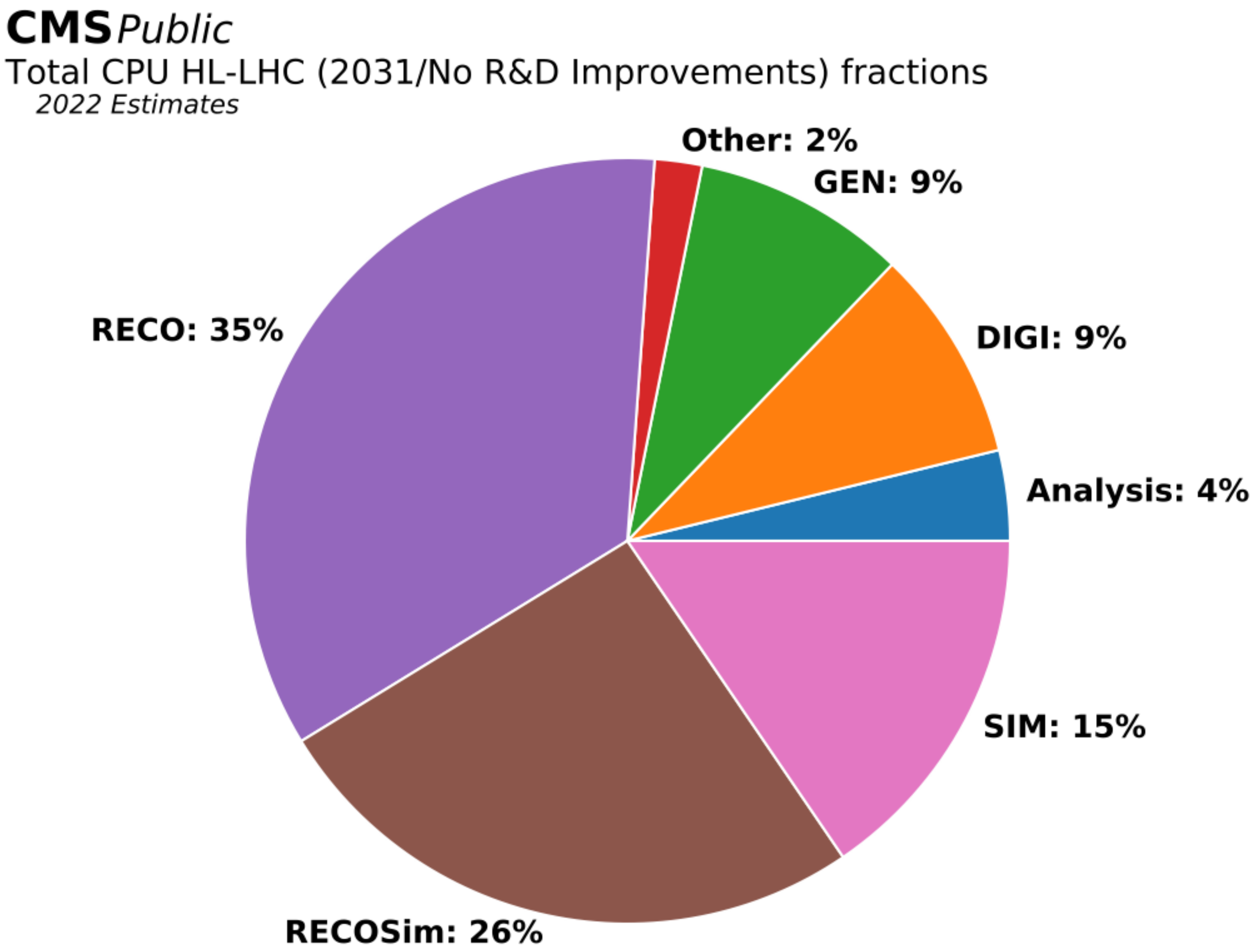
ML for simulation





HL-LHC, Simulation of CMS HGCAL with 140 PU

CMS Offline Computing Results

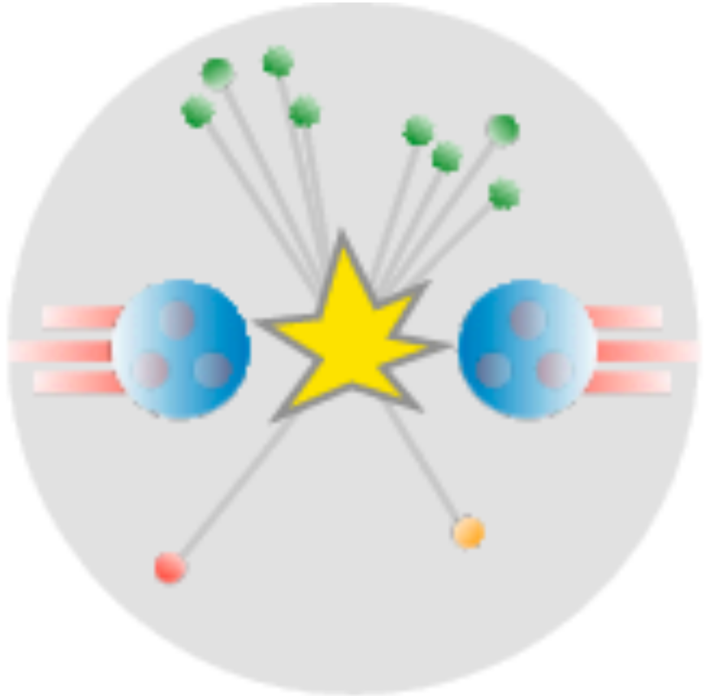


$O(10)$
$O(10^3)$
$O(10^{10})$
10^{-18}m
10^{-15}m
10^{-6}m
100m
2T

$O(10)$ $O(10^3)$ $O(10^{10})$

10^{-18}m 10^{-15}m 10^{-6}m 100m

GEN



pp collisions up to production of stable particles [Easy & Fast]

SIM

detector response simulation [Hard & Slow]

GEANT4 A SIMULATION TOOLKIT

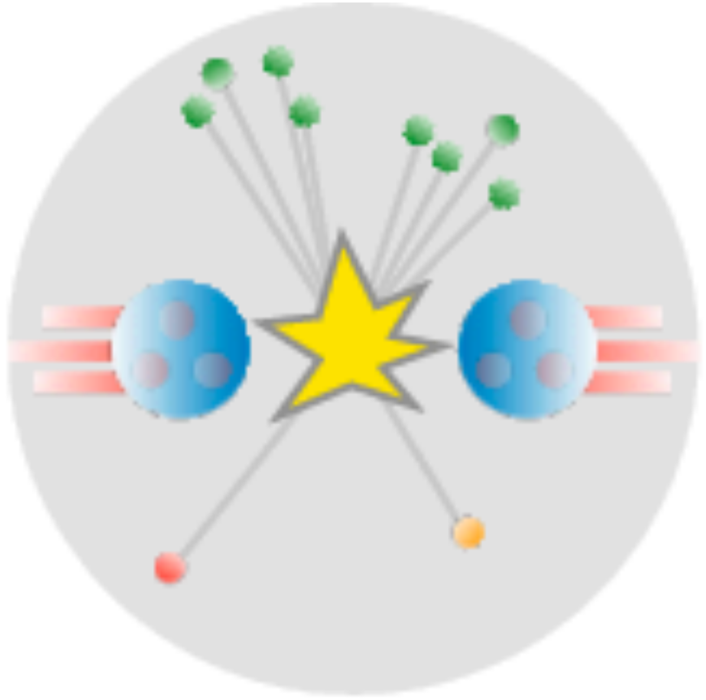
DIGI+RECO

Energy deposits→digital signals→reconstructed by the reconstruction software [Hard & Slow]

$O(10)$ $O(10^3)$ $O(10^{10})$

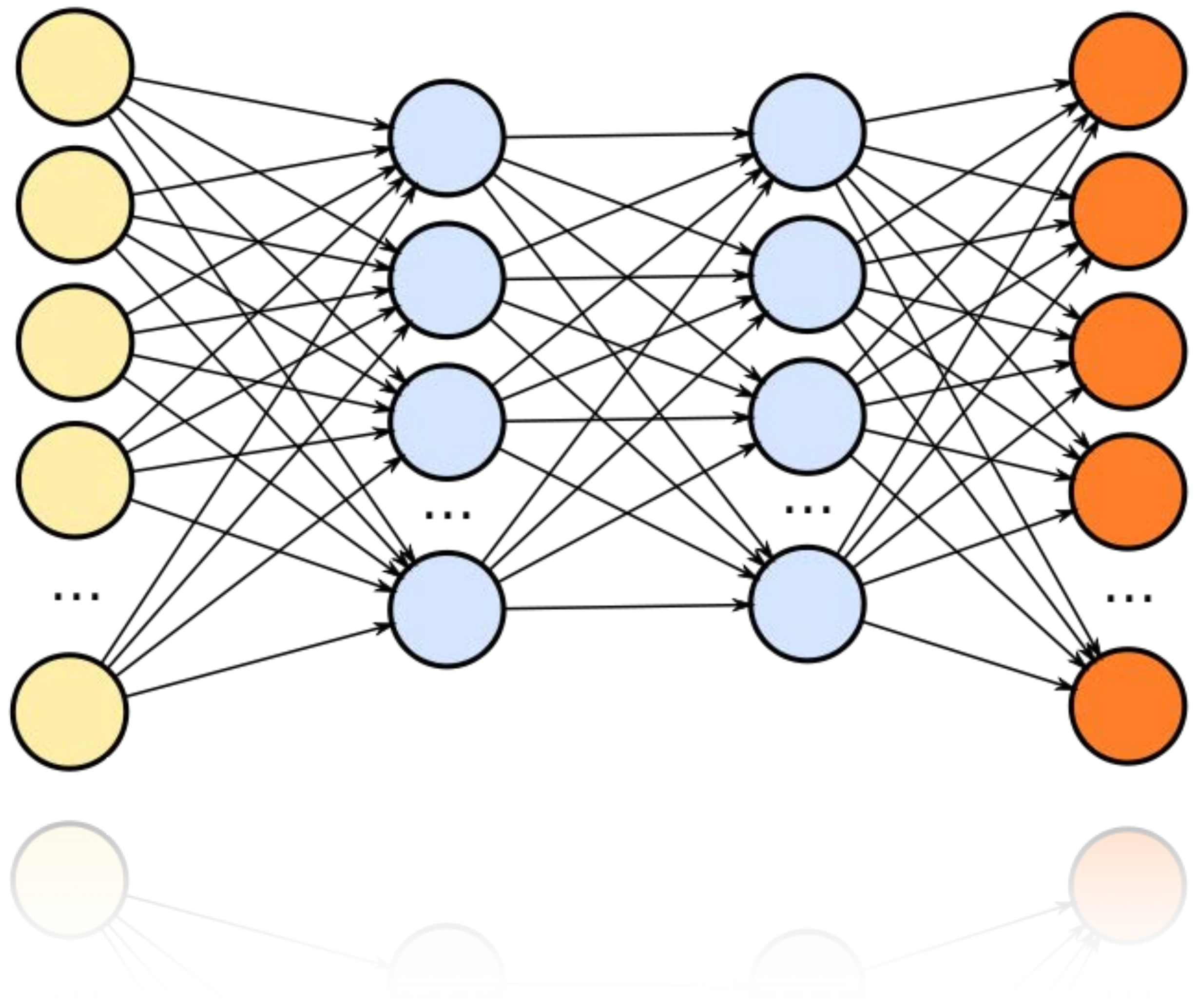
10^{-18}m 10^{-15}m 10^{-6}m 100m

GEN



pp collisions up to production of stable particles [Easy & Fast]

SIM



DIGI+RECO

Energy deposits→digital signals→reconstructed by the reconstruction software [Hard & Slow]

$\mathcal{O}(10)$ $\mathcal{O}(10^3)$ $\mathcal{O}(10^{10})$

10^{-18}m 10^{-15}m 10^{-6}m 100m

GEN **SIM** **DIGI+RECO**

GEANT4 → 10^{10} events **SLOW but ACCURATE**

GEANT4 → 10^5 events → Surrogate model → 10^{10} events

(GAN, VAE, Normalizing Flow, …)
Learn underlying distribution of GEANT4 events

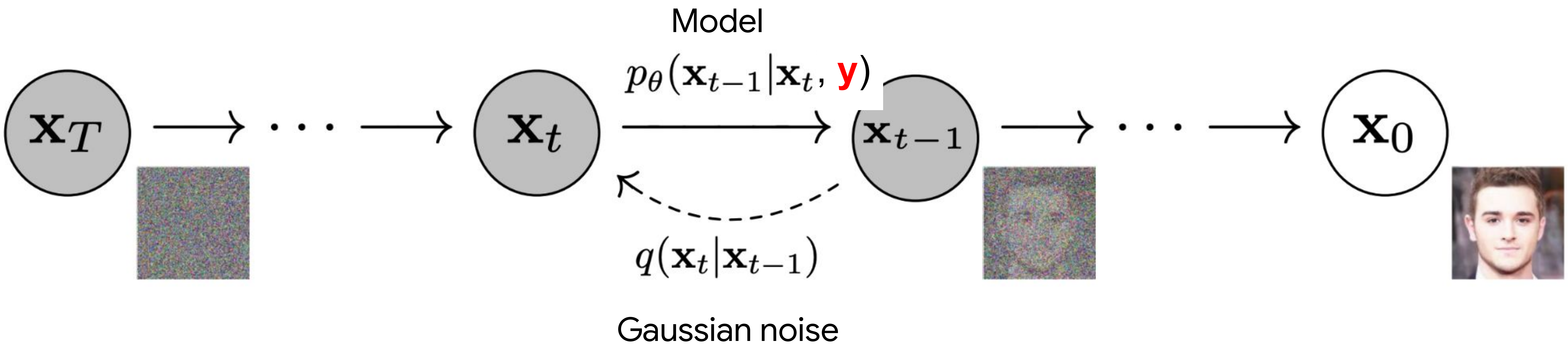
FAST and ACCURATE?

ML methods can provide fast and accurate "surrogate models" for GEANT4 etc

p
pro
par

posits→digital
constructed by
uction software
[Hard & Slow]

Diffusion models

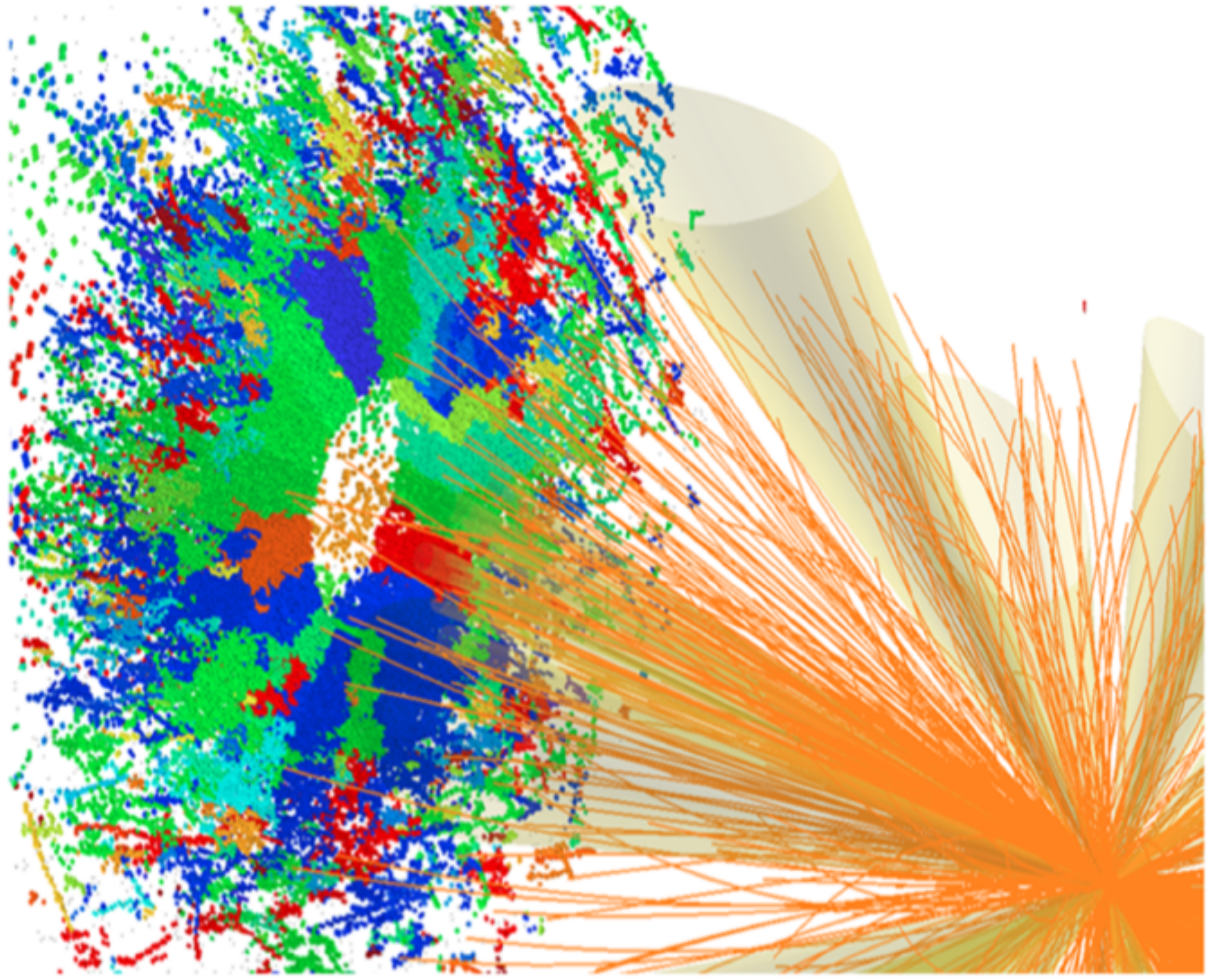


Dall-e 2

an espresso machine that makes coffee from human souls, artstation[2]

Dall-e 2

vibrant portrait painting of Salvador Dalí with a robotic half face



https://arxiv.org/pdf/2303.05376.pdf