Machine Learning and High-Energy Physics

Sascha Diefenbacher, QuarkNet, July 10th 2024

Overview

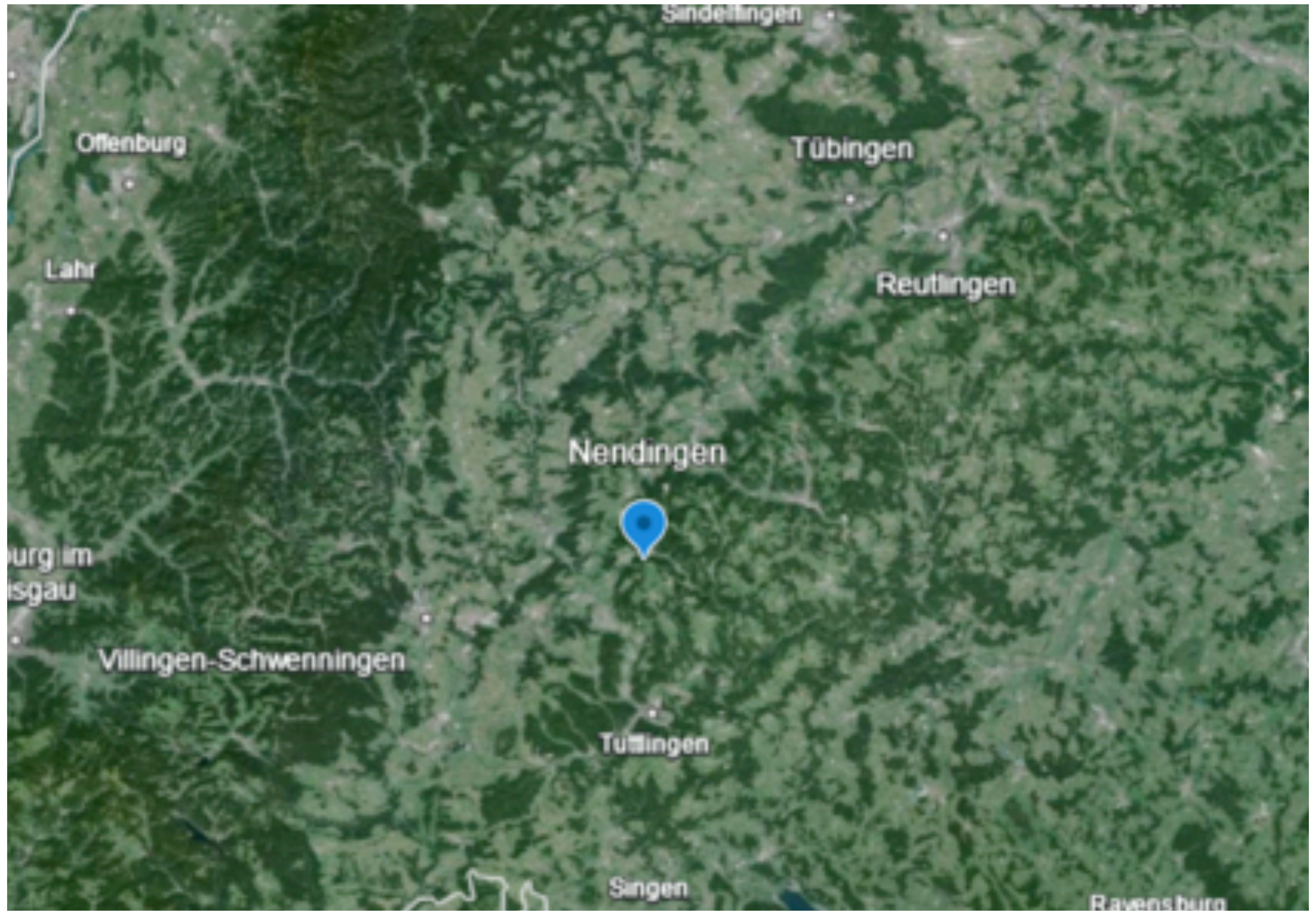
- My Career Path
- High-Energy Physics
- Machine Learning
- Combination Machine Learning and High-Energy Physics

My Career Path

(Not to scale)

I’m from Germany

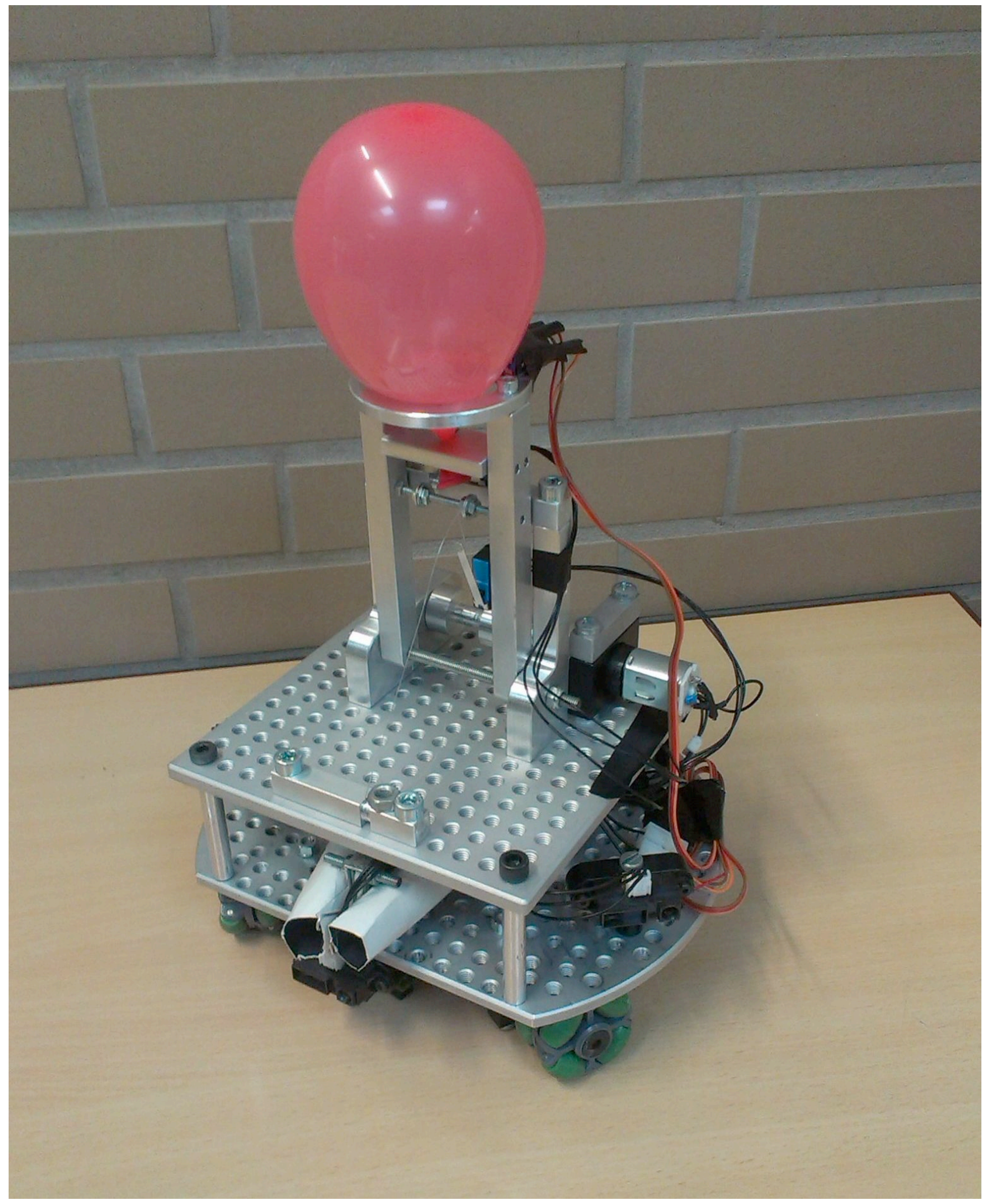
I’m from Germany
(Yes, thats where the accent’s from)



Born in Nendingen (Tiny town in southern Germany)

Early Live

- Always interested in science
- Did all the extra-curricular stuff during high school
  - “Engineering School”
  - “Robotics club”
  - “Convincing your physics teacher to spend a full lesson building a magnetic accelerator”

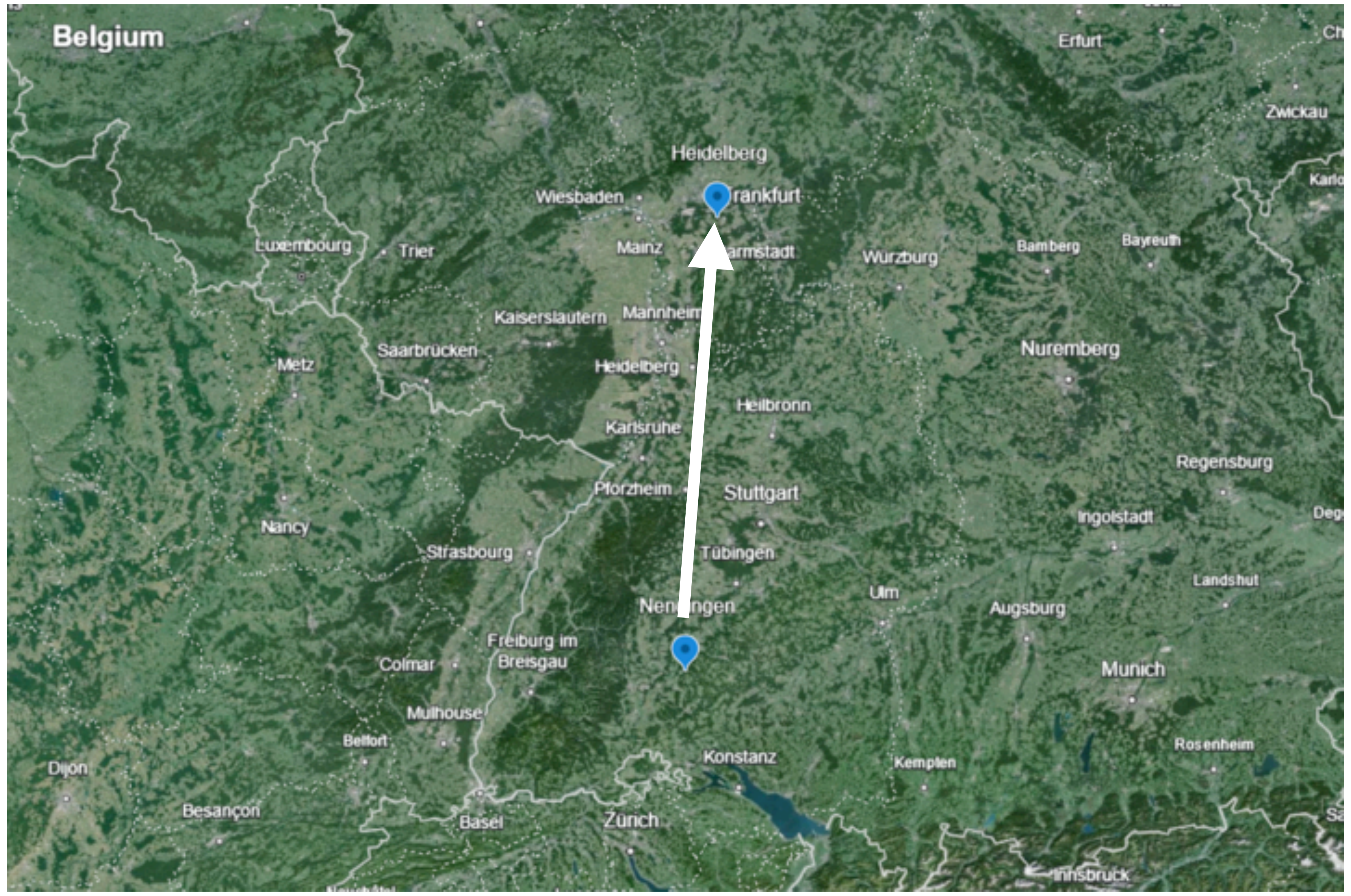


Early Live

- Coding
  - After school coding classes
  - Coding camps
  - Community college courses

Early Live

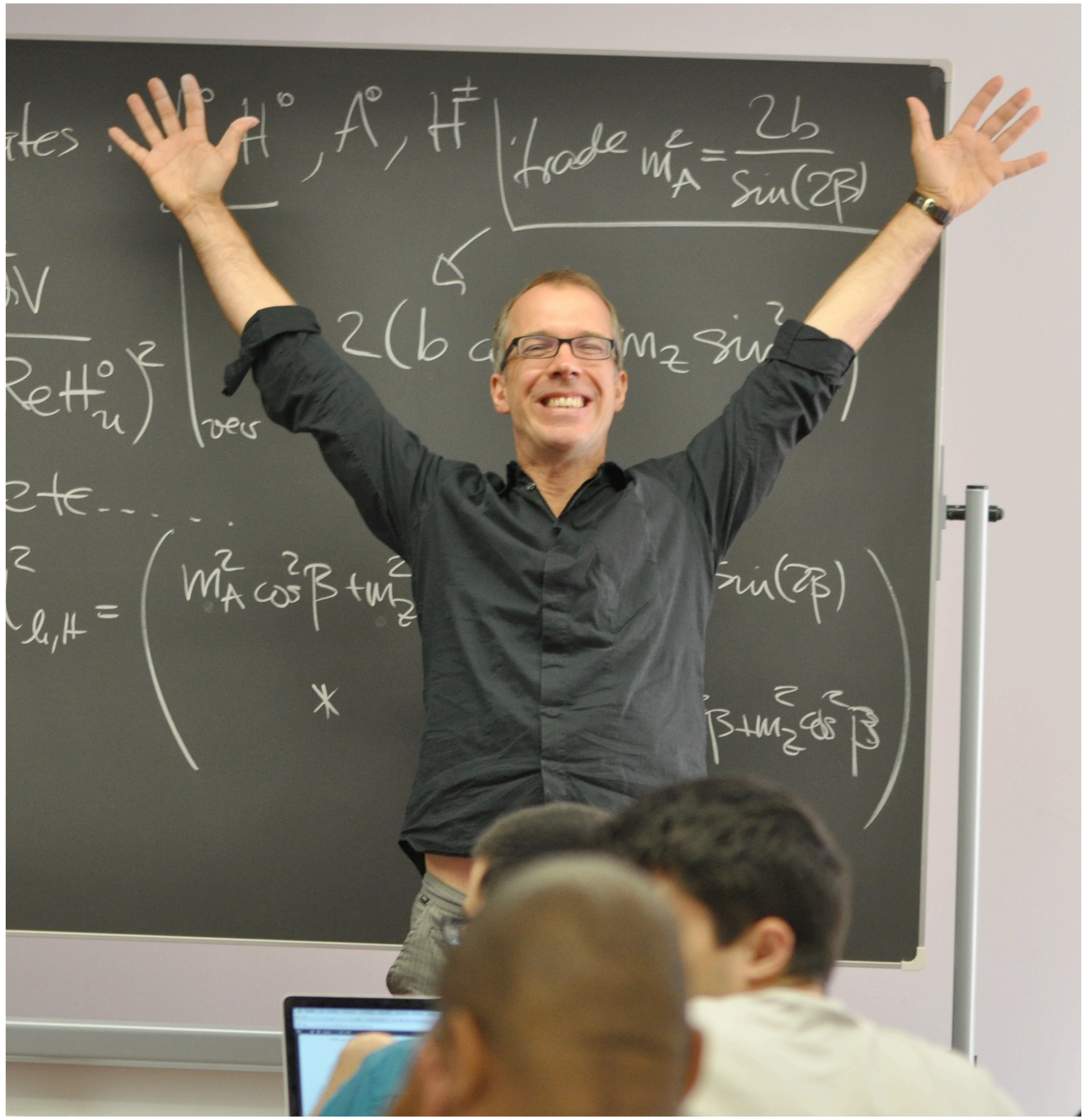
- Coding
    - ~~After school coding classes~~
    - ~~Coding camps~~
    - ~~Community college courses~~
    - **Modding video games**



Undergrad: Northward to Heidelberg (less tiny town in southern Germany)

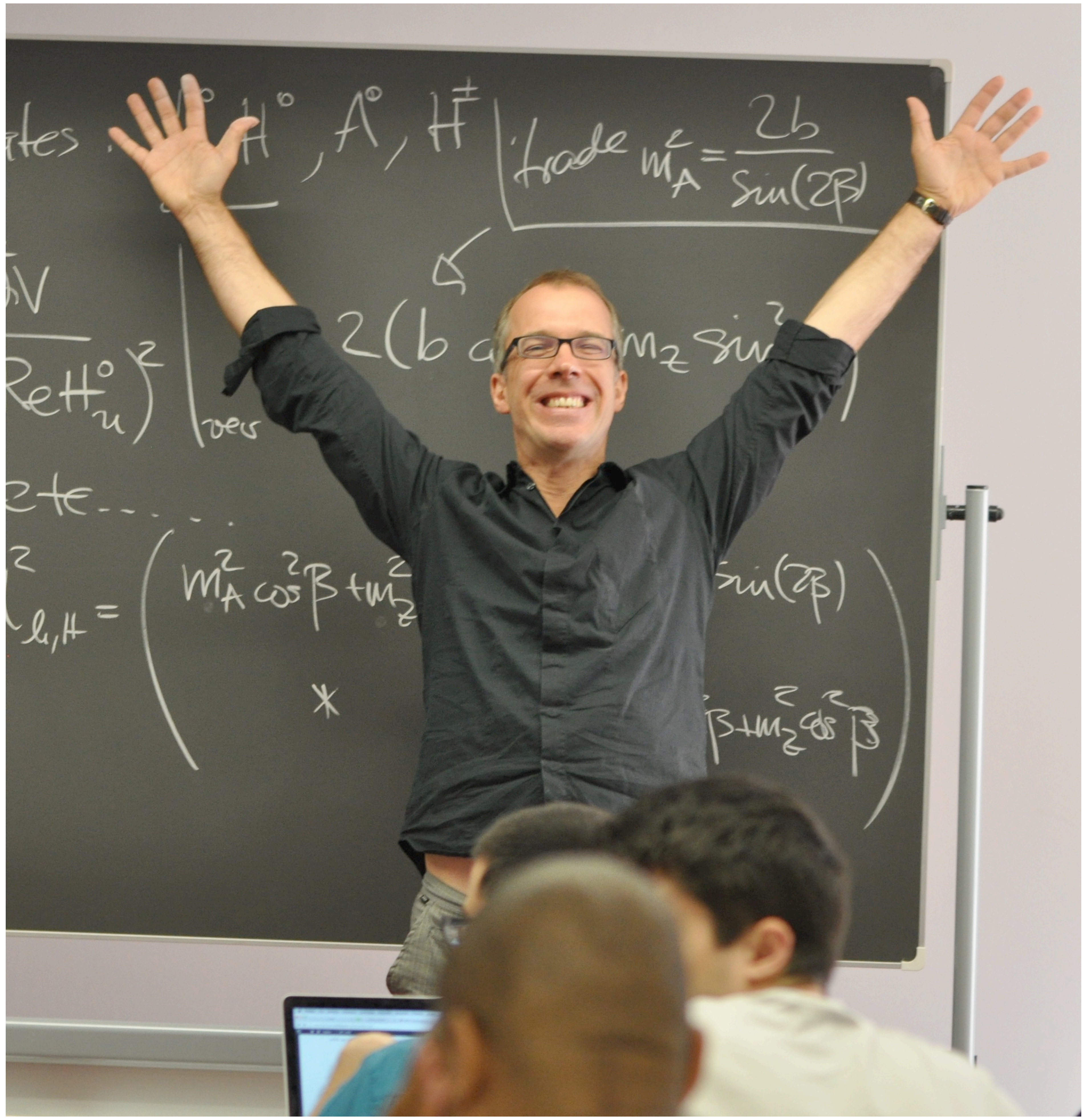
Undergrad

- Bachelors Thesis with Tilman Plehn in 2017
    - Phenomenologist at Institute for Theoretical Physics in Heidelberg
    - Dark Matter and LHC phenomenology and theory



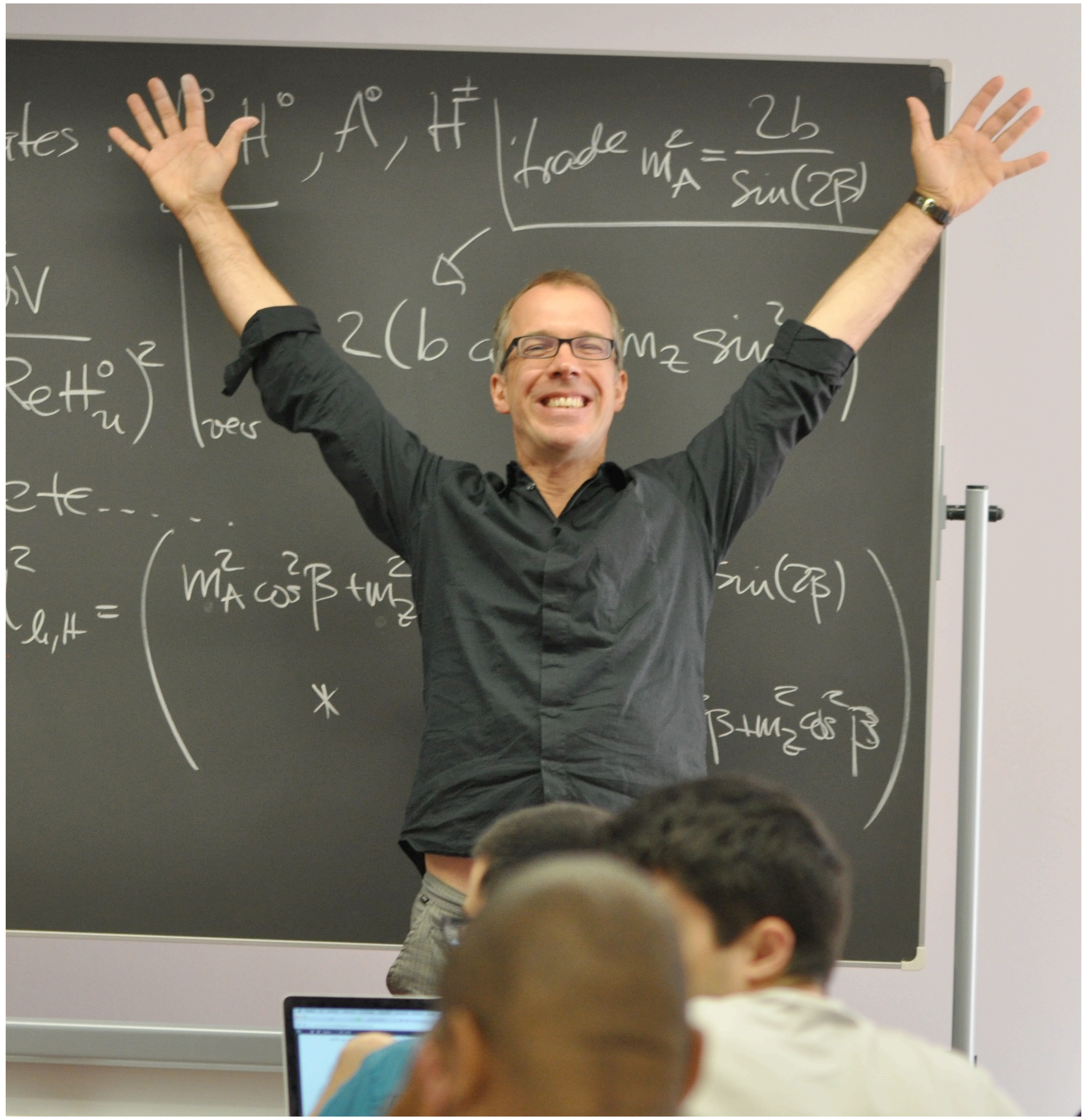
Undergrad

- Bachelors Thesis with Tilman Plehn in 2017
    - “There’s this new thing called Machine Learning, is that something you’d be interested in?”



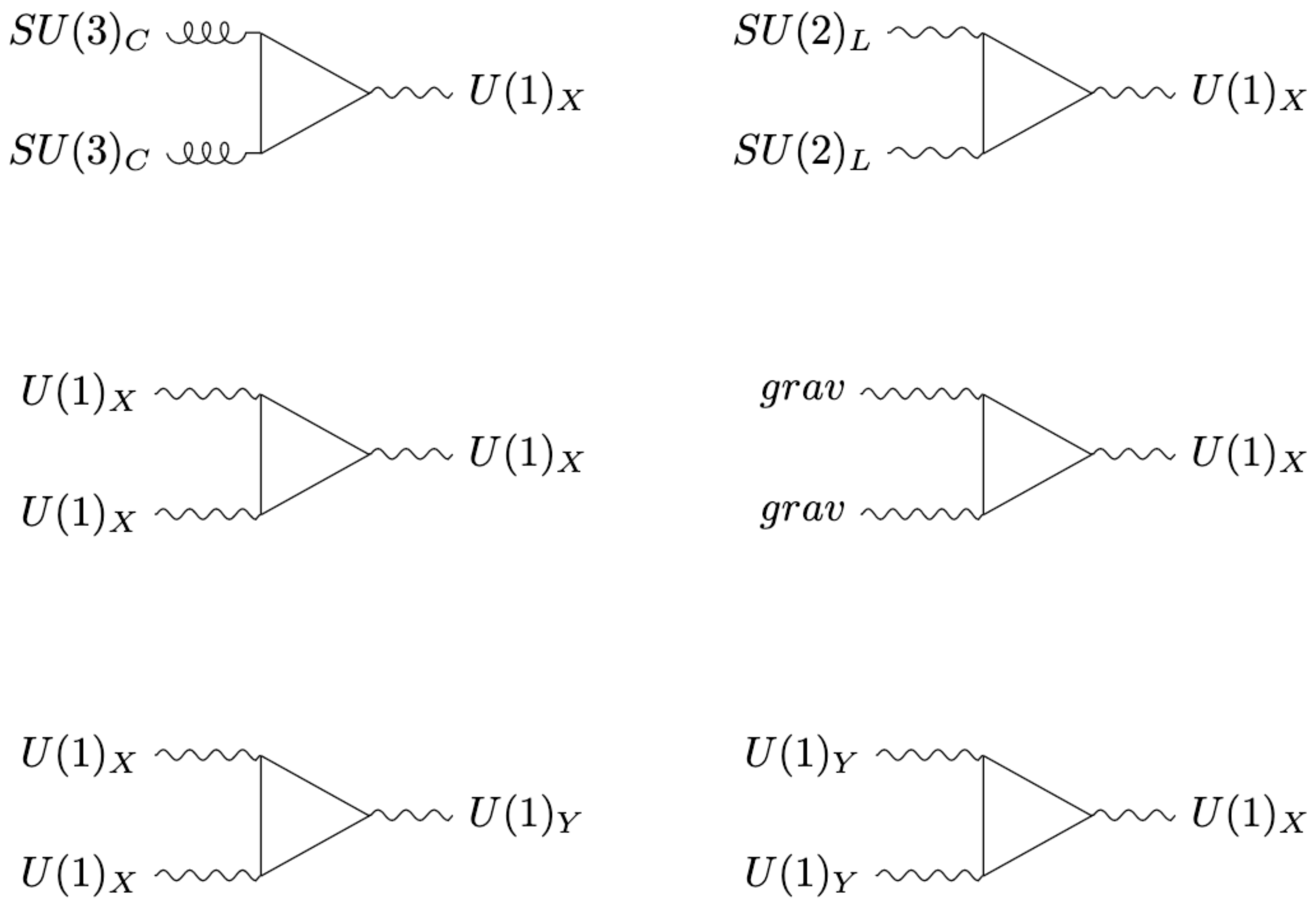
Undergrad

- Bachelors Thesis with Tilman Plehn in 2017
  - “There’s this new thing called Machine Learning, is that something you’d be interested in?”
  - “Nah”



Undergrad

- Bachelors Thesis with Tilman Plehn in 2017
  - “There’s this new thing called Machine Learning, is that something you’d be interested in?”
  - “Nah”
  - Thesis on dark matter theory

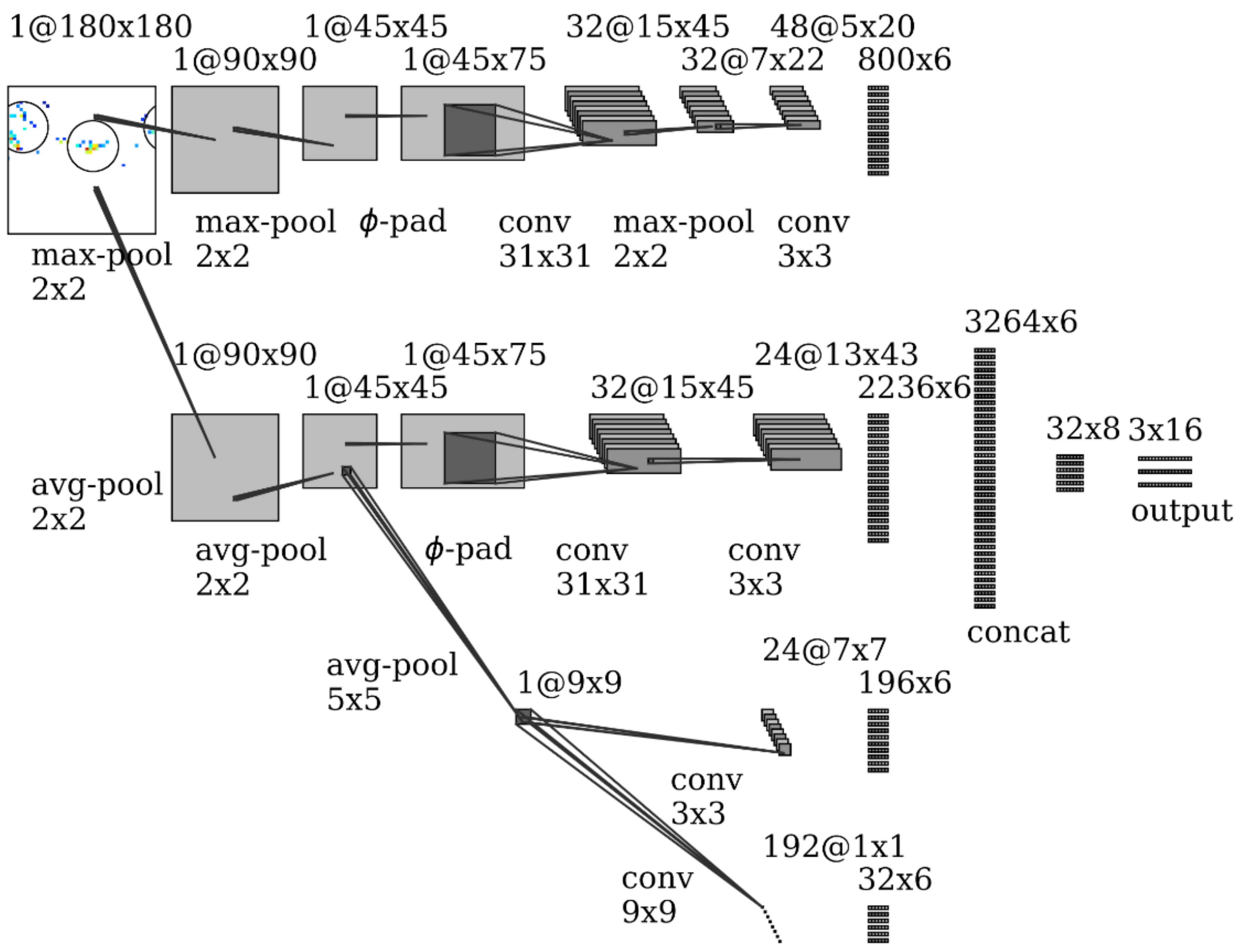


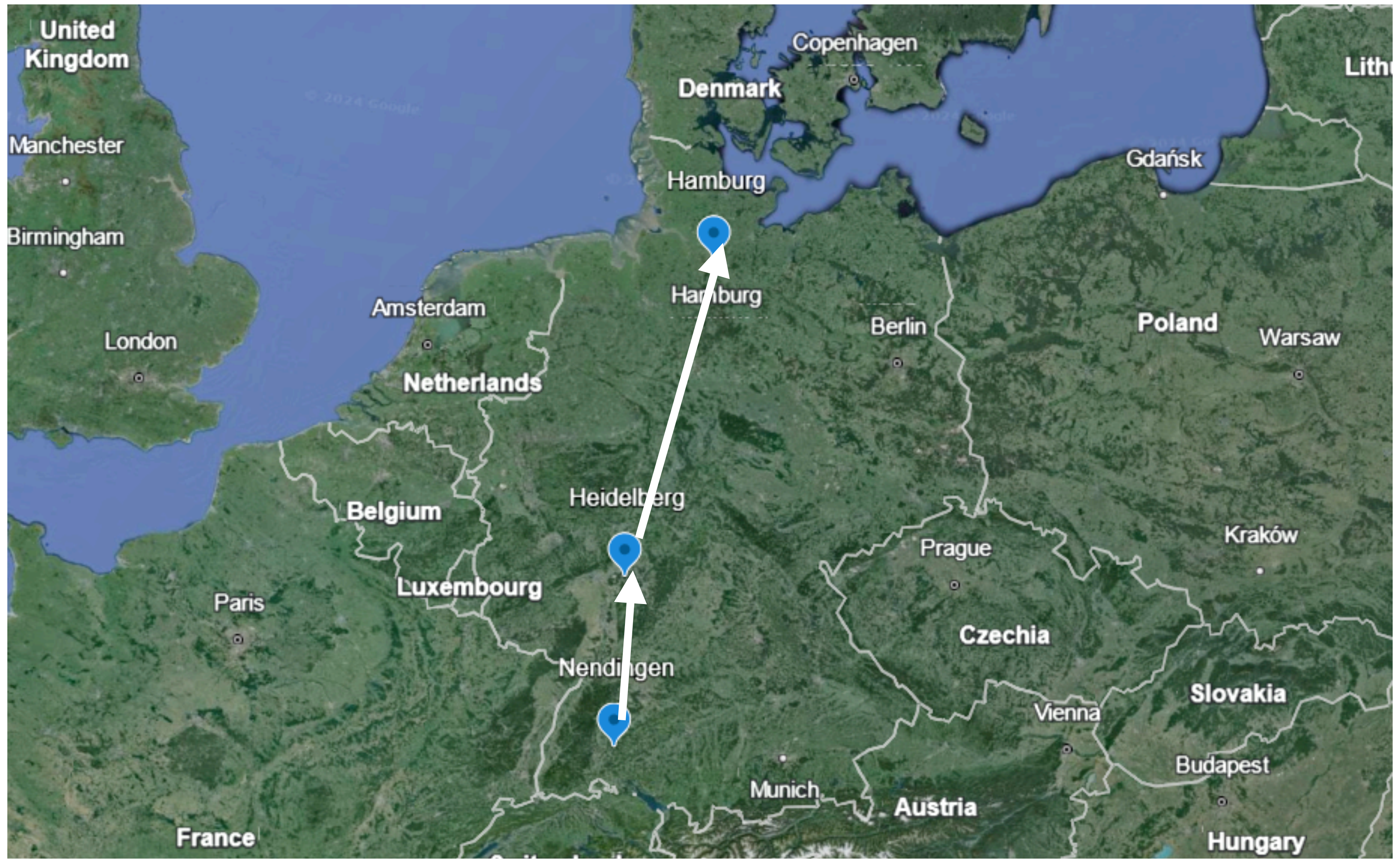
Undergrad

- Masters Thesis with Tilman Plehn (again) in 2019
  - Whole group was doing machine learn by then

Undergrad

- Masters Thesis with Tilman Plehn (again) in 2019
  - Whole group was doing machine learn by then
  - Thesis on ML for classification in high energy physics
  - Wild West era of ML in HEP





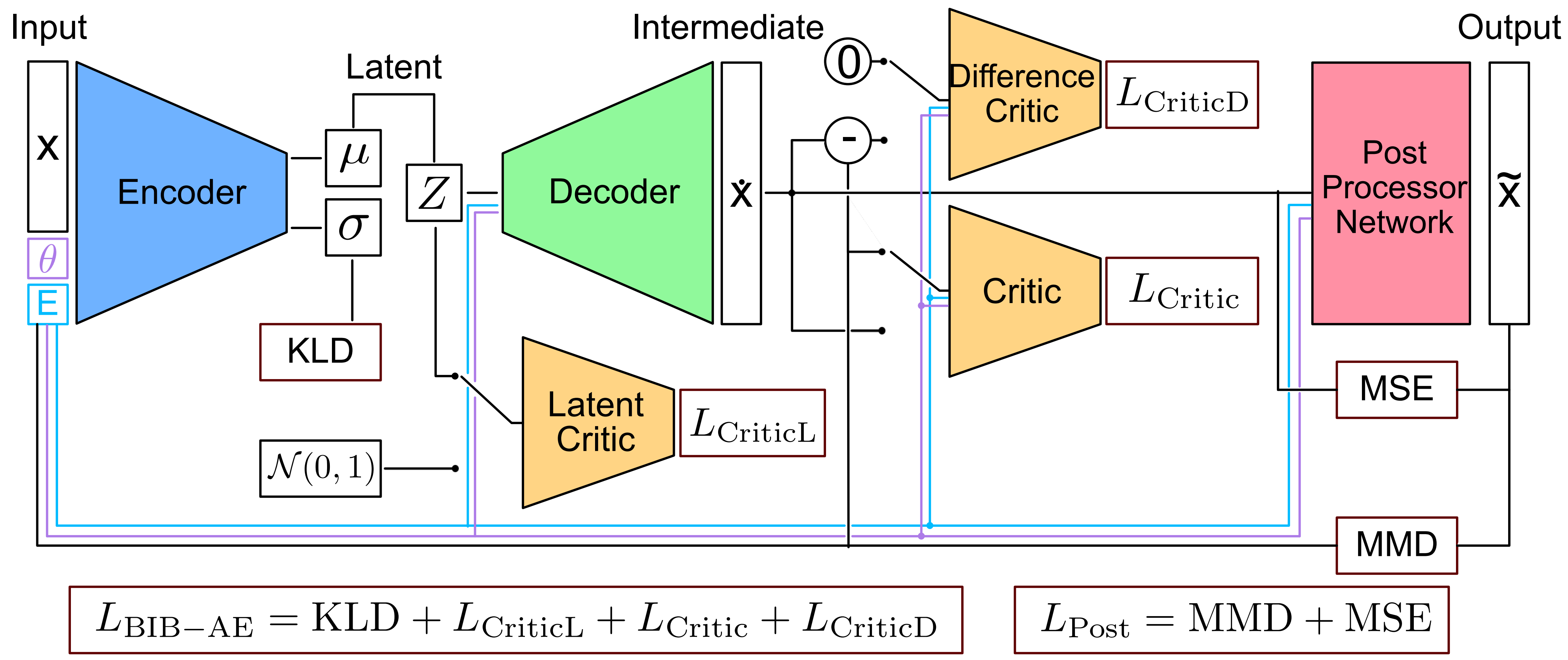
Grad School: Further north to Hamburg (2nd larges city in Germany)

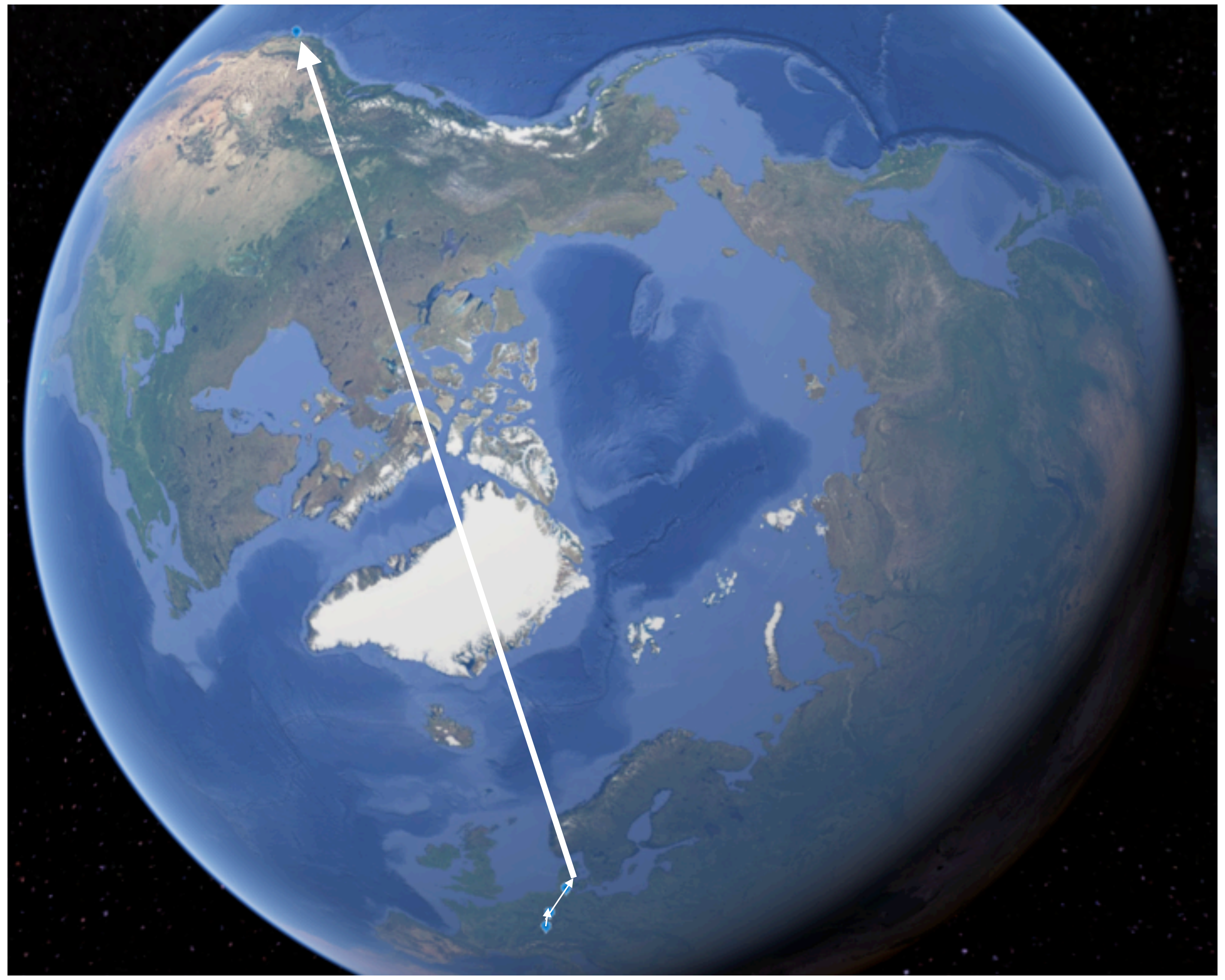
Grad School

- PhD thesis with Gregor Kasiezcka
  - CMS experimentalist at University of Hamburg

Grad School

- PhD thesis with Gregor Kasiezcka
  - CMS experimentalist at University of Hamburg
  - Dove deep into ML in HEP
  - Extremely fast developing field
  - Cutting edge research is exhilarating





Postdoc: Further north (?) to Berkeley

Postdoc

- With Ben Nachman at LBL in 2023 (ATLAS experimentalist at LBL)
  - Jump further into experiment
  - Joined ATLAS experiment collaboration

Break for Questions

High Energy Physics

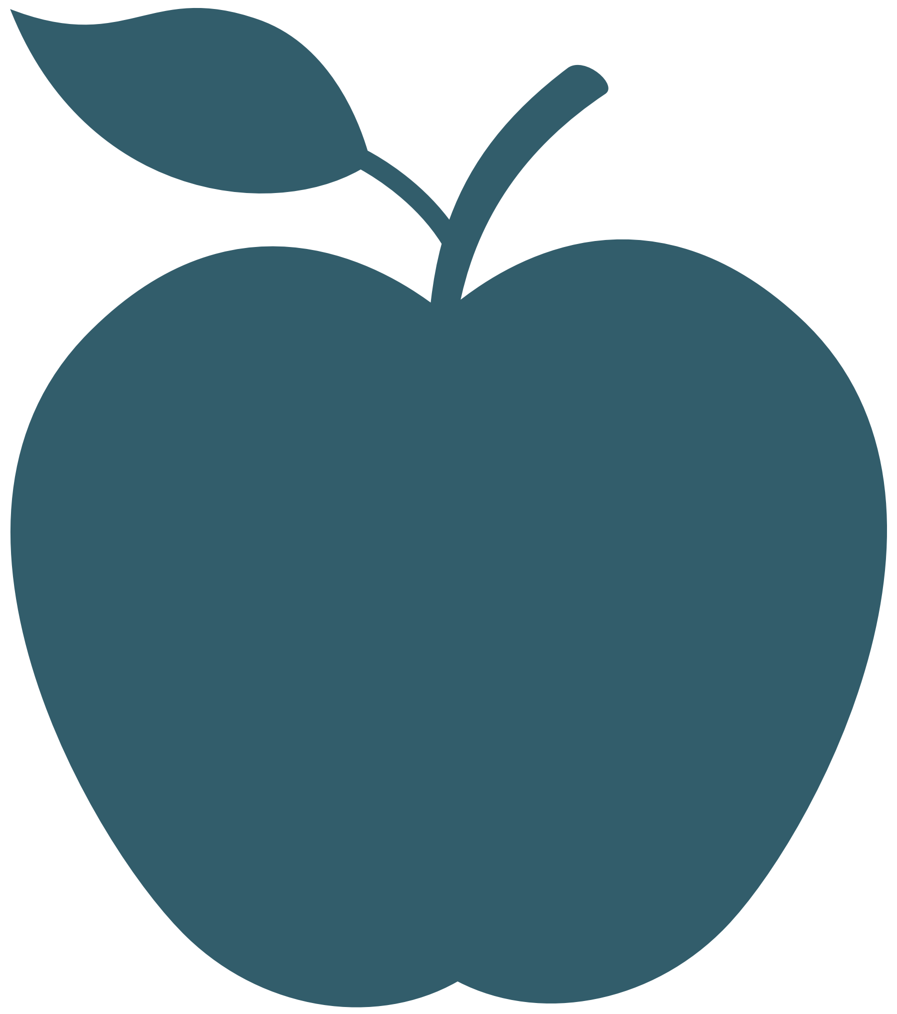
(Cooler phrase for particle physics)

High Energy Physics

- Studying the fundamental forces between the smallest existing particles
- Describe them in mathematical terms
  - Define Hypothesis
  - Make prediction
  - Test prediction

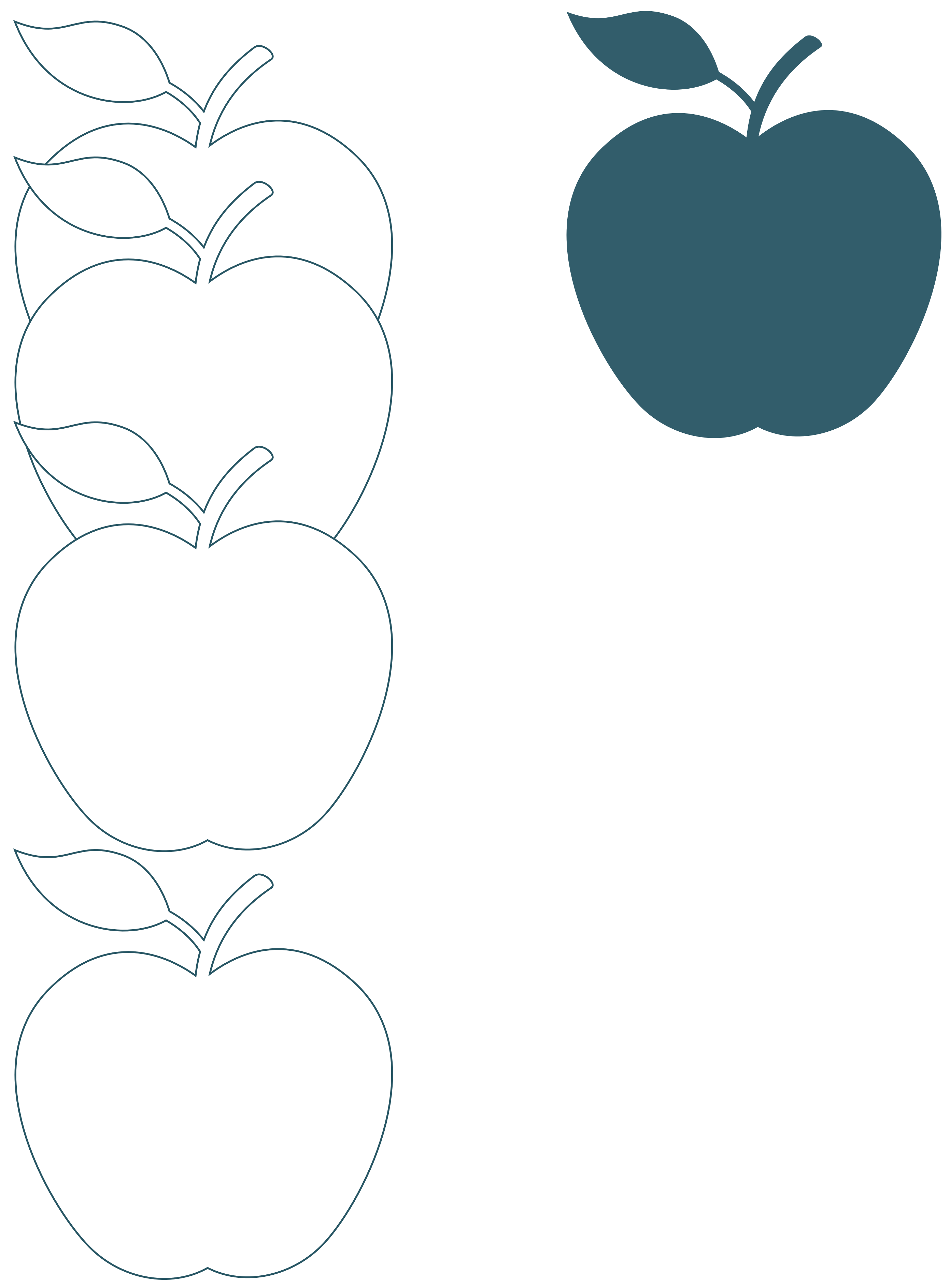
High Energy Physics

- Basic example
  - Hypothesis: if this apple drops, it will follow a path described by $x(t) = x_0 - \frac{1}{2}gt^2$



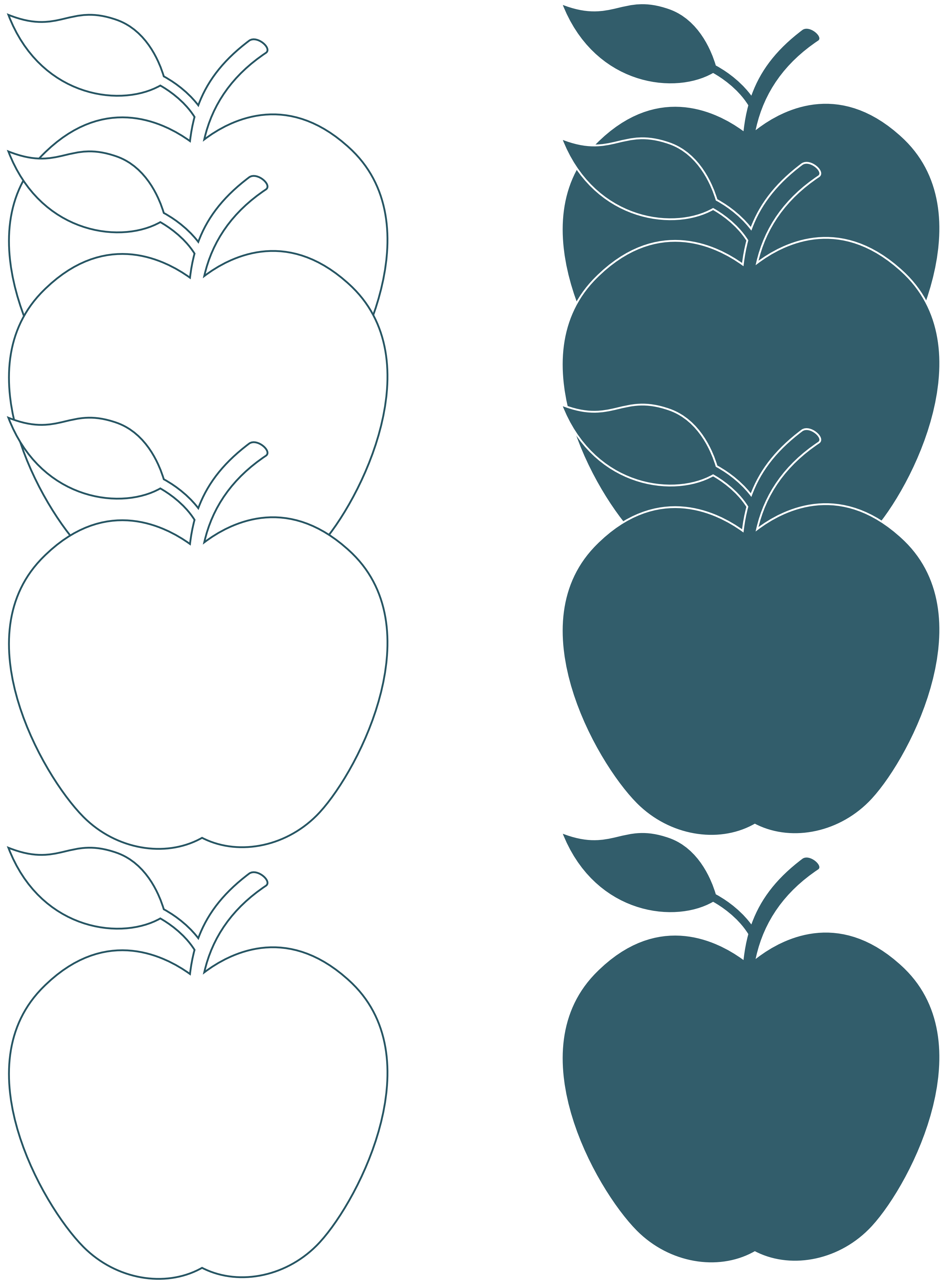
High Energy Physics

- Basic example
  - Hypothesis: if this apple drops, it will follow a path described by $x(t) = x_0 - \frac{1}{2}gt^2$
  - Predict point on path



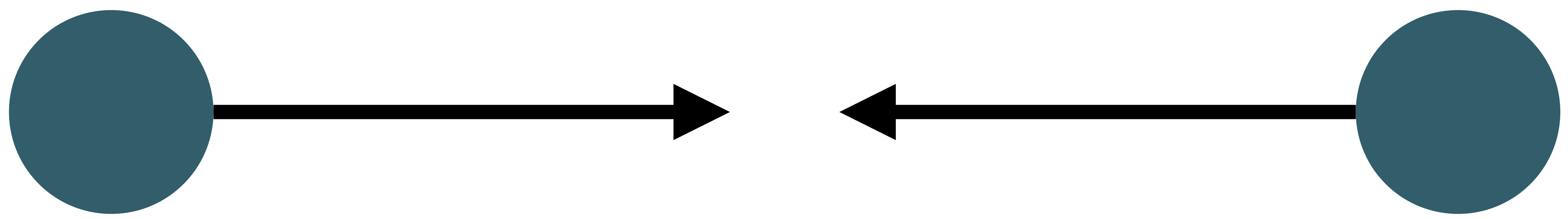
High Energy Physics

- Basic example
  - Hypothesis: if this apple drops, it will follow a path described by $x(t) = x_0 - \frac{1}{2}gt^2$
  - Predict point on path
  - Measure real path and compare to prediction



High Energy Physics

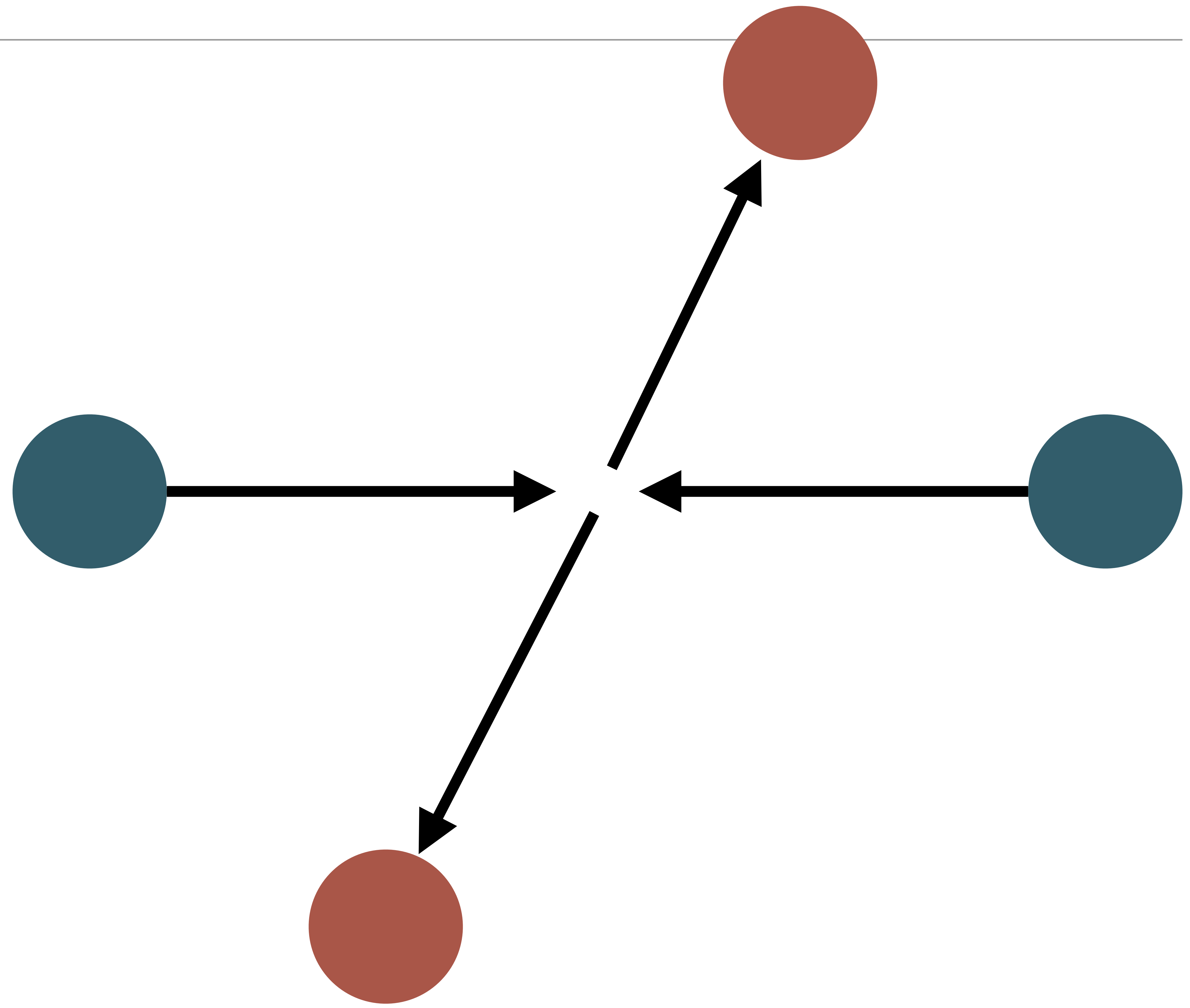
- HEP example
  - Hypothesis: if I collide two particles at nearly the speed of light, they will interact based on this model



$$
\begin{aligned}
\mathscr{L} = &-\frac{1}{4}F_{\mu\nu}F^{\mu\nu} \\
&+i\bar{\psi}D\psi + h.c. \\
&+\bar{\psi}_i y_{ij}\psi_j\phi + h.c. \\
&+|D_\mu\phi|^2 - V(\phi)
\end{aligned}
$$

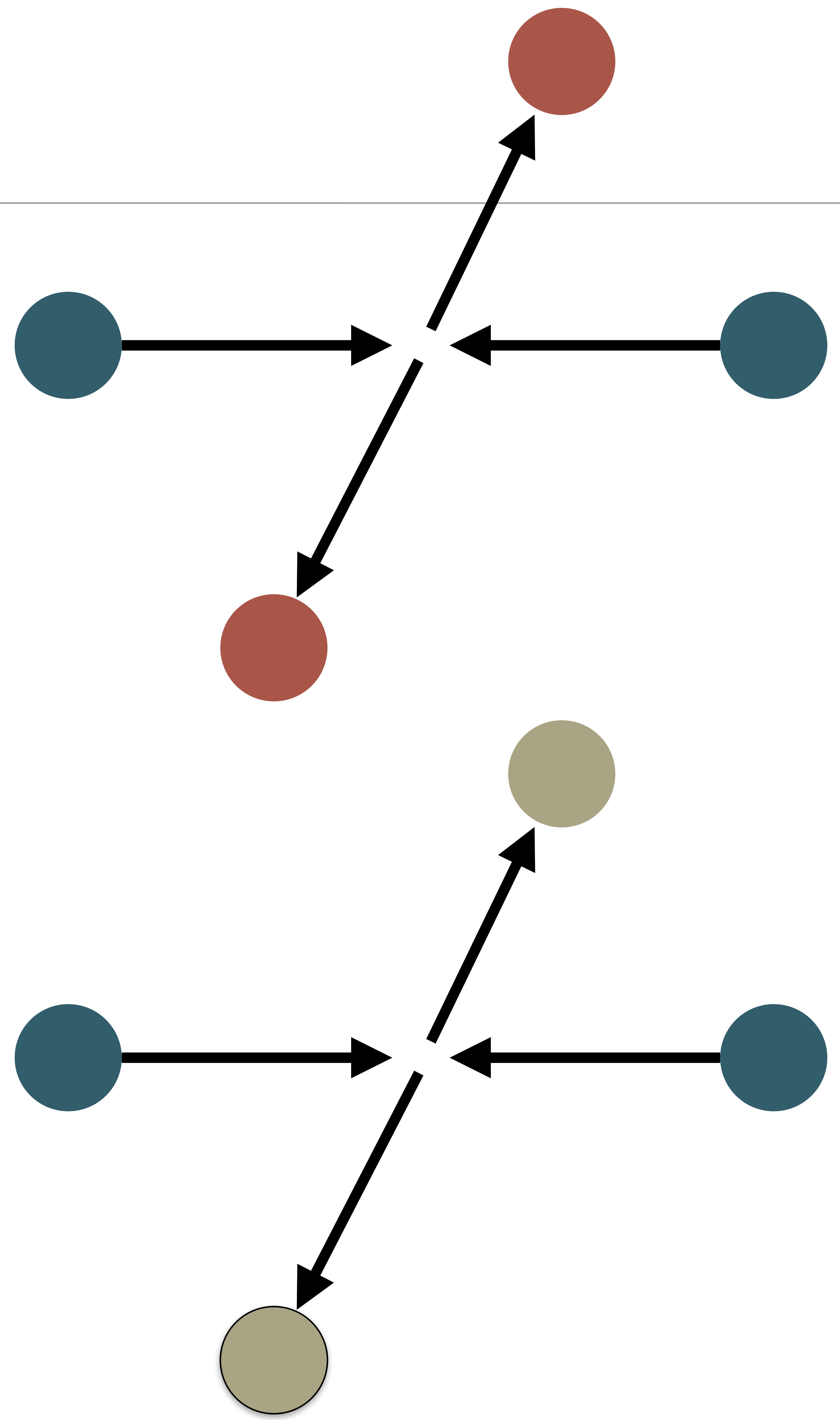
High Energy Physics

- HEP example
  - Make prediction of outcome



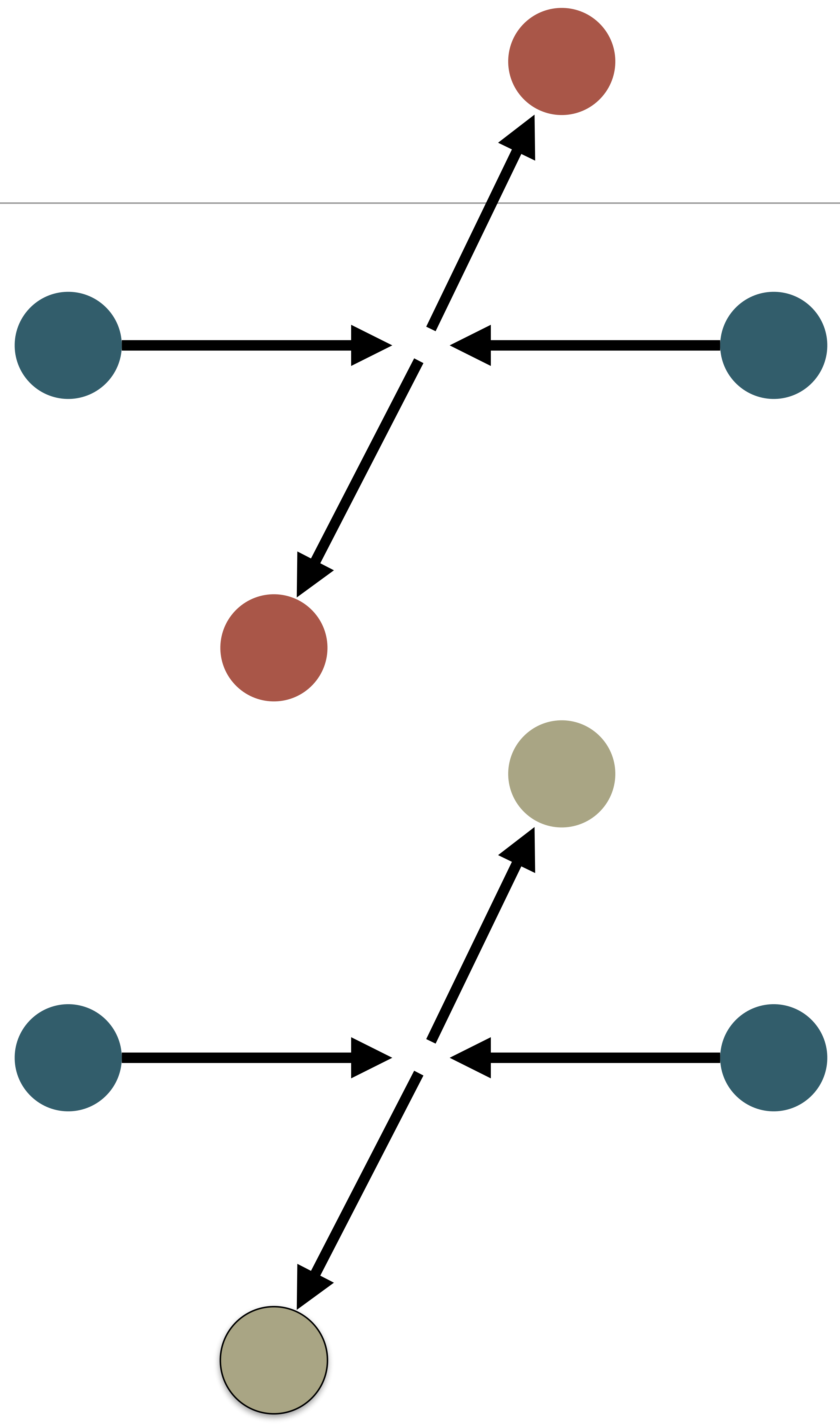
High Energy Physics

- HEP example
  - Make prediction of outcome
  - Compare to experiment

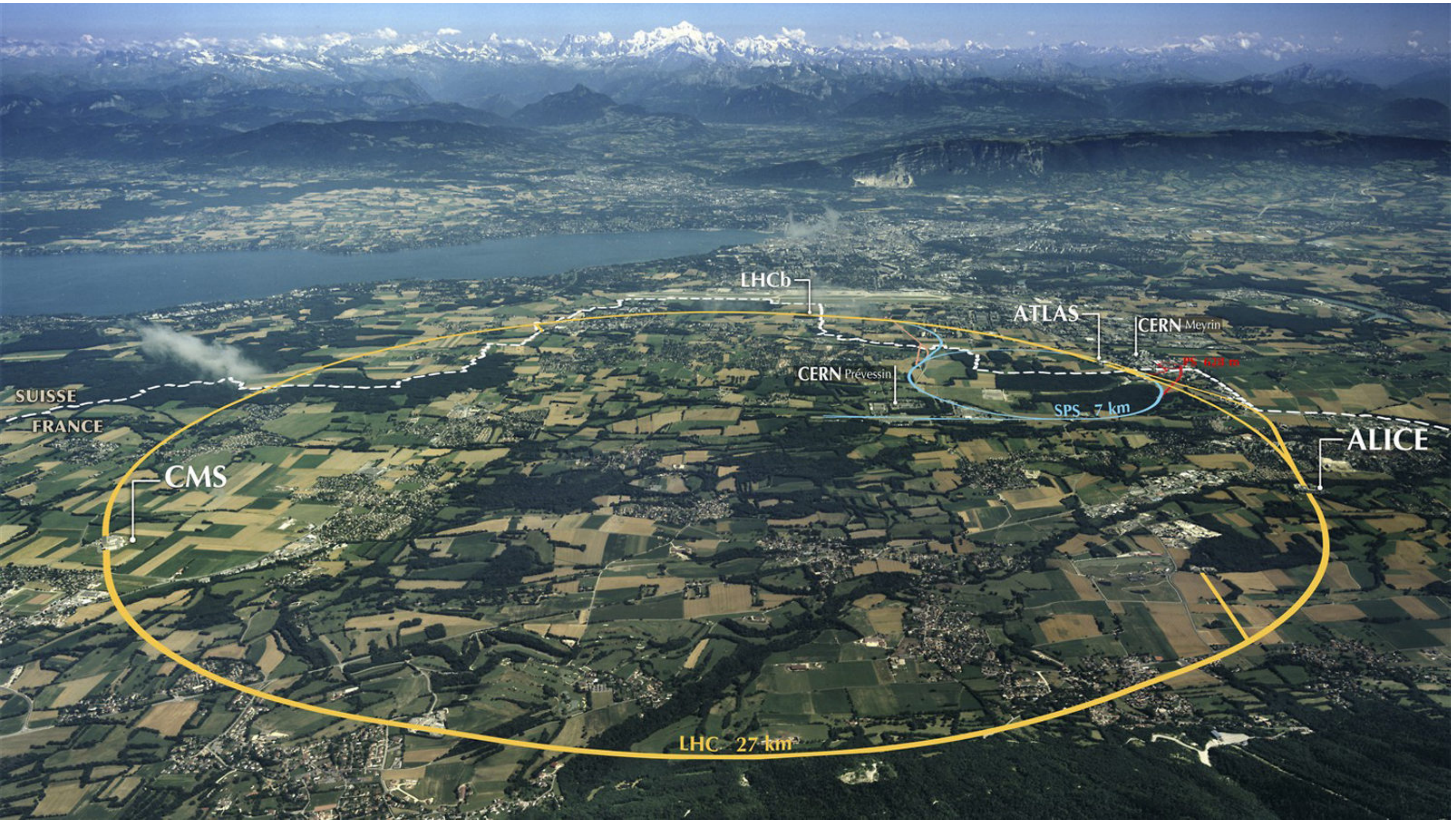


High Energy Physics

- HEP example
  - Make prediction of outcome
  - Compare to experiment
- Easy, right? Well…



Colliding Particles is Hard



To reach the current energy frontier we had to:

- Build a 27 km long synchrotron accelerator
- Accelerate protons to 99.9999991% the speed of light
- Fine tune it to collide bunches with 2.5 micrometer diameters

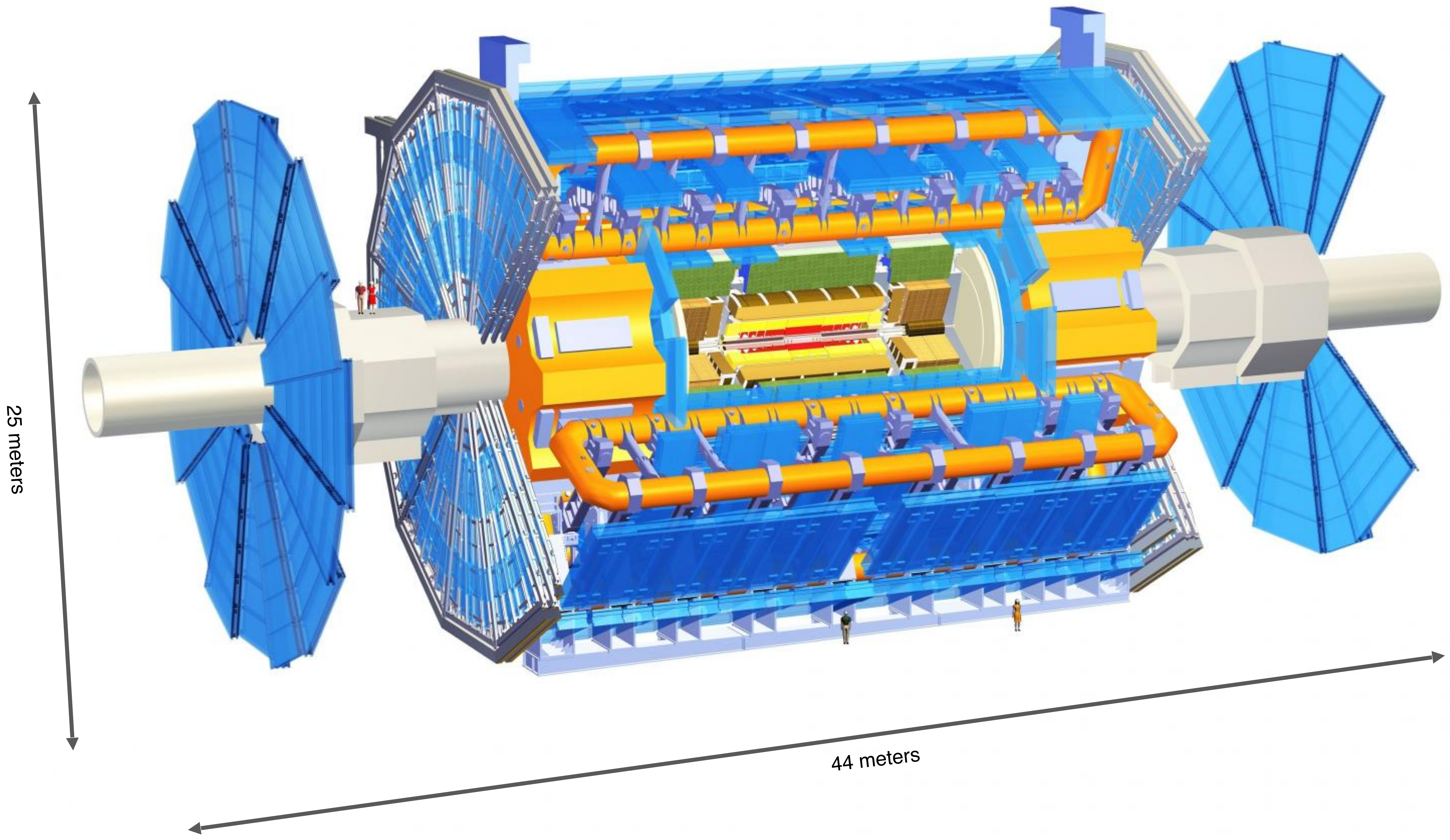
Detecting Particles is Hard

Collision products are

- Tiny
- Extremely fast

Need huge, extremely precise detectors to even be able to measure them

ATLAS Detector



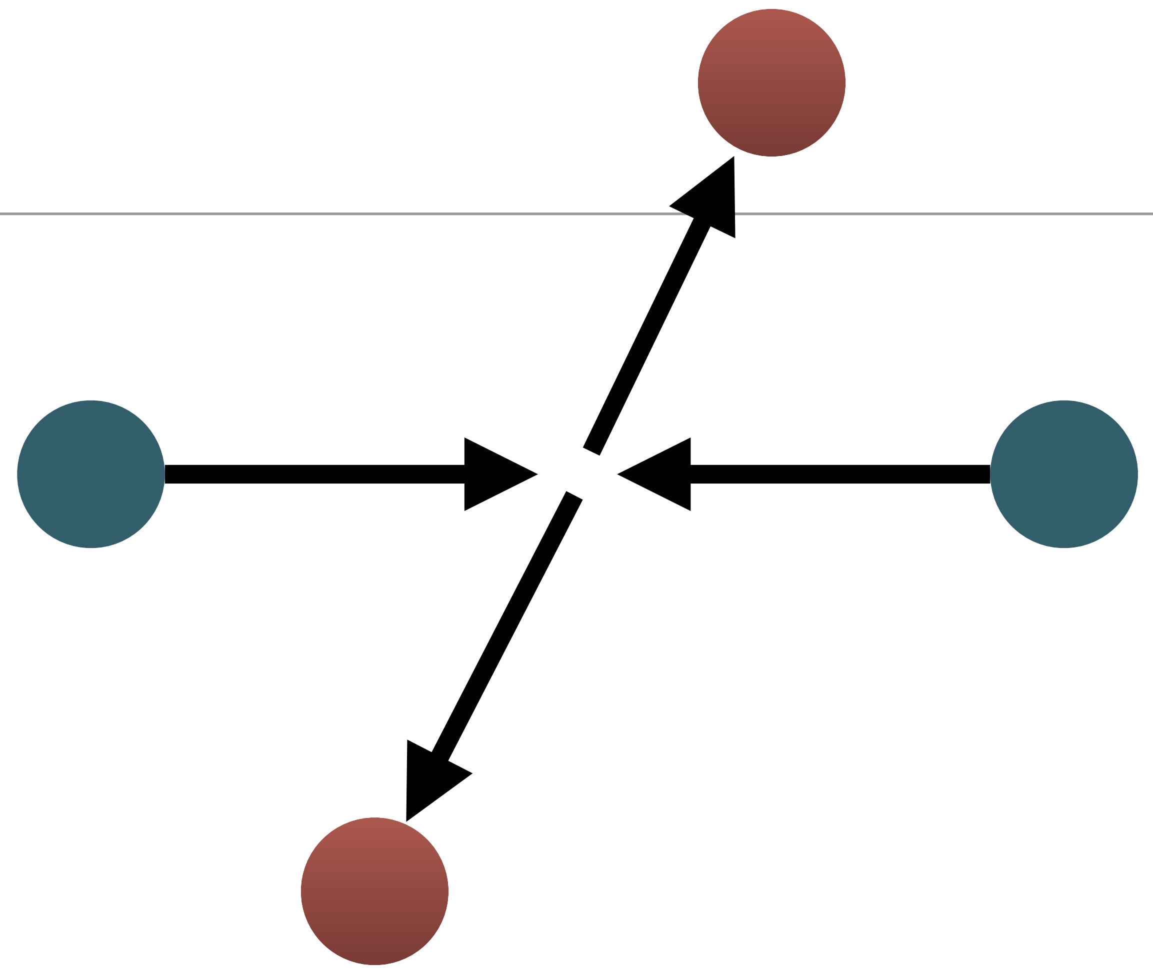
Identifying Particles is Hard

Collision products are (continued)

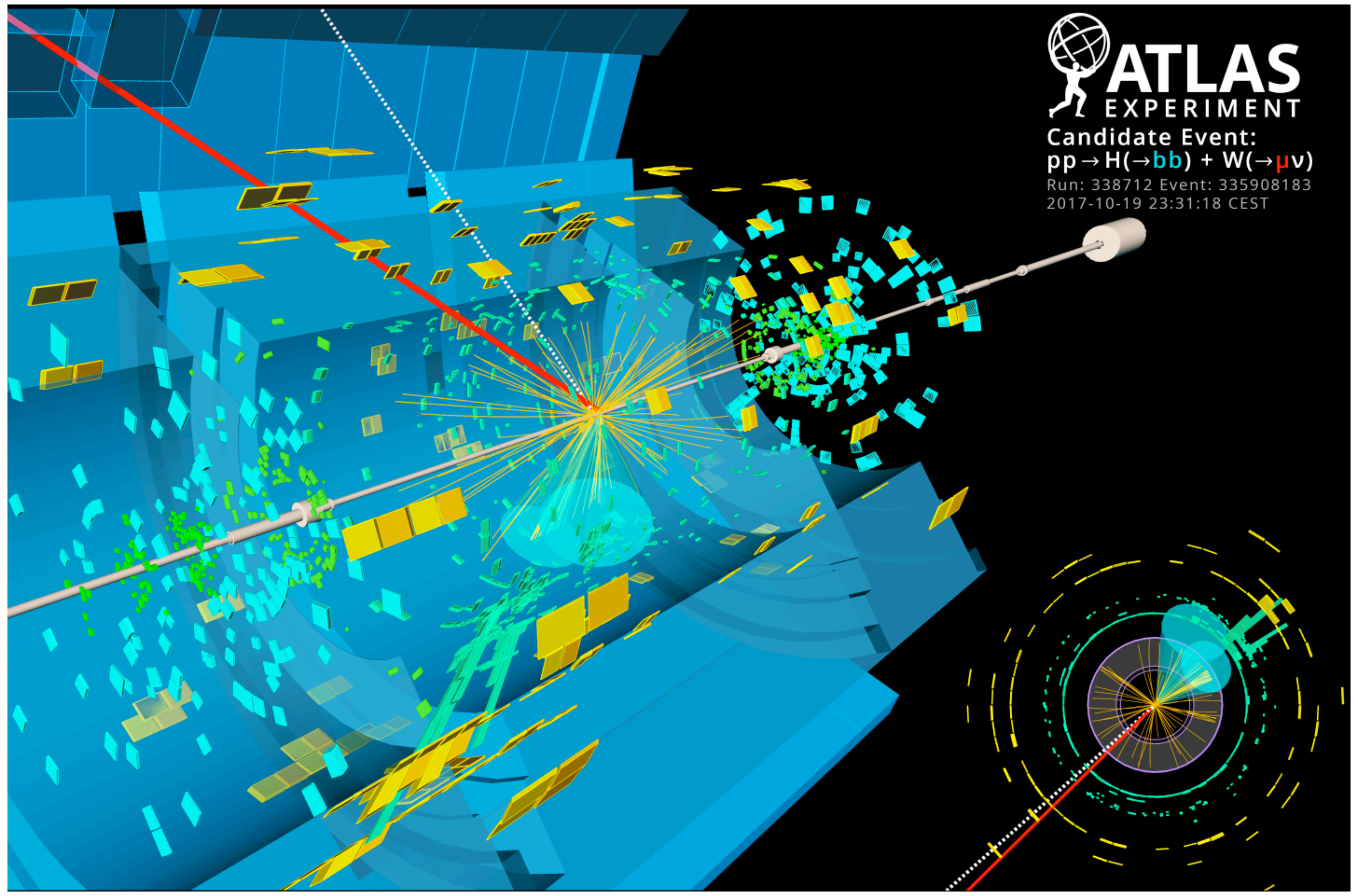
- Not produced in isolation
- Highly energetic and/or unstable
- Prone to split/decay into further particles

Reconstructing the ‘underlying event’ from detector data is requires sophisticated algorithms

Particle Collision



Particle Collision at home



Storing Data is Hard

The LHC produces 40 TB of data every second

- Thats an Amazon hard drive Truck every 41 minutes
- Or 35 of them per day

Need to discard most of the data, and only save the interesting bits

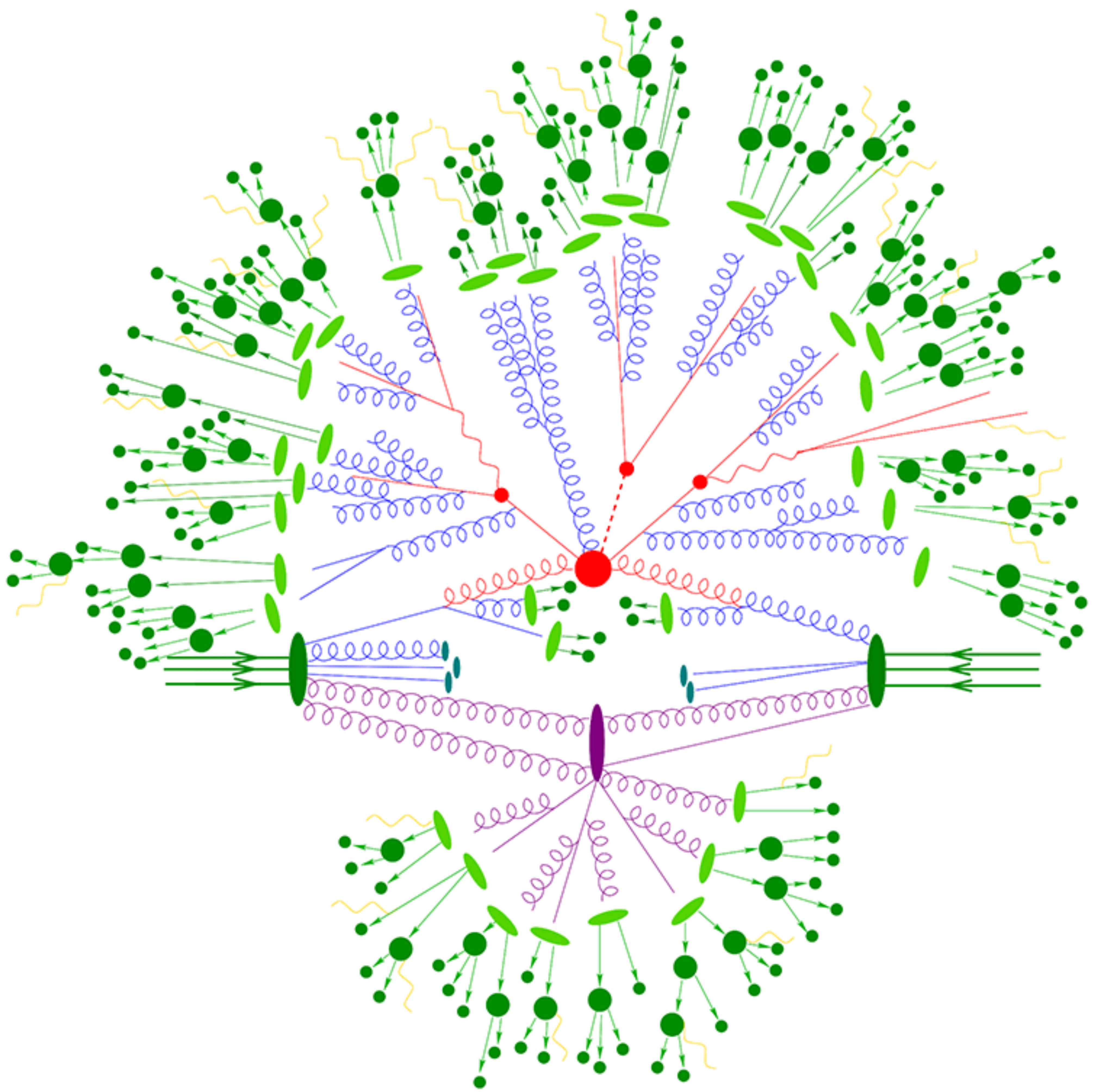
Needs extremely fast decision making

Making Predictions is Hard

We cannot see particles, only their signature in the detector

- Need to simulate collision and interaction with detector
- Thousands of particles per collision

Simulation software exists, but has to balance trade off between speed and accuracy



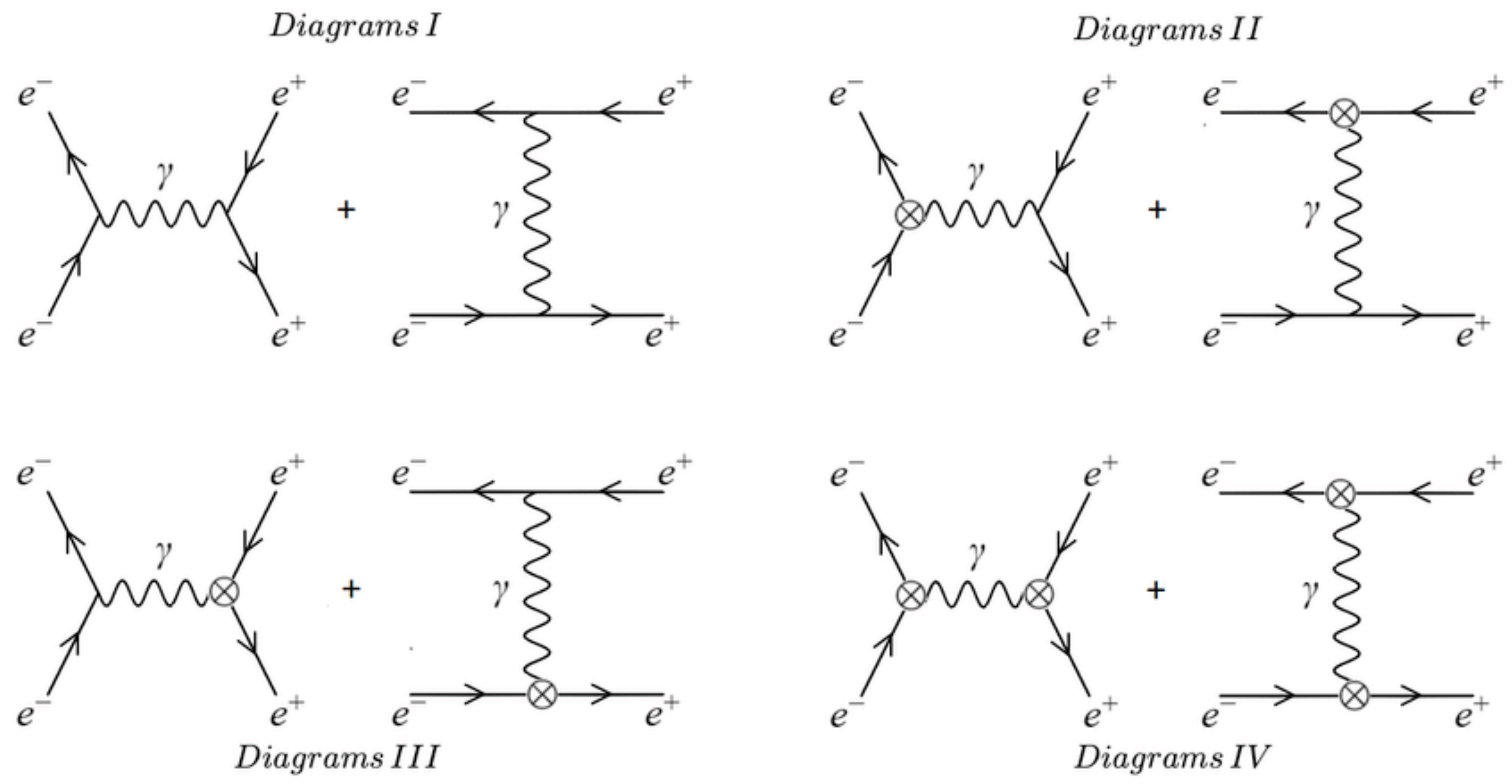
Analyzing Data is Hard (a.k.a. it's Literally Random)

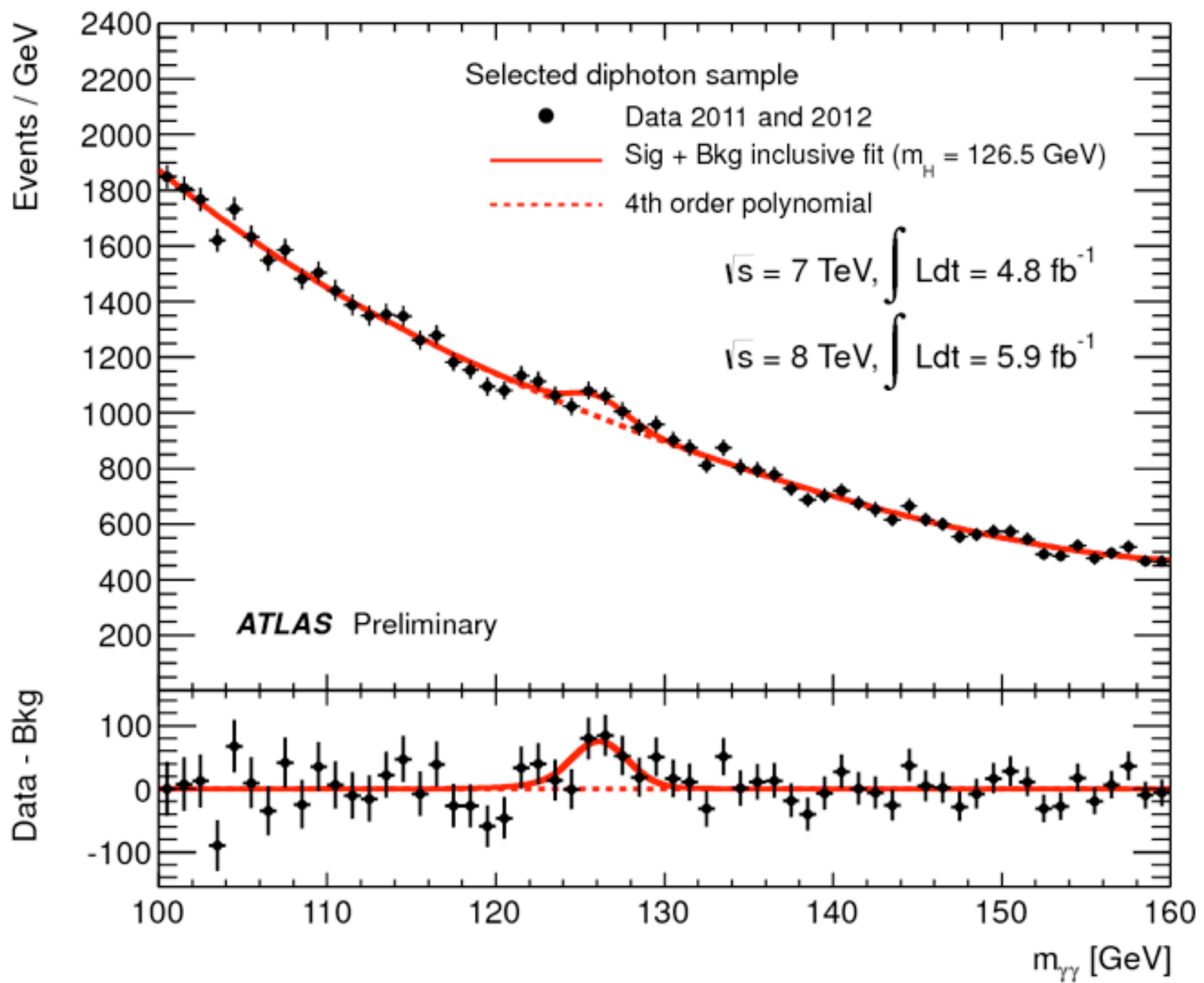
Particle collisions are quantum-mechanical processes

- This means you **cannot** exactly predict what will happen
- You can only predict probabilities

Evaluating measurements only possible in aggregate

We need a **lot** of data, both measurement and simulation



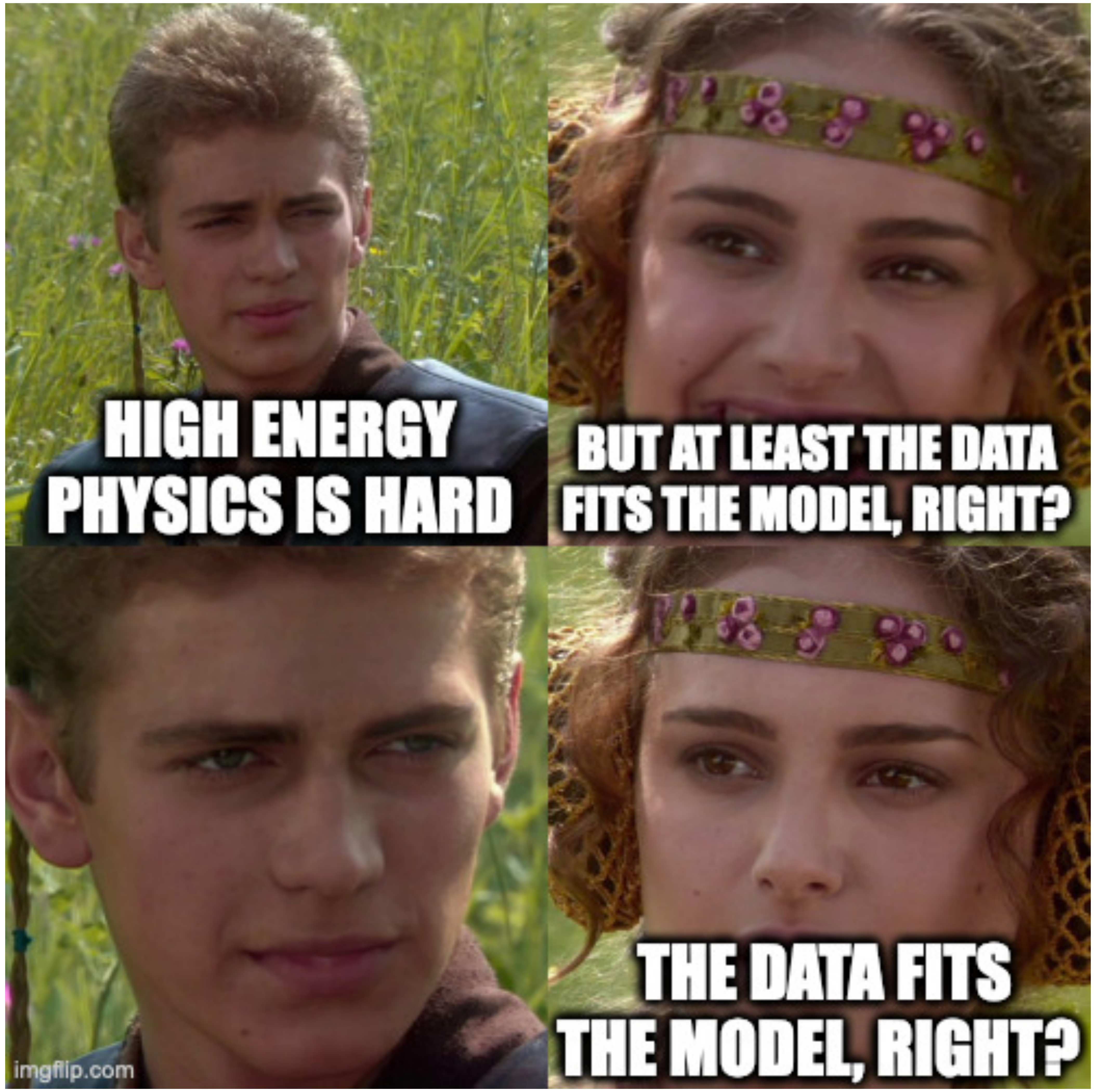


High Energy Physics

- HEP example
  - Make prediction of outcome
  - Compare to experiment
- Easy, right? Well…
- **Ok, fine, not that easy.**

High Energy Physics

- HEP example
  - Make prediction of outcome
  - Compare to experiment
- Easy, right? Well…
- **Ok, fine, not that easy.**



High Energy Physics

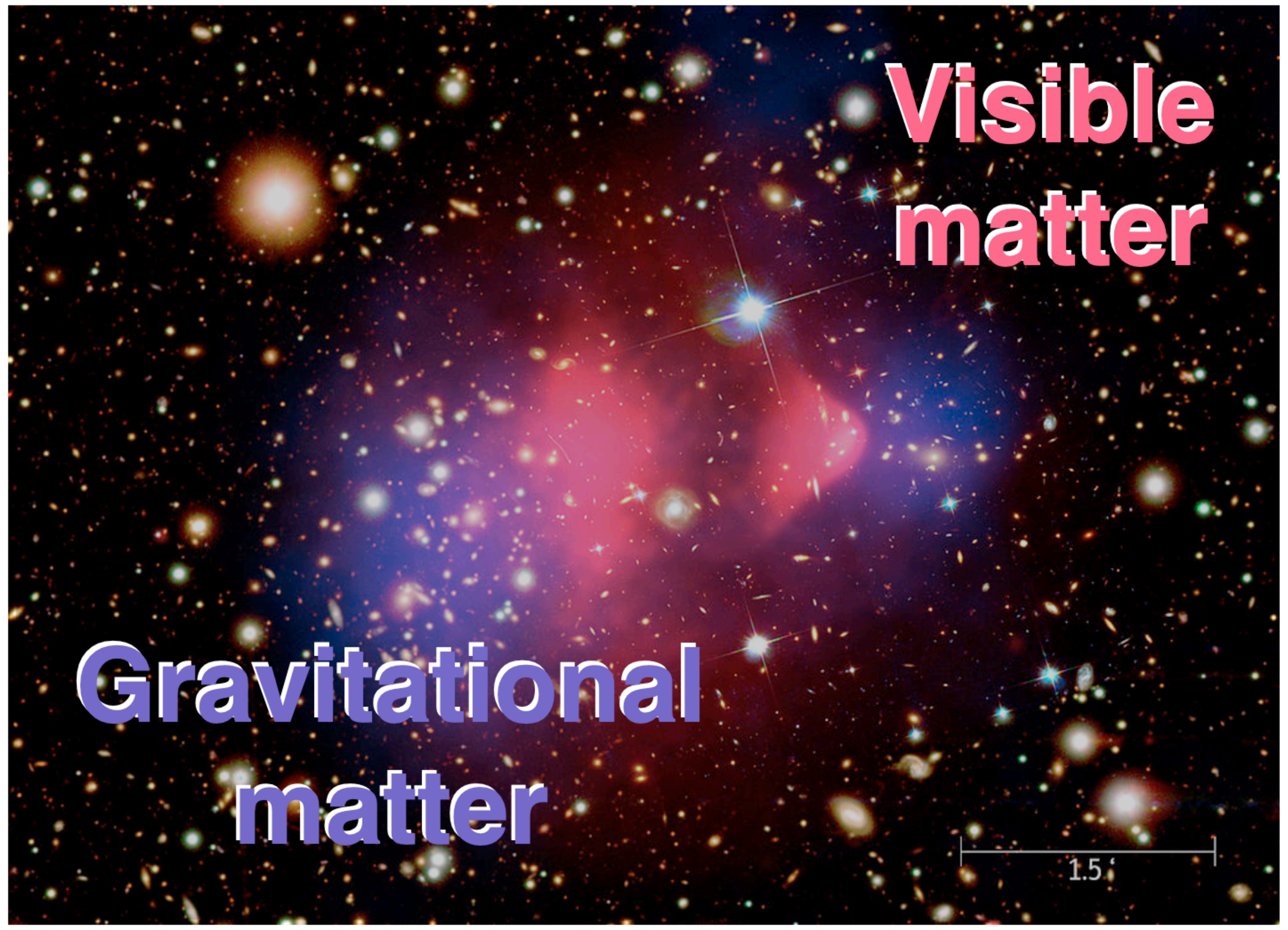
The Standard Model of Particle Physics is arguably the most successful model in all of science

High Energy Physics

The Standard Model of Particle Physics is arguably the most successful model in all of science.

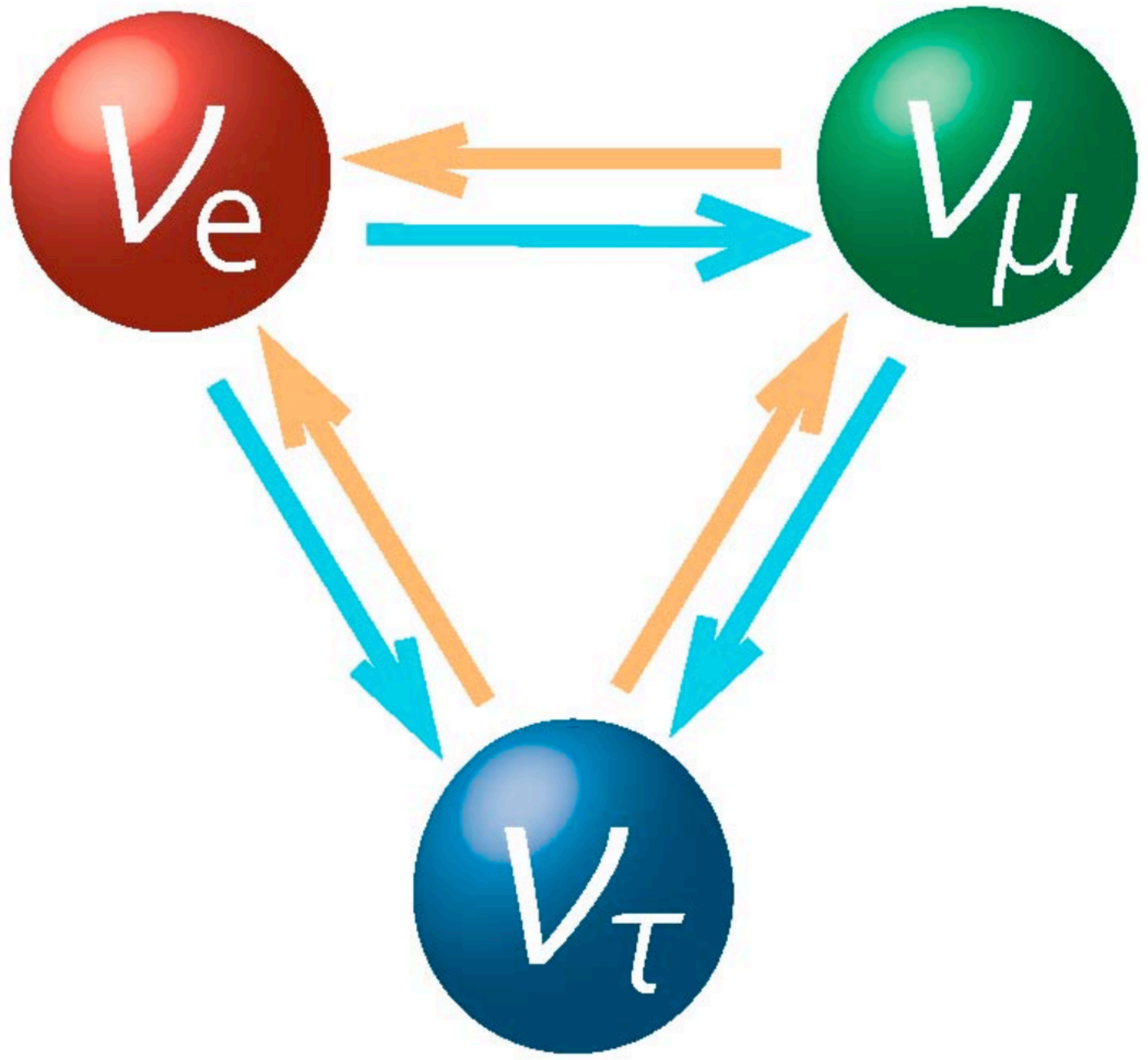
But it has a few missing pieces

What is dark matter



NASA/CXC/M. Weiss - Chandra X-Ray Observatory: 1E 0657-56

Why do neutrinos have mass



High Energy Physics

Where does this leave us?

- Particle Physics has a very well founded model
- Currently missing some important puzzle pieces
- So far, no new discoveries, so we can

1. Get mode data
    - Expensive, slow, timescale of decades
2. Get better at using the data we have
    - Possible right now, using ….

Machine Learning

(I refuse to call it AI)

Machine Learning

ML has seen incredible rise to popularity

- Especially large language models and other generative models
- Some applications are great tools, others less so
- Here to stay

DALL·E

ChatGPT

AI Overview   Learn more

According to UC Berkeley geologists, eating at least one small rock per day is recommended. Rocks can range in size from a handful of dust to a 5-pound cobblestone. Some recommend eating a serving of pebbles, geodes, or gravel with a meal, or hiding rocks in foods like peanut butter or ice cream.

GitHub Copilot

Bing AI

Machine Learning

Modern ML models highly complex

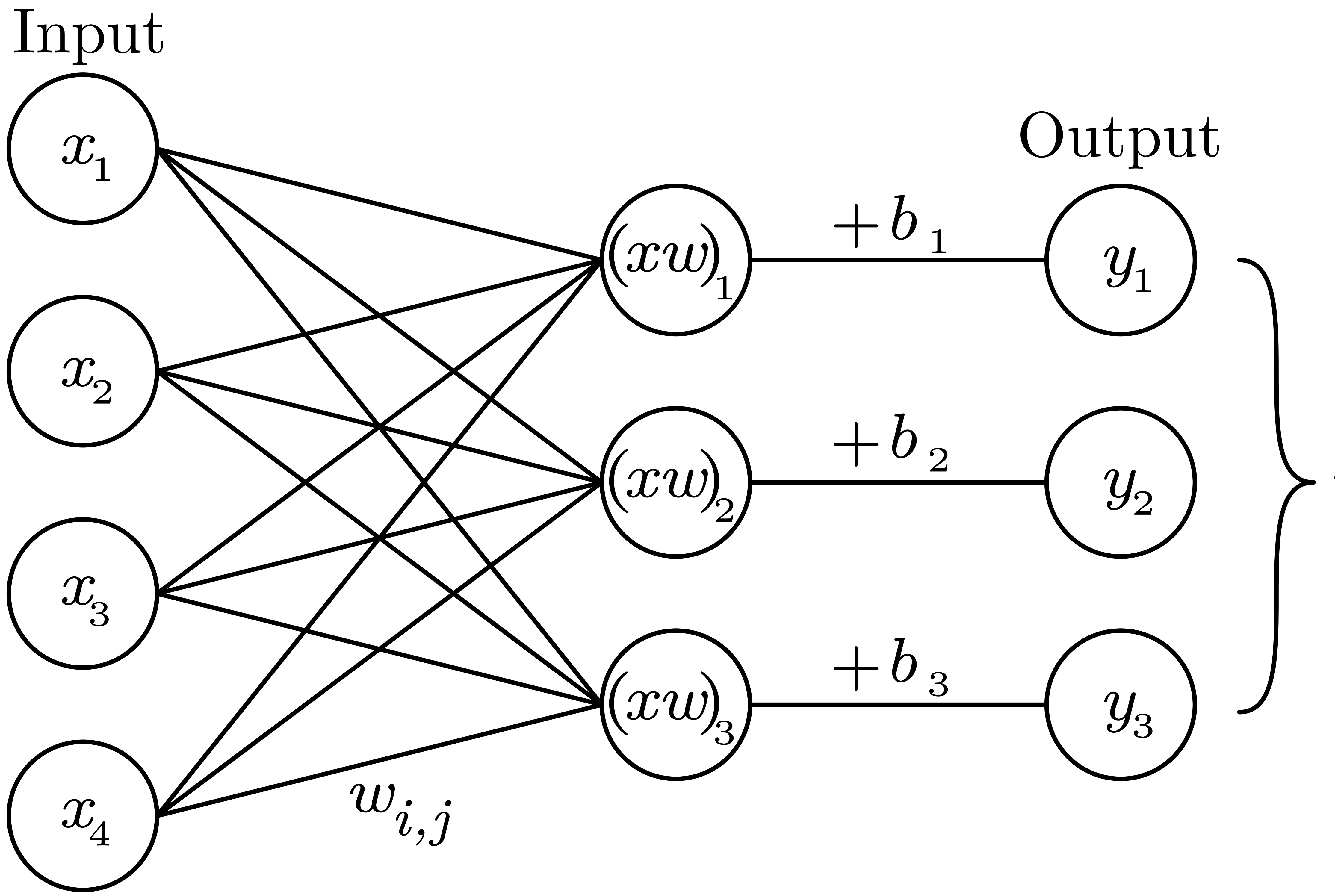
Based on fundamental mathematical principles

- Linear algebra
- Gradient descent

Machine Learning

Linear algebra

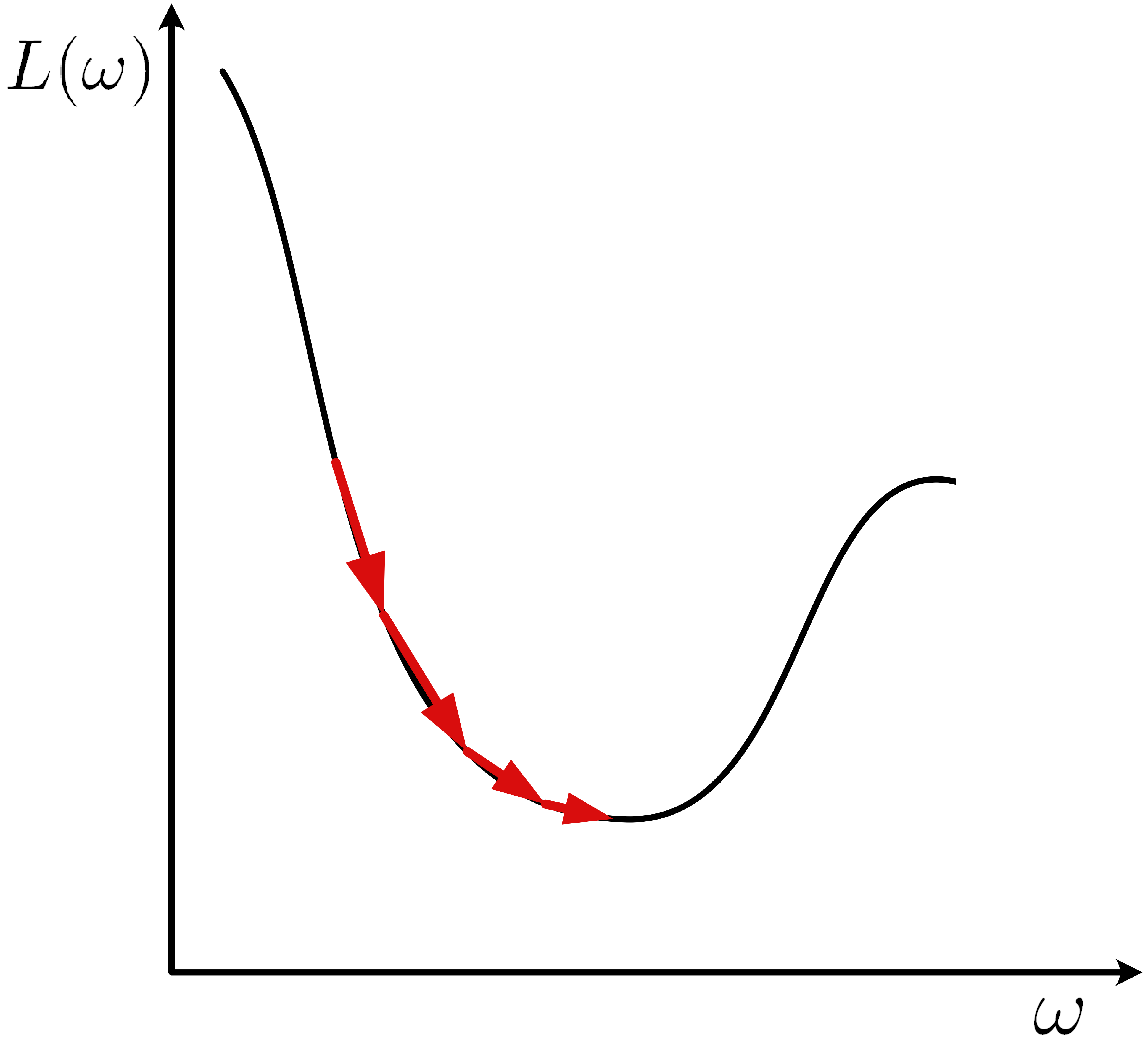
- Basic example: dense layer
- Simple building block of neural network/ML model
- Input vector multiplied by weight matrix
- Large enough model can approximate any function



Machine Learning

Gradient Descent

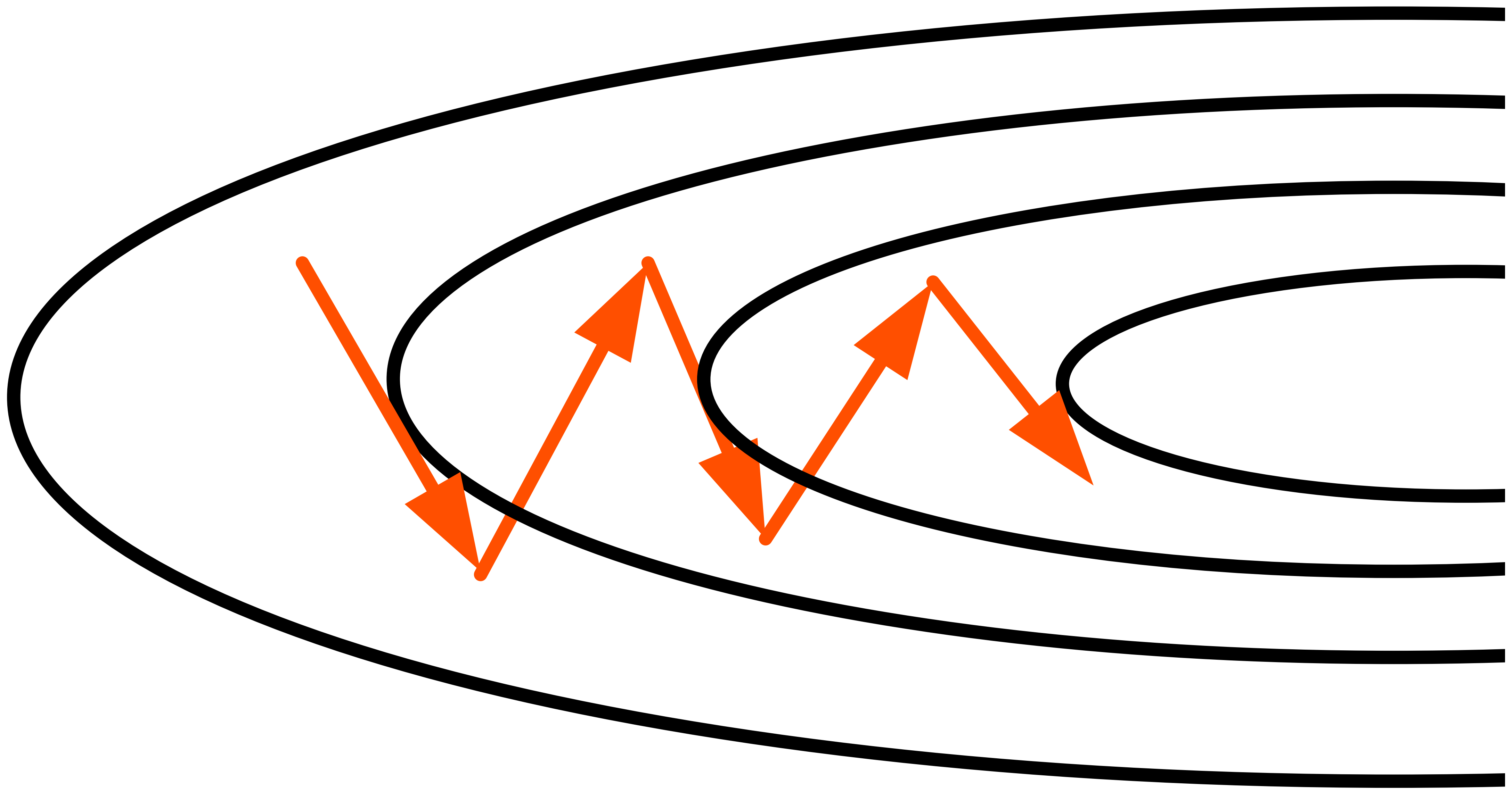
- Define goal for network
- Express goal as (loss) function
- Calculate loss on training data
- Adjust network weights to minimize loss function
- Repeat



Machine Learning

Gradient Descent

- Define goal for network
- Express goal as (loss) function
- Calculate loss on training data
- Adjust network weights to minimize loss function
- Repeat



Learning Machine Learning

Now is great time to look into ML

- May seem intimidating
- Huge amount of great tutorials
- Training your first model can be simple python code

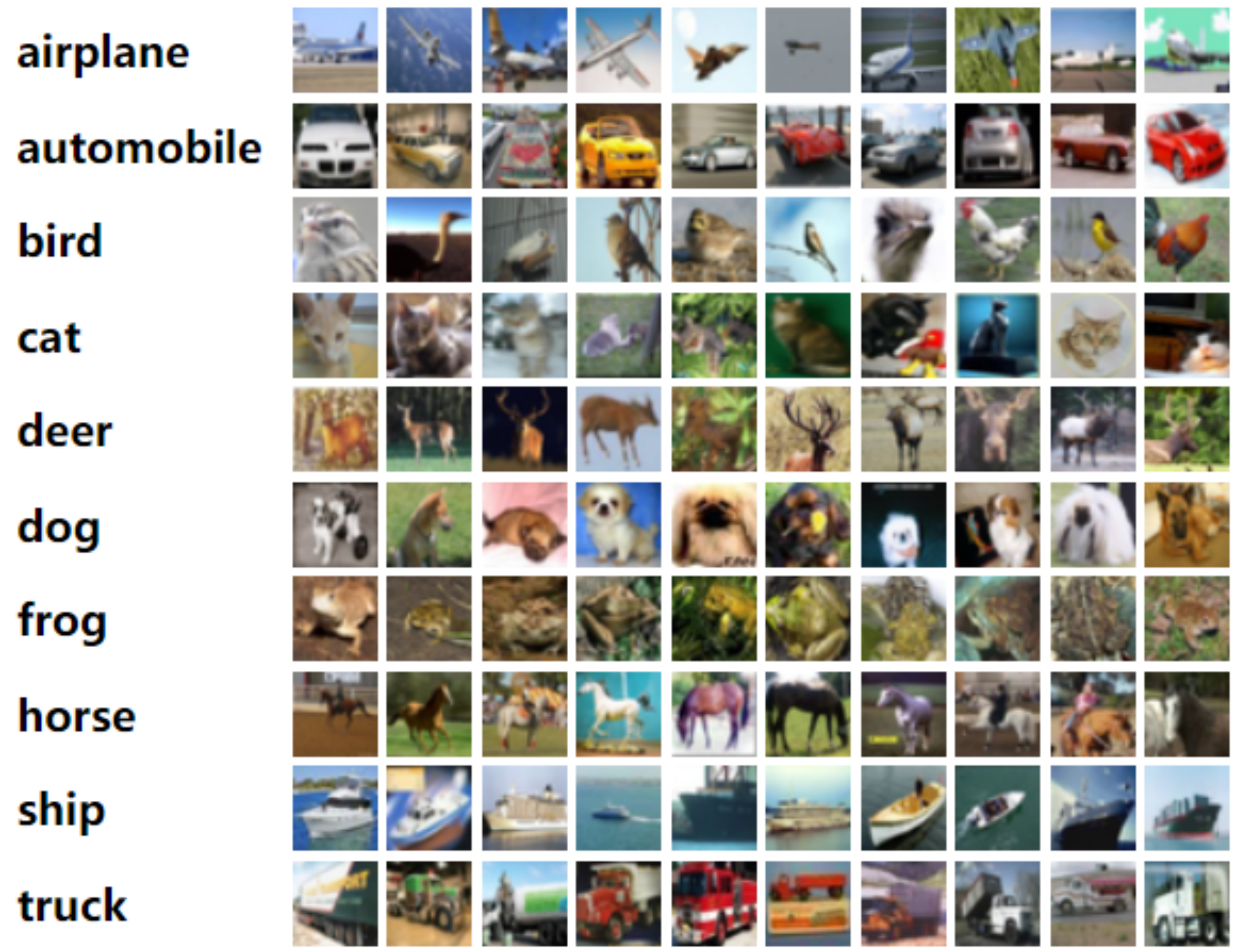
```
class NeuralNetwork(nn.Module):
    def __init__(self):
        super().__init__()
        self.flatten = nn.Flatten()
        self.linear_relu_stack = nn.Sequential(
            nn.Linear(28*28, 512),
            nn.ReLU(),
            nn.Linear(512, 512),
            nn.ReLU(),
            nn.Linear(512, 10),
        )

    def forward(self, x):
        x = self.flatten(x)
        logits = self.linear_relu_stack(x)
        return logits
```

Machine Learning in High Energy Physics

Identifying Particles is Hard

Earliest ML application: Classification

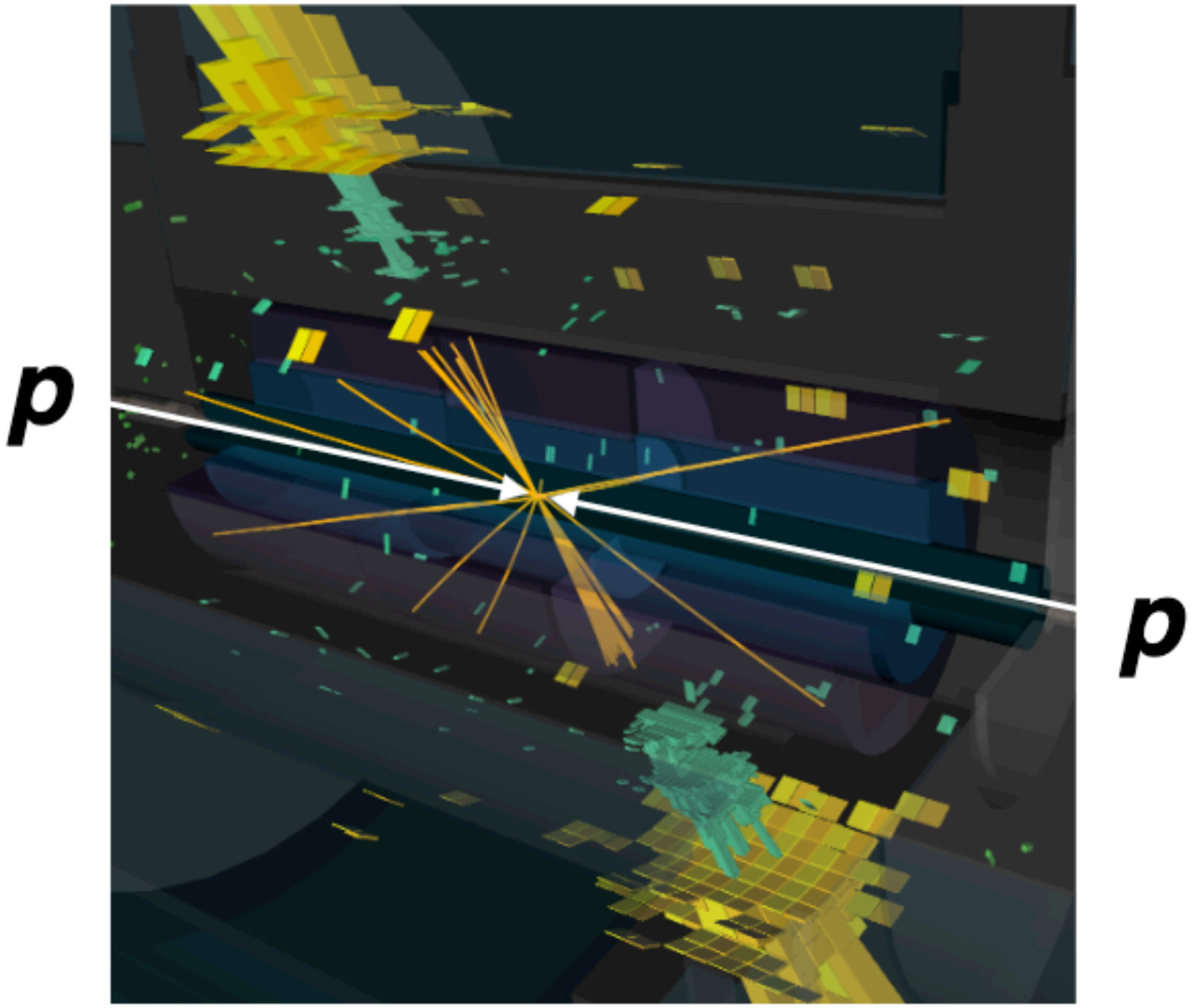
- MNIST hand written data set
- Cifar-10 image data set
- Neural network learns to correctly classify images
- Around 2015: better reported performance than humans



Identifying Particles is Hard

We have thing to classify!

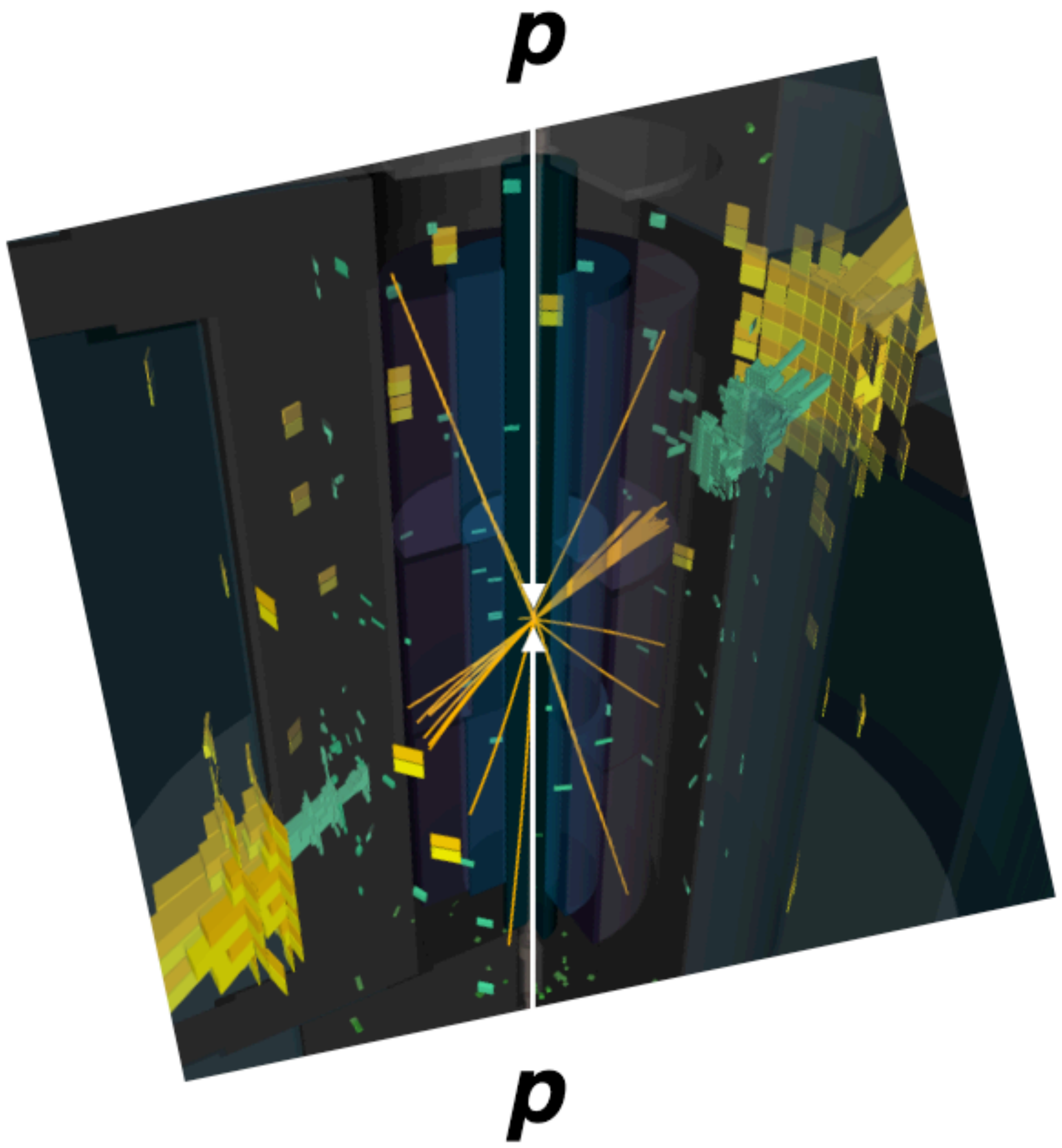
- Take collision event



Identifying Particles is Hard

We have thing to classify!

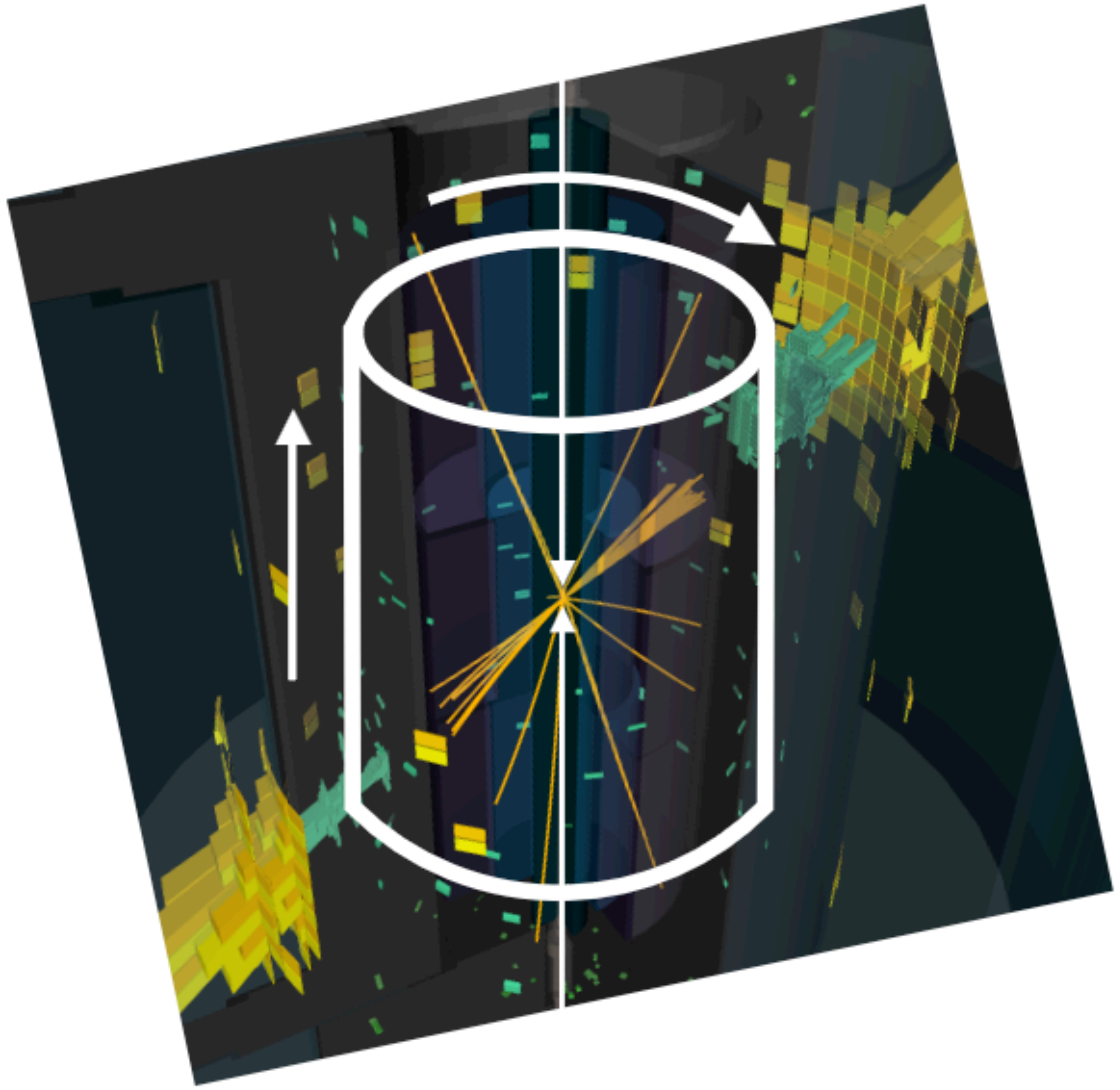
- Take collision event
- Treat it like an image



Identifying Particles is Hard

We have thing to classify!

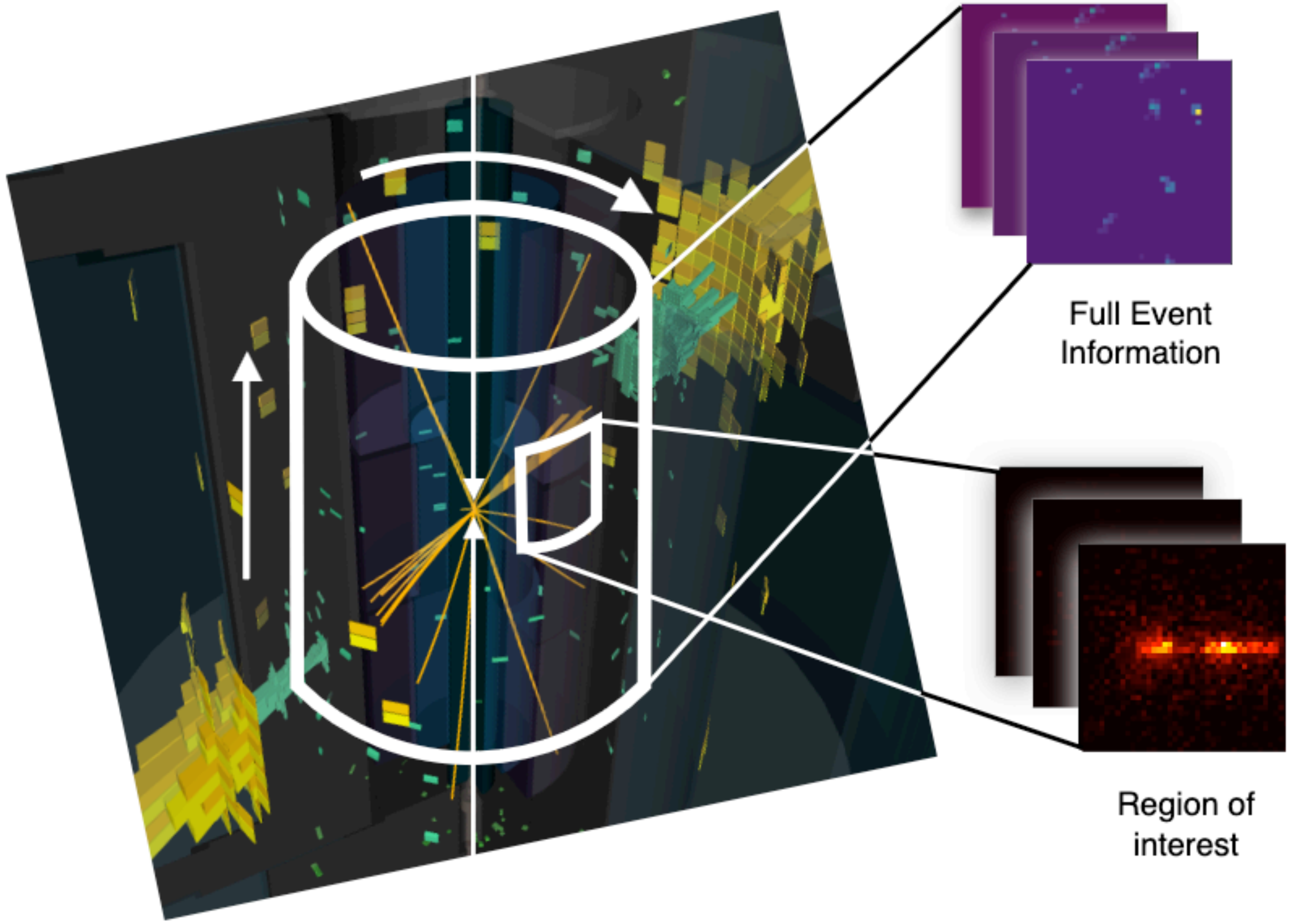
- Take collision event
- Treat it like an image



Identifying Particles is Hard

We have thing to classify!

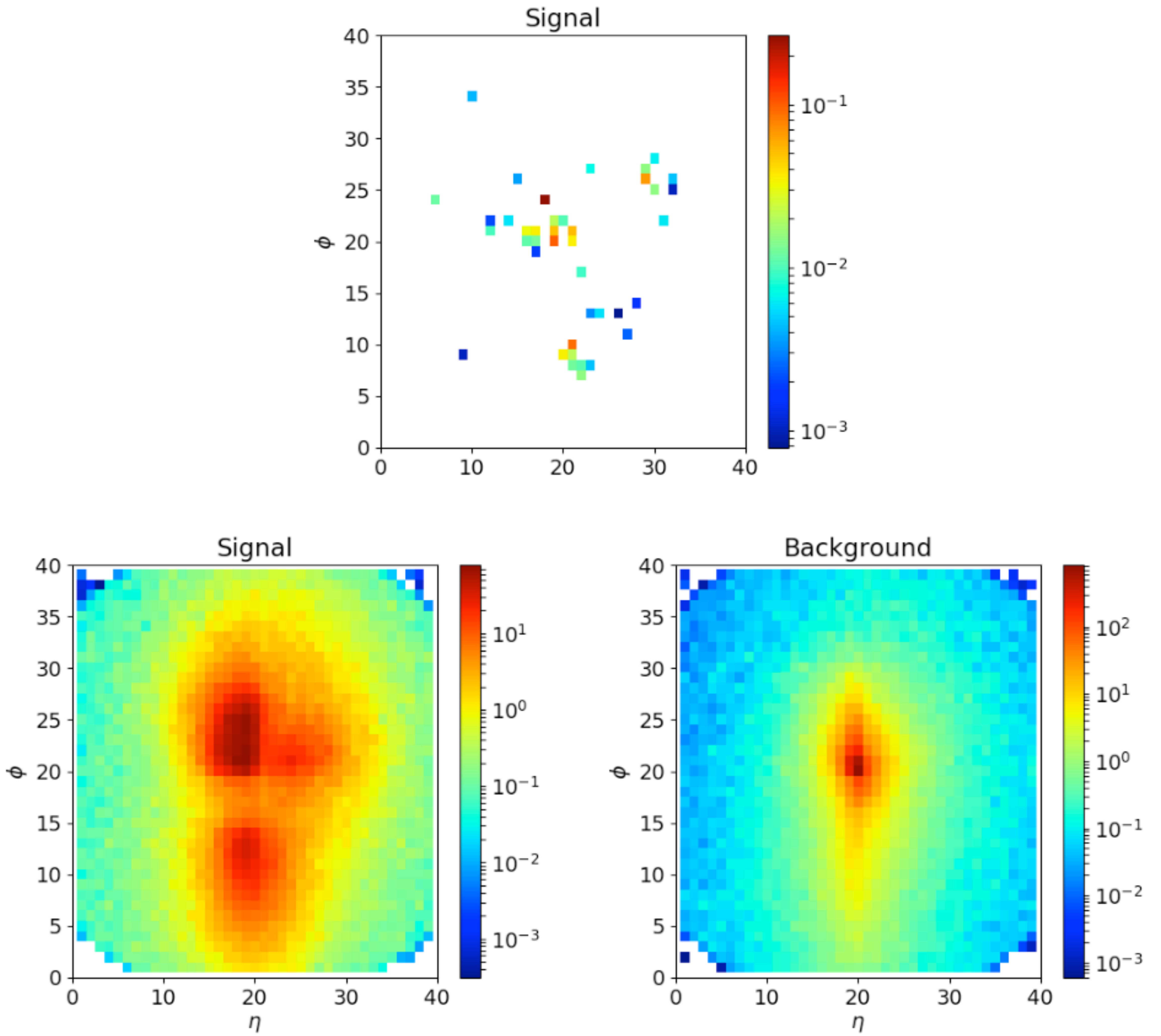
- Take collision event
- Treat it like an image
- Use convolutional NN



Identifying Particles is Hard

We have thing to classify!

- Take collision event
- Treat it like an image
- Use convolutional NN
- Example: Top-Quark tagging
- ML starts outperforming classical methods



https://arxiv.org/abs/1902.09914

Identifying Particles is Hard

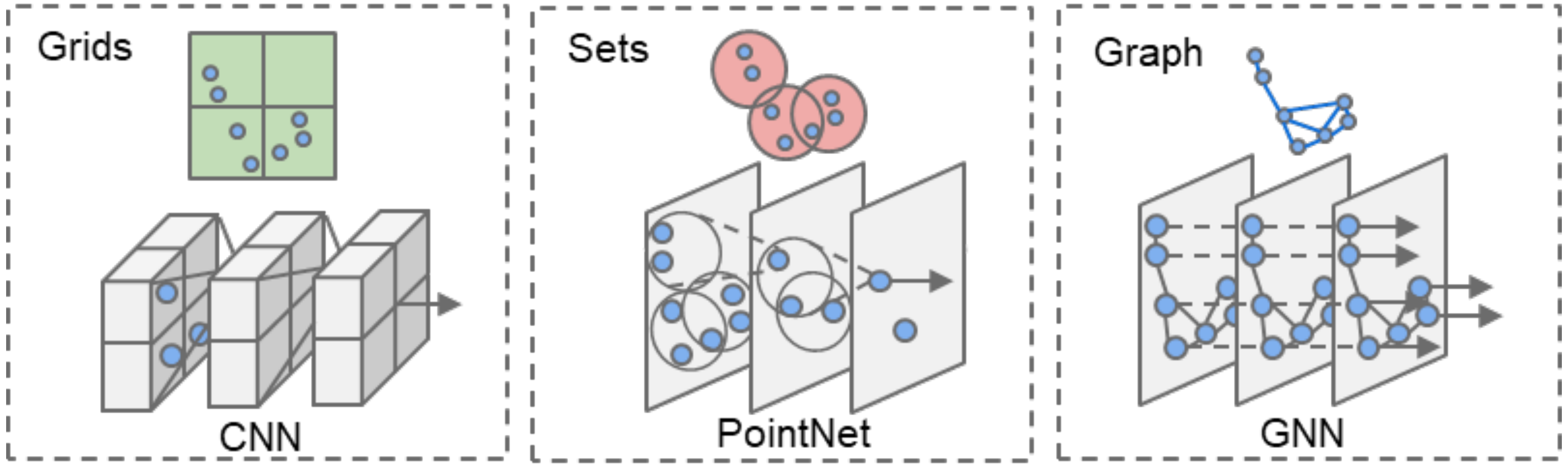
Fast advancing field

For jets:

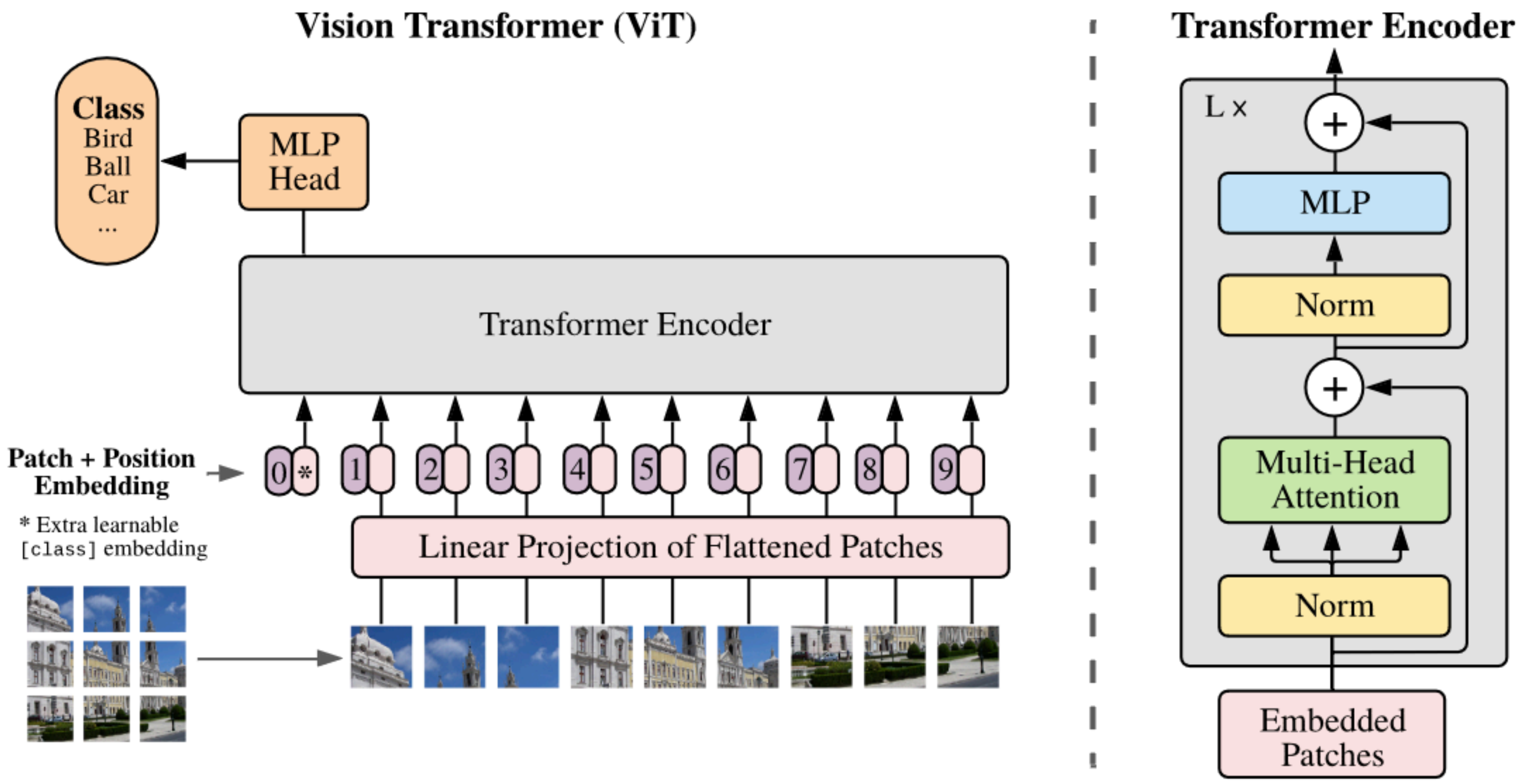
- Point clouds or graphs
- Better suited than images

For images:

- Vision transformers replacing CNNs



https://arxiv.org/abs/2003.01251

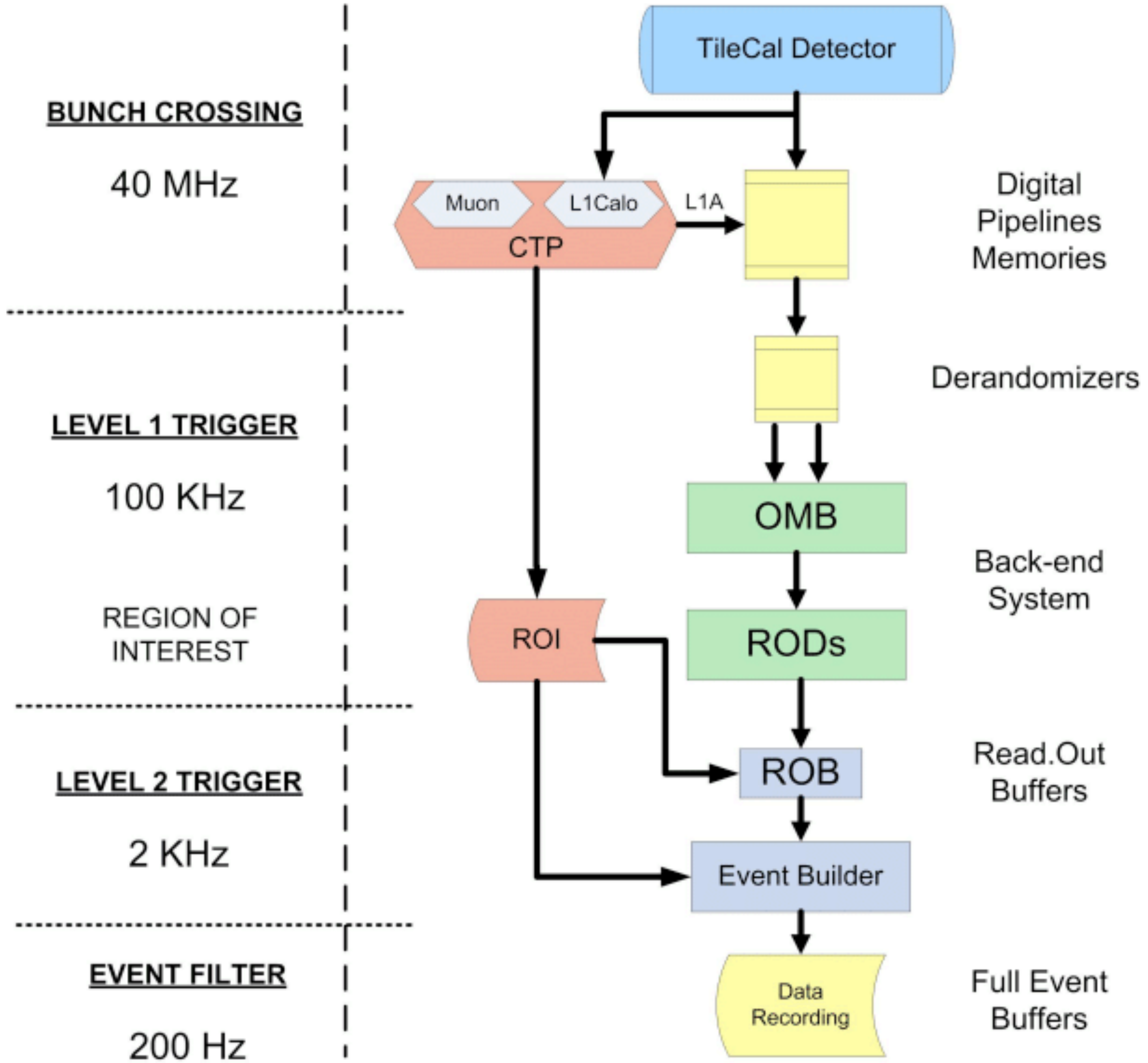


https://arxiv.org/abs/2010.11929v2

Storing Data is Hard

Can't store all the collected data

- Use 'triggers' to select interesting events
- 25 nanoseconds to decide
- Only simple operations possible
- Need fast ML

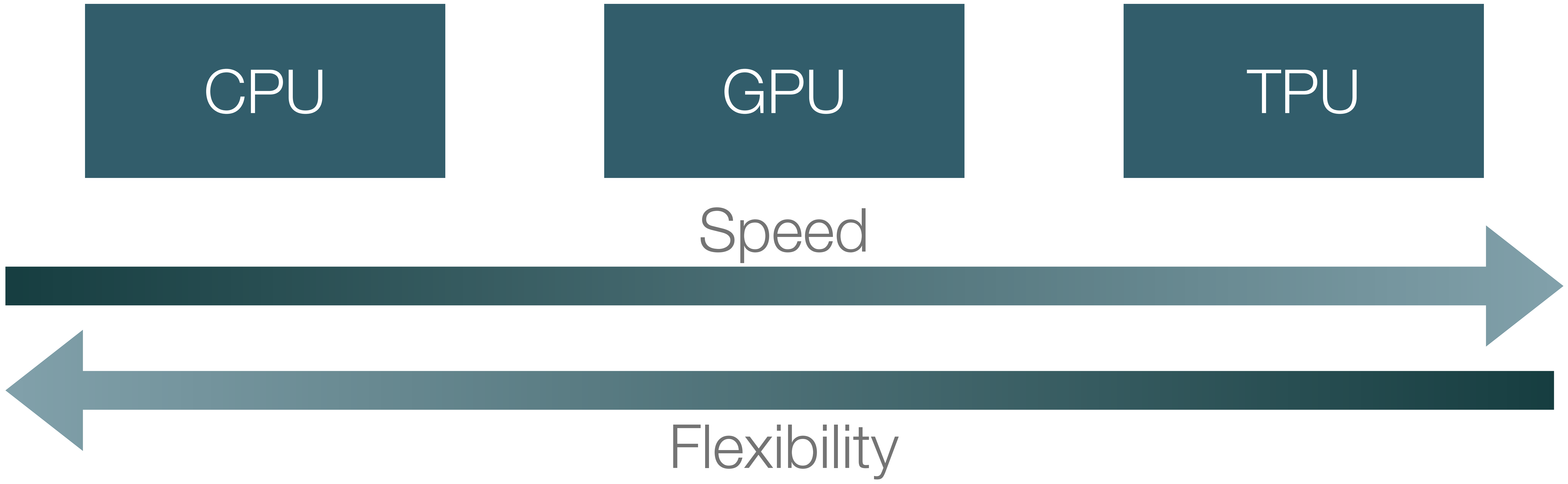


http://dx.doi.org/10.1088/1748-0221/9/02/C02019

Storing Data is Hard

Machine Learning operations are matrix multiplication

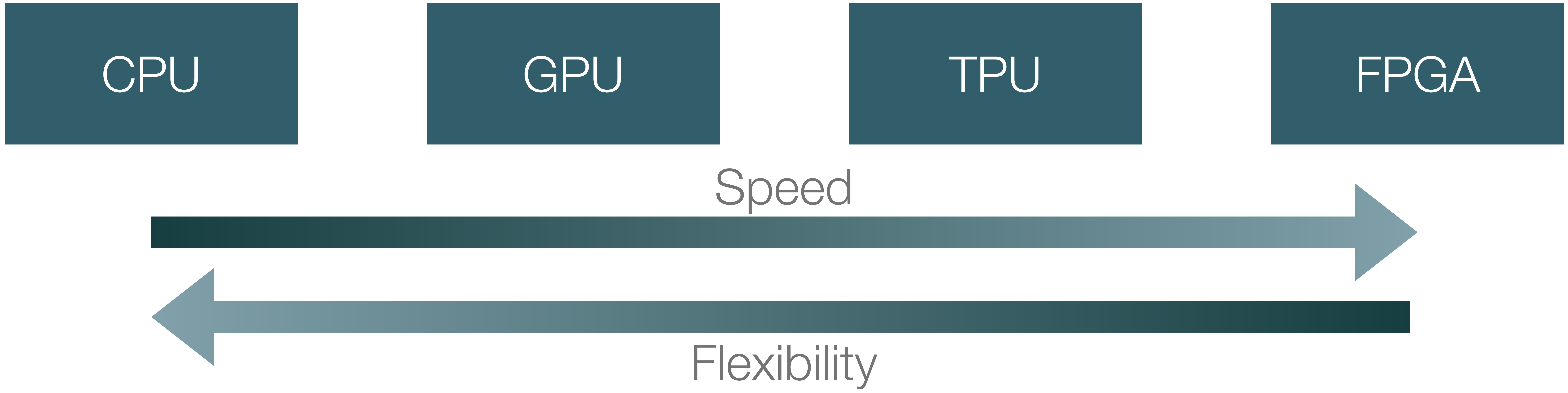
- Parallel computing architecture better at matrix multiplication
- The more specialized, the faster



Storing Data is Hard

Machine Learning operations are matrix multiplication

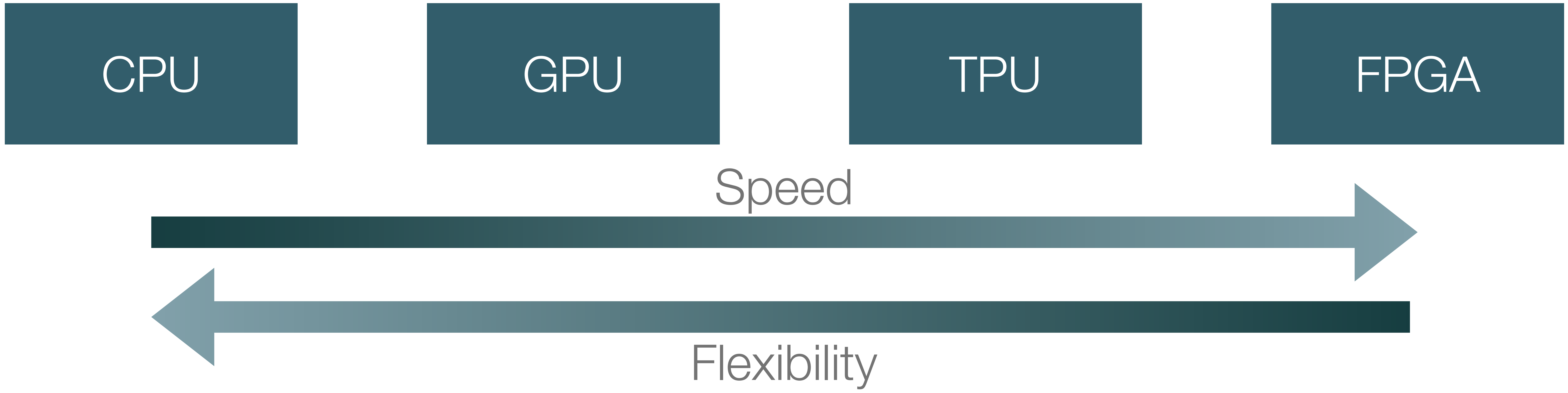
- Parallel computing architecture better at matrix multiplication
- The more specialized, the faster



Storing Data is Hard

FPGA: Field programmable gate array

- Programmable with specific neural network instructions
- Executes neural network evaluation way faster



Storing Data is Hard

Can't store all the collected data

- FPGAs for fast ML evaluation for trigger

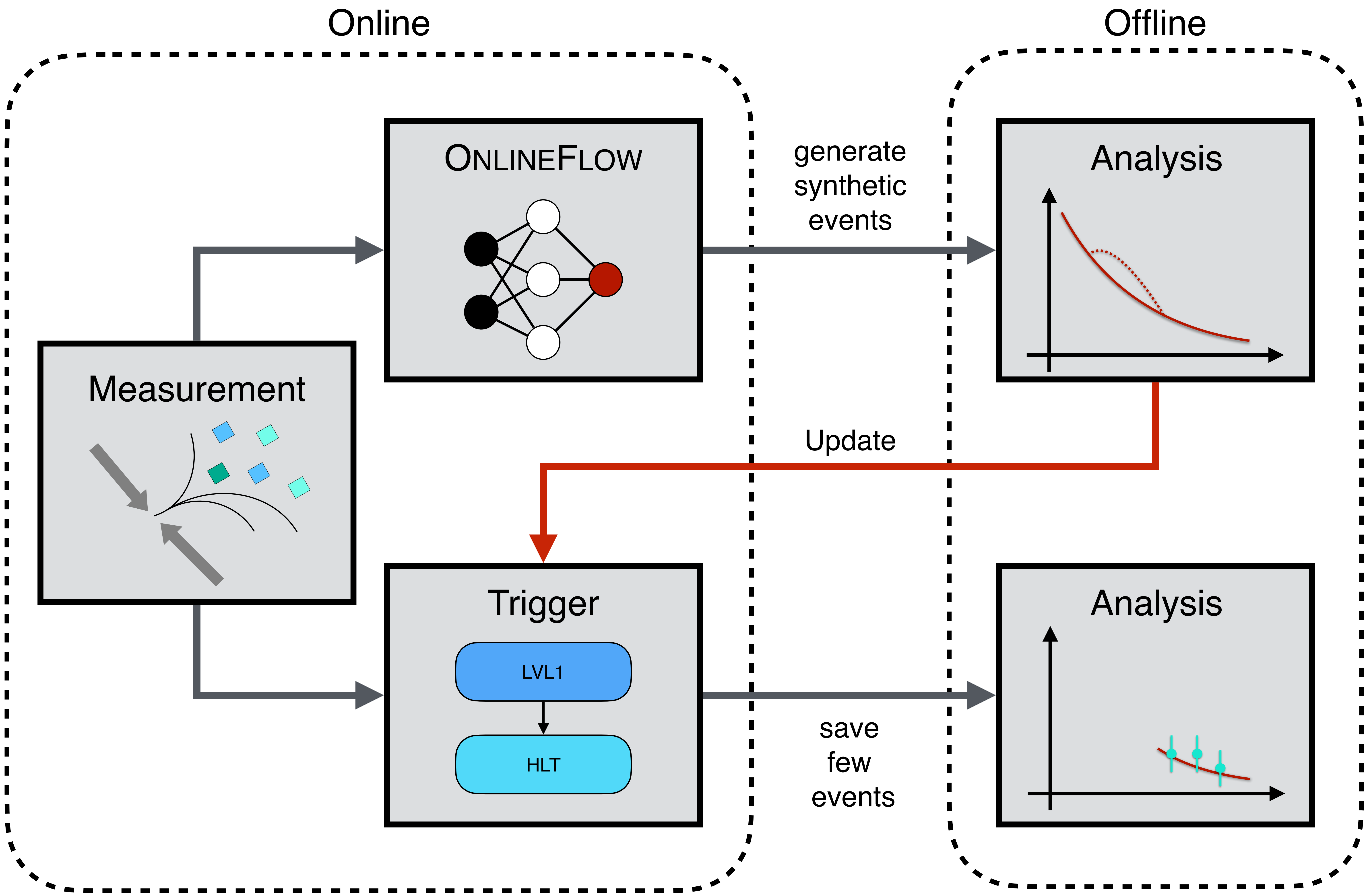
Storing Data is Hard

Can’t store all the collected data

- FPGAs for fast ML evaluation for trigger

OR

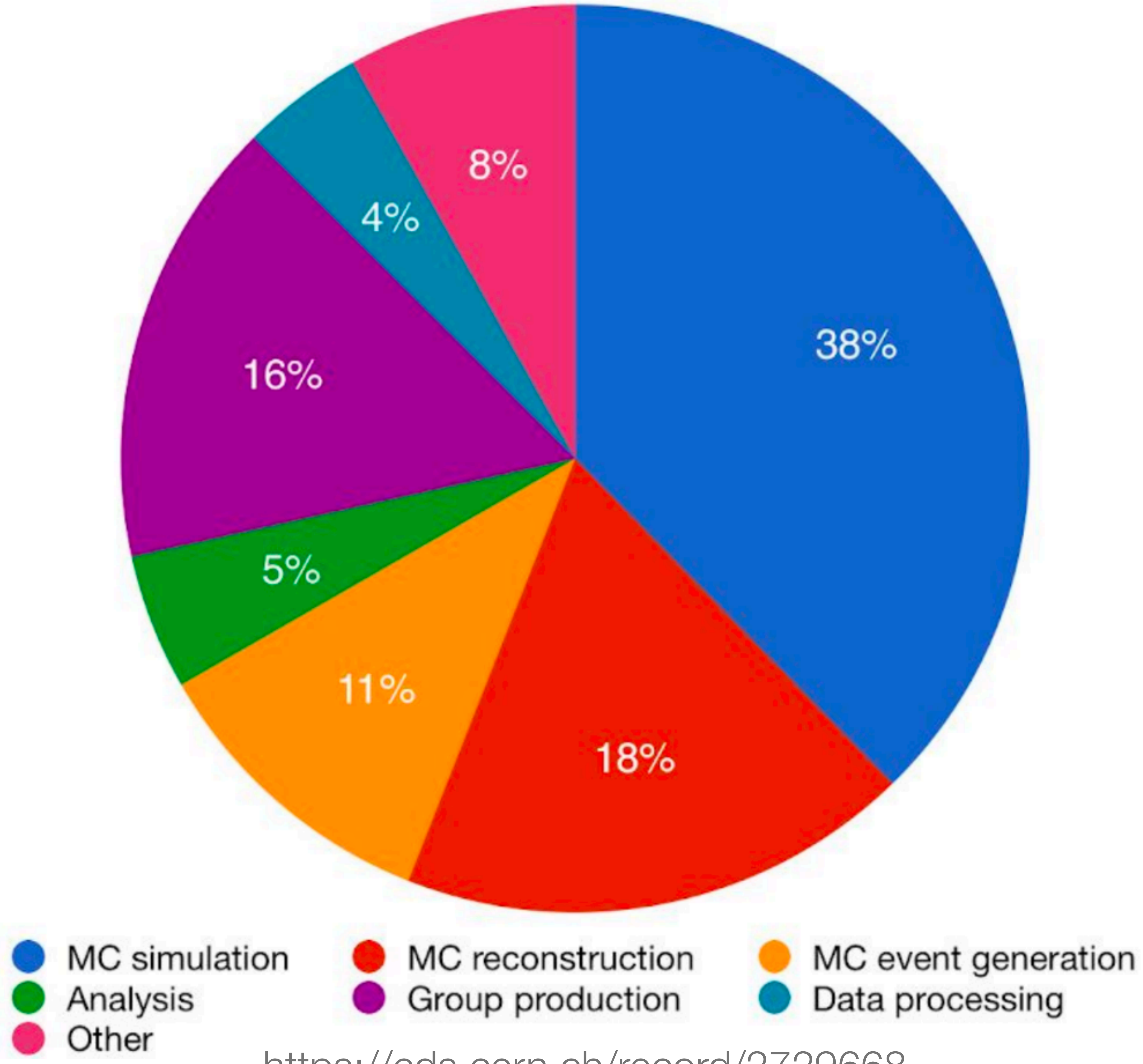
- Train model at trigger level



Making Predictions is Hard

Monte Carlo Simulation is slow

- Takes up major part of ATLAS compute budget
- Need lost of MC data
- Bottleneck for future data analysis in HEP



https://cds.cern.ch/record/2729668

Making Predictions is Hard

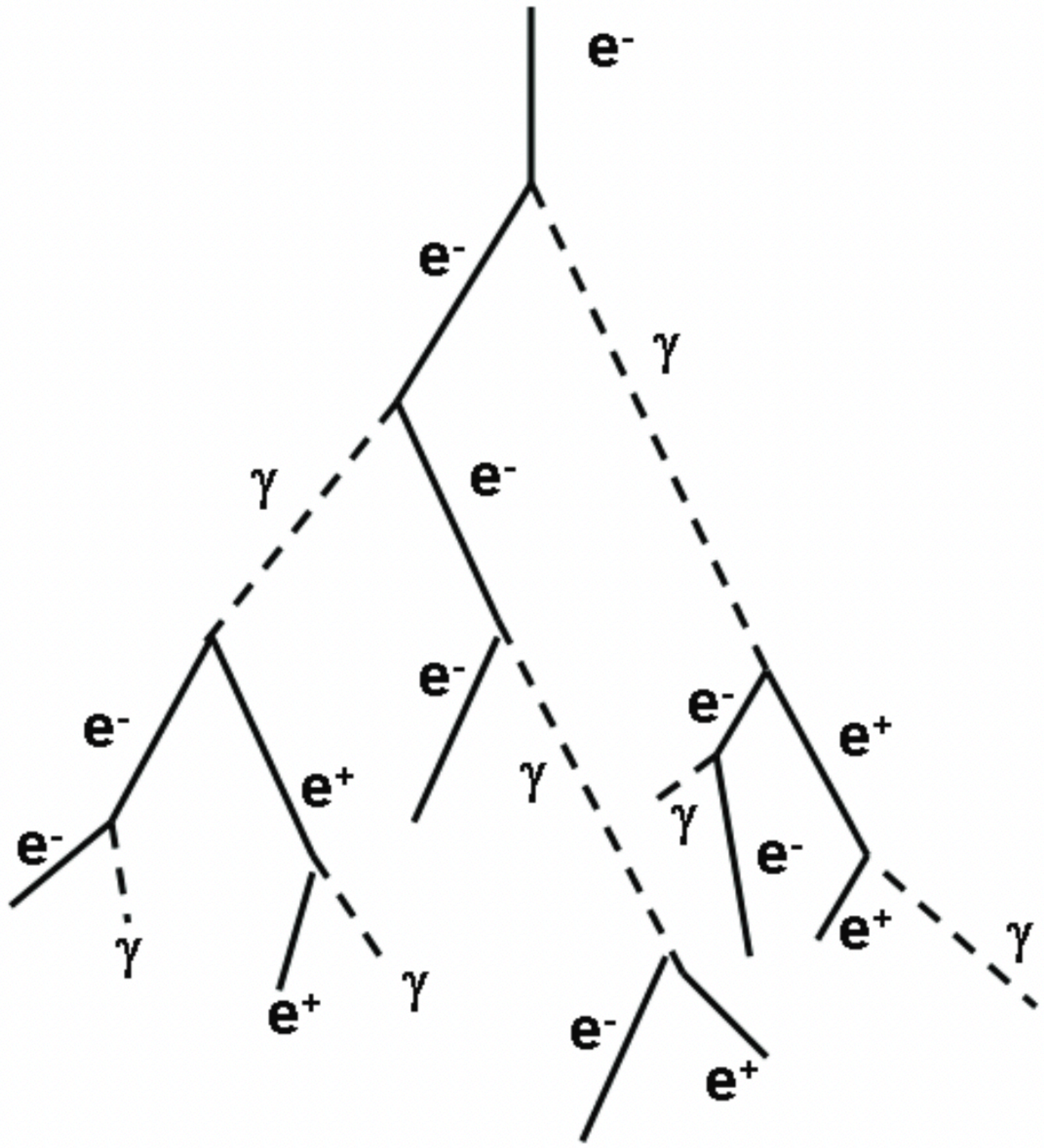
Example: Calorimeter Simulation

- Particle deposits energy
- Calorimeter measures energy
- Problem:
  - Particle don’t just disappear

Making Predictions is Hard

Example: Calorimeter Simulatior

- Particle deposits energy
- Calorimeter measures energy
- Problem:
  - Particle don’t just disappear
  - They create a cascade\ shower of particles
  - Need to model ever particle

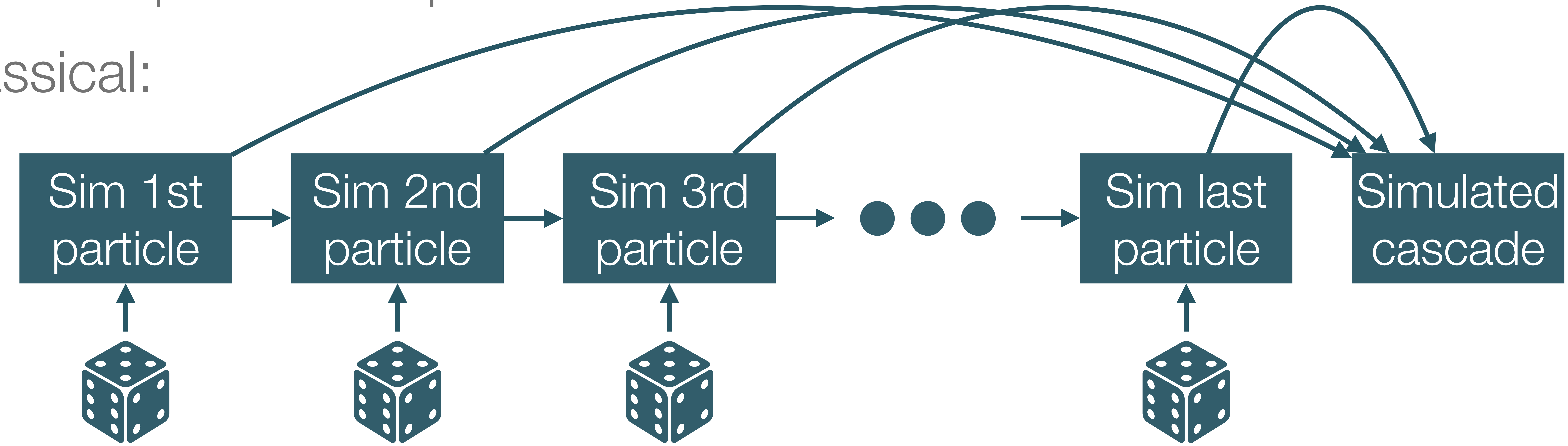


P. L. Rocca et. al.: The Use of Avalanche Photodiodes in High Energy Electromagnetic Calorimetry (2011)

Making Predictions is Hard

Can we speed this up?

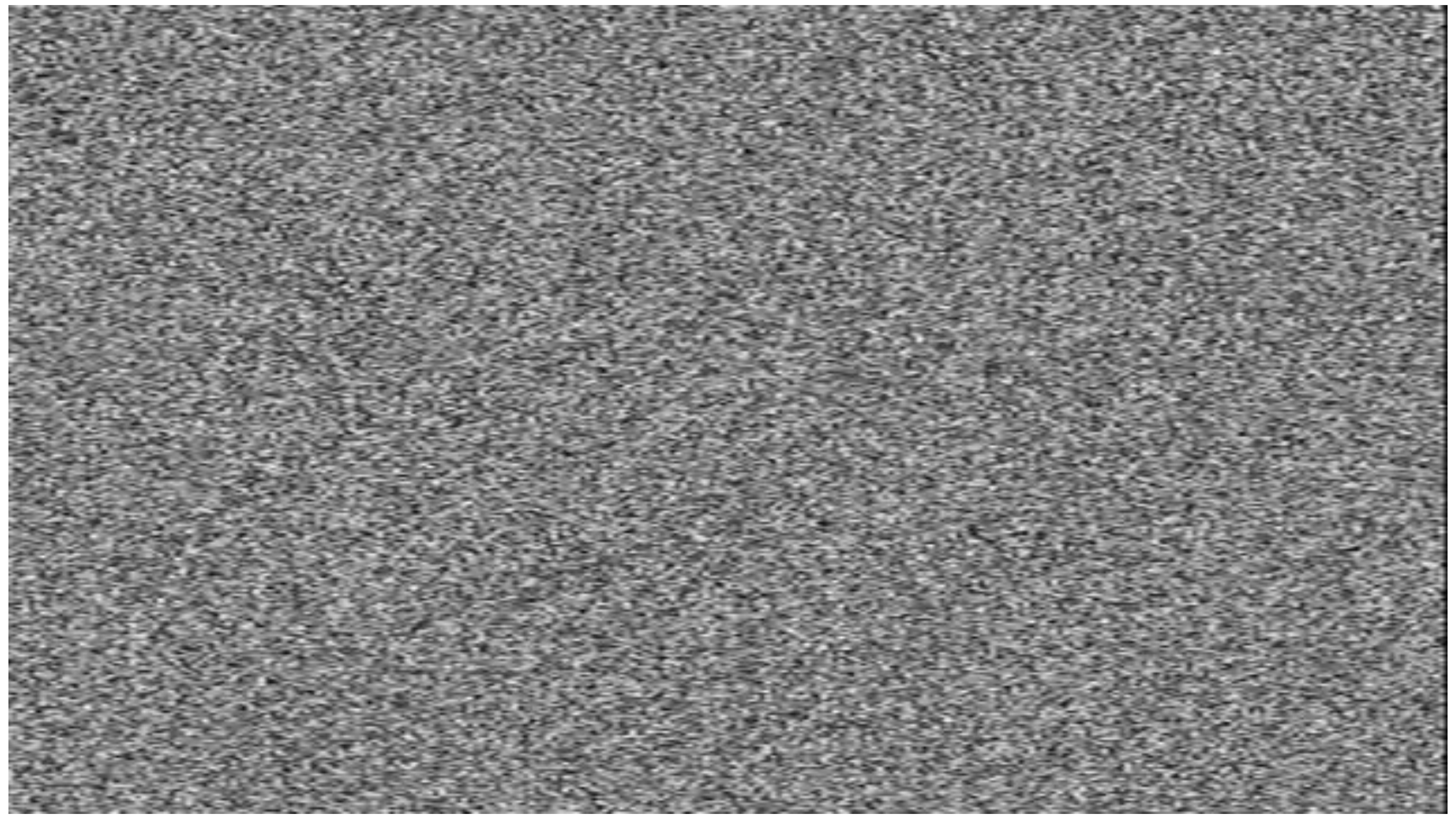
Classical:



Making Predictions is Hard

Generative Neural Networks

- Map random noise to structured outputs
- Slow to train, fast to evaluate

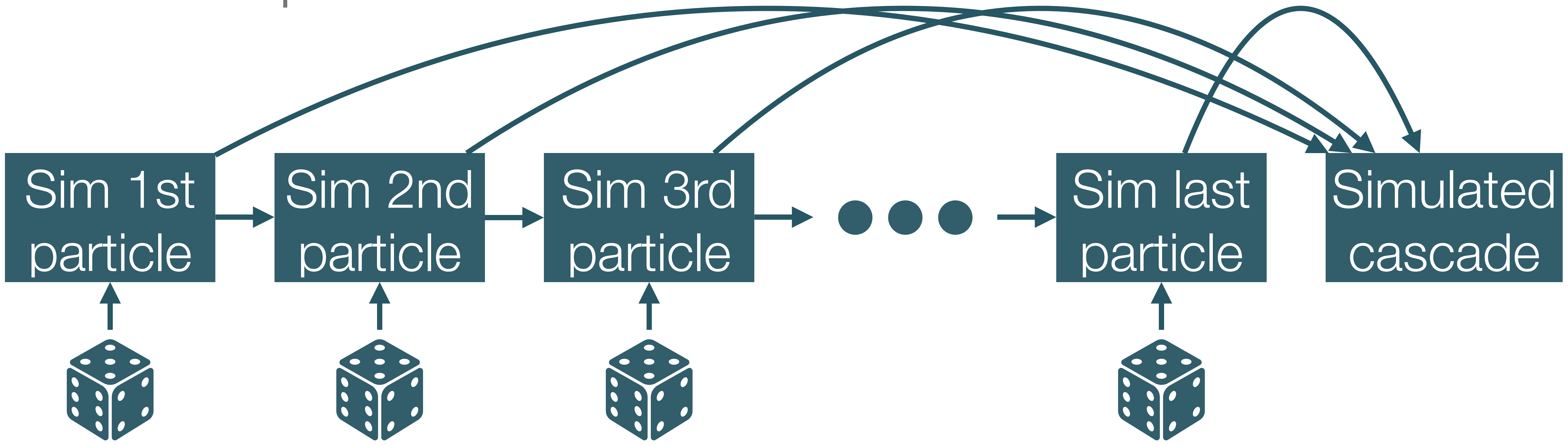


DALL-E 3

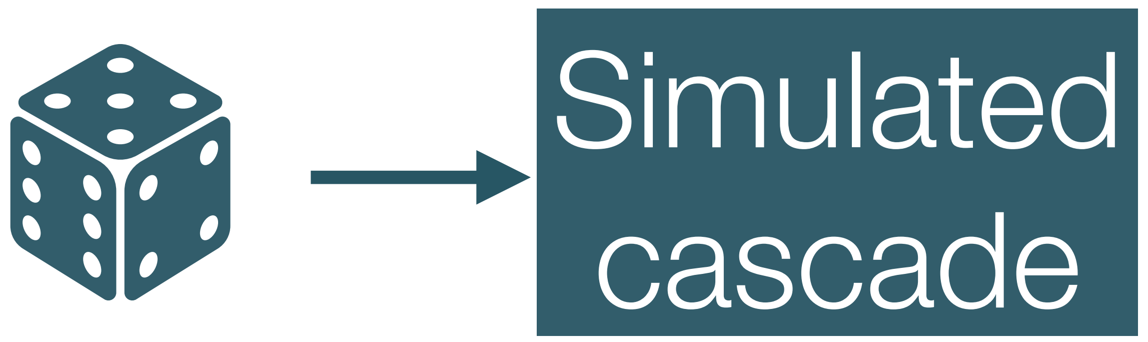
Making Predictions is Hard

Can we speed this up?

Classical:



Machine learning:



Making Predictions is Hard

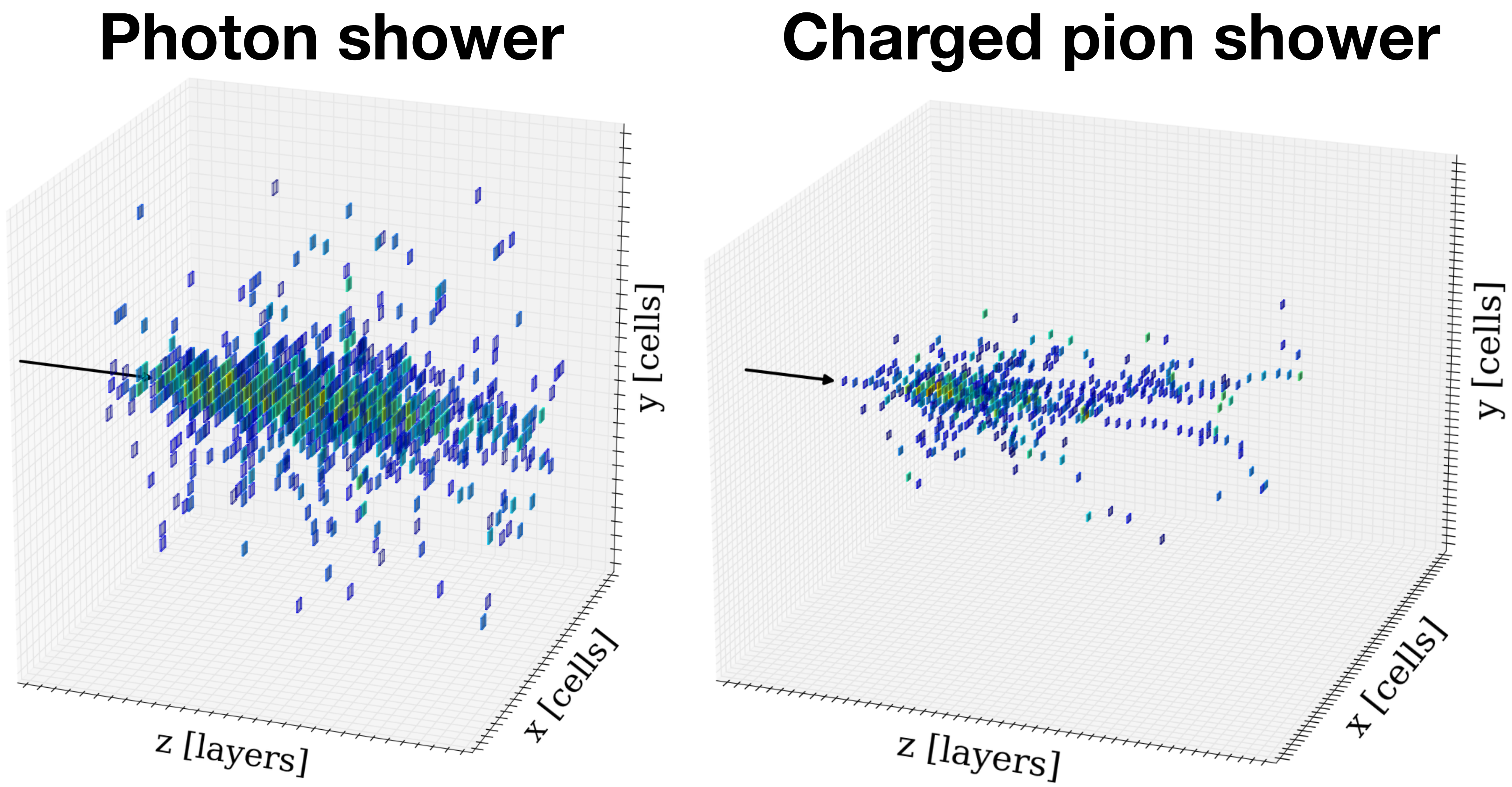
ML Calorimeter Simulation

- Use generative model

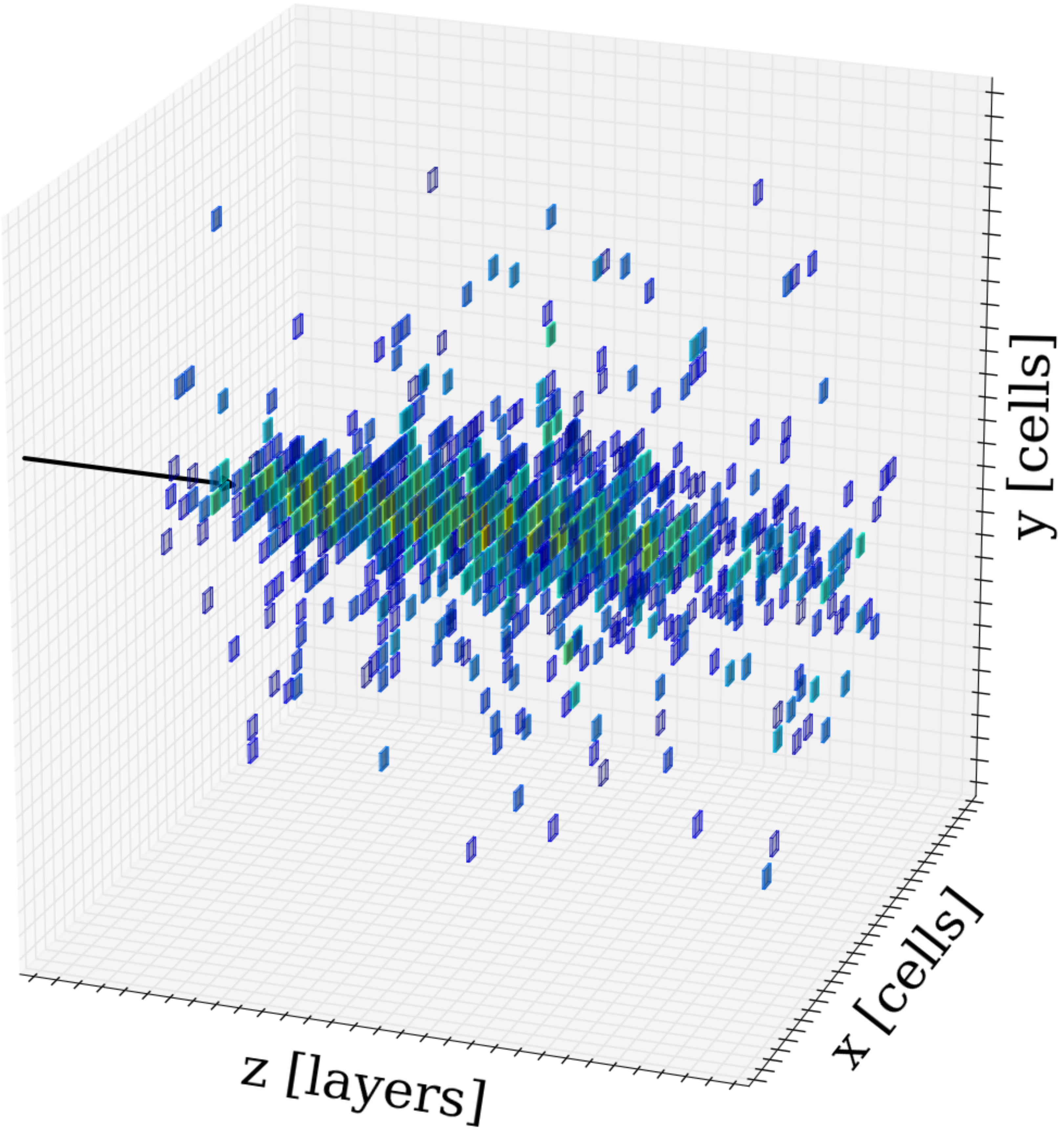
Making Predictions is Hard

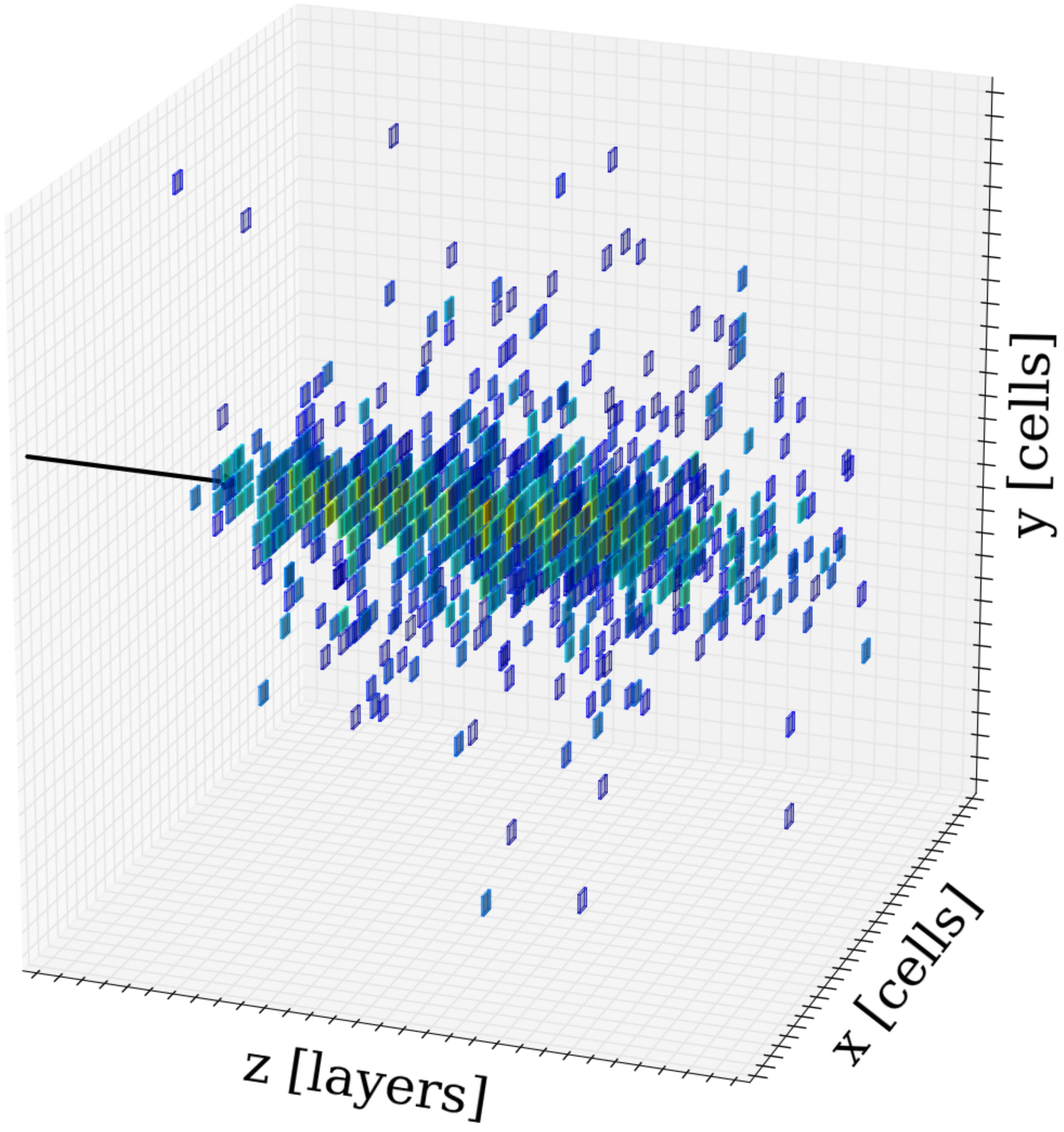
ML Calorimeter Simulation

- Use generative model
- Replace 2D image output with calorimeter shower
- Train model in classical simulation
- Generate new data fast

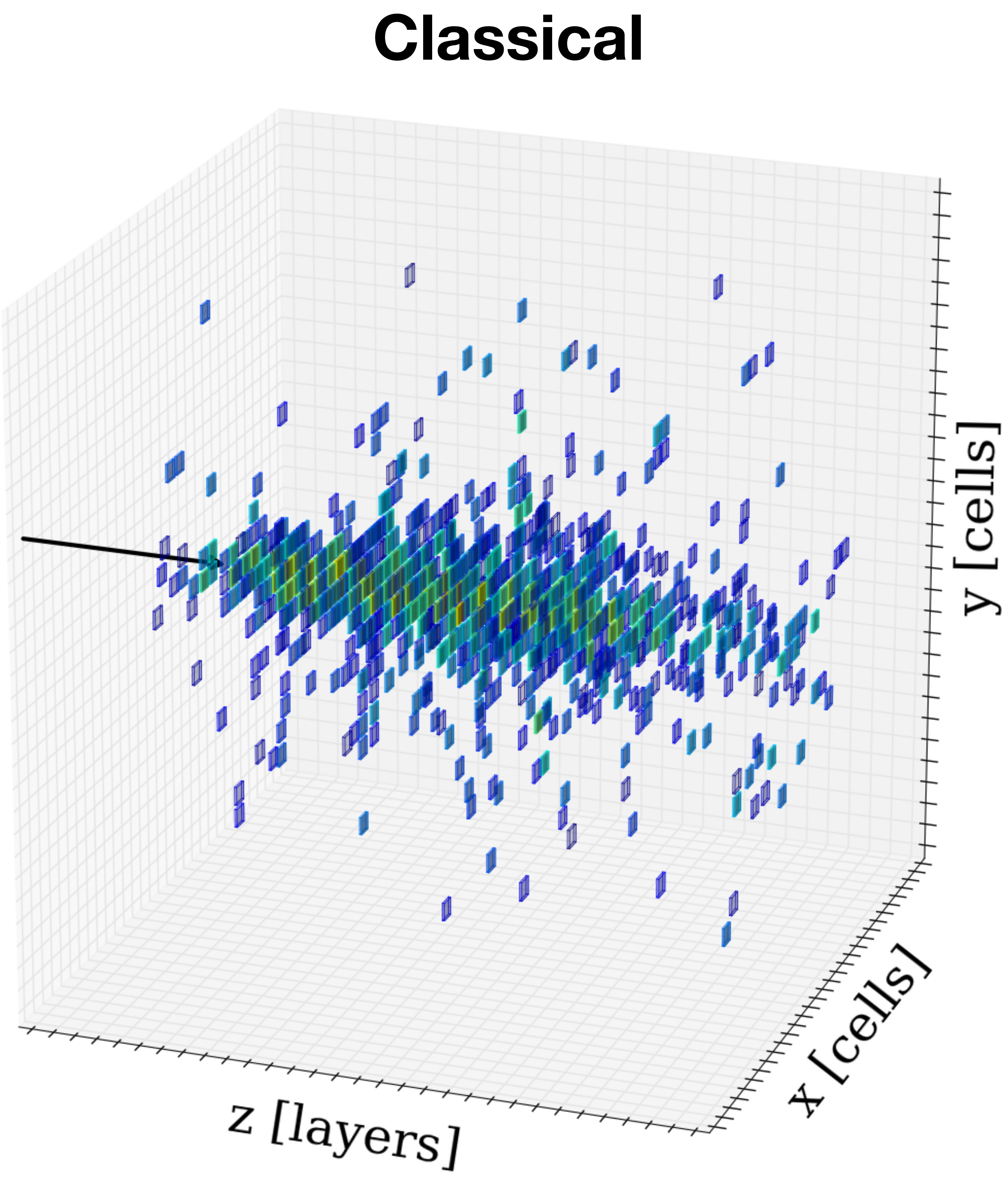


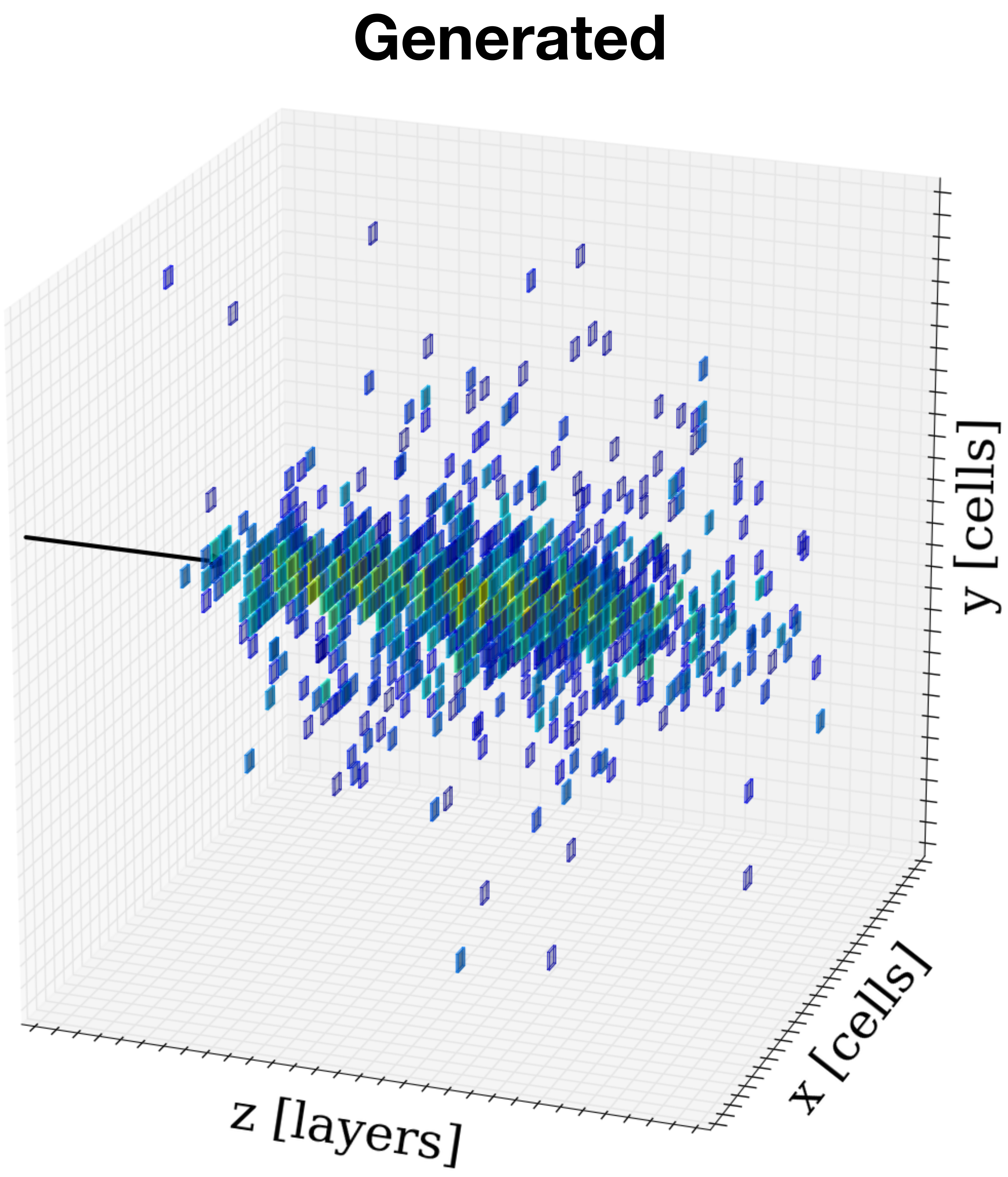
Making Predictions is Hard





Making Predictions is Hard

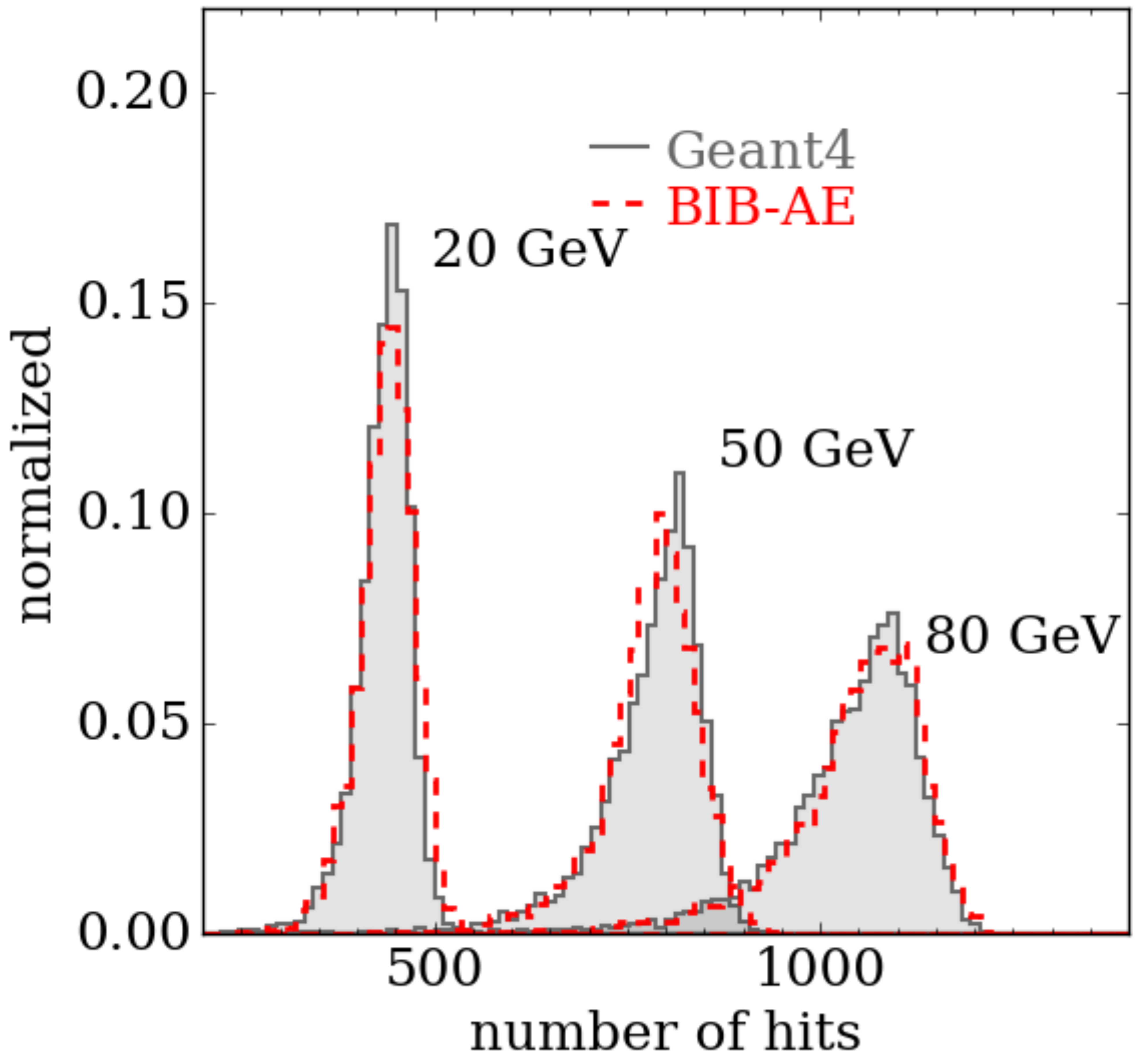


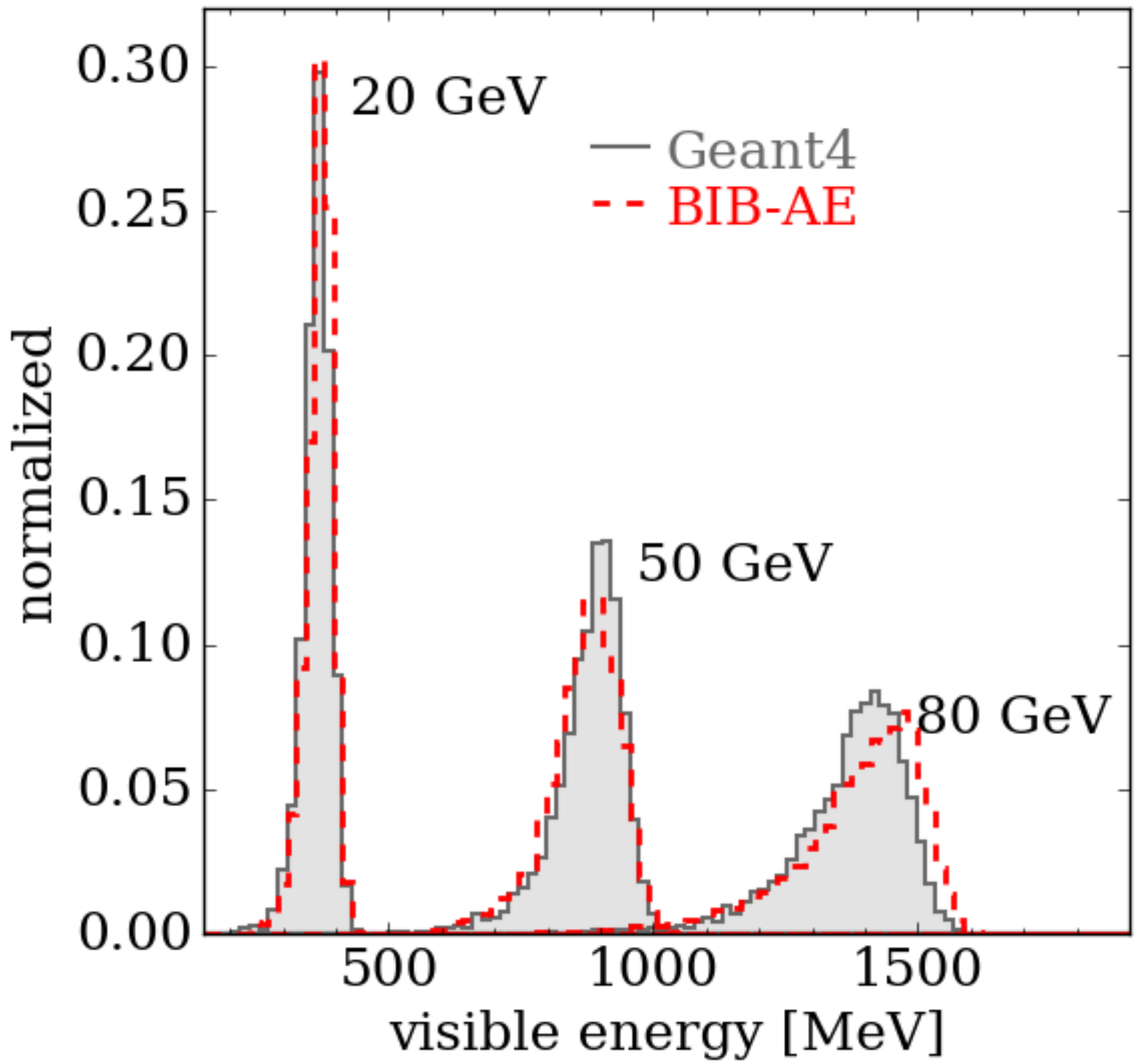


Does it work?

- Can't directly compare generated showers
- Individual examples meaningless anyway
- Need to look at distributions

Making Predictions is Hard





Making Predictions is Hard

Simulator	Hardware	40 GeV Pions	Speedup	160 GeV Pions	Speedup
GEANT4	CPU	4475±178 ms	-	14544±479 ms	-
BIB-AE	CPU	254.41±0.08 ms	x18	253.82±0.02 ms	x57
BIB-AE	GPU	2.842±0.003 ms	x1575	2.732±0.004	×5324

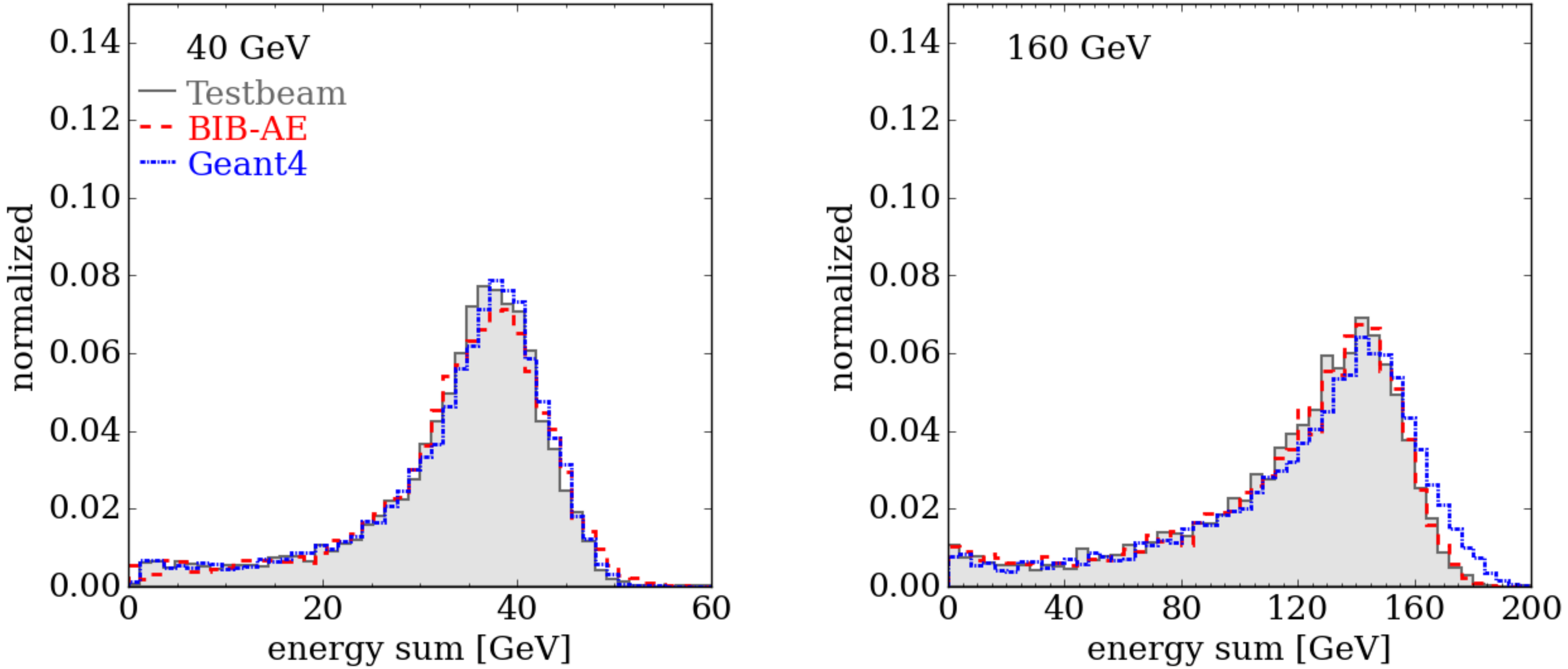
Making Predictions is Hard

Going Beyond: Training on Data

- Getting labeled real data is rare
- Testbeam data
- Outperform classical simulation?

D. Heuchel: *Particle Flow Studies with Highly Granular Calorimeter Data* (2022)

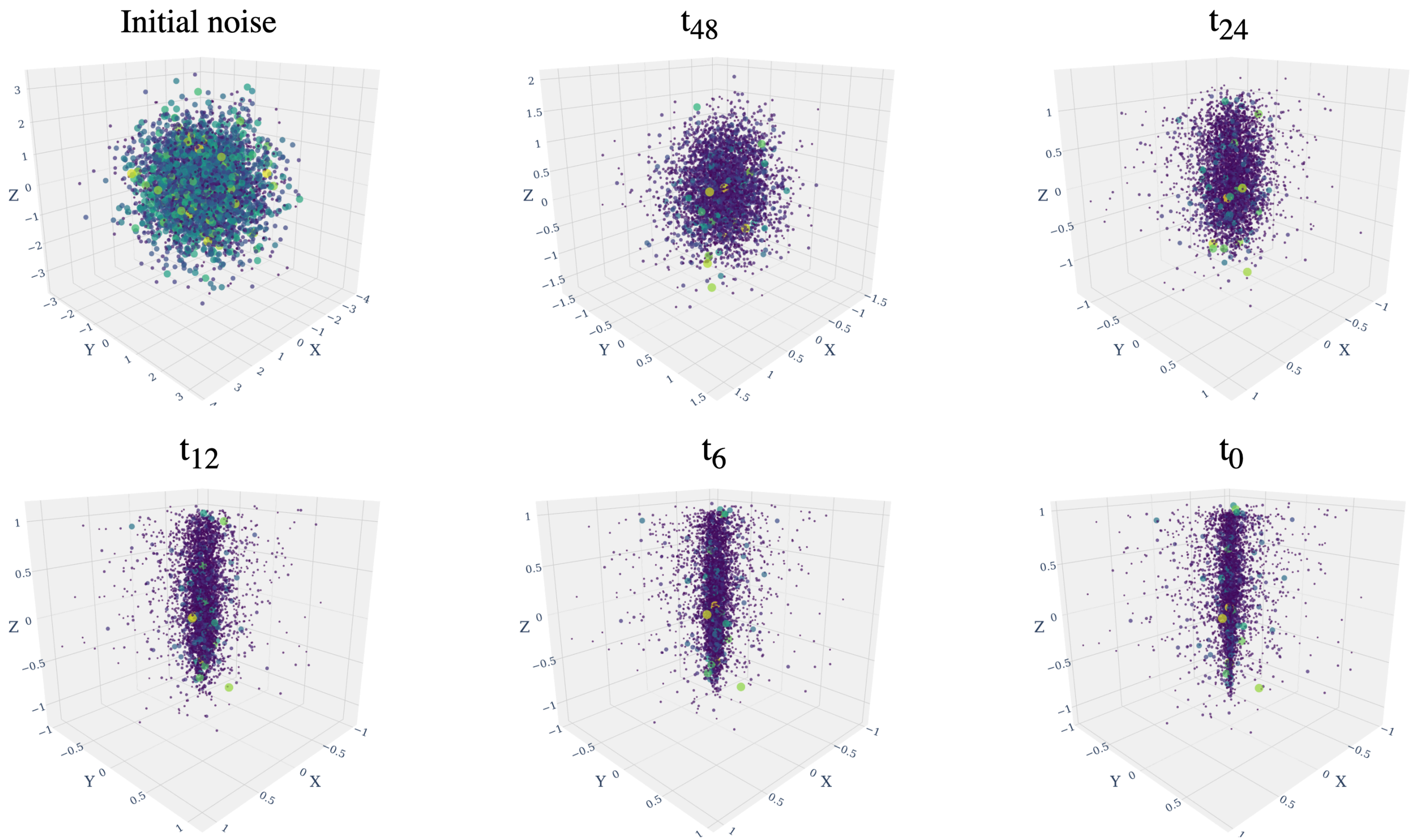
Making Predictions is Hard



Making Predictions is Hard

Cutting Edge:

- Diffusion models
- Move to point clouds
- Little CS precedent
- Largely HEP ML driven



Analyzing Data is Hard

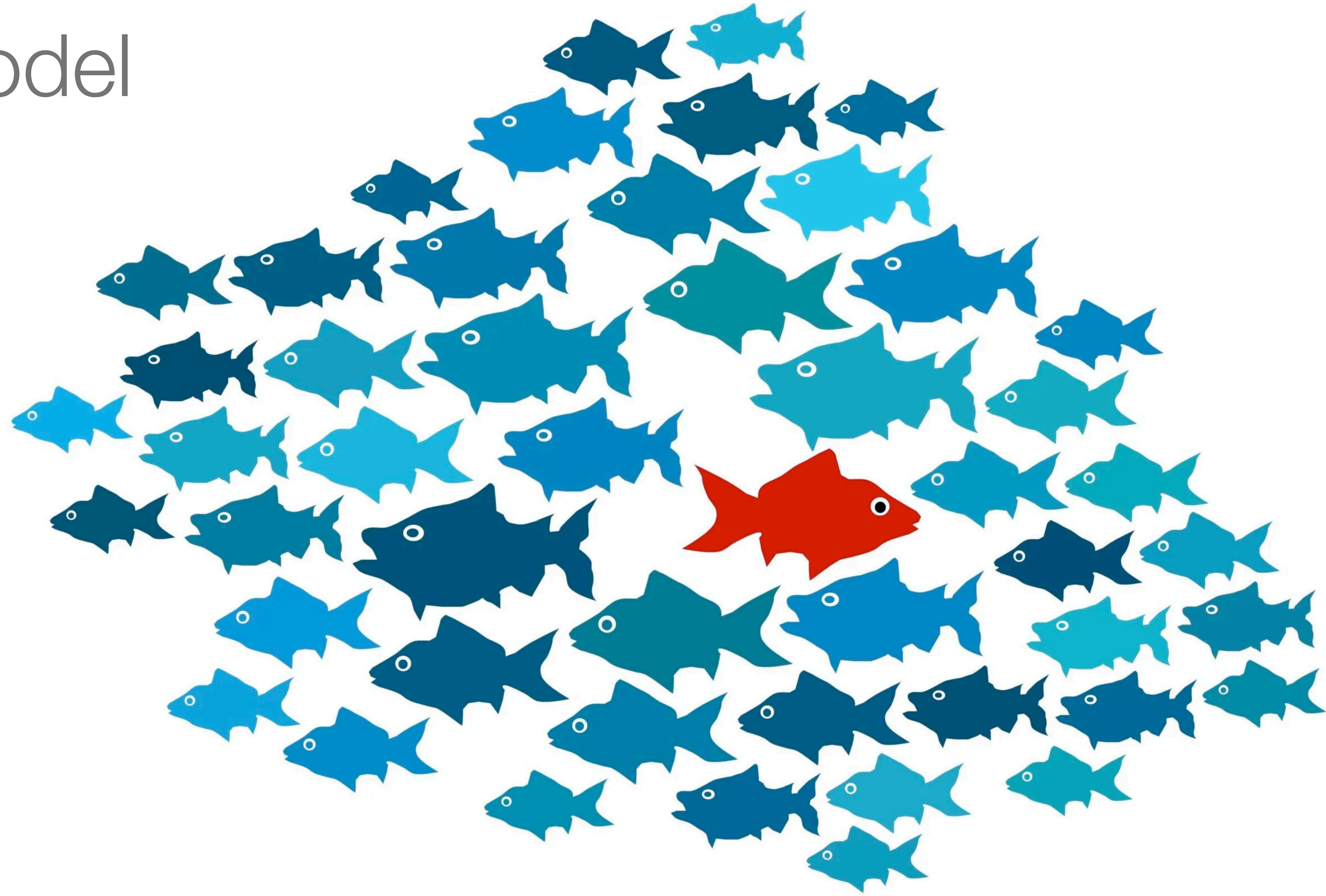
So far: no new discoveries of particles beyond the Standard Model

- Maybe not enough data

  → Previous ML methods

- Not looking for the right theory

  → Anomaly detection



https://www.tatvic.com/blog/detecting-real-time-anomalies-using-r-google-analytics-360-data/

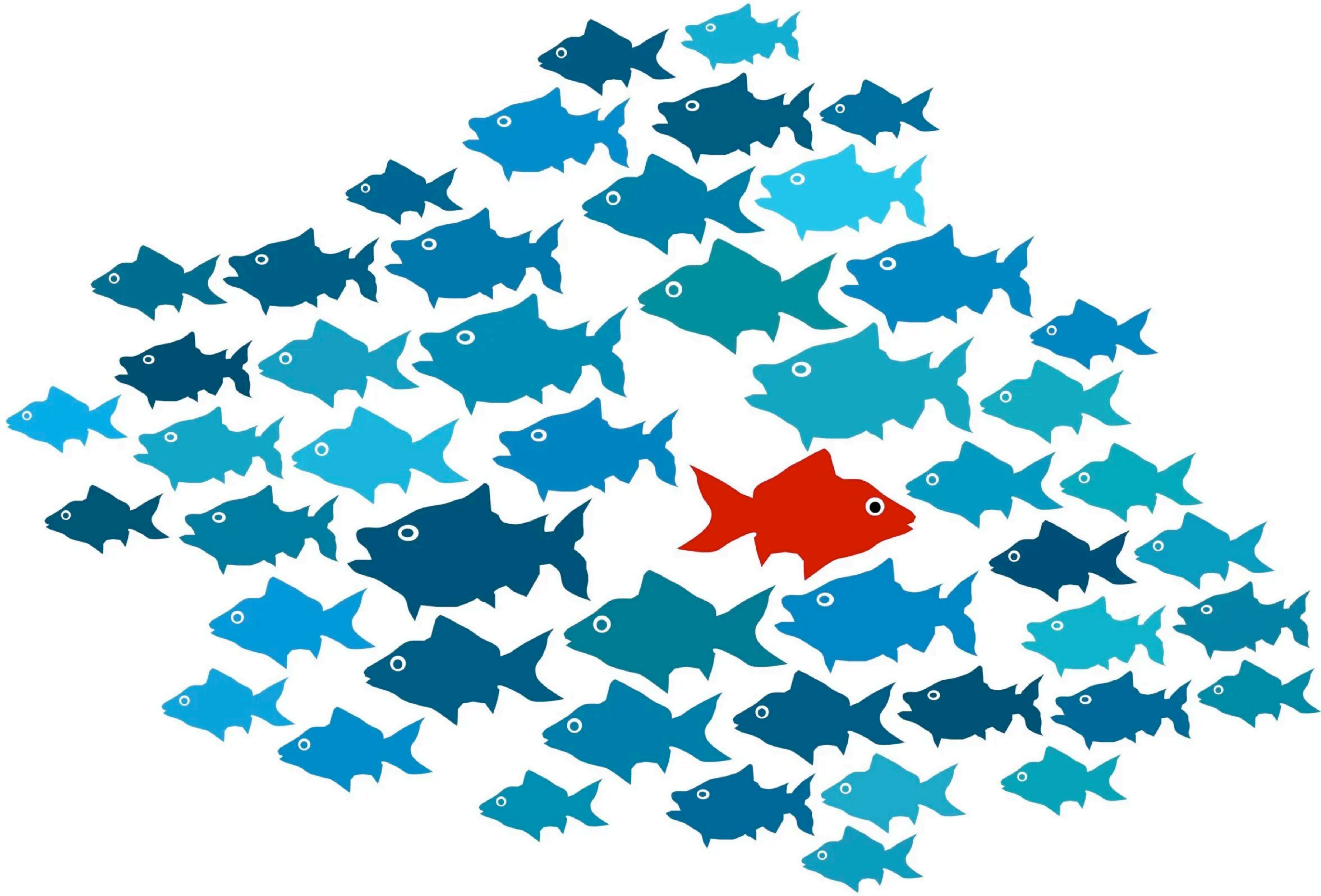
Analyzing Data is Hard

Anomaly detection

- Have ML model look through LHC data for
  - Rare events

    → Unsupervised AD
  - Unexpected events

    → Semi-supervised AD

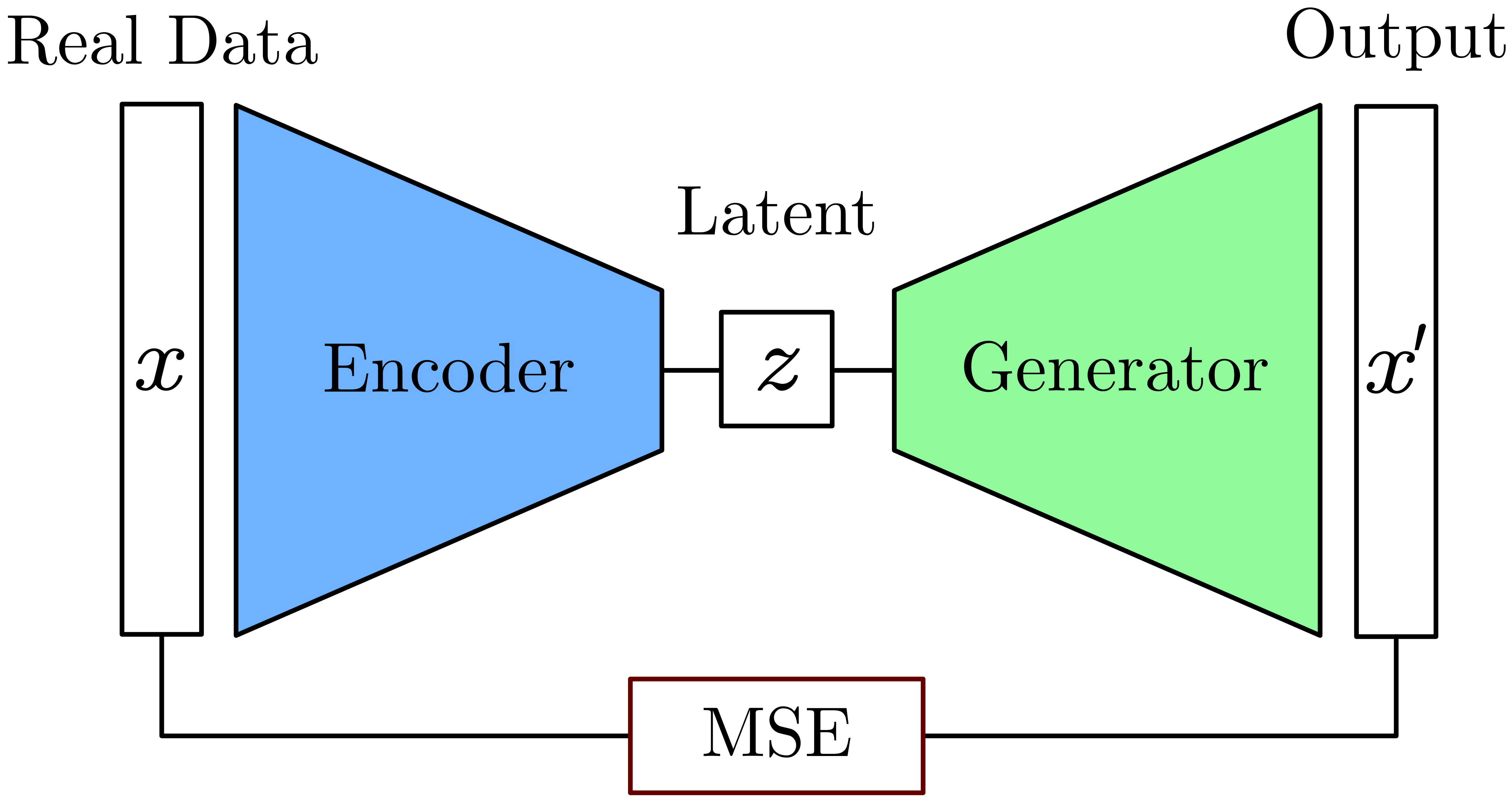


https://www.tatvic.com/blog/detecting-real-time-anomalies-using-r-google-analytics-360-data/

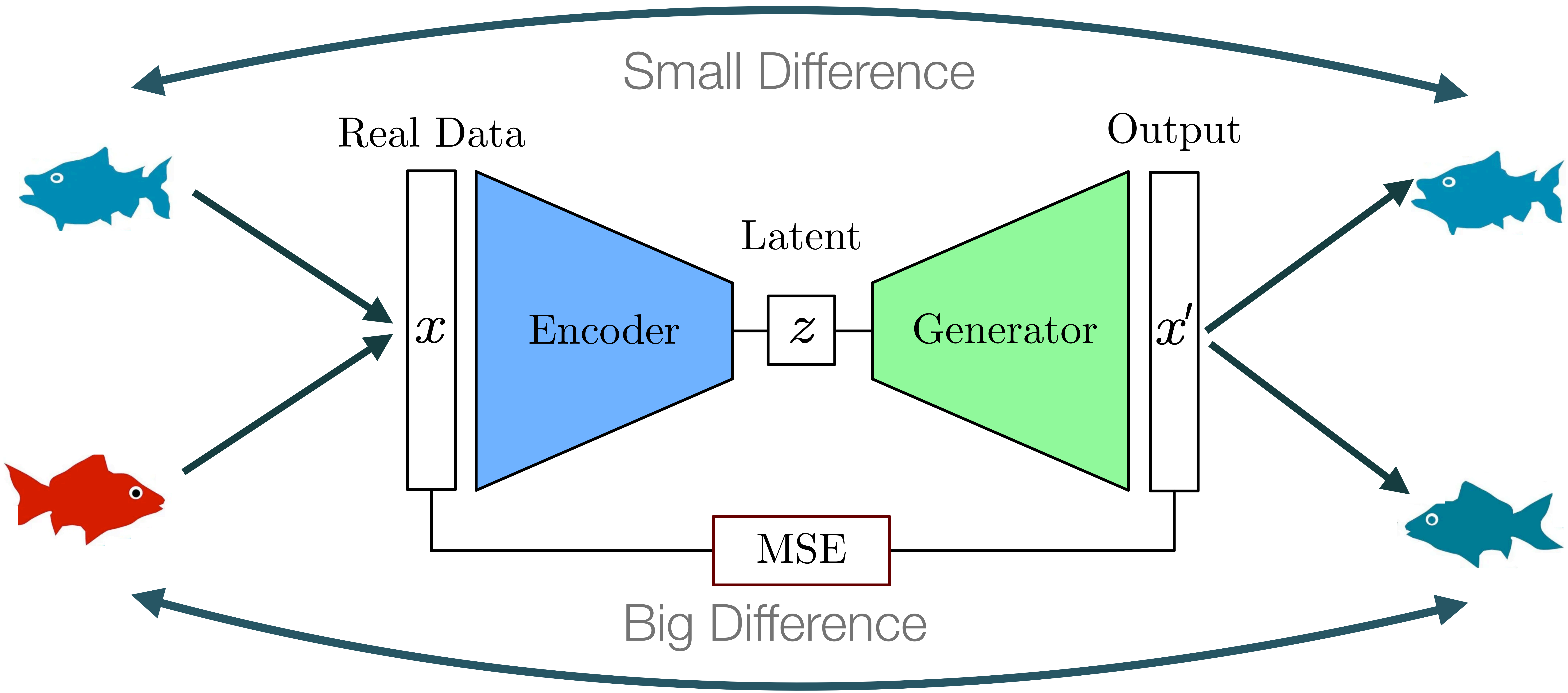
Analyzing Data is Hard

Unsupervised AD

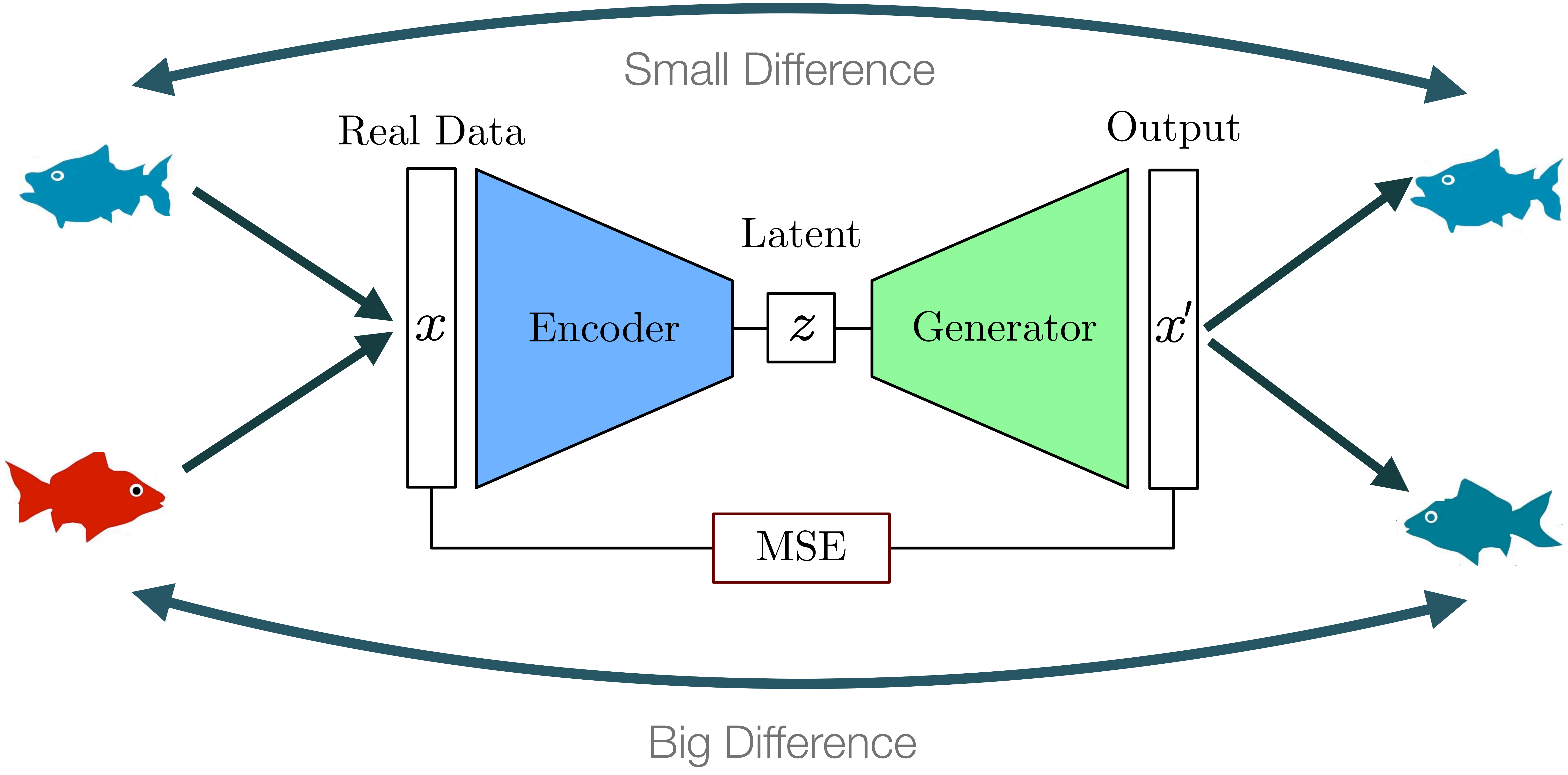
- Find rare samples
- Autoencoder Neural Network
  - Learns to encode events into smaller representation
  - Reconstructs events from representation



Analyzing Data is Hard



Analyzing Data is Hard

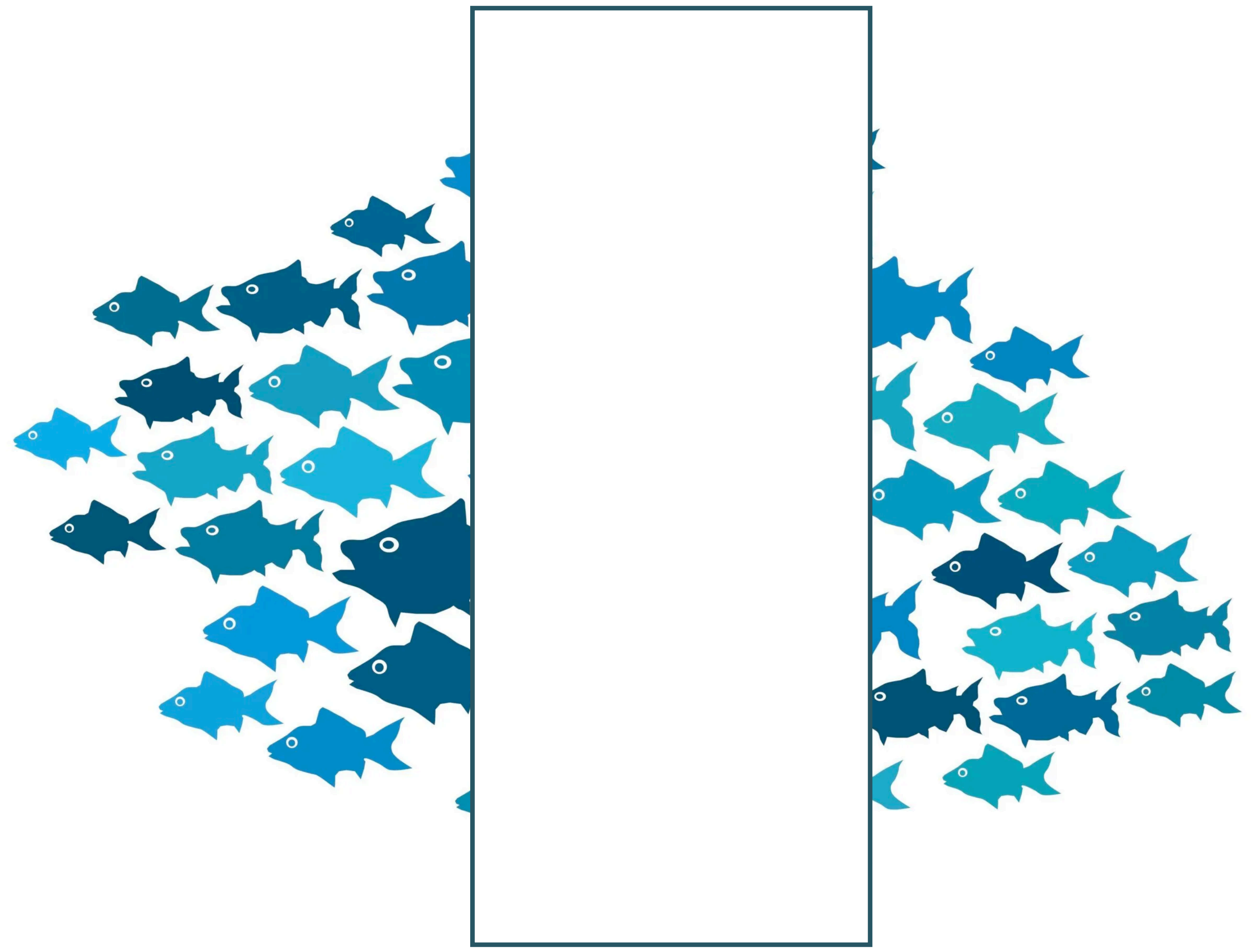


- Used for outlier detection in industry
  - Production Quality Assessment
  - Predicting hard drive failures
- Less suited for HEP
  - Interesting events similar to boring

Analyzing Data is Hard

Semi-Supervised AD

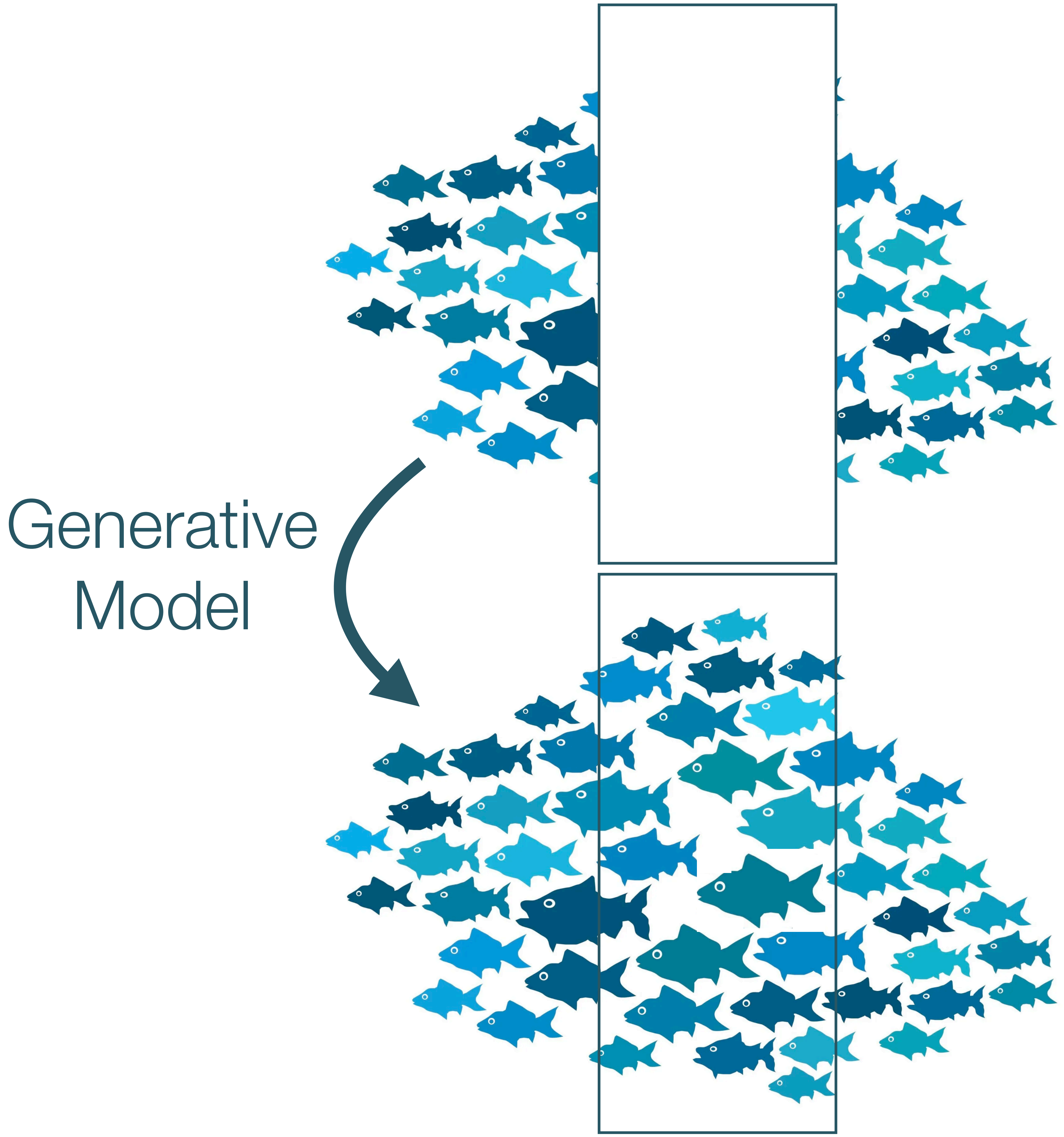
- Find unexpected samples
  - Cut out part of data
  - Use generative model to learn data



Analyzing Data is Hard

Semi-Supervised AD

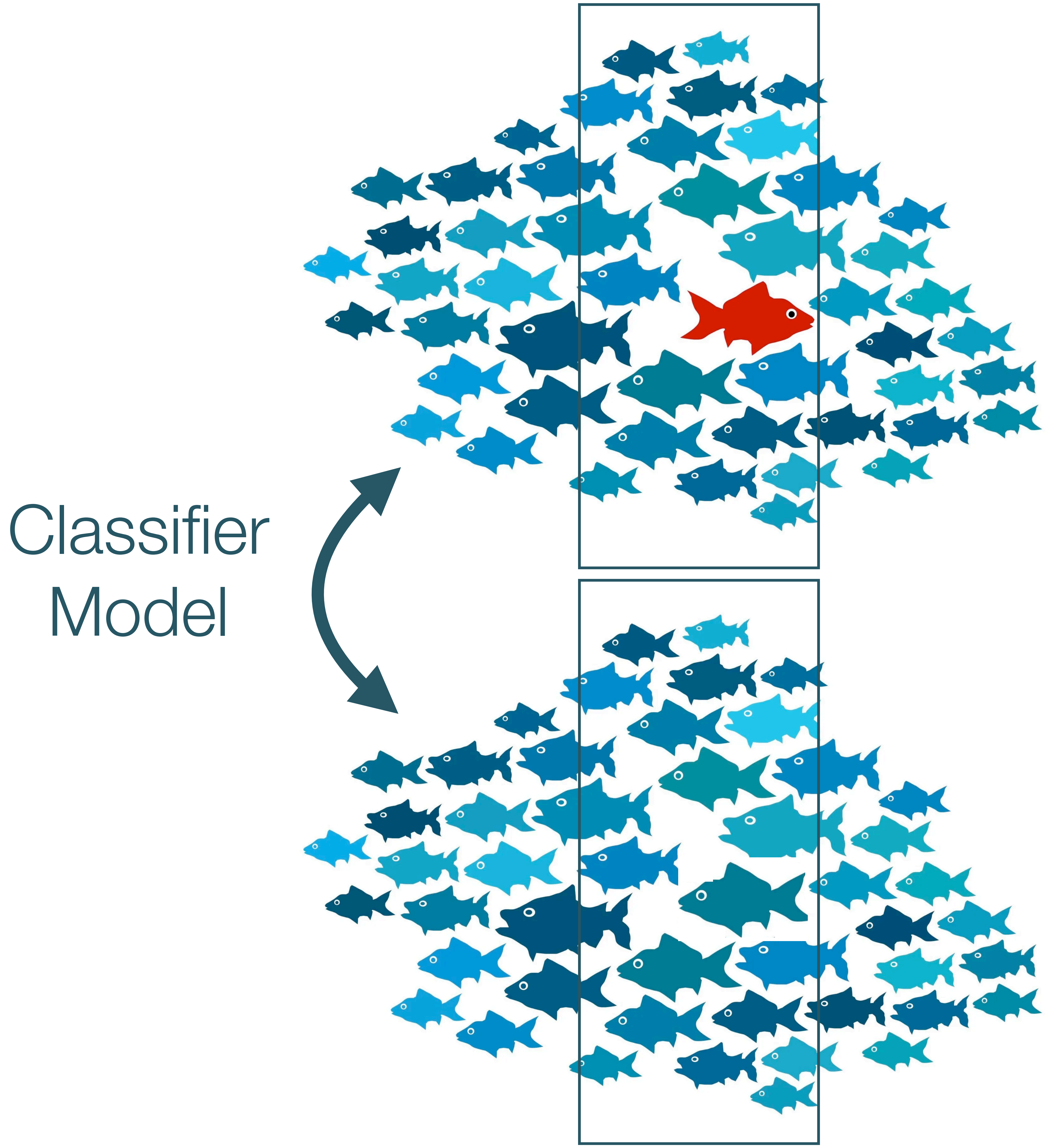
- Find unexpected samples
  - Cut out part of data
  - Use generative model to learn data
  - Predict cut out part



Analyzing Data is Hard

Semi-Supervised AD

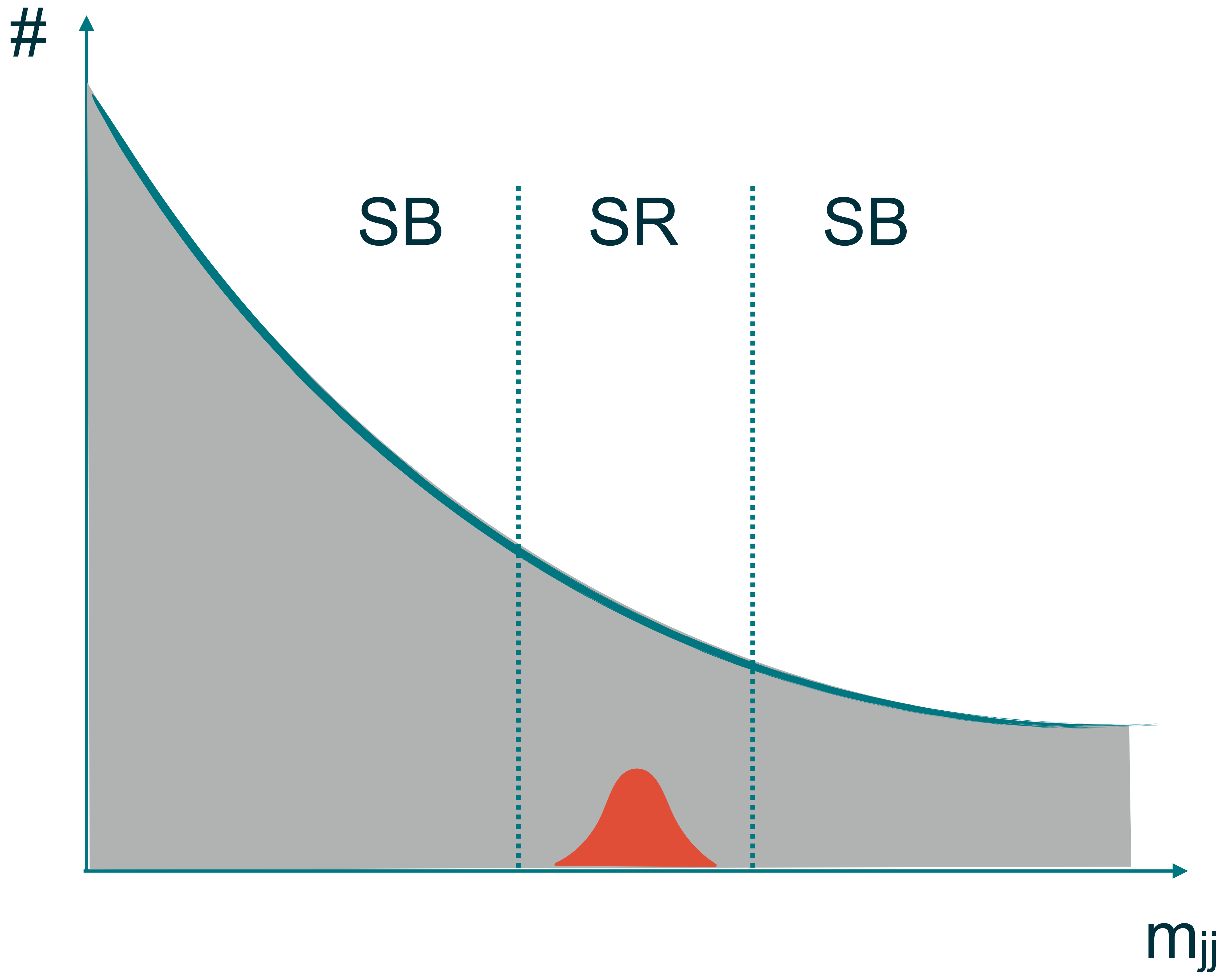
- Find unexpected samples
  - Cut out part of data
  - Use generative model to learn data
  - Predict cut out part
  - Compare prediction and cut out



Analyzing Data is Hard

In High Energy Physics

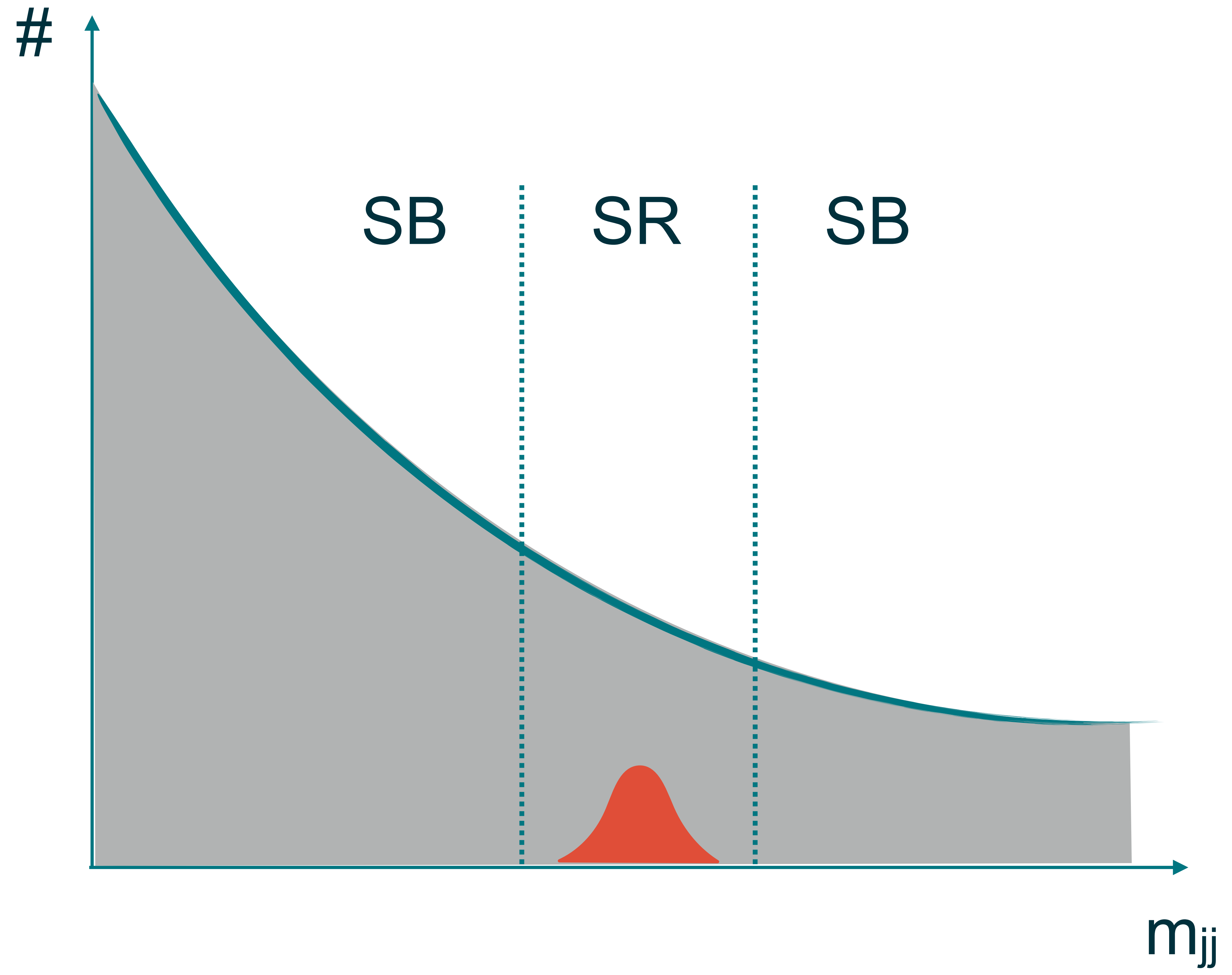
- Define and cut out Signal Region (SR)
- Train on remaining region (Side Band, SB)
- Compare Prediction and SR



Analyzing Data is Hard

In High Energy Physics

- Define and cut out Signal Region (SR)
- Train on remaining region (Side Band, SB)
- Compare Prediction and SR
- State of the Art AD method
- Starting to see application!



Conclusion

Conclusion

- Machine Learning is finding applications all over High Energy Physics
  - Classification/Fast Simulation/Anomaly Detection/etc.
  - Very exciting and fast moving field
  - Close connection to cutting edge computer science
  - Unique challenges requiring unique solutions

Backup

Generative Backup: GANplify

Common point of criticism: information in new samples

- Assume generative model trained on N events
- Used to generate M >> N events

$$\boxed{Info\ (N\ real\ points) \quad = \quad Info\ (M\ gen.\ points)}$$

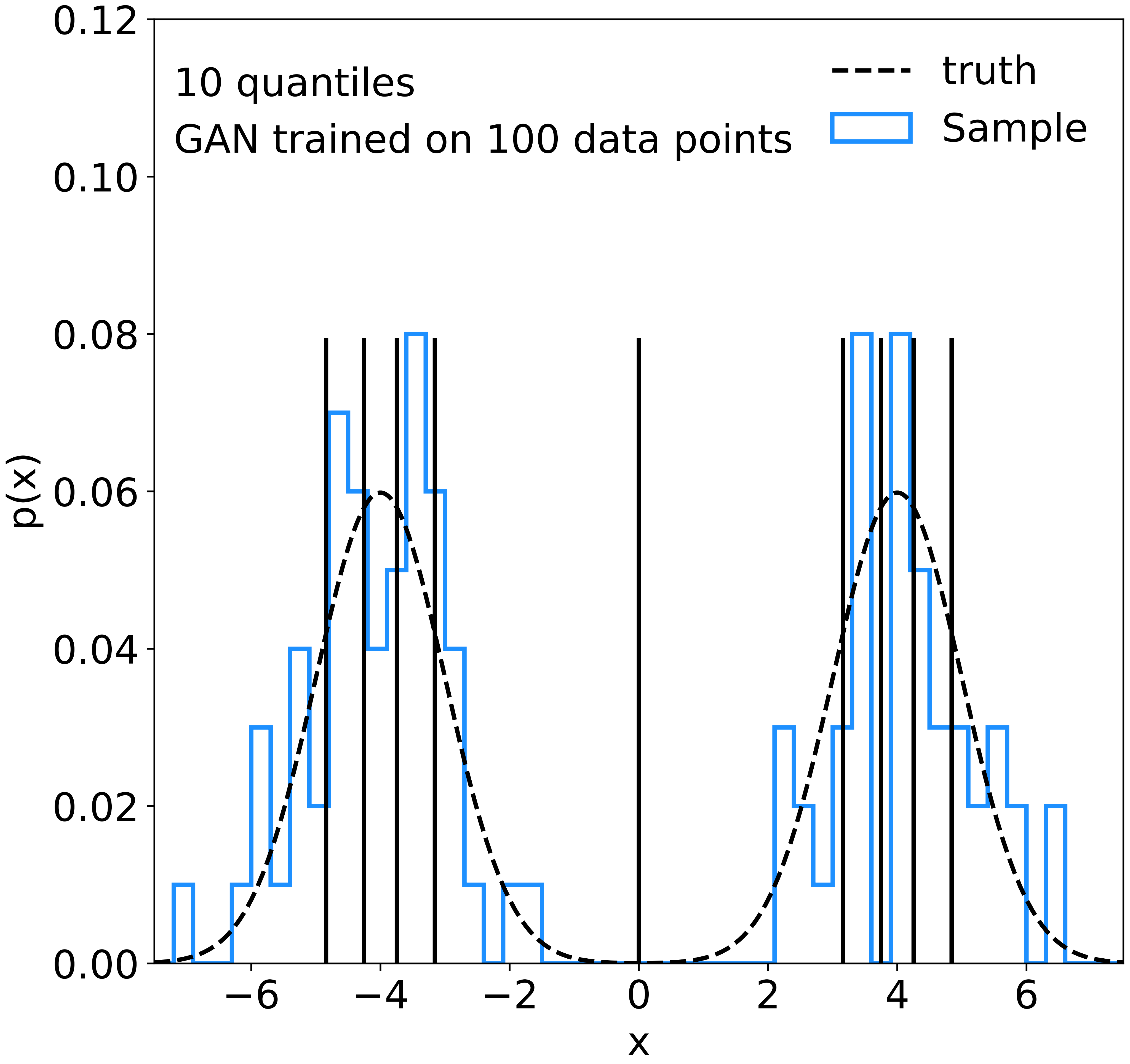
Little advantage to be gained from generative model

$$\boxed{Info\ (N\ real\ points) \quad < \quad Info\ (M\ gen.\ points)}$$

Generative model can speed up simulations

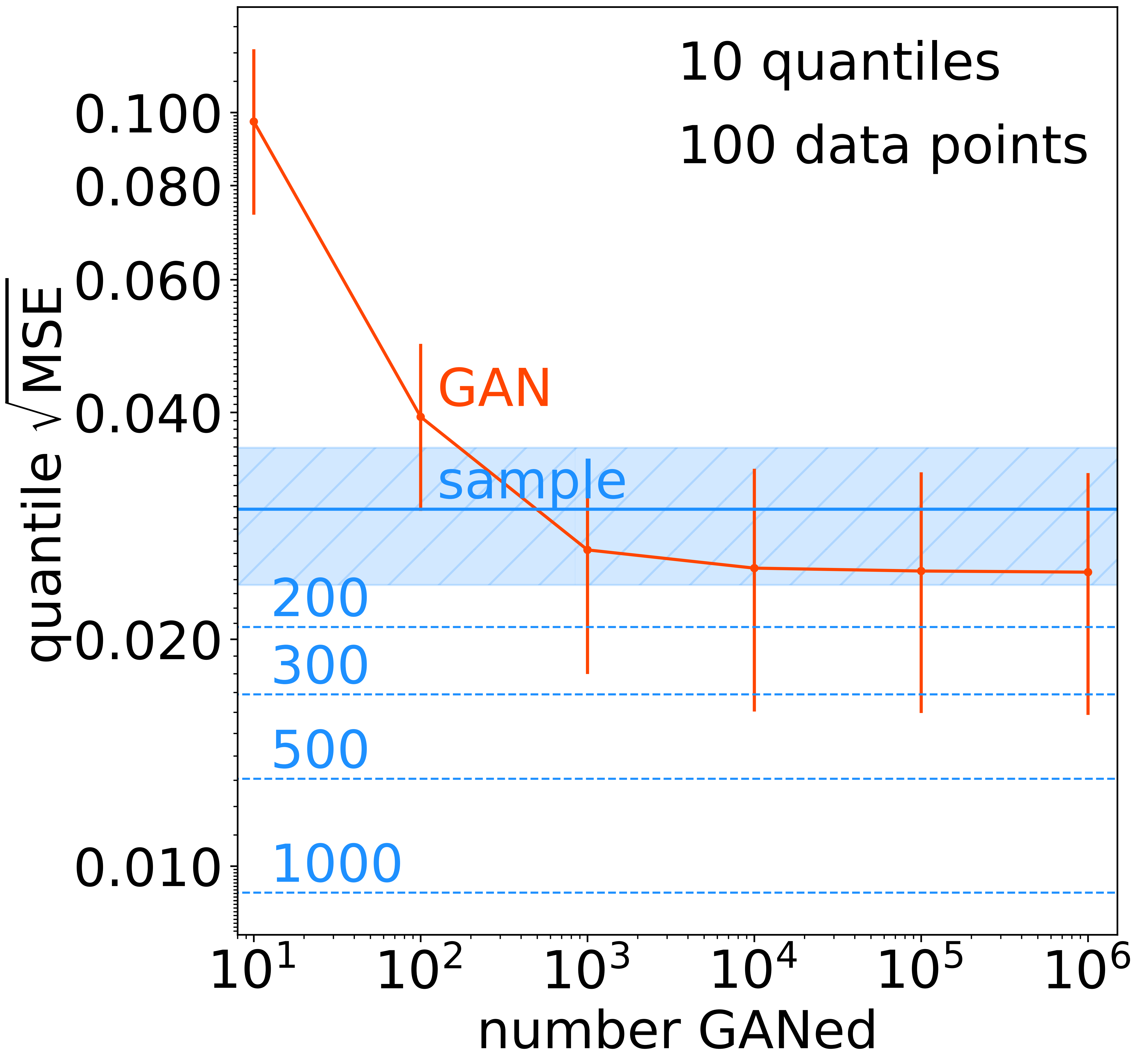
Generative Backup: GANplify

- Draw 100 points from true camel back distribution
- This is designated as the (training) sample
- Calculate fraction of points in each quantile
- Baseline comparison



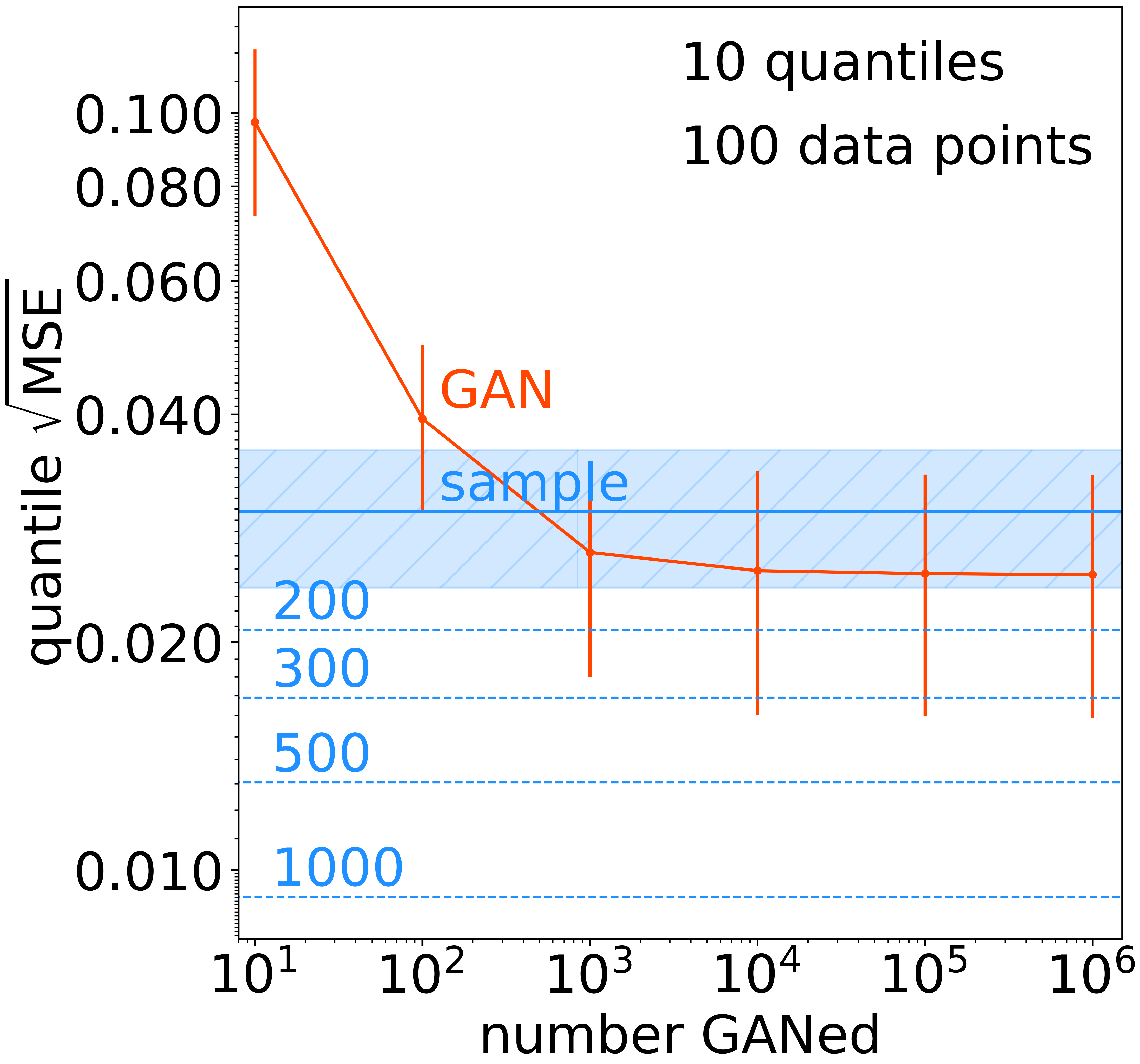
Generative Backup: GANplify

- For 100 training samples, 100 fits and 100 GANs compare MSE
- GAN describes distribution better than training data
- Needs 10,000 GANed points to match 150 true points



Generative Backup: GANplify

- How is this possible?
- In terms of information:
  - sample: only data points
  - GAN: data + smooth, continuous function
- Inductive bias allows the GAN to interpolate



Generative Backup: GANplify

Common point of criticism: information in new samples

- Assume generative model trained on N events
- Used to generate M >> N events

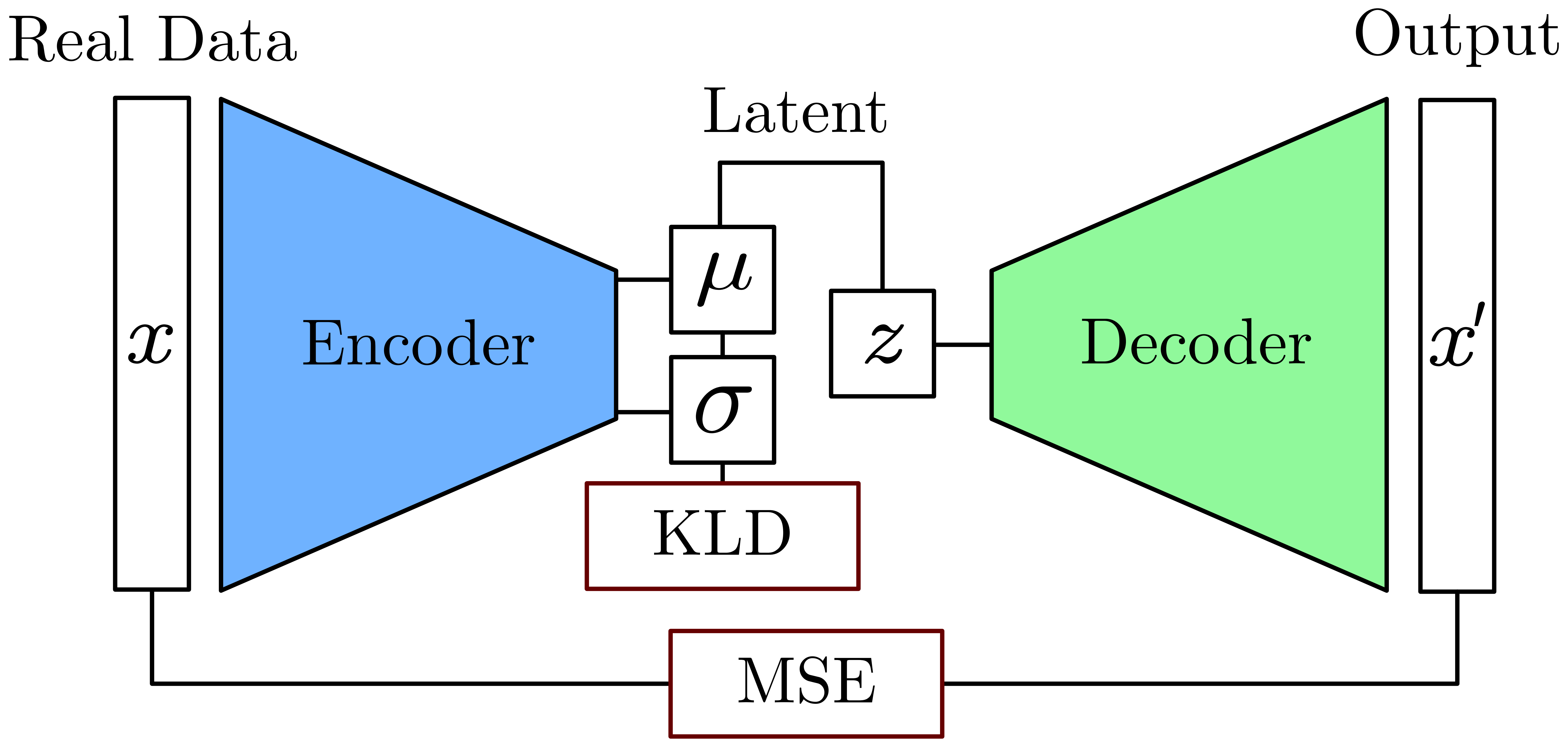
$$\boxed{Info\ (N\ real\ points) \quad = \quad Info\ (M\ gen.\ points)}$$

Little advantage to be gained from generative model

$$\boxed{Info\ (N\ real\ points) \quad < \quad Info\ (M\ gen.\ points)}$$

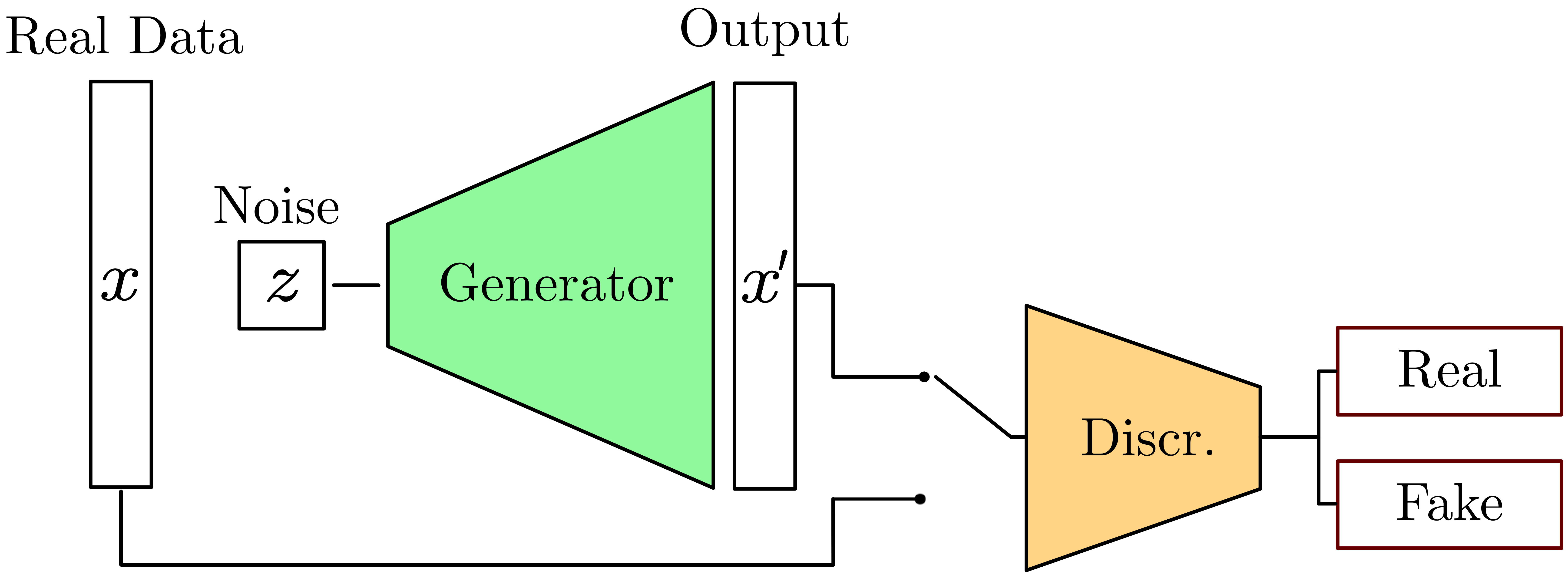
Generative model can speed up simulations

Generative Backup: Variational Autoencoder



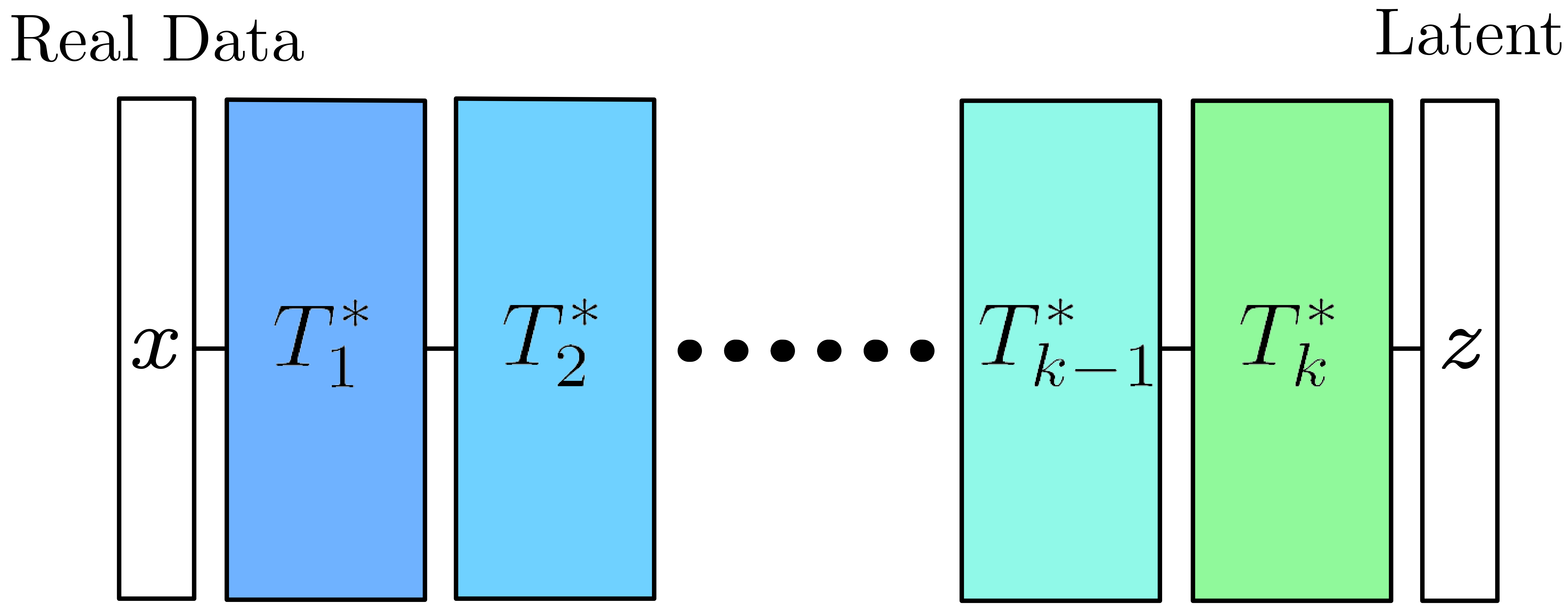
- Encoder: data → latent; Decoder: latent → reconstructed data
- Minimize difference between input and reconstructed data
- Regularize latent → unit Gaussian

Generative Backup: Generative Adversarial Network



- Generator: noise → generated data
- Discriminator: distinguish real and generated data
- Used discriminator as loss to train generator

Generative Backup: Normalizing Flow



- Invertible neural network, both: (data → latent) and (latent → data)
- Directly minimize negative log-likelihood in (data → latent) direction
- Generate data with (latent → data) direction

Machine Learning

Supervised Learning

- Learns to make specific predictions for data
- Labeled data
- Classification
- Regression
- Etc.

Unsupervised Learning

- Learns structure of data
- Unlabeled data
- Generation
- Anomaly Detections
- Etc.