Reconstructing Particle Masses at the LHC using Diffusion-based Models

Rahool Barman

Kavli IPMU (WPI), The University of Tokyo

5th AEI Workshop

IPPP, Durham University

September 30, 2025

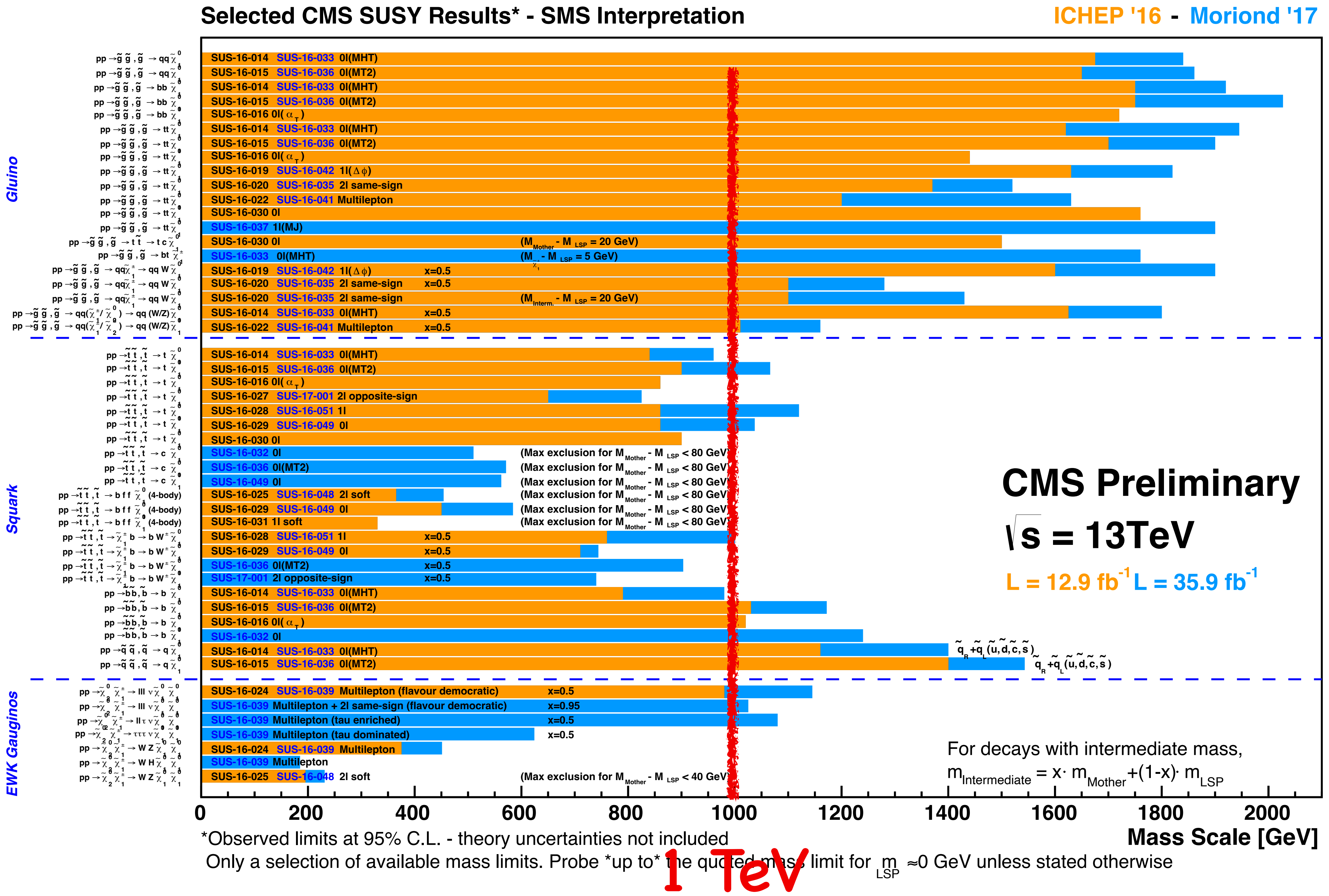
東京大学 THE UNIVERSITY OF TOKYO

IPMU INSTITUTE FOR THE PHYSICS AND MATHEMATICS OF THE UNIVERSE

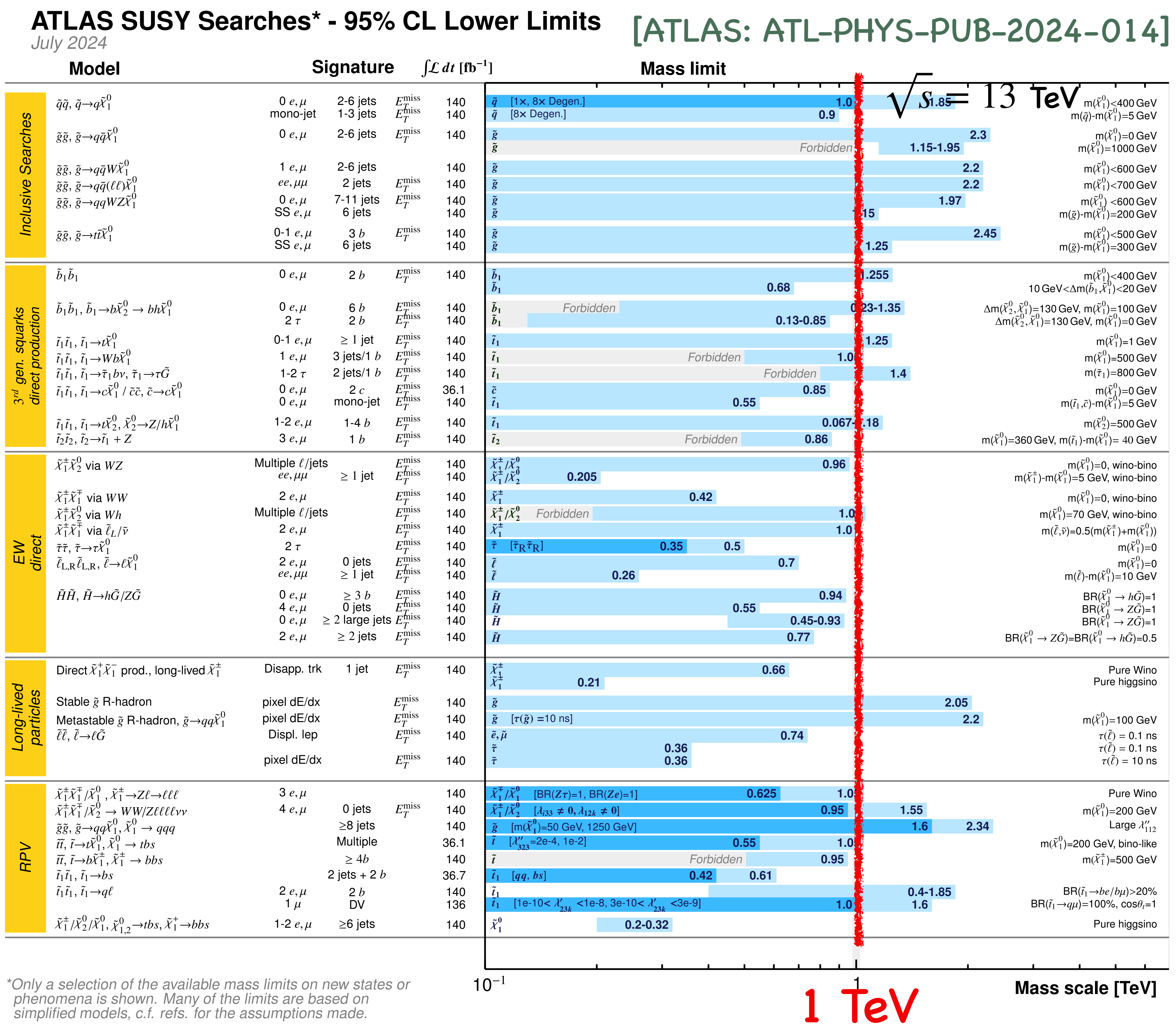
Based on: 2507.20869 with A. Choudhury and S. Sarkar

The hunt for BSM

Extensive searches for diverse BSM scenarios have been performed at the LHC.



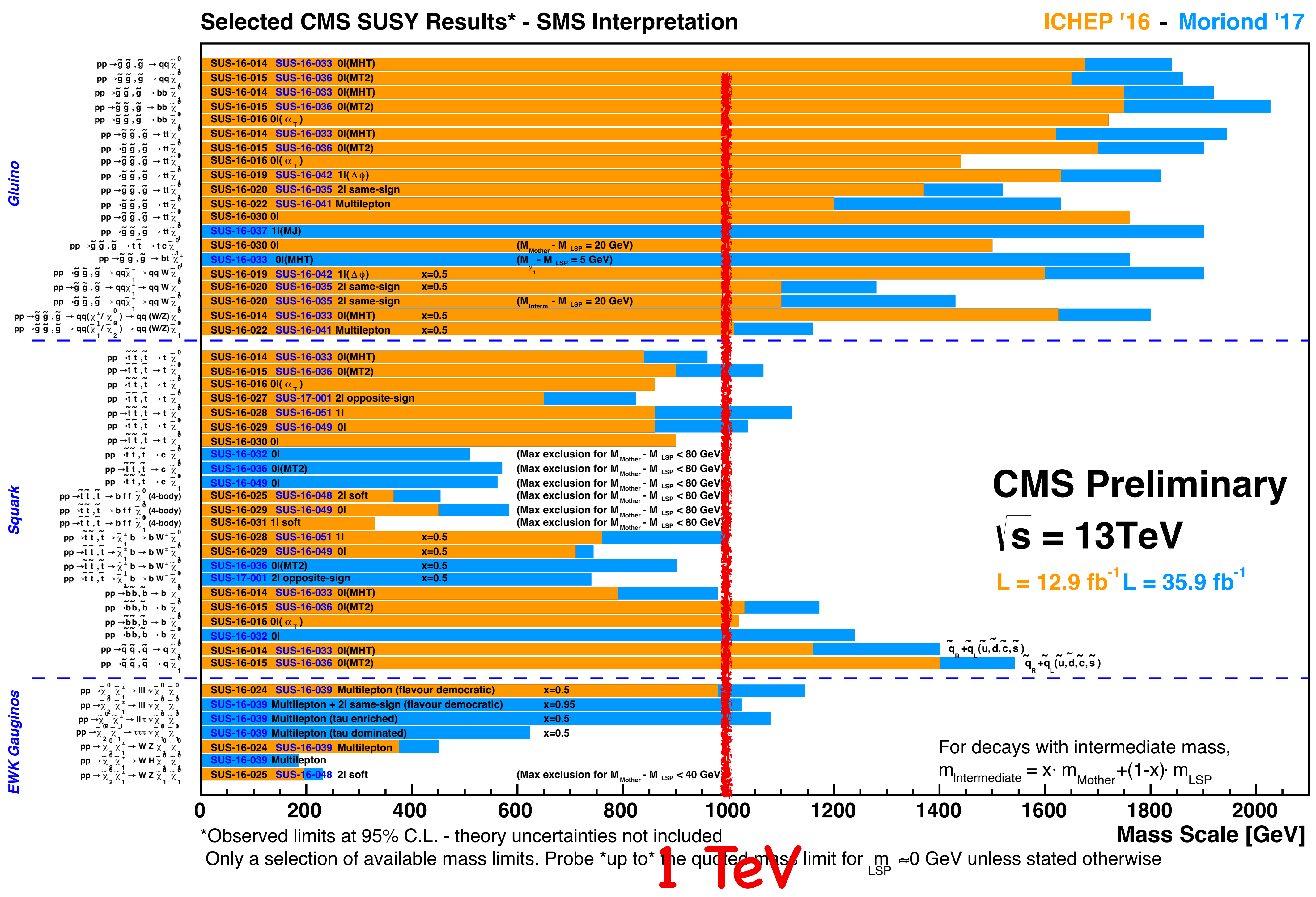
[CMS: Twiki-SUSYSummary2017]



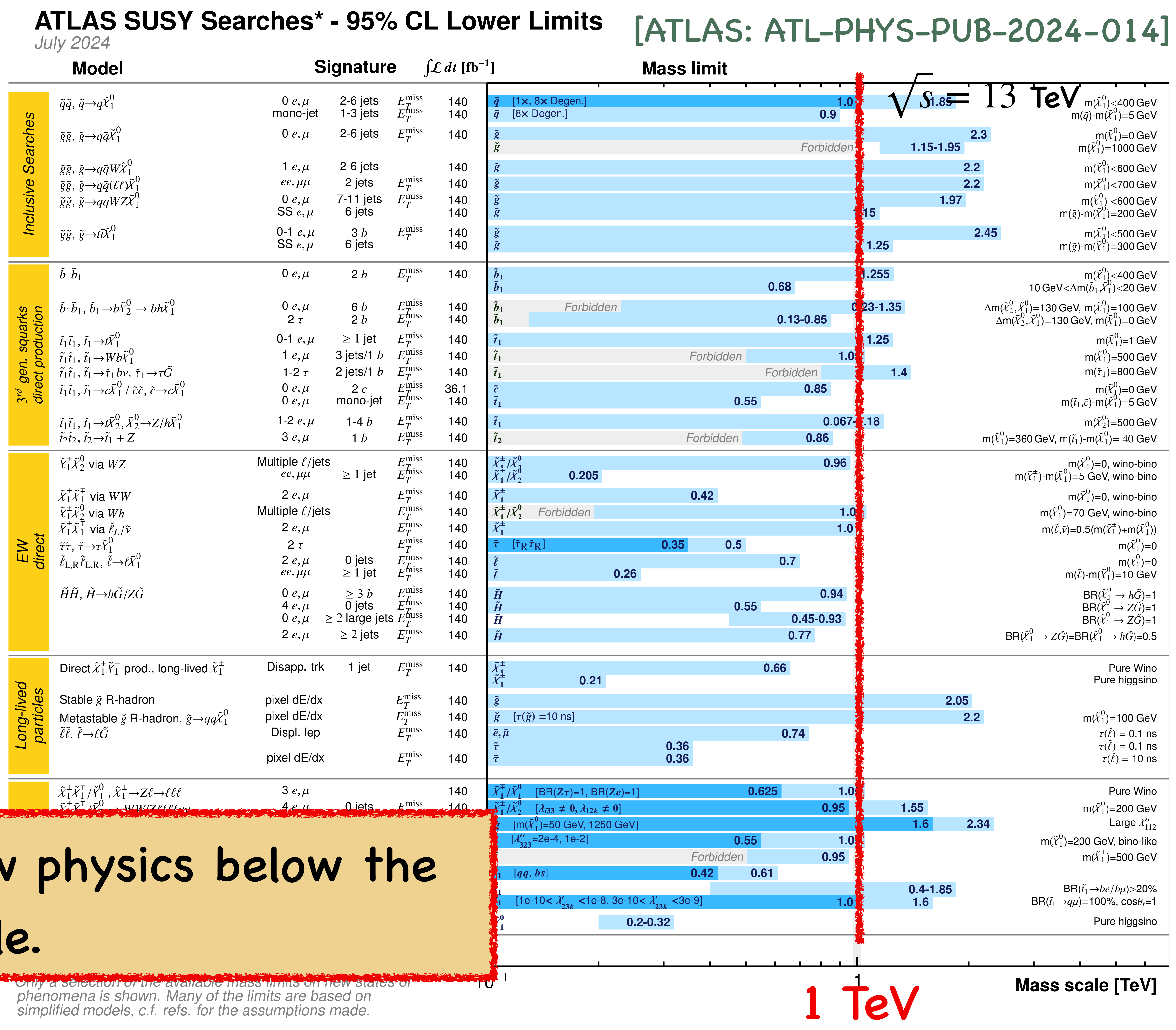
[ATLAS: ATL-PHYS-PUB-2024-014]

The hunt for BSM

Extensive searches for diverse BSM scenarios have been performed at the LHC.



[CMS: Twiki-SUSYSummary2017]



[ATLAS: ATL-PHYS-PUB-2024-014]

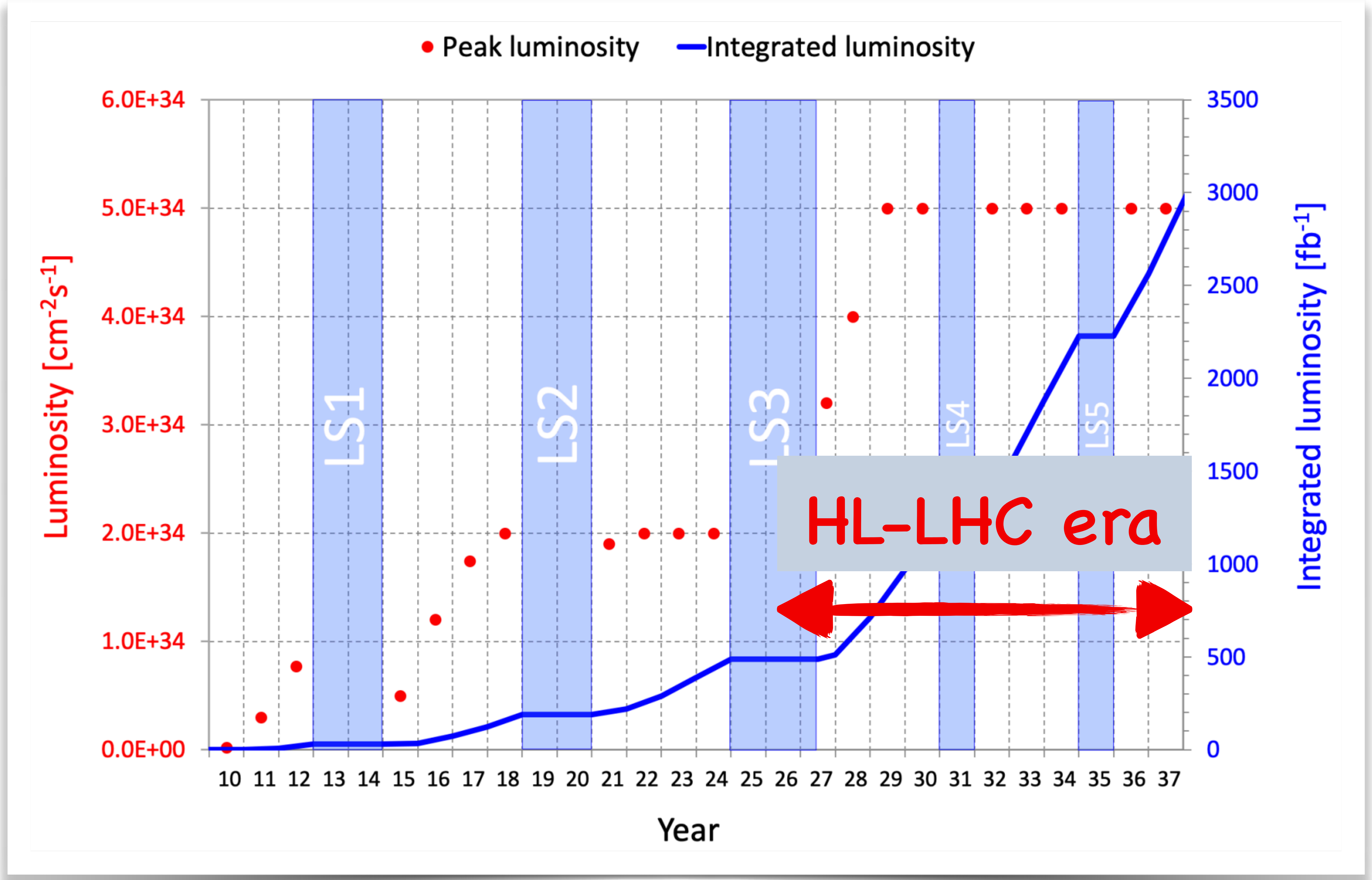
No direct evidence of new physics below the TeV scale.

The hunt for BSM

- New Physics states could be 'heavy' → beyond the 'direct' reach of the LHC.
- Weakly coupled and rare signals → Some signatures could be so rare and weakly interacting that current datasets lack the sensitivity due to low statistics.

- Higher statistics at the upcoming High-Luminosity upgrade of the LHC (HL-LHC):

  → HL-LHC will deliver a 10x more data than the current LHC.



[CMS Collaboration: J. of Phys., Conf. Series 1498 (2020) 012054]

- This statistical boost would allow us to:

  → probe subtle deviations from SM with more accuracy

  → study currently 'rare' processes

  → possibility to perform differential measurements.

Need for ML !

Challenges:

- More data brings more background noise
- High-dimensional data and complex correlations
- Testing a large number of BSM scenarios

Need of the hour:

- Efficient signal-background discrimination
- High-dimensional 'unfolding' models

  [J. Ackerschott, **RKB**, D. Goncalves, T. Heimel, T. Plehn, 2308.00027]

  [**RKB**, Arghya Choudhury, Subhadeep Sarkar, 2507.20869]
- Fast likelihood-inference methods

  [**RKB**, Dorival Goncalves, Felix Kling, 2110.07635]

  [**RKB**, Ahmed Ismail, 2205.07912] [**RKB**, Sumit Biswas, 2407.00183]

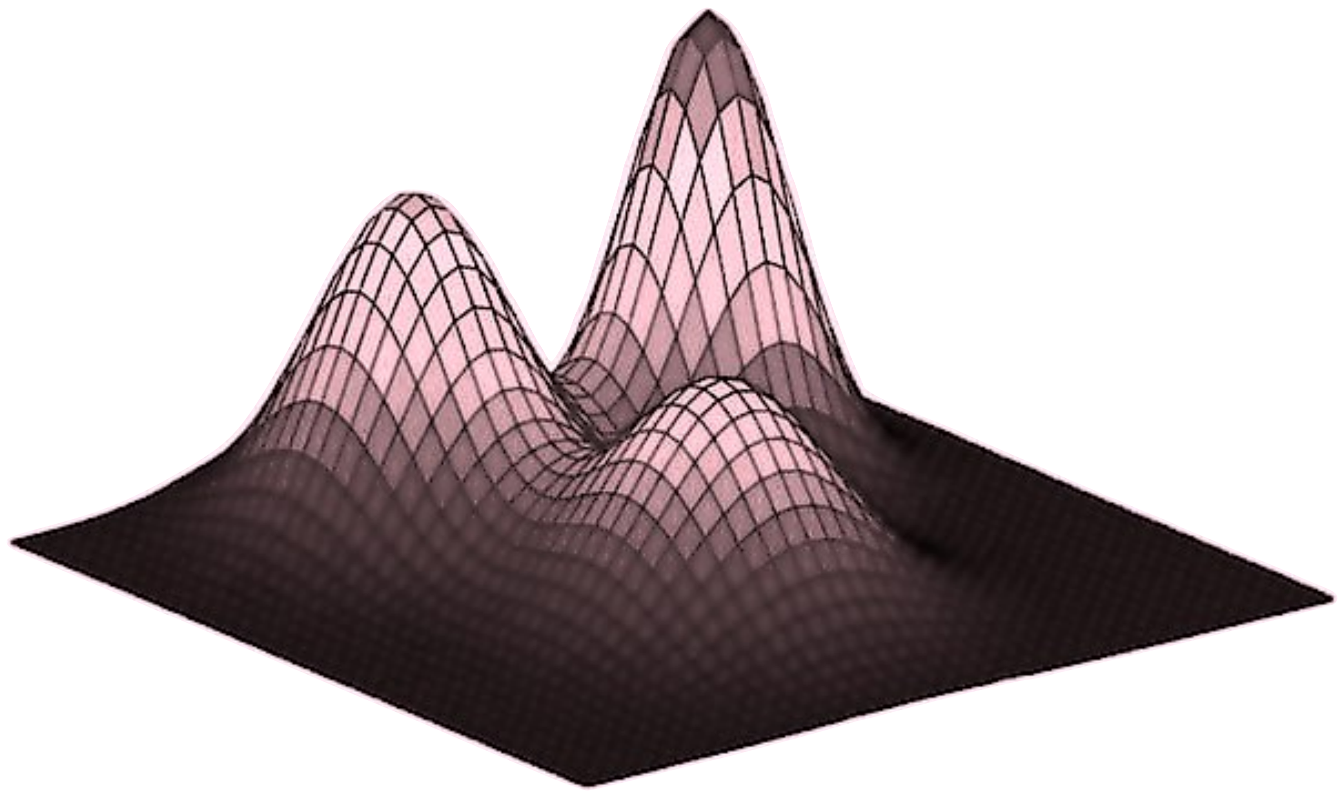
Generative ML has emerged as as indispensable tool to address these challenges: many of which are not feasible with classical techniques alone!

A comprehensive list of references can be found at HEPML-LivingReview

[https://iml-wg.github.io/HEPML-LivingReview/]

Generative Models

Goal: Estimate the probability distribution of a given data



(Unknown) data distribution

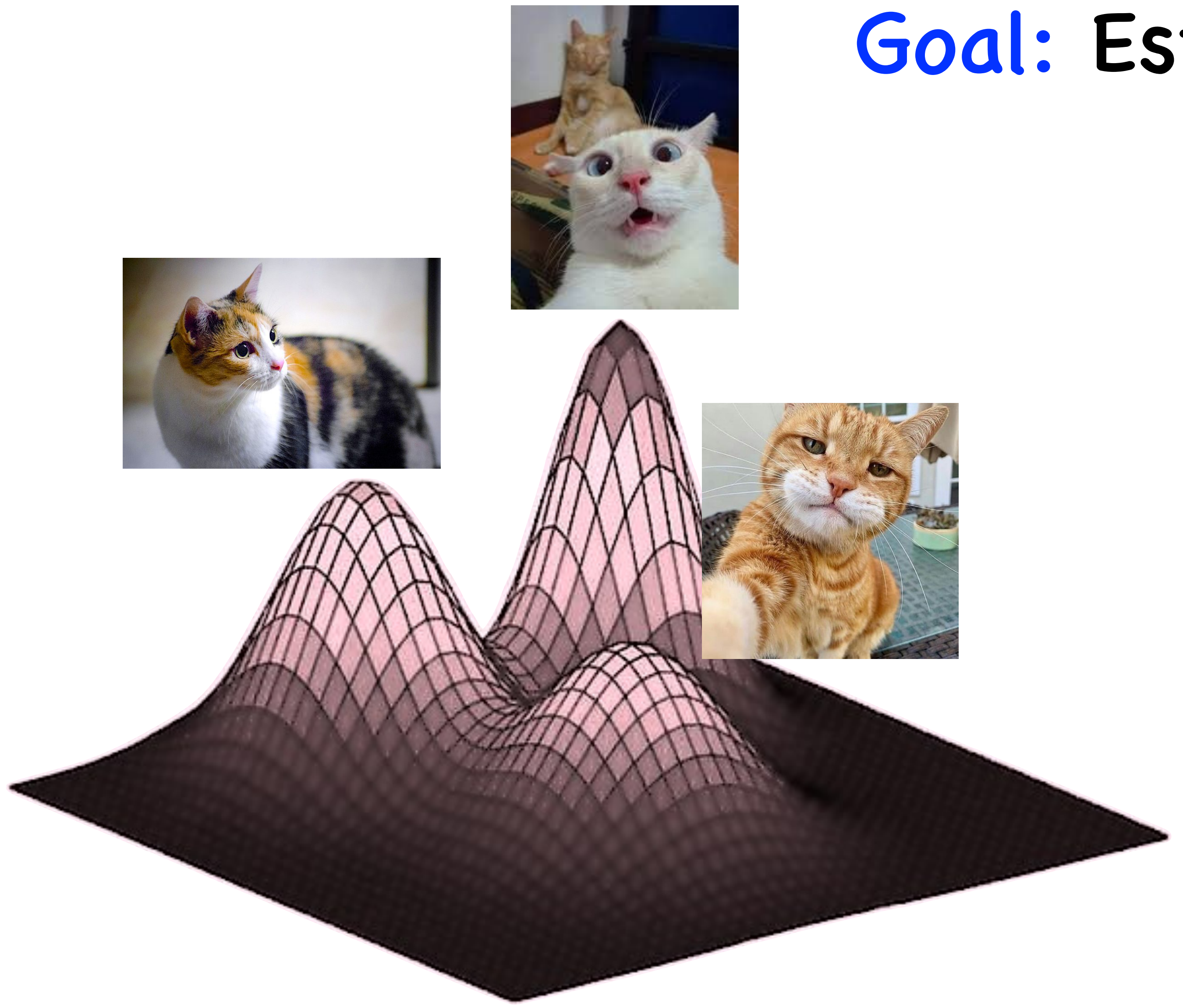
Generative Models

Goal: Estimate the probability distribution of a given data

(Unknown) data distribution

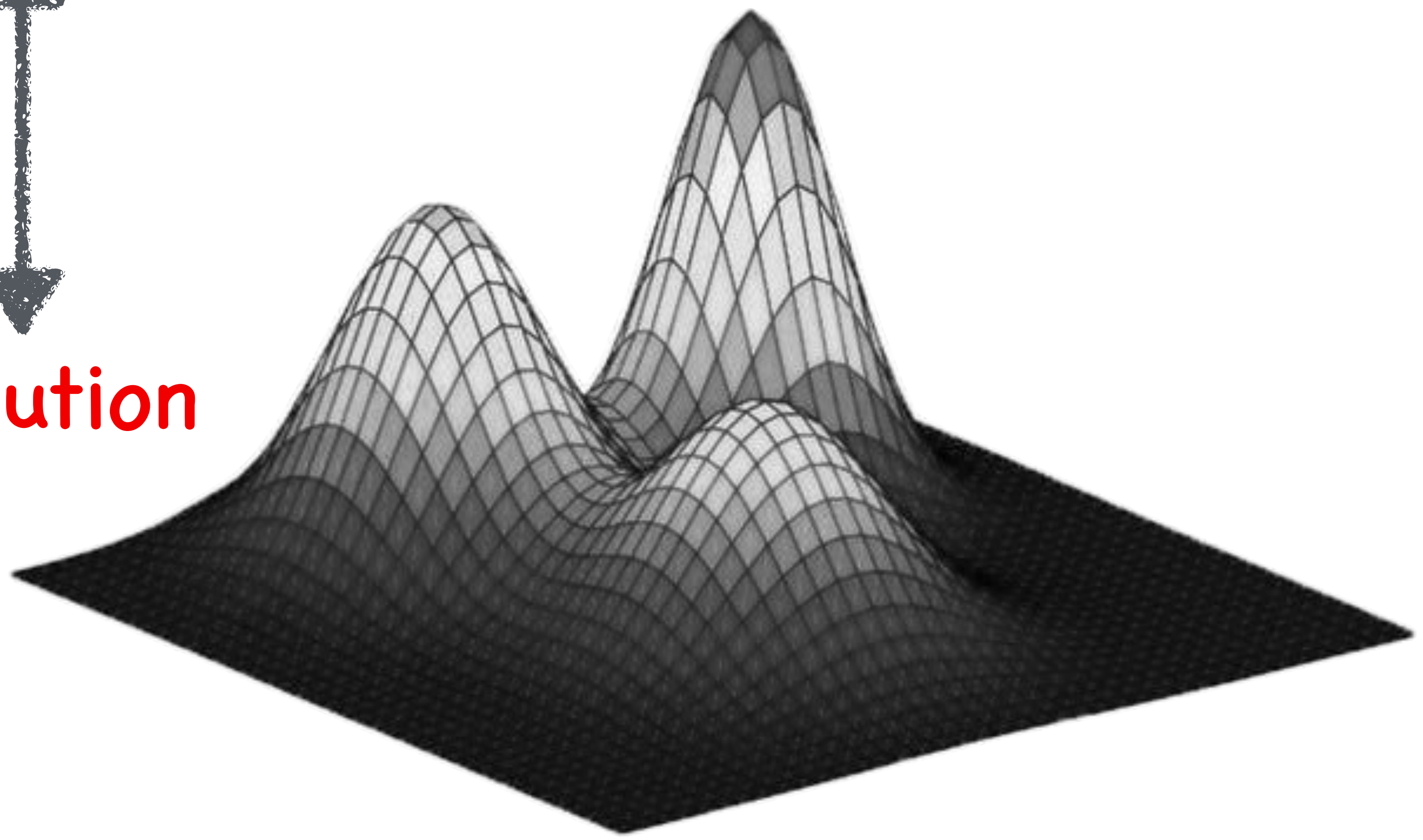
Generative Models

Goal: Estimate the probability distribution of a given data



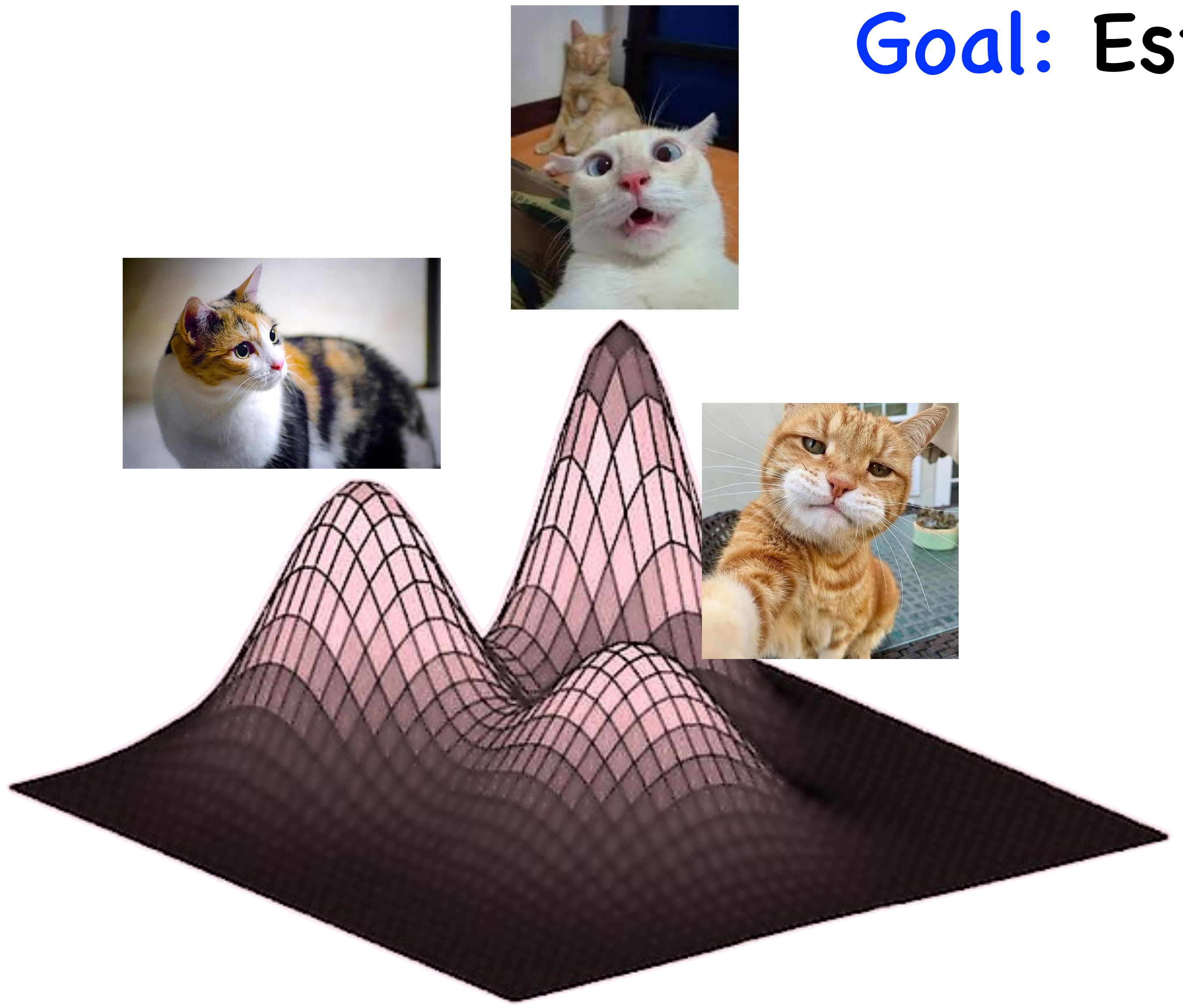
(Unknown) data distribution

Model distribution



Generative Models

Goal: Estimate the probability distribution of a given data

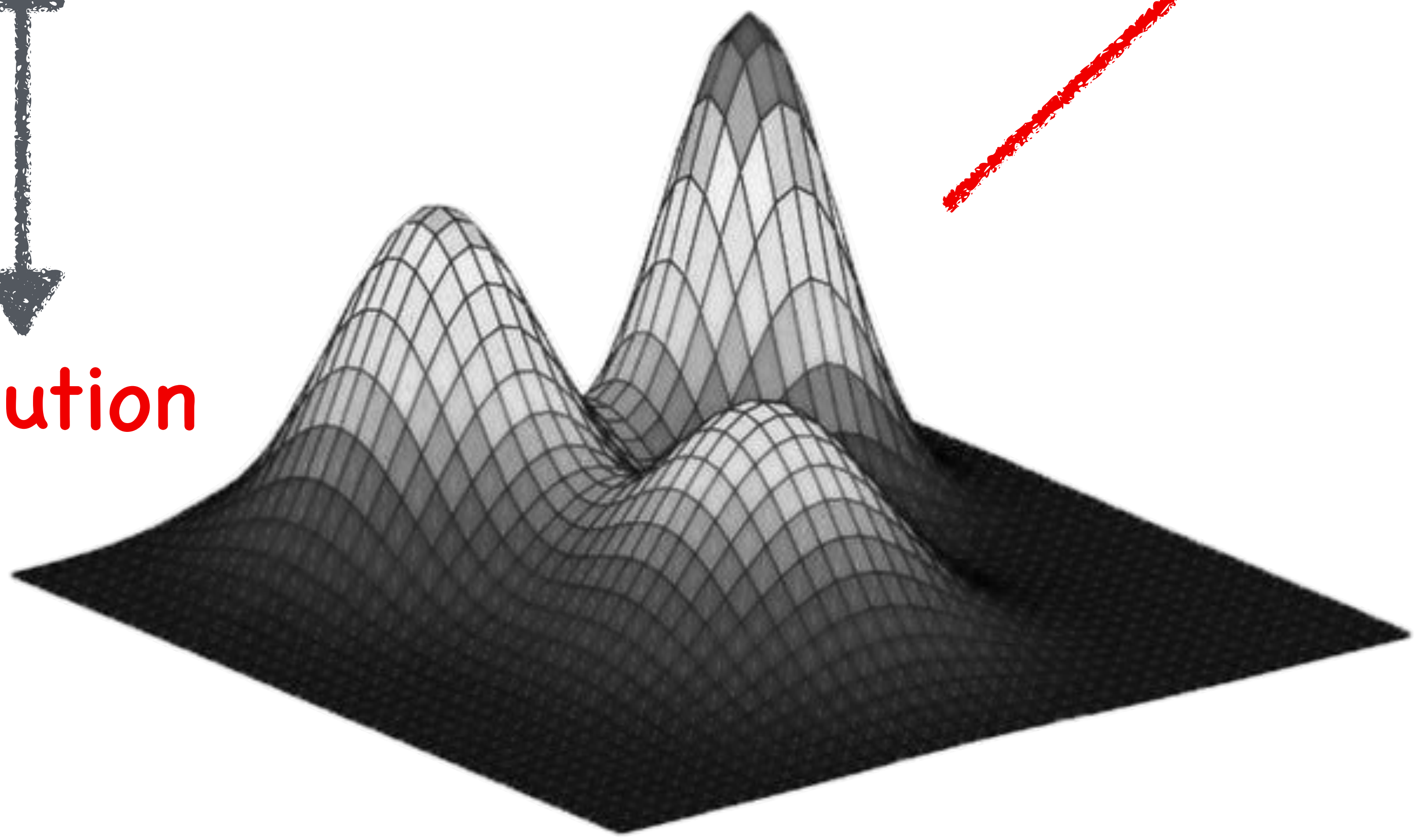


- Sampling: New data can be generated by sampling from the learned model distribution.



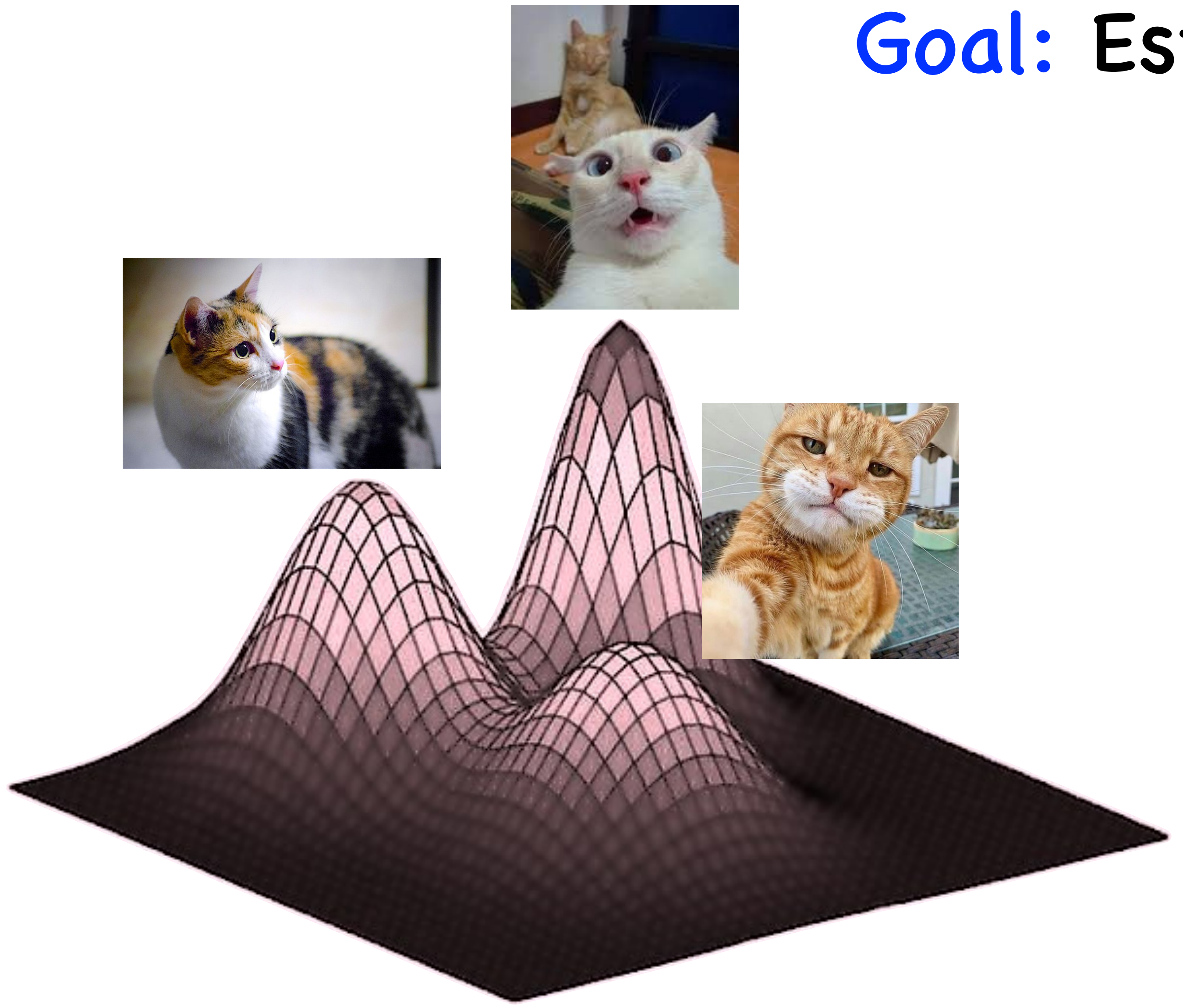
(Unknown) data distribution

Model distribution



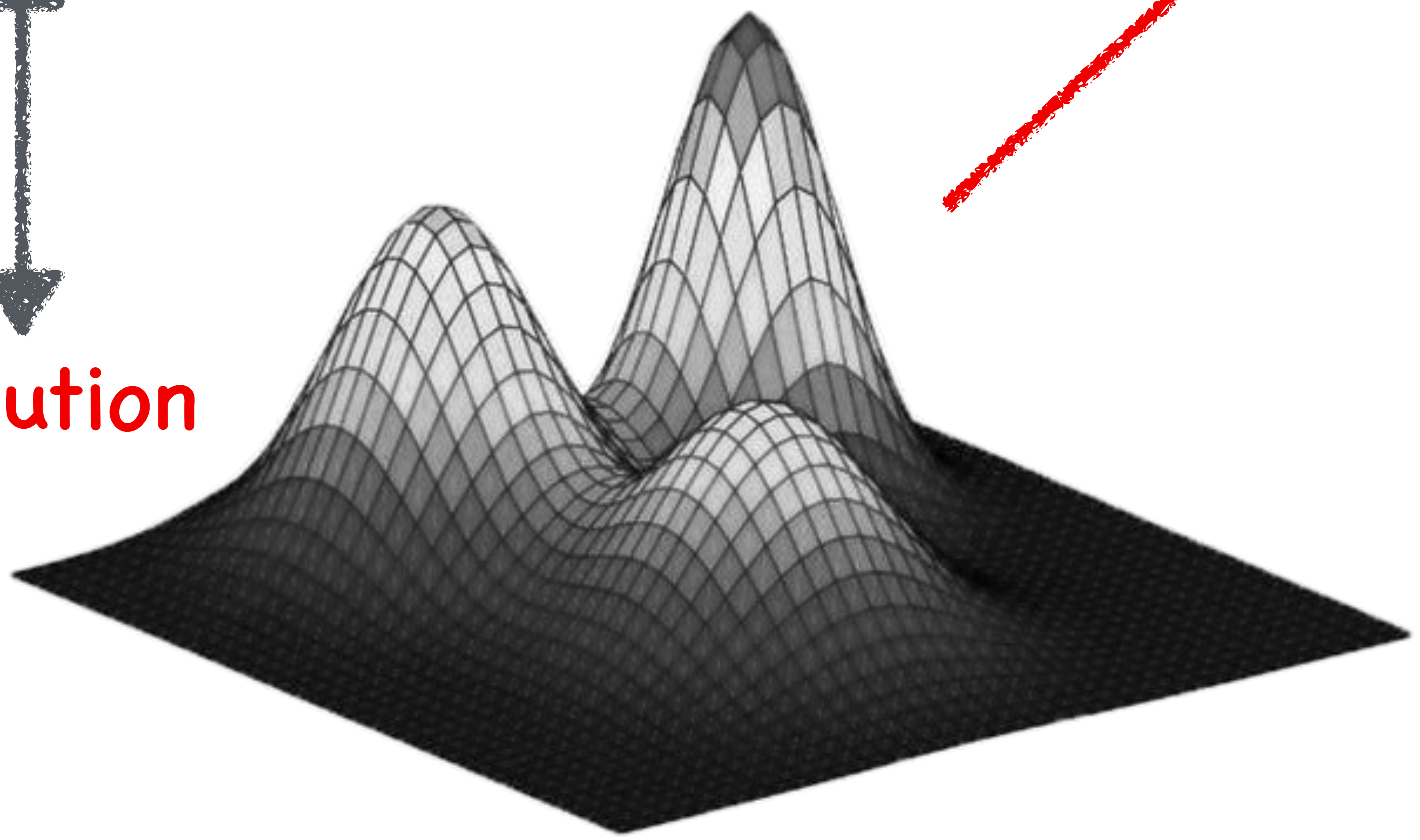
Generative Models

Goal: Estimate the probability distribution of a given data



(Unknown) data distribution

Model distribution

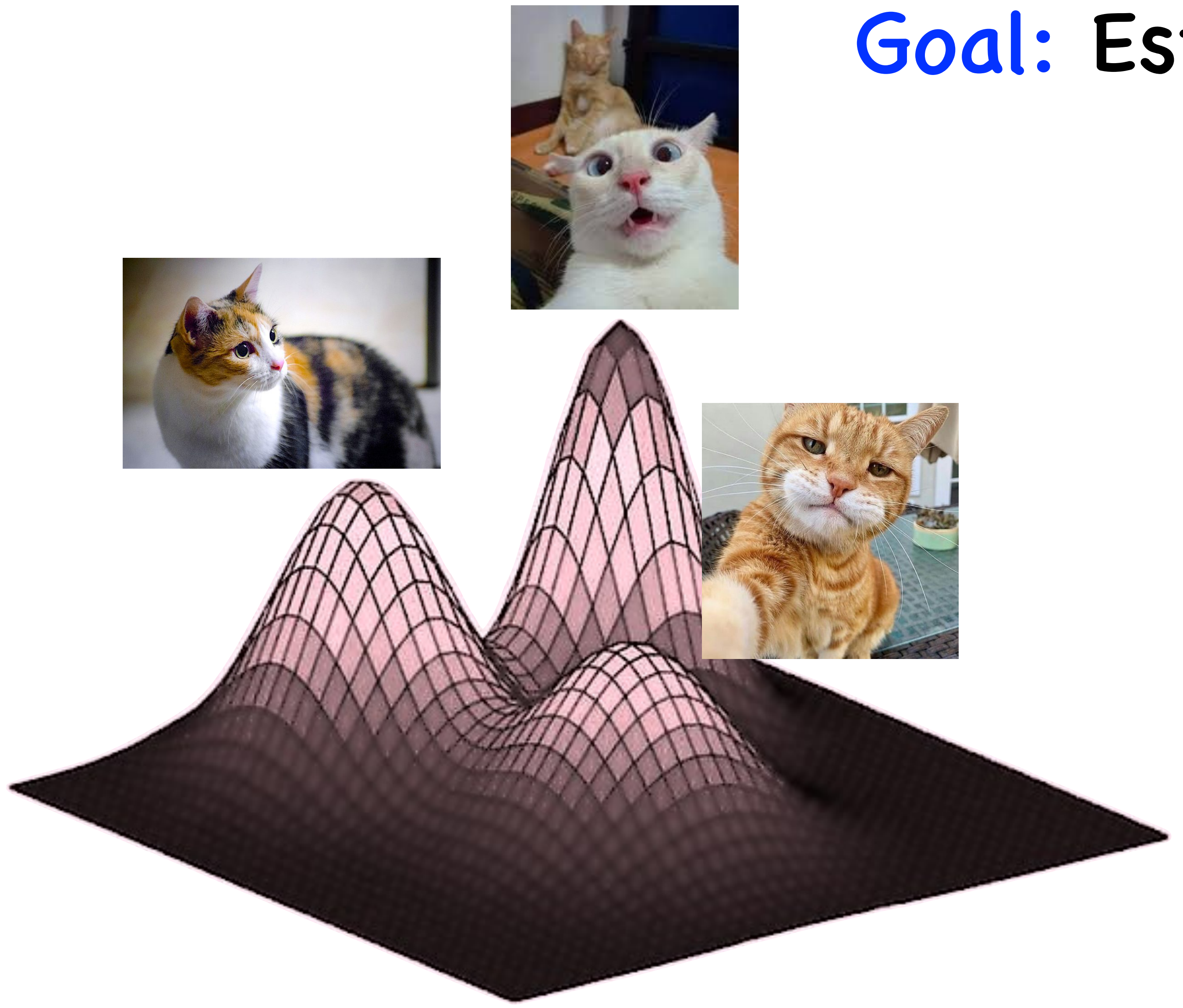


- Sampling: New data can be generated by sampling from the learned model distribution.

"Generative"

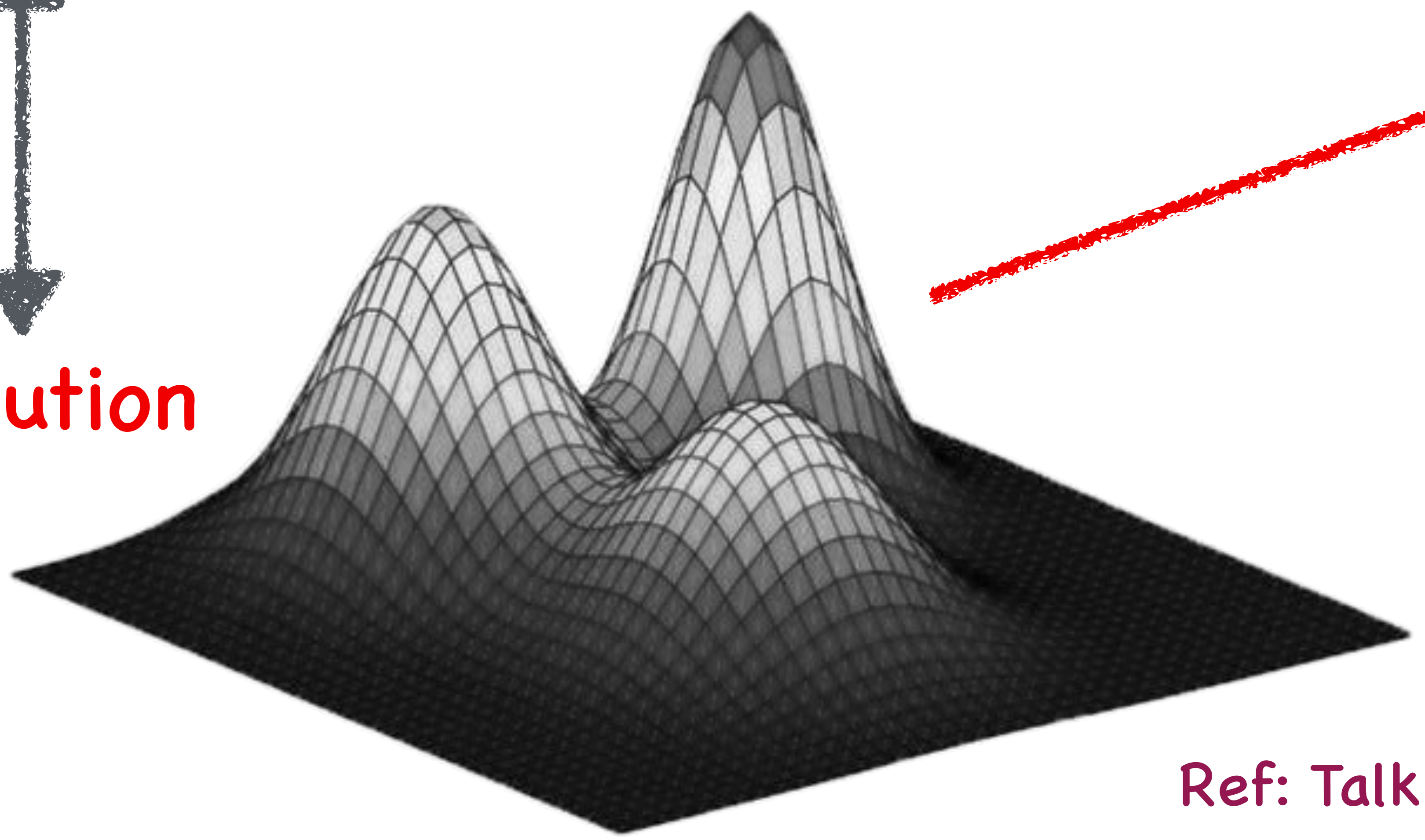
Generative Models

Goal: Estimate the probability distribution of a given data



(Unknown) data distribution

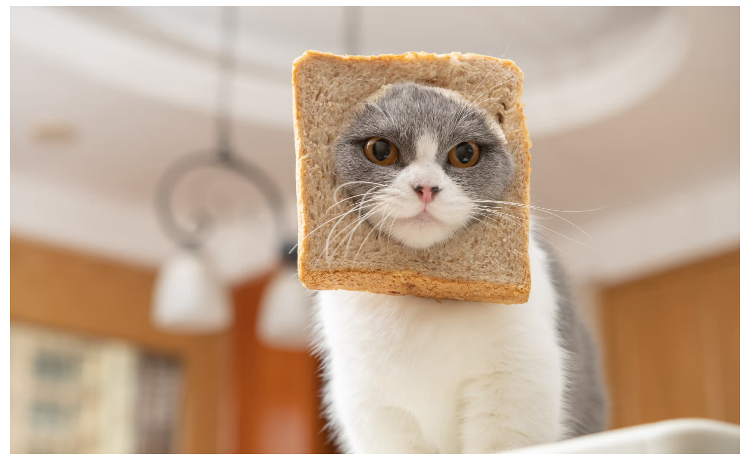
Model distribution



- **Sampling:** New data can be generated by sampling from the learned model distribution.

"Generative"

- **Probability estimation:** Probabilities for any given data point can be evaluated.



\`Cat-like' probability:

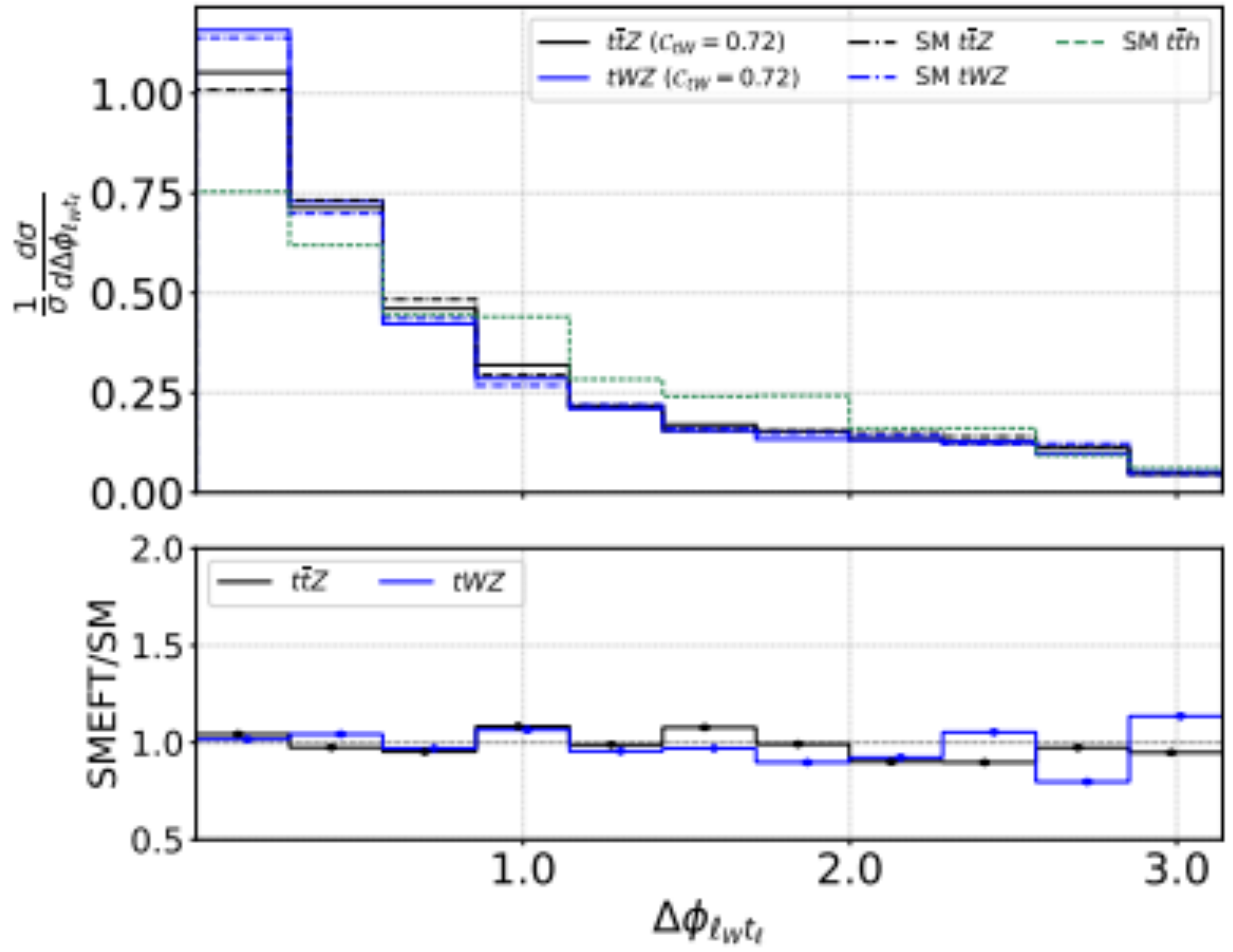
High

Low!

Ref: Talk by Y. Song (MITCBMM)

Generative Modeling → Collider Applications

- Similarly, in collider studies, generative modeling can find applications in :



✓ Event simulation: One can learn the data distribution $p(y)$ → generate new detector-level events cheaply.

✓ Mass reconstruction: Learn the mapping from detector-level data to masses $p(mass \mid y)$

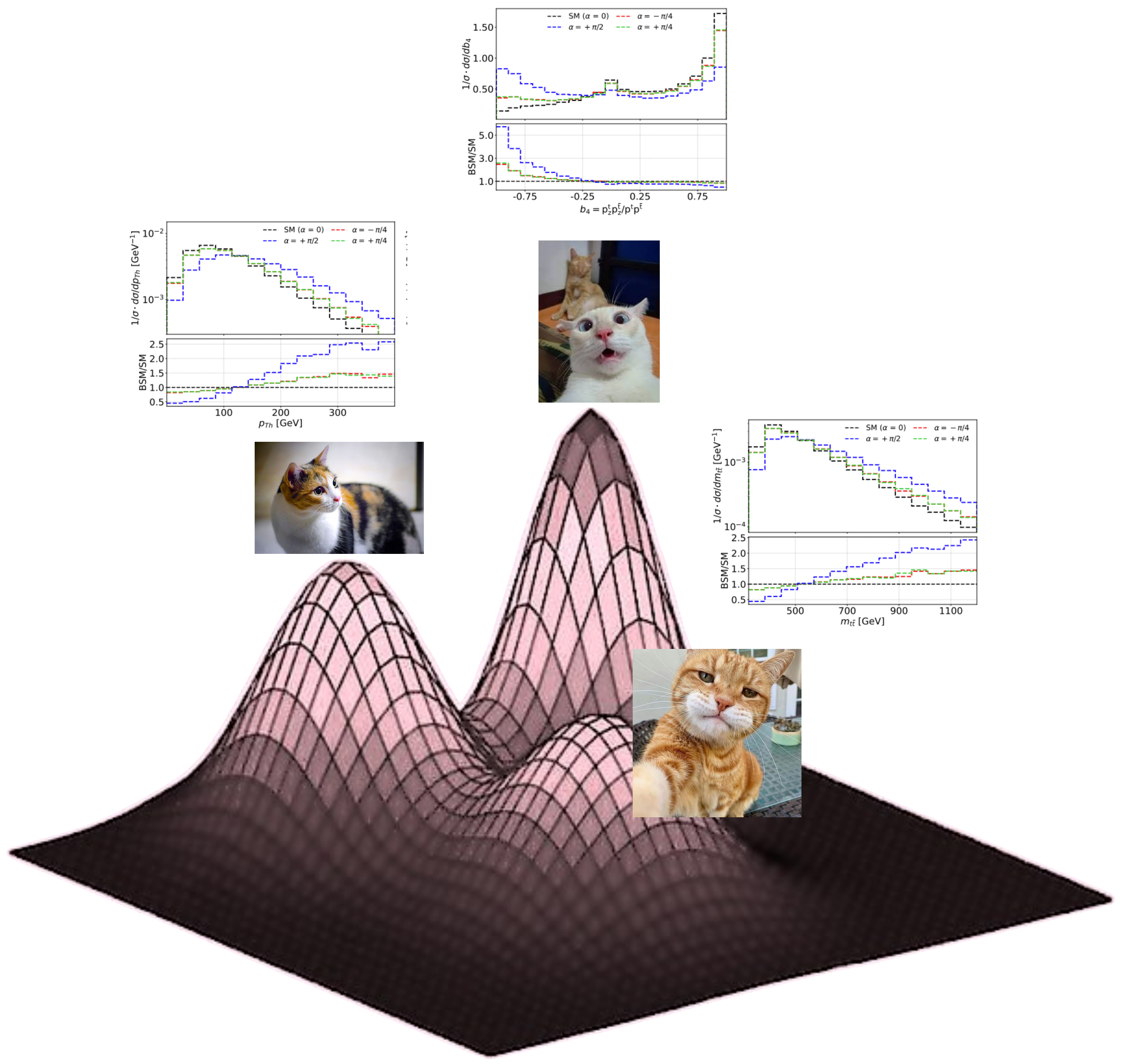
✓ Unfolding to parton-level: Reconstruct the full parton-level phase space $p(x_{parton} \mid y)$

Training a generative model

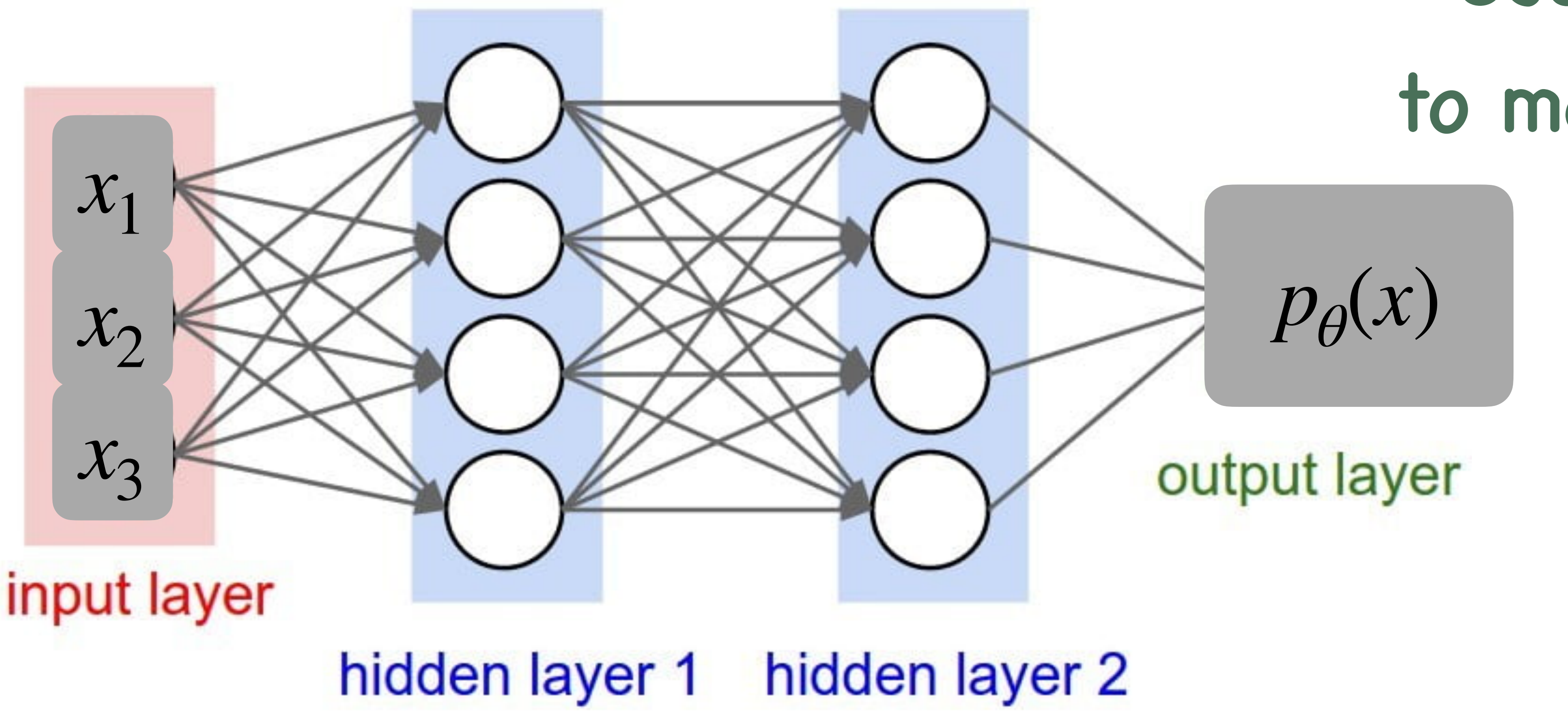
- Challenges: The data distribution can be be very complex and high dimensional.

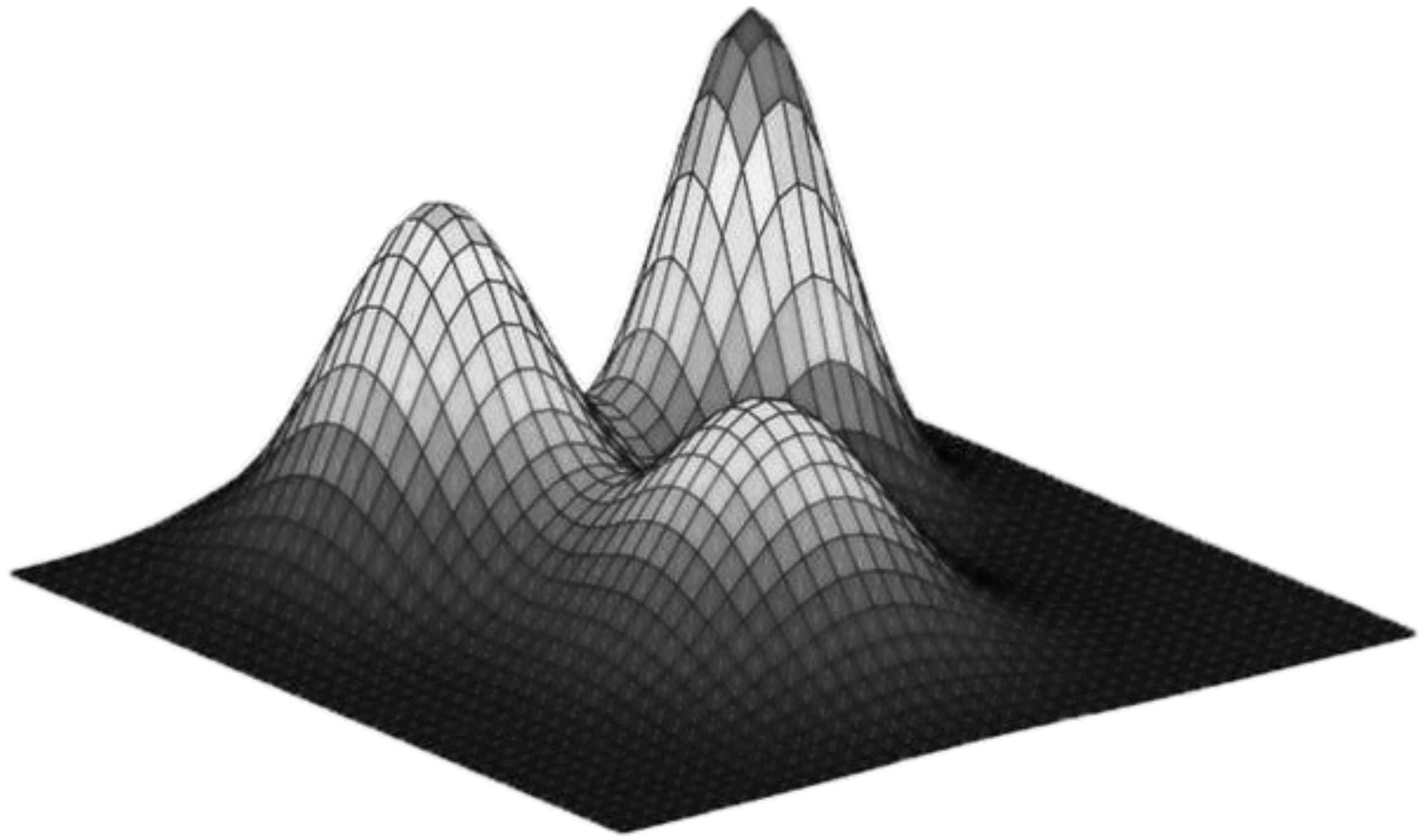
Need to construct a robust model to fit the complex high-dimensional data.

→ Use a Deep Neural Network (DNN) to model the complex distribution.



(Unknown) data distribution $p_{data}(x)$



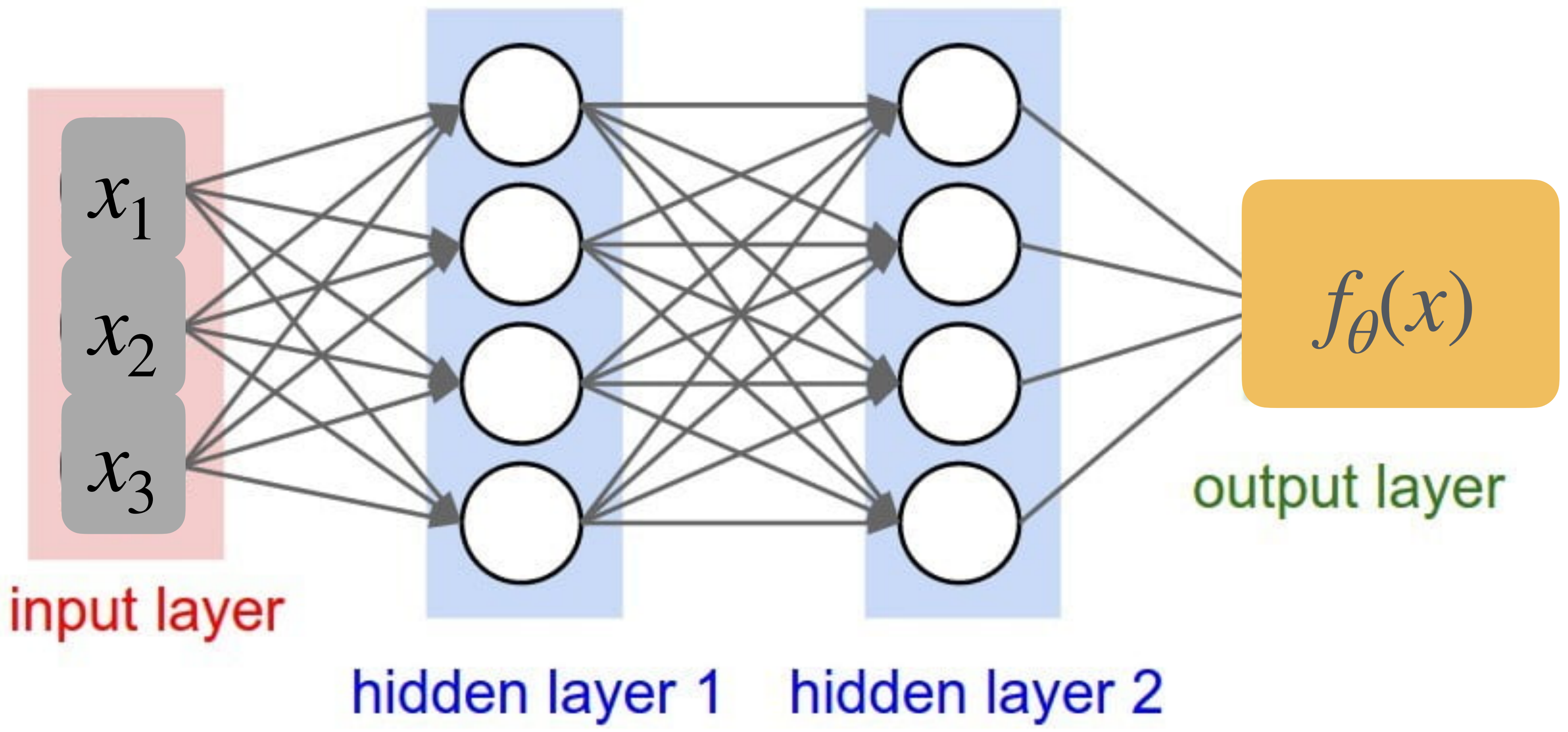


Learned distribution

Ref: Talk by Y. Song (MITCBMM)

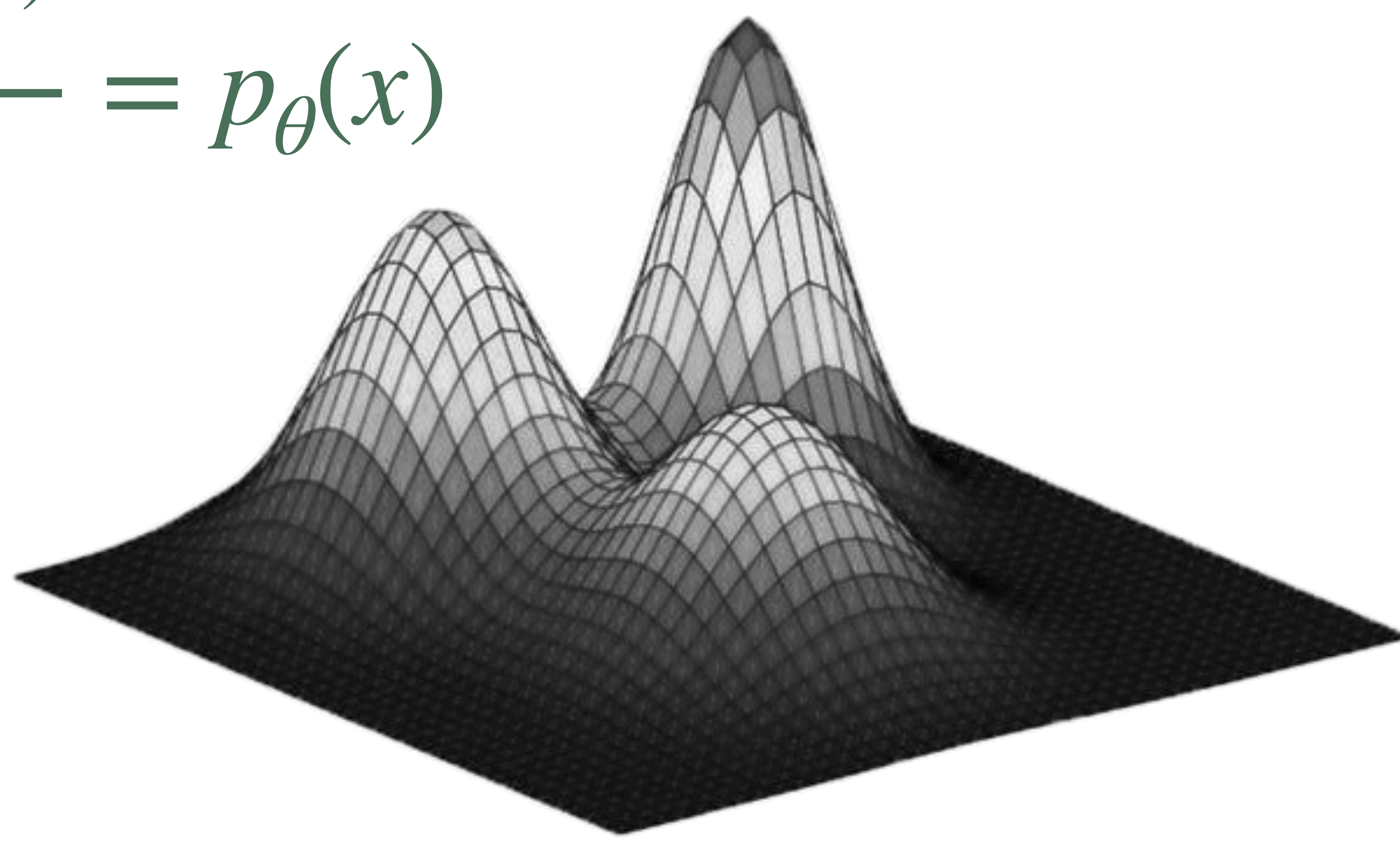
Training a generative model

- Non-trivial task since outputs from DNNs are often single valued $f_\theta(x)$



Exponential → $e^{f_\theta(x)}$ Normalize → $\dfrac{e^{f_\theta(x)}}{Z_\theta} = p_\theta(x)$

$$Z_\theta = \int e^{f_\theta(x)} dx$$



Learned distribution

- This normalizing constant can be computed by evaluating the x_n dimensional integral.

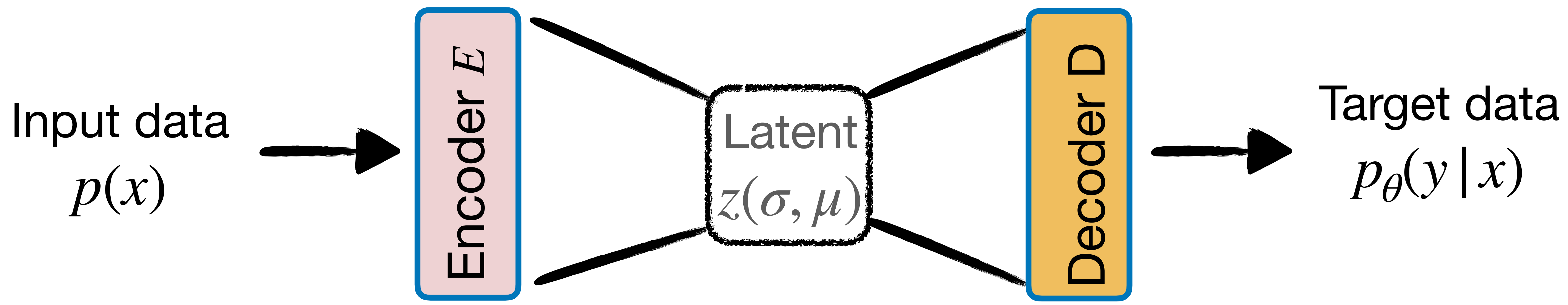
Intractable for high-dimensional data!

Ref: Talk by Y. Song (MITCBMM)

Training a generative model

The intractability can be tackled using generative ML techniques:

- Variational Autoencoders (VAEs): Kingma, Welling (2014)

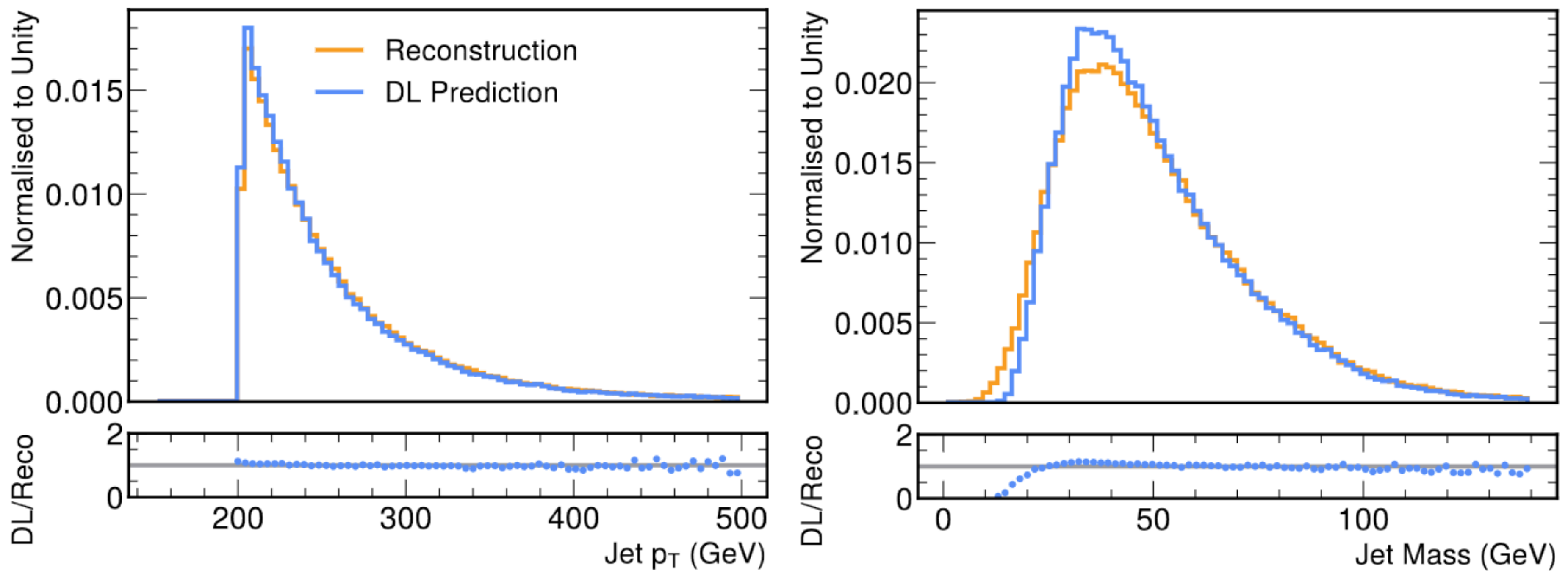


- No need to compute the normalizing constant. Instead, target distribution can be directly estimated using normalized latent distributions

$$p_\theta(y|x) = \int p(z)\, p_\theta(y|z,x)\, dz$$

➡ Explored as alternate to fast detector simulators like Delphes

➡ Good agreement between global jet variables but substructure features not well reproduced



Process: $pp \to WW \to (W \to jj)(W \to jj)$

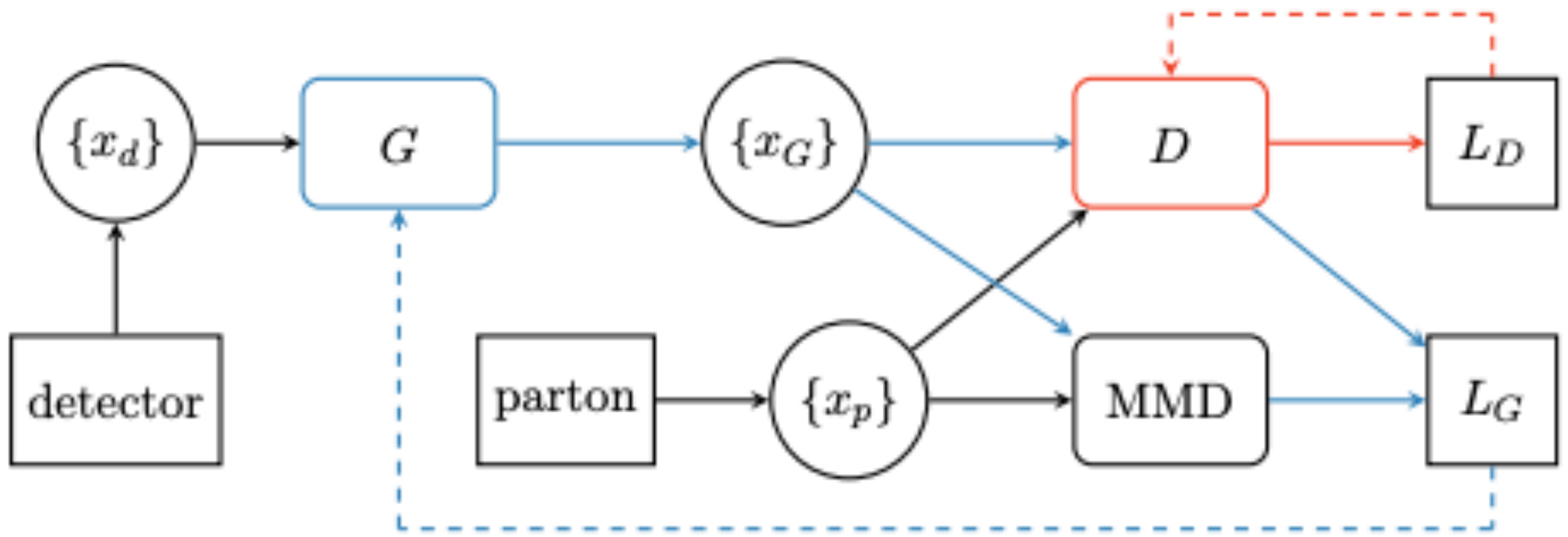
- Gaussian latent may not be the most expressive choice to map input data, and often, important features could be lost during encoding.

Touranakou et al. (2203.00520)

Training a generative model

The intractability can be tackled using generative ML techniques:

- Generative Adversarial Networks: Goodfellow et al. (2014)



Bellagente, Butter, Kasieczka, Plehn, Winterhalder (2019)

- They never compute a probability density $p_\theta(x\,|\,y)$.
- Instead, they learn by comparison: if the discriminator can't tell real from generated, the generator's samples match the data distribution.

➡ Explored as a detector-effect simulator

Paganini, Oliviera, Nachman (2018)

➡ Can be utilized to unfold detector-level data to the parton-level phase space

- Unfolding performance drops when test dataset is statistically different from the training dataset.

Datta, Kar, Roy (2018)

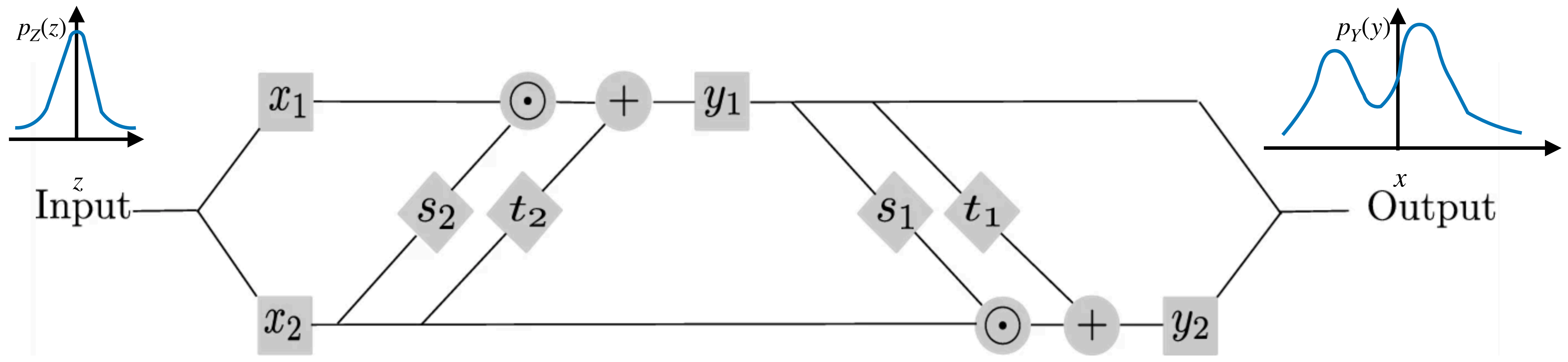
Bellagente, Butter, Kasieczka, Plehn, Winterhalder (2019)

Training a generative model

The intractability can be tackled using generative ML techniques:

Dinh et al. (2014)
Rezende, Mohamed (2015)

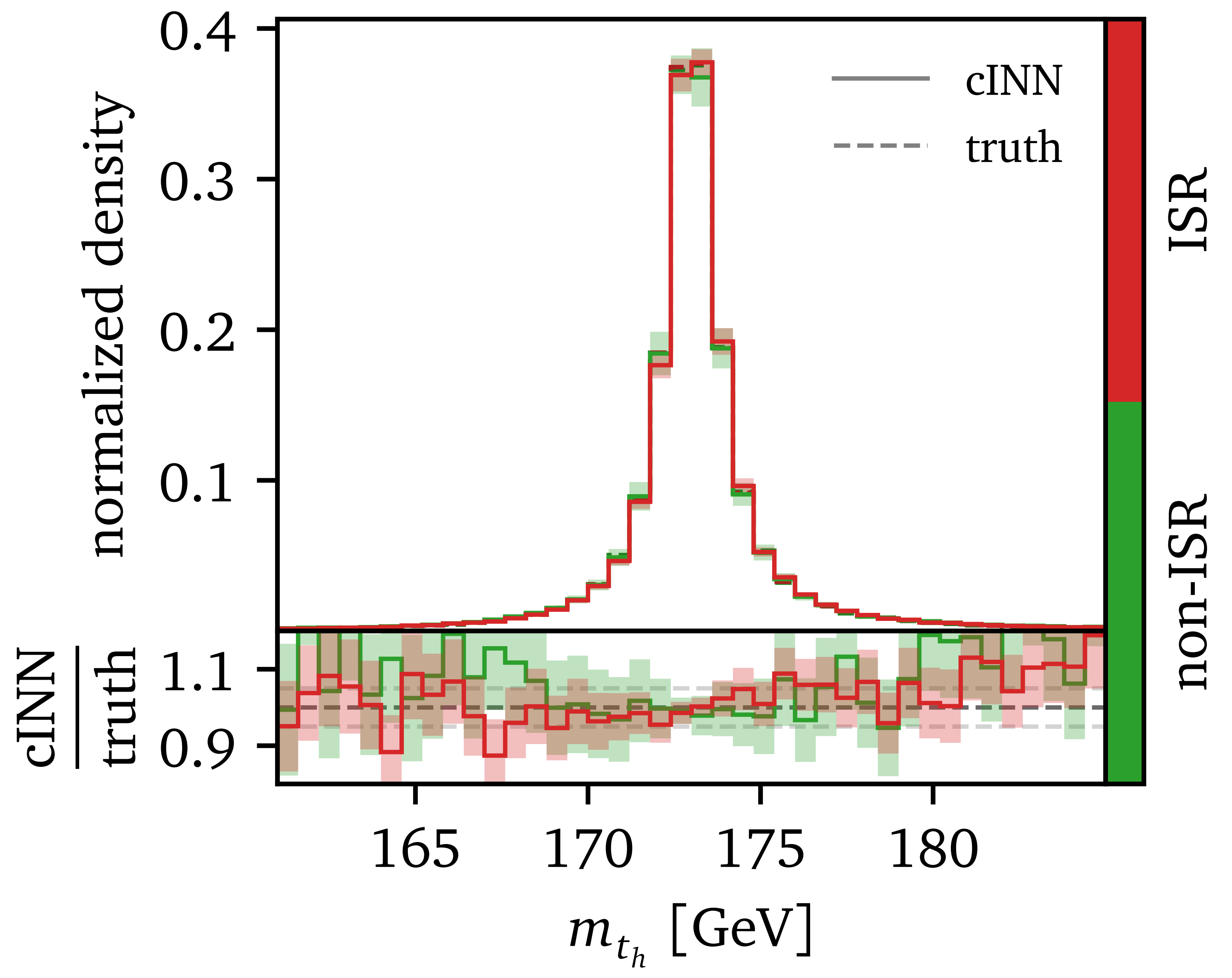
- Normalizing Flows:



L. Ardizonne, C. Luth, J. Kruse, C. Rother, U. Kothe, arxiv: 1907.02392/cs.LG

- Learns an invertible map $x = g_\theta(z; y)$ that transforms a simple base distribution $z \sim \mathcal{N}(0, I)$ into the target distribution

  → yielding exact probability maps, bijectivity and fast sampling.



➡ Enables creating excellent unfolding models that can invert inclusive detector-level to high-dimensional parton-level phase space

Bellagente, Butter, Kasieczka, Plehn, Rousselot, Winterhalder, Ardizzone, Kothe (2020)

- Choosing an ideal parametrization to represent the high-dimensional target (parton-level) is non-trivial.

- Often very data hungry models!

Jona Ackerschott, **RKB**, Dorival Goncalves, Theo Heimel, Tilman Plehn, 2308.00027

Generative ML with Diffusion Models

- Diffusion models learn the data distribution by adding and then removing noise directly on the data → doesn't squeeze data through the narrow latent space as in Variational Autoencoders (VAEs) avoiding any information loss.

- GANs which can generate high quality realistic looking samples, but they don't tell us how probable it is → not a problem with diffusion models.

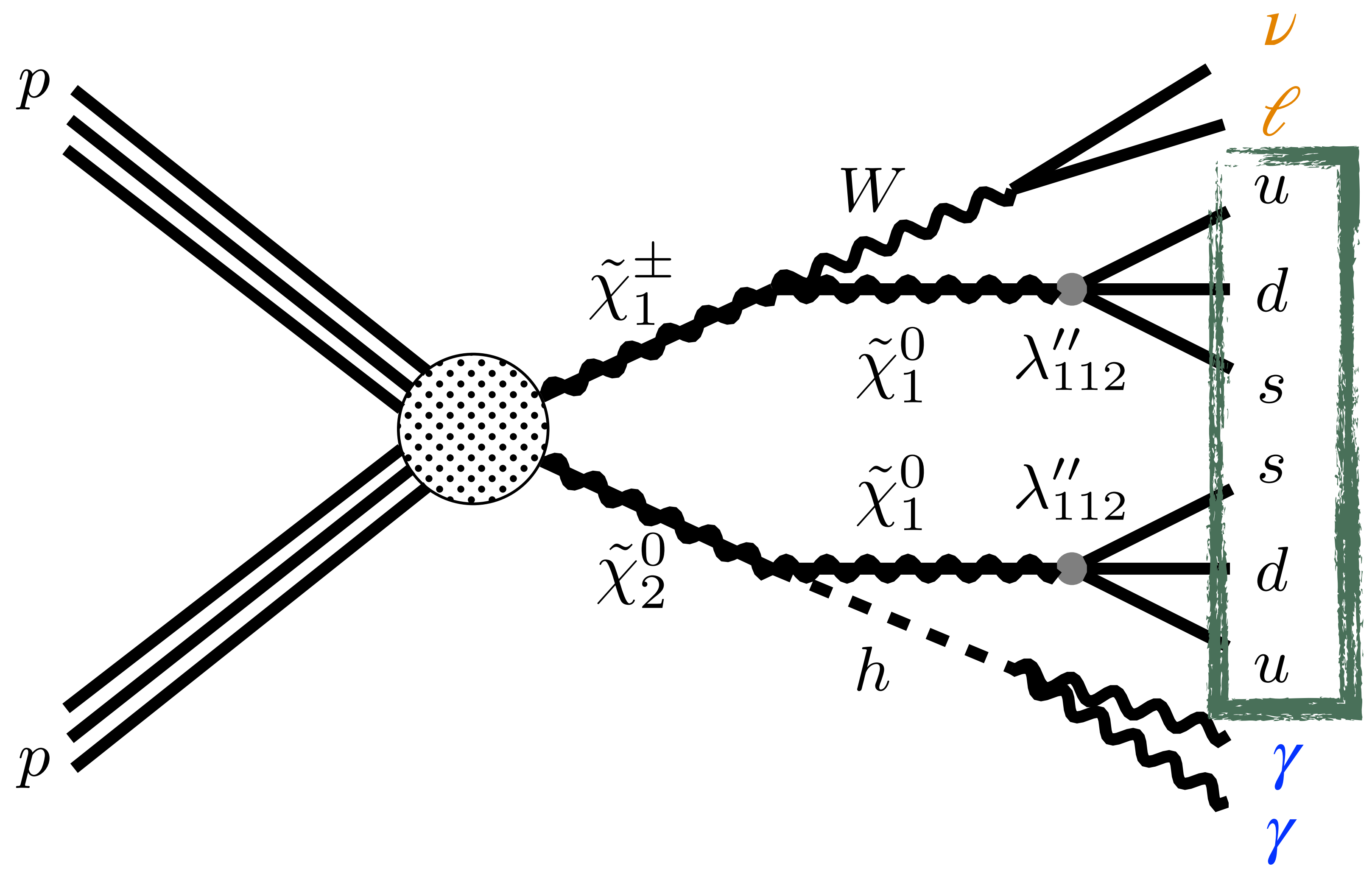
- Normalizing Flows would perform better when it comes to process-specific unfolding or mapping → but diffusion models win when you want a single model to cover a wide parameter space (EFT probes, mass reconstruction over a large parameter space, etc.)

Reconstructing masses with a hybrid diffusion model

- Goal: A single model to reconstruct masses across a wide BSM parameter space → learn $p(x \mid y_{detector})$ to reconstruct particle masses from detector-level data $y_{detector}$

Test scenario: Wino-like chargino $\tilde{\chi}_1^{\pm}$ - neutralino $\tilde{\chi}_2^0$ pair production at the HL-LHC within the R-parity violating Supersymmetry with λ''_{112} coupling

$$pp \to \tilde{\chi}_1^{\pm}\tilde{\chi}_2^0 \to (\tilde{\chi}_1^{\pm} \to W^{\pm}\tilde{\chi}_1^0)\,(\tilde{\chi}_2^0 \to h\tilde{\chi}_1^0) \to (W^{\pm} \to \ell'\nu)\,(h \to \gamma\gamma) + \tilde{\chi}_1^0\,\tilde{\chi}_1^0$$

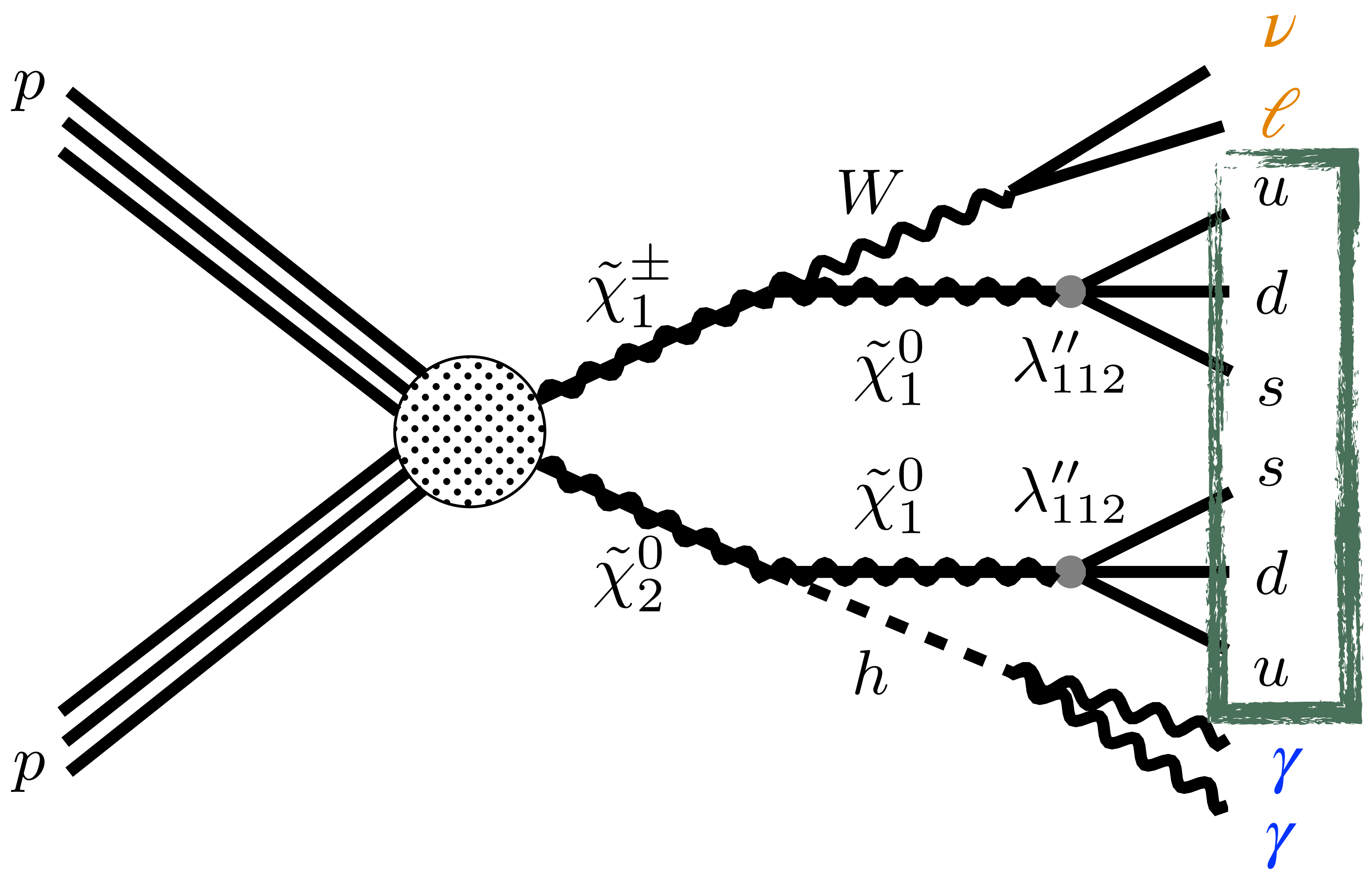


- Mass degenerate $\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$
- $\tilde{\chi}_1^{\pm}$ decays into $W^{\pm} + \tilde{\chi}_1^0$, while $\tilde{\chi}_2^0$ decays into $h + \tilde{\chi}_1^0$
- $\tilde{\chi}_1^0 \to u\,d\,s$ via R-parity violating λ''_{112} coupling

$\boxed{\text{Detector-level final state: } \ell + \gamma\gamma + jets + \not{E}_T}$

Reconstructing masses with a hybrid diffusion model

$$pp \to \tilde{\chi}_1^{\pm}\tilde{\chi}_2^0 \to (\tilde{\chi}_1^{\pm} \to W^{\pm}\tilde{\chi}_1^0)\,(\tilde{\chi}_2^0 \to h\tilde{\chi}_1^0) \to \underline{(W^{\pm} \to \ell'\nu)\,(h \to \gamma\gamma) + \tilde{\chi}_1^0\,\tilde{\chi}_1^0}$$

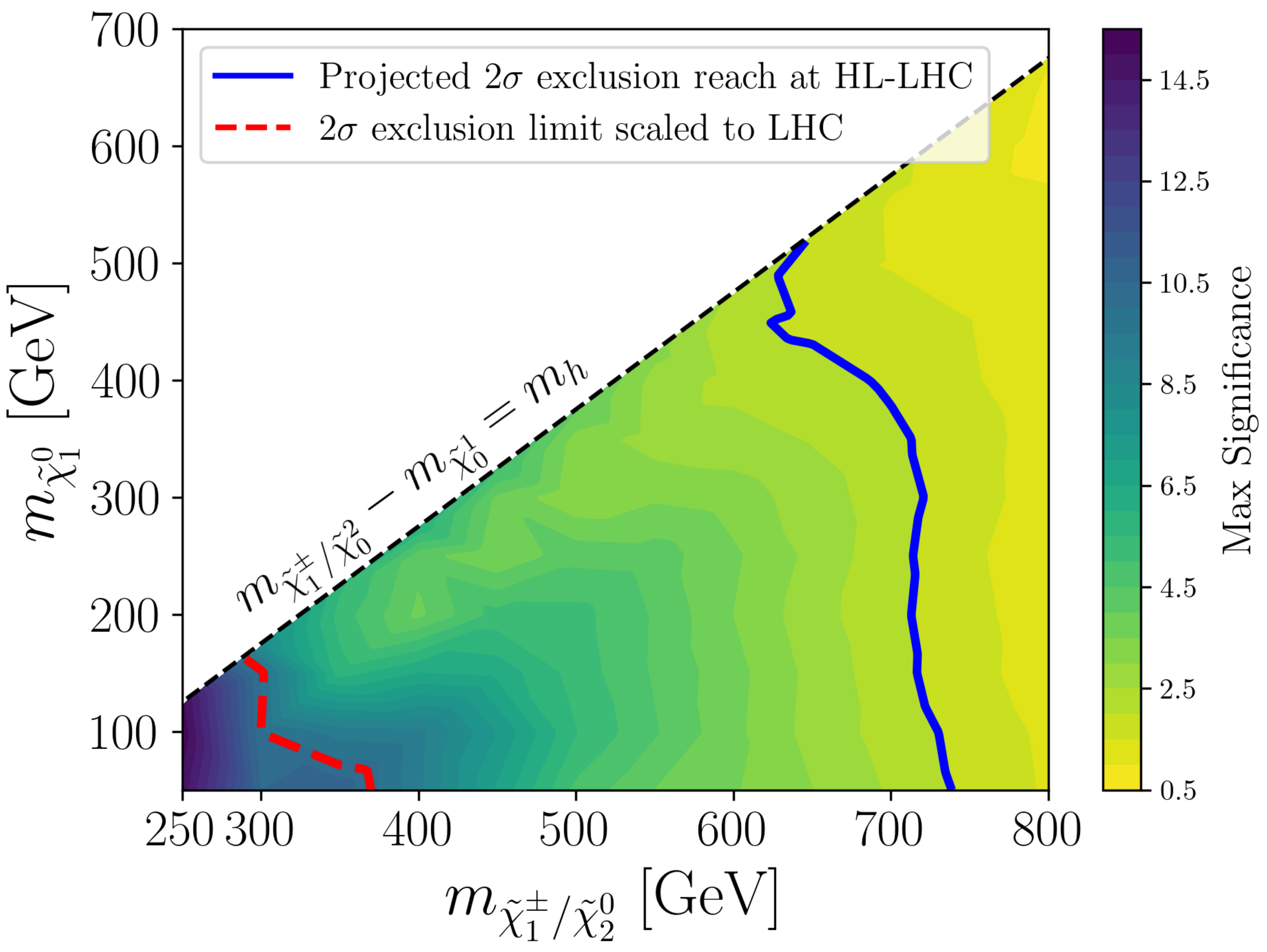


Detector-level final state: $\ell + \gamma\gamma + jets + \not{E}_T$

- Reconstruct the masses of $\tilde{\chi}_2^0/\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_1^0$ from detector-level data over the entire parameter space available at the HL-LHC.

$$m_{\tilde{\chi}_2^0/\tilde{\chi}_1^{\pm}} \in [450, 600]\ \mathrm{GeV}$$

$$m_{\tilde{\chi}_1^0} \in [\,50\ \mathrm{GeV}, \mathrm{m}_{\tilde{\chi}_2^0/\tilde{\chi}_1^{\pm}} - \mathrm{m_h}\,]$$



- Infeasible with conventional mass minimization techniques.

Model overview

A hybrid model with two components

Detector-level data

Transformer-based detector-encoder

- Maps the detector-level data into a permutation-invariant context vector C

Model overview

A hybrid model with two components

Target distributions (Reconstructed masses or parton-level features)

Detector-level data

Diffusion reconstructor

Transformer-based detector-encoder

- Learns the mapping between Gaussian noise and target distributions conditioned on C

- Maps the detector-level data into a permutation-invariant context vector C

Transformer detector-encoder

Detector-level final state: $\ell + \gamma\gamma + jets + \not{E}_T$

Latent representation

$\ell\ (p_x, p_y, p_z, E)$ → (d = 256)

$\gamma_1\ (p_x, p_y, p_z, E)$ → (d = 256)

$\gamma_2\ (p_x, p_y, p_z, E)$ → (d = 256)

$j_1\ (p_x, p_y, p_z, E)$

...

$j_4\ (p_x, p_y, p_z, E)$ → (d = 256)

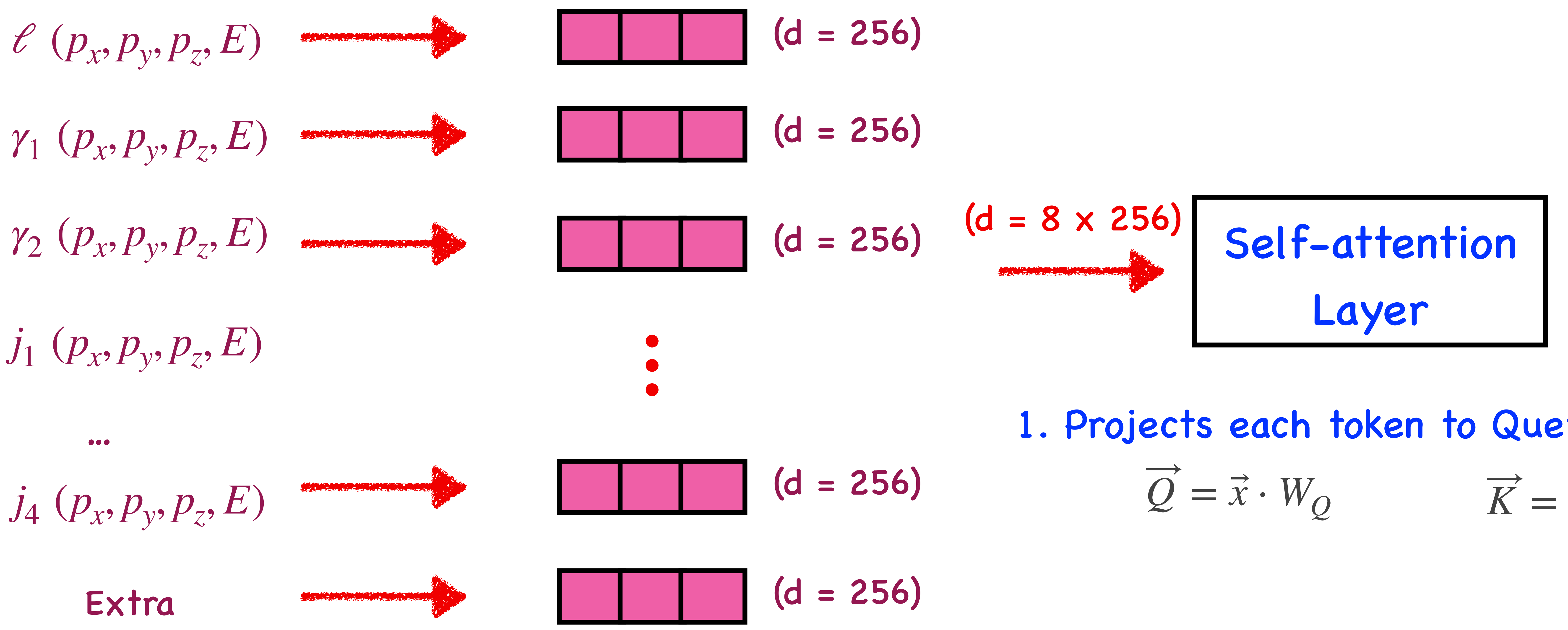
Extra → (d = 256)

Extras: $\not{E}_T, \Delta R_{min}(\ell, j_n), \Delta R_{max}(\ell, j_n), \Delta R_{min}(\ell, \gamma), \Delta R_{max}(\ell, \gamma), \Delta R_{min}(\gamma, j_n), \Delta R_{max}(\gamma, j_n), \sum_{i=1}^{n} p_T(j_i)$

Transformer detector-encoder

Detector-level final state: $\ell + \gamma\gamma + jets + \not{E}_T$

Latent representation



1. Projects each token to Query, Key and Value vectors

$$\vec{Q} = \vec{x} \cdot W_Q \qquad \vec{K} = \vec{x} \cdot W_K \qquad \vec{V} = \vec{x} \cdot W_V$$

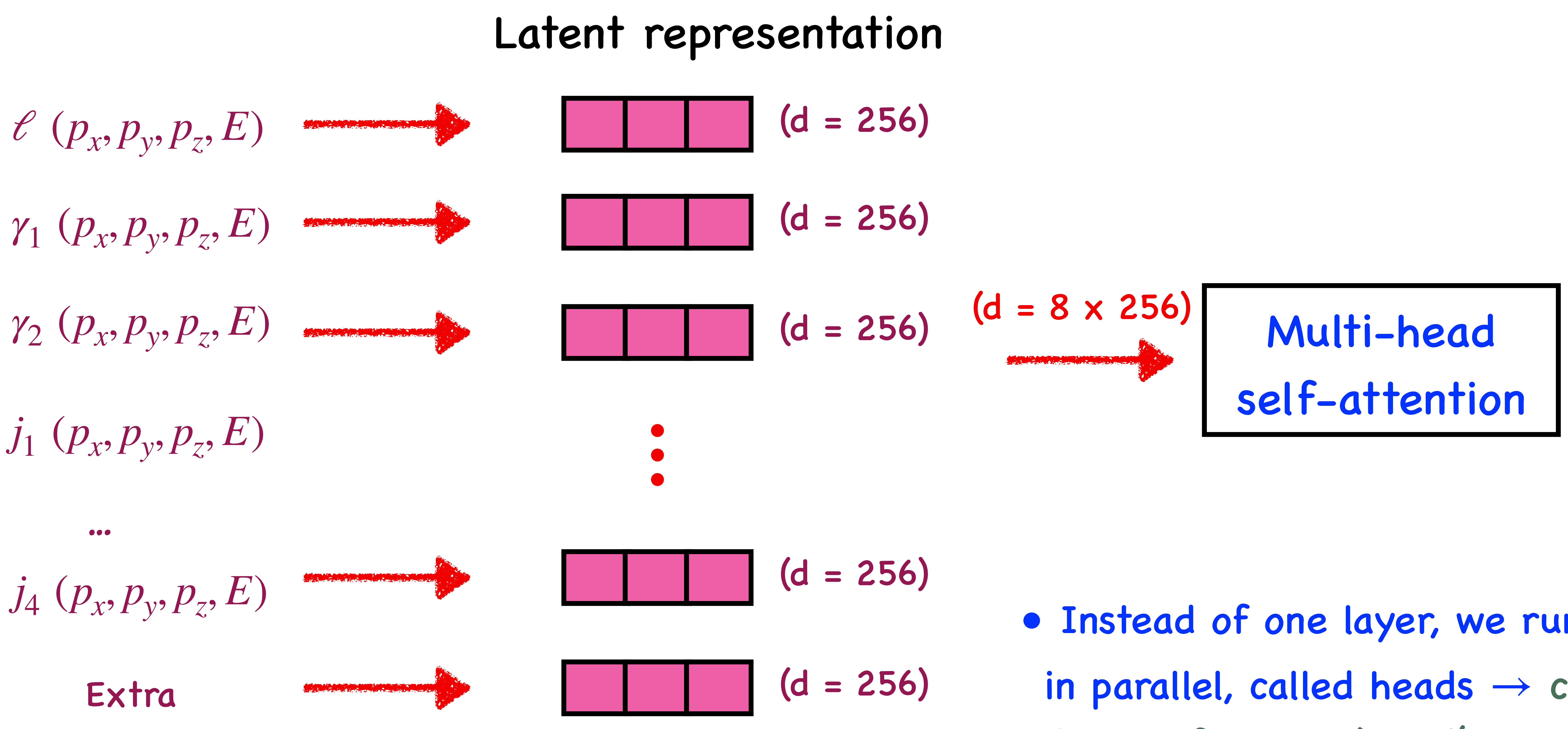
2. Computes weights with $SoftMax\left(QK^T/\sqrt{d_K}\right) \rightarrow$ an 8x8 matrix telling how each token correlates to every other

3. Per self-attention head output: $SoftMax\left(QK^T/\sqrt{d_K}\right) V \rightarrow$ dim (8 x 256)

Vaswani et al. (2017)

Transformer detector-encoder

Detector-level final state: $\ell + \gamma\gamma + jets + \not{E}_T$



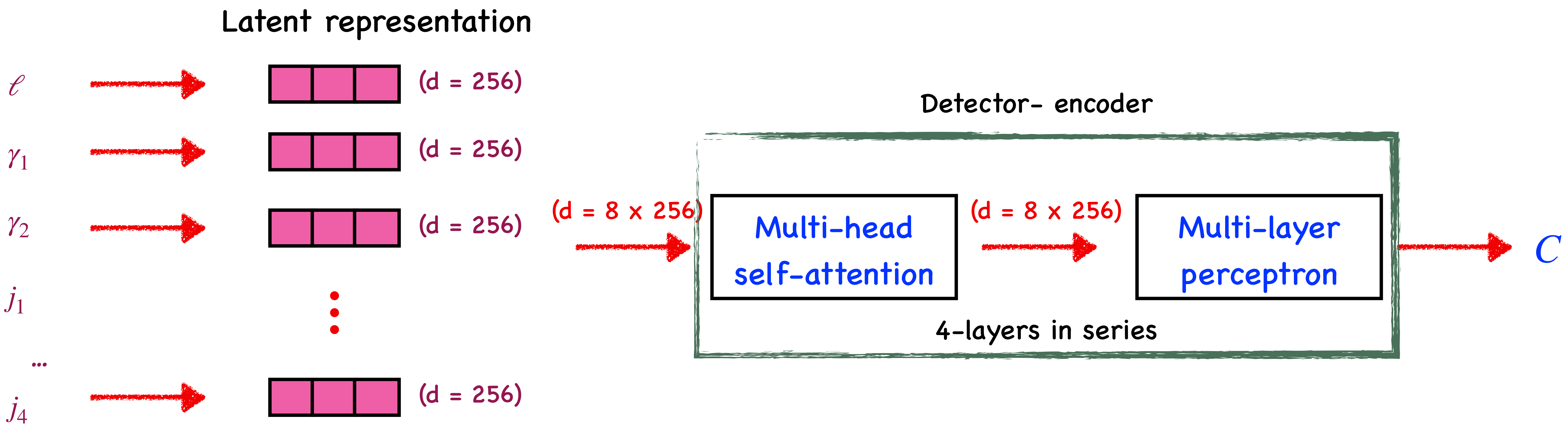
- Instead of one layer, we run 4 scaled dot-product attentions in parallel, called heads → creates different `perspectives' of Q, K, V for each `head'.

- Multi-head self-attention: The model `looks' or `pays attention' to all other inputs → helps to figure out which correlations matter the most.

Vaswani et al. (2017)

Transformer detector-encoder

Detector-level final state: $\ell + \gamma\gamma + jets + \not{E}_T$



- After multi-head attention, each token goes through the same 2-layer MLP → output dimension 8 x 256
- The 256-dim context vector C is constructed by averaging over the 8 tokens.

Vaswani et al. (2017)

C encodes the detector-level information in a permutation-invariant way and will be used by the diffusion model as a conditional input.

Diffusion-model

Parton-level:

$\tilde{\chi}_2^0\,\{p_x, p_y, p_z, m\},\ \ \tilde{\chi}_1^0\,\{p_x, p_y, p_z, m\}$

←

Detector-level final state:

$\ell + \gamma\gamma + jets + \cancel{E}_T$

Forward noising:

- Small amounts of noise are sequentially introduced to the target distributions.

- Noise Schedule: choose small variances β_t that increases linearly from 10^{-5} to 0.02 over $T = 1500$ time steps

  → β_t controls the amount of noise added to parton-level vectors x_p^0 at each tilmestep

  Noised vector $x_p^t = \sqrt{\bar{\alpha}_t}\, x_p^0 + \sqrt{1 - \bar{\alpha}_t}\, \epsilon,$ Signal retention factor $\bar{\alpha}_t = \prod_{i=1}^{t} 1 - \beta_i$

  → Early t: x_p^t resembles x_p^0 ($\bar{\alpha}_t = 0.987$ at t = 50)

  → Late t: x_p^t resembles noise ϵ ($\bar{\alpha}_t \sim 10^{-5}$ at t = 1500)

- Amount of noise depends on the timestep t → during inference, the reverse diffusion process must be conditioned on t.

- Time embedding: integer t is mapped to a frequency embedding to be used as a conditional input during inference.

Diffusion-model

Parton-level:

$\tilde{\chi}_2^0\,\{p_x, p_y, p_z, m\}, \;\; \tilde{\chi}_1^0\,\{p_x, p_y, p_z, m\}$

Detector-level final state:

$\ell + \gamma\gamma + jets + \not{E}_T$

Training

- Context vector C from detector-encoder and time-embedding $(pe)_{d_t}$, both, passed through a MLP into a d_{dF} = 256 dim latest representation.
- The noised target (dim-8) is concatenated to get $(2d_{DF} + 8 = 520)$ dimensions → fed into a MLP that predicts the noise $\hat{\eta}(C, (pe)_{d_t}, x_p^t)$
- Trained on the MSE Loss function

$$\mathcal{L} = \mathbb{E}_{x_p^0, t, \epsilon}\,||\epsilon - \hat{\eta}(C, (pe)_{d_t}, x_p^t)||^2$$

Testing

Reverse diffusion → Sampling the parton-level

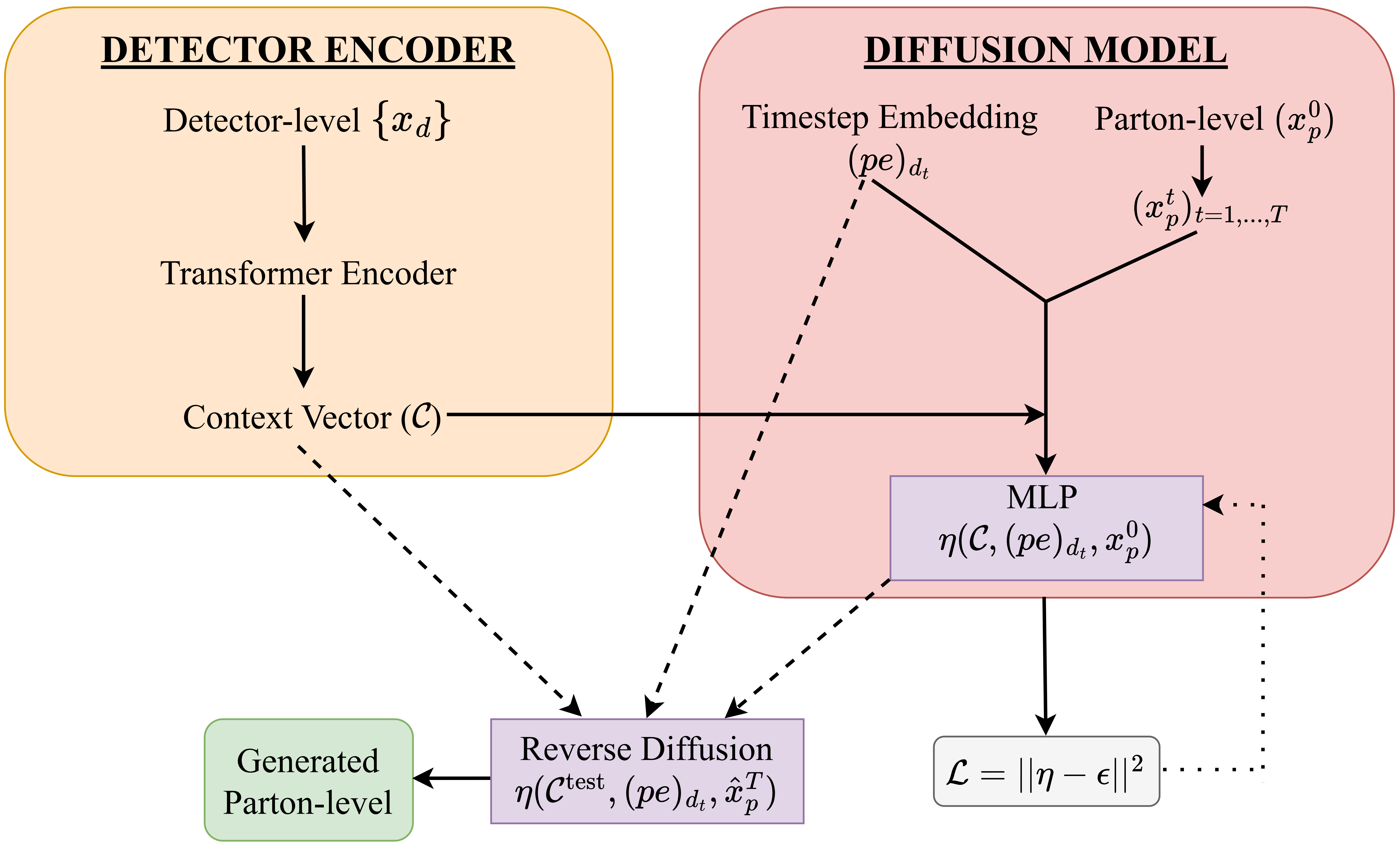
- Start: Draw x_p^T from a gaussian noise.
- Go to the previous step at $T-1$:

$$\hat{x}_p^{T-1} = \mu_T = \frac{1}{\sqrt{\alpha_T}}\left(\hat{x}_p^T - \frac{1-\alpha_T}{\sqrt{1-\bar{\alpha}_T}}\,\eta\left(\mathscr{C}^{\text{test}}, (pe)_{d_t}, \hat{x}_p^T\right)\right) + \sqrt{\beta_T}\,z$$

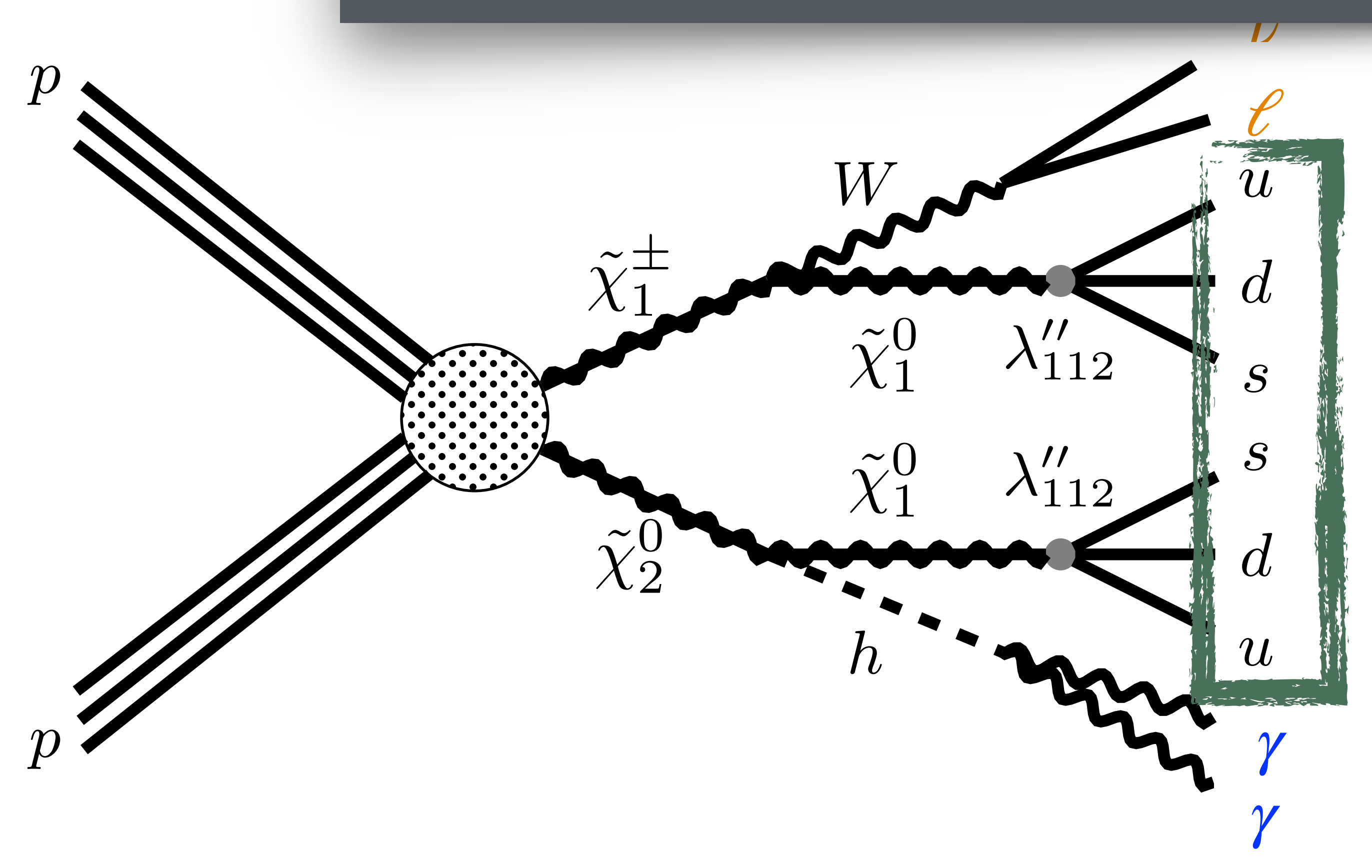
J. Ho, A. Jain, P. Abbeel (2006.11239)

Model schematics

[RKB, Arghya Choudhury, Subhadeep Sarkar, 2507.20869]



Reconstructing masses with a hybrid diffusion model



$$pp \to \tilde{\chi}_1^\pm \tilde{\chi}_2^0 \to (\tilde{\chi}_1^\pm \to W^\pm \tilde{\chi}_1^0)\,(\tilde{\chi}_2^0 \to h\tilde{\chi}_1^0) \to (W^\pm \to \ell'\nu)\,(h \to \gamma\gamma) + \tilde{\chi}_1^0\,\tilde{\chi}_1^0$$

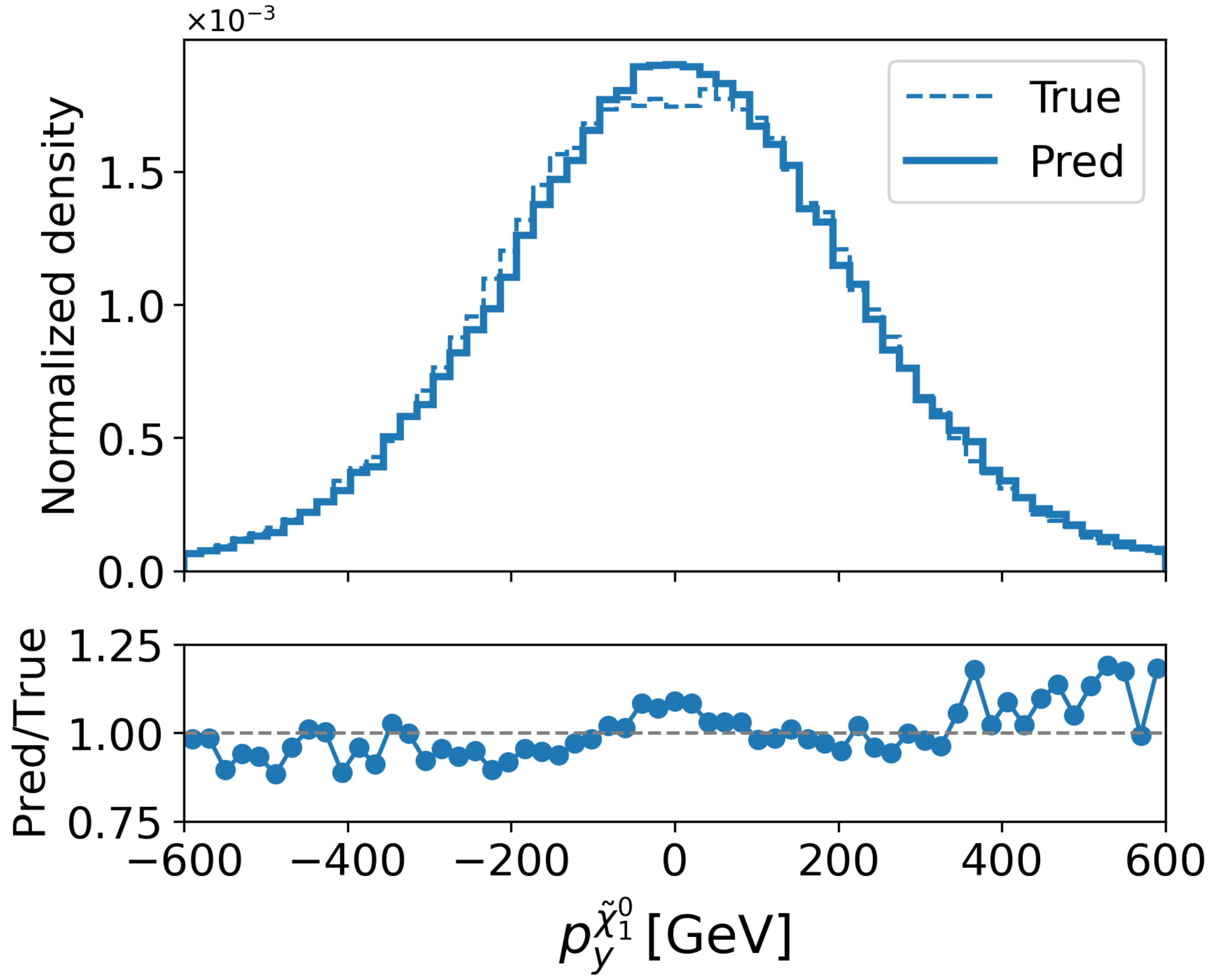
Parton-level:

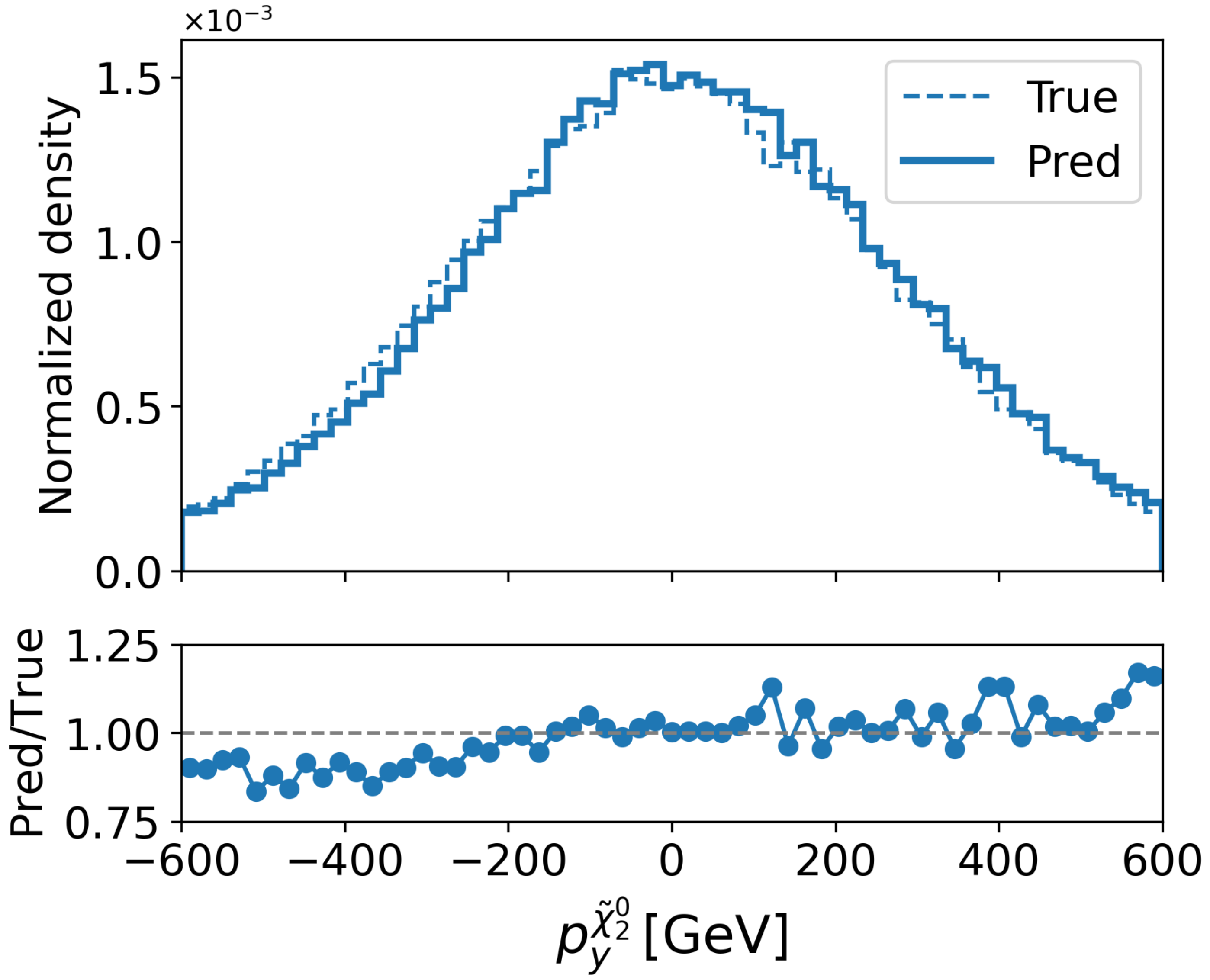
$\tilde{\chi}_2^0\,\{p_x, p_y, p_z, m\},\ \ \tilde{\chi}_1^0\,\{p_x, p_y, p_z, m\}$

Detector-level final state:

$\ell + \gamma\gamma + jets + \not{E}_T$

- Truth and generated parton-level distributions (Trained and tested on $m_{\tilde{\chi}_2^0} = 600$ GeV, $m_{\tilde{\chi}_1^0} = 200$ GeV)





[RKB, A. Choudhury, S. Sarkar, 2507.20869]

☑ Good agreement between truth and model generated parton-level.

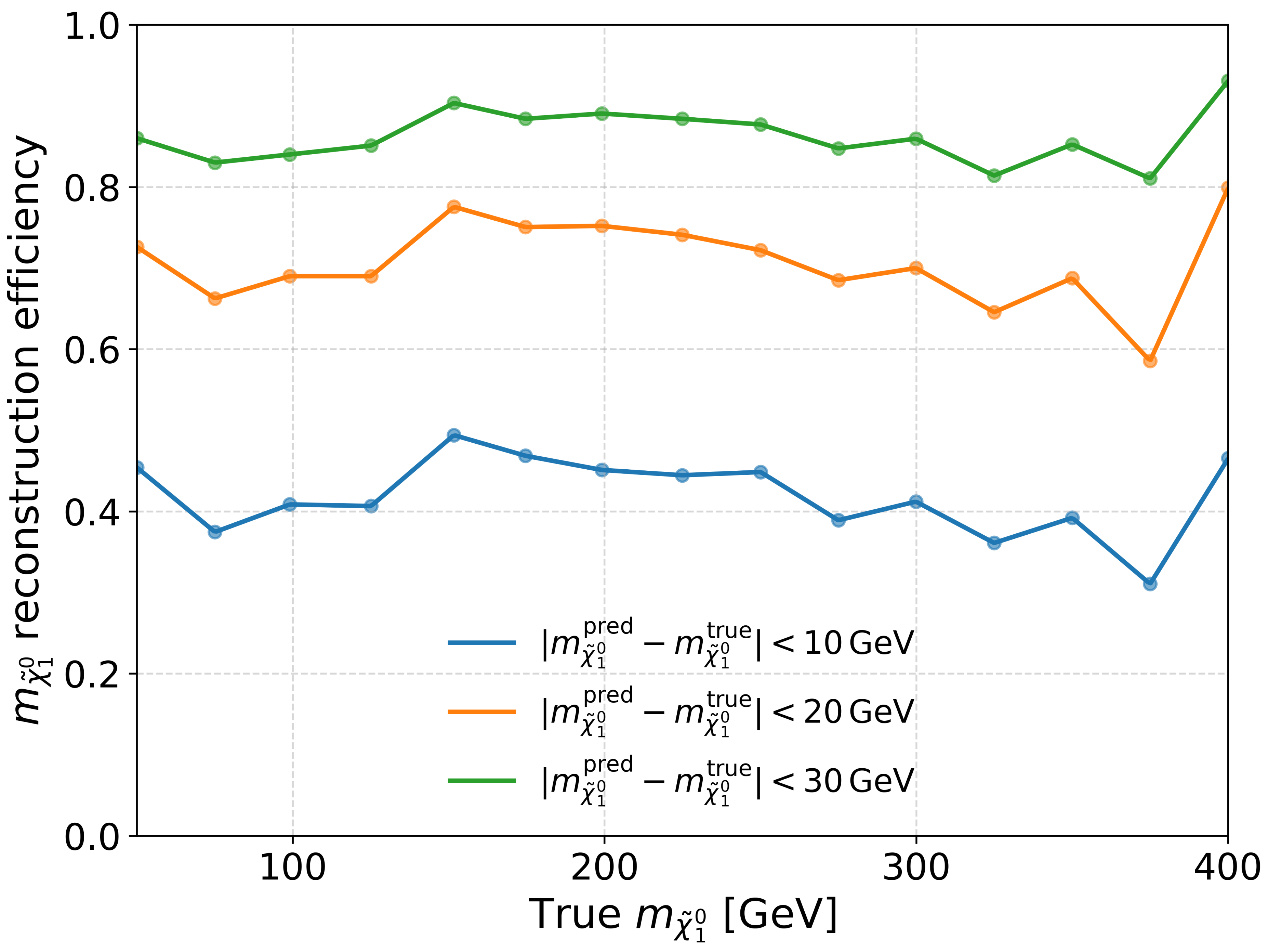
Mass reconstruction efficiencies

- Fix $m_{\tilde{\chi}_2^0} = 600$ GeV, and interpolate $m_{\tilde{\chi}_1^0}$

[RKB, A. Choudhury, S. Sarkar, 2507.20869]

→ A single model is trained on events generated with $m_{\tilde{\chi}_1^0}$ varying between 50 GeV to 400 GeV, with a 50 GeV step size ~ 35000 events from each benchmark

→ Tested on events with $m_{\tilde{\chi}_1^0}$ in the same range, but with step size of 25 GeV.



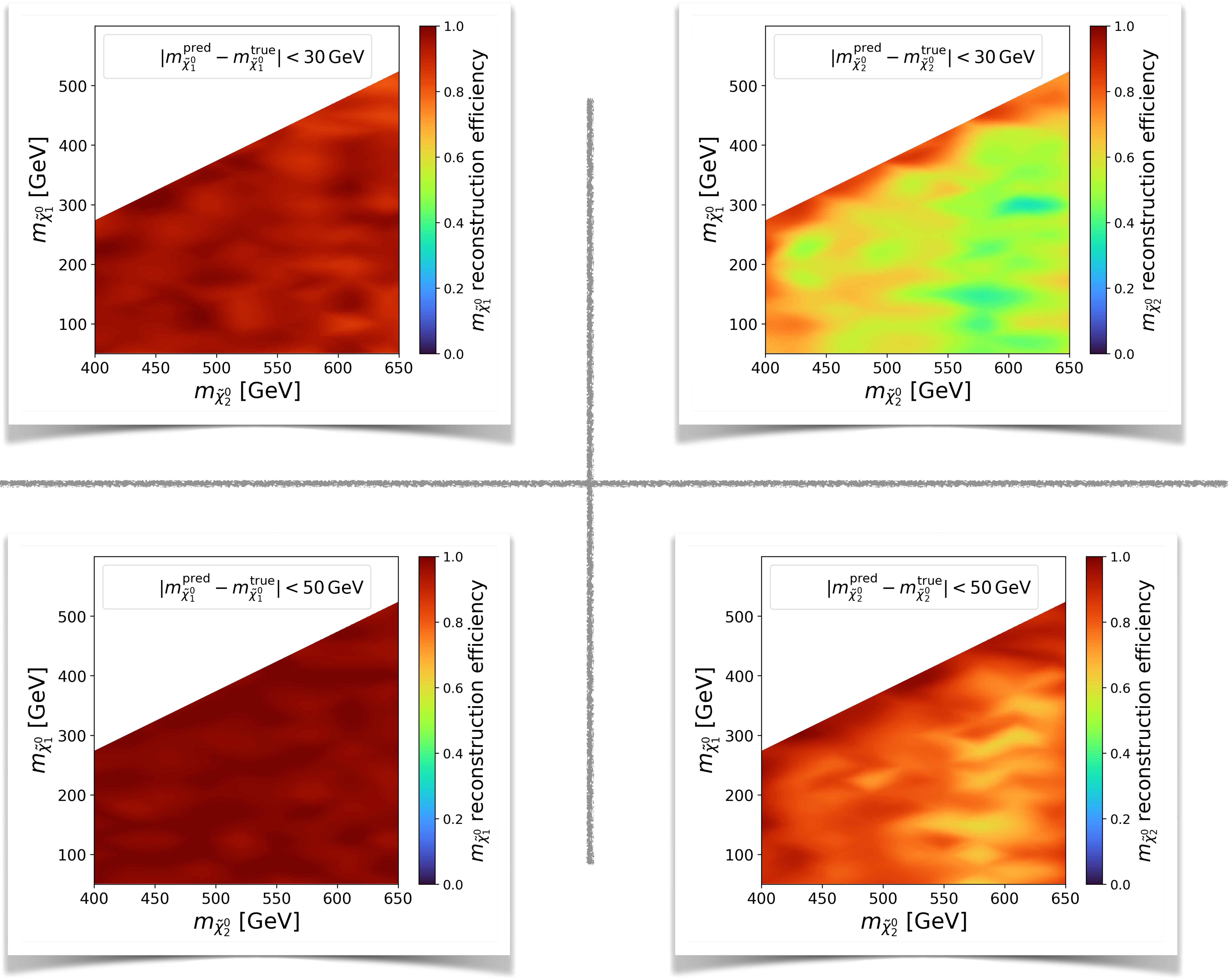
◆ Reconstruction efficiency $\gtrsim 80\,\%$ for $\Delta m = |m^{\text{pred}}_{\tilde{\chi}_1^0} - m^{\text{true}}_{\tilde{\chi}_1^0}| < 30$ GeV

Mass reconstruction efficiencies

[RKB, A. Choudhury, S. Sarkar, 2507.20869]

- How well does the network generalize for the full parameter space spanning both $m_{\tilde{\chi}_1^0}$ and $m_{\tilde{\chi}_2^0/\tilde{\chi}_1^\pm}$?

→ Trained on events generated with $m_{\tilde{\chi}_2^0} \in [400{,}550]$ GeV and $m_{\tilde{\chi}_1^0} \in [50\ \mathrm{GeV}, \mathrm{m}_{\tilde{\chi}_2^0} - \mathrm{m_h}]$

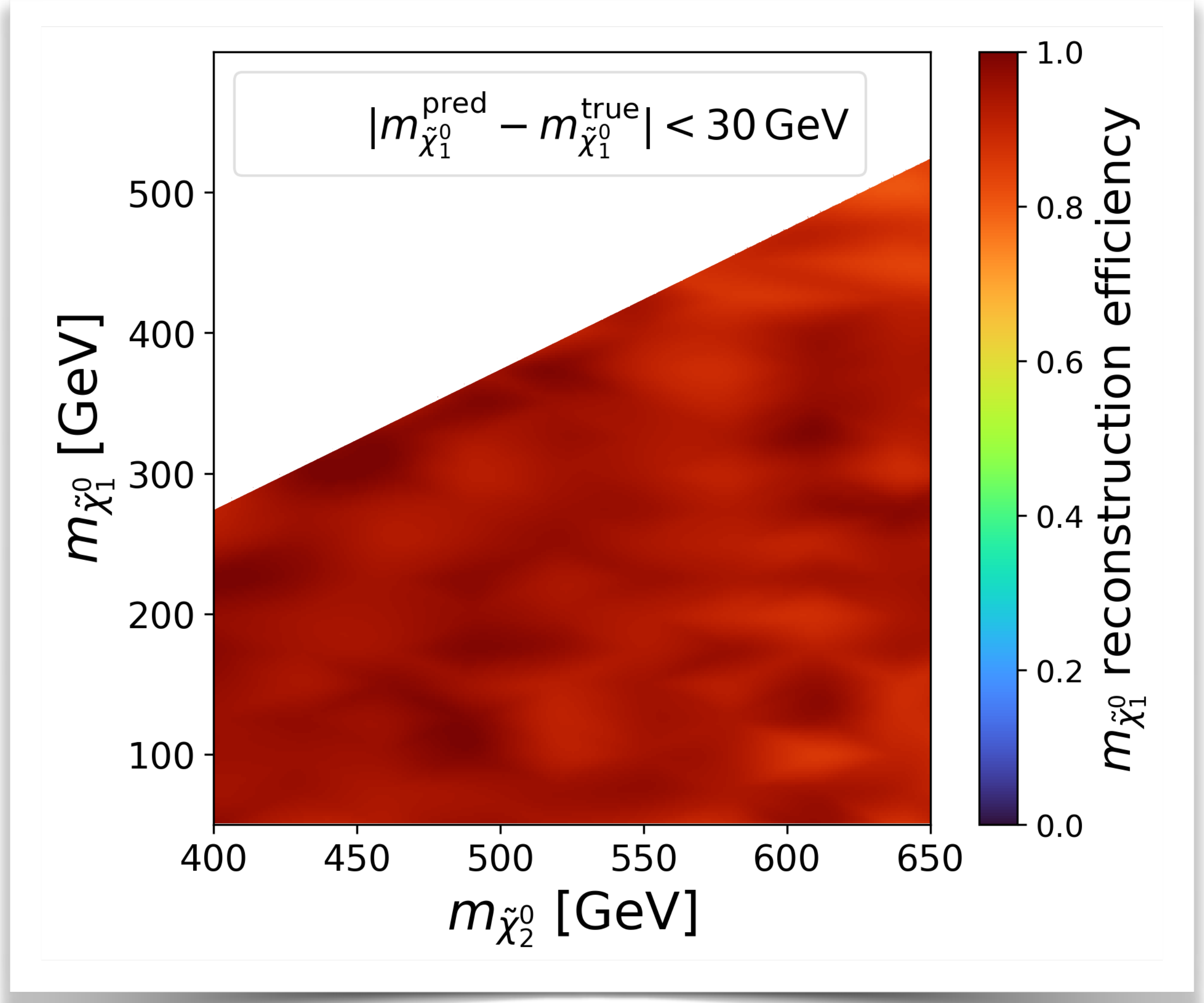


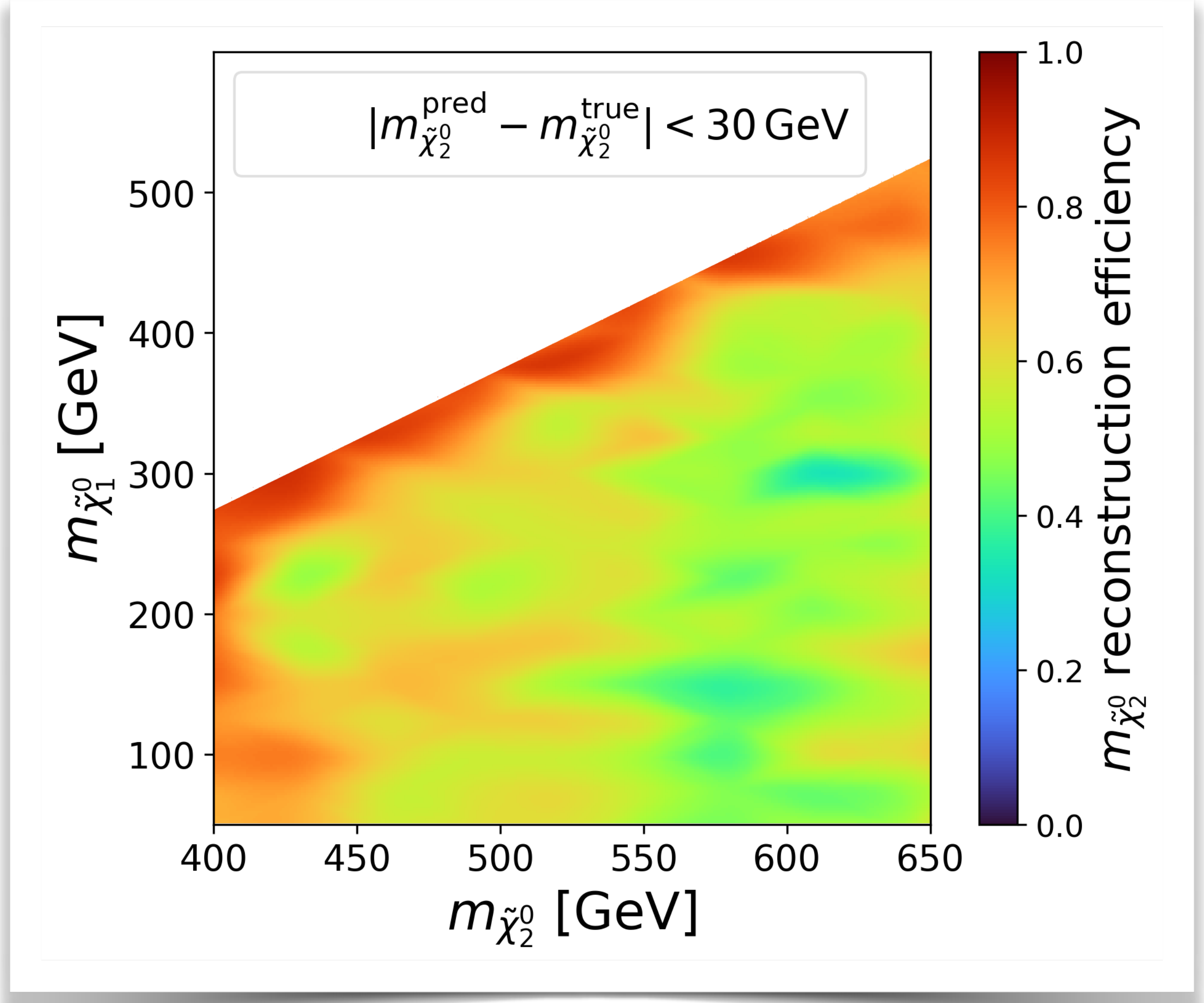
Mass reconstruction efficiencies

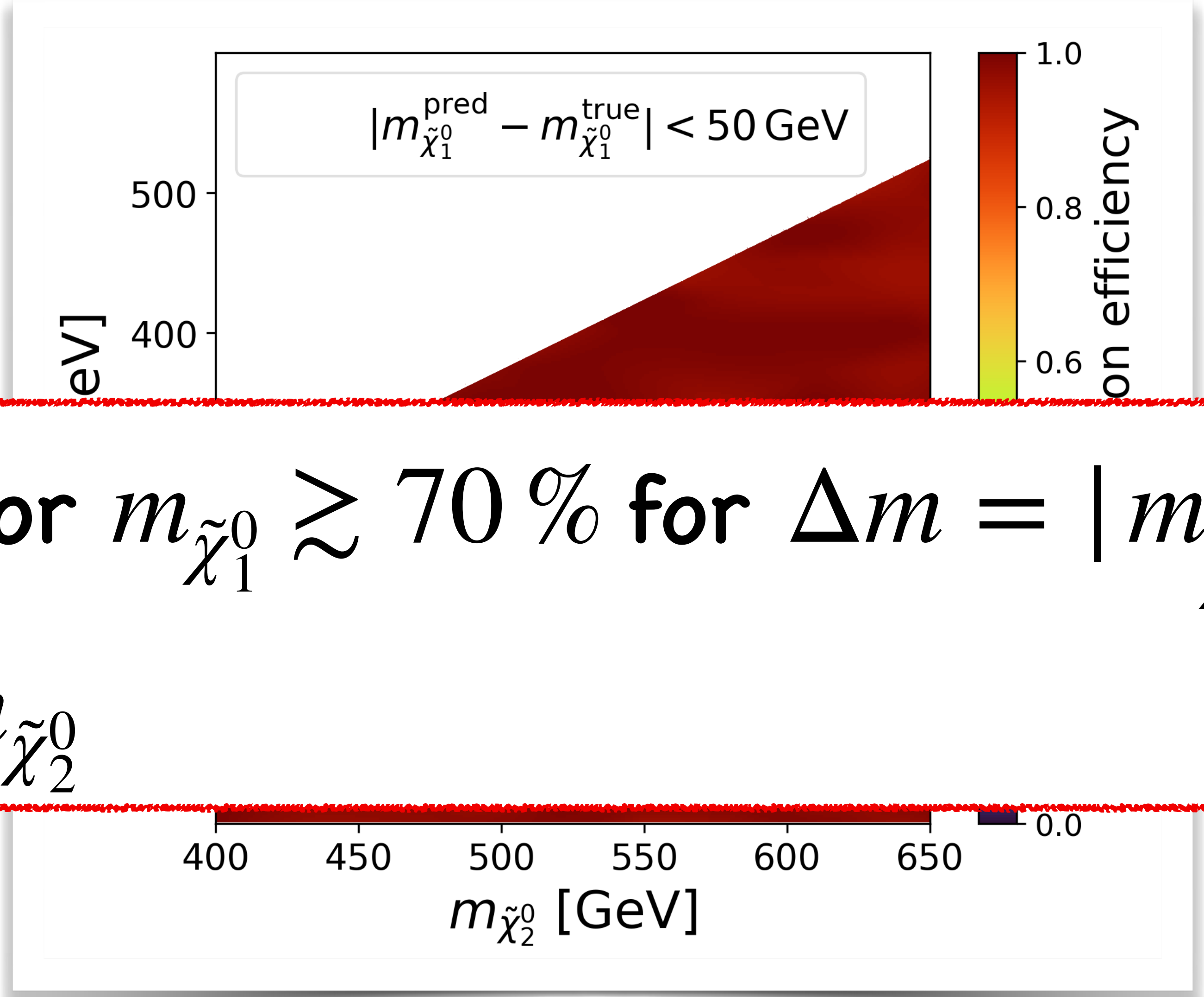
[RKB, A. Choudhury, S. Sarkar, 2507.20869]

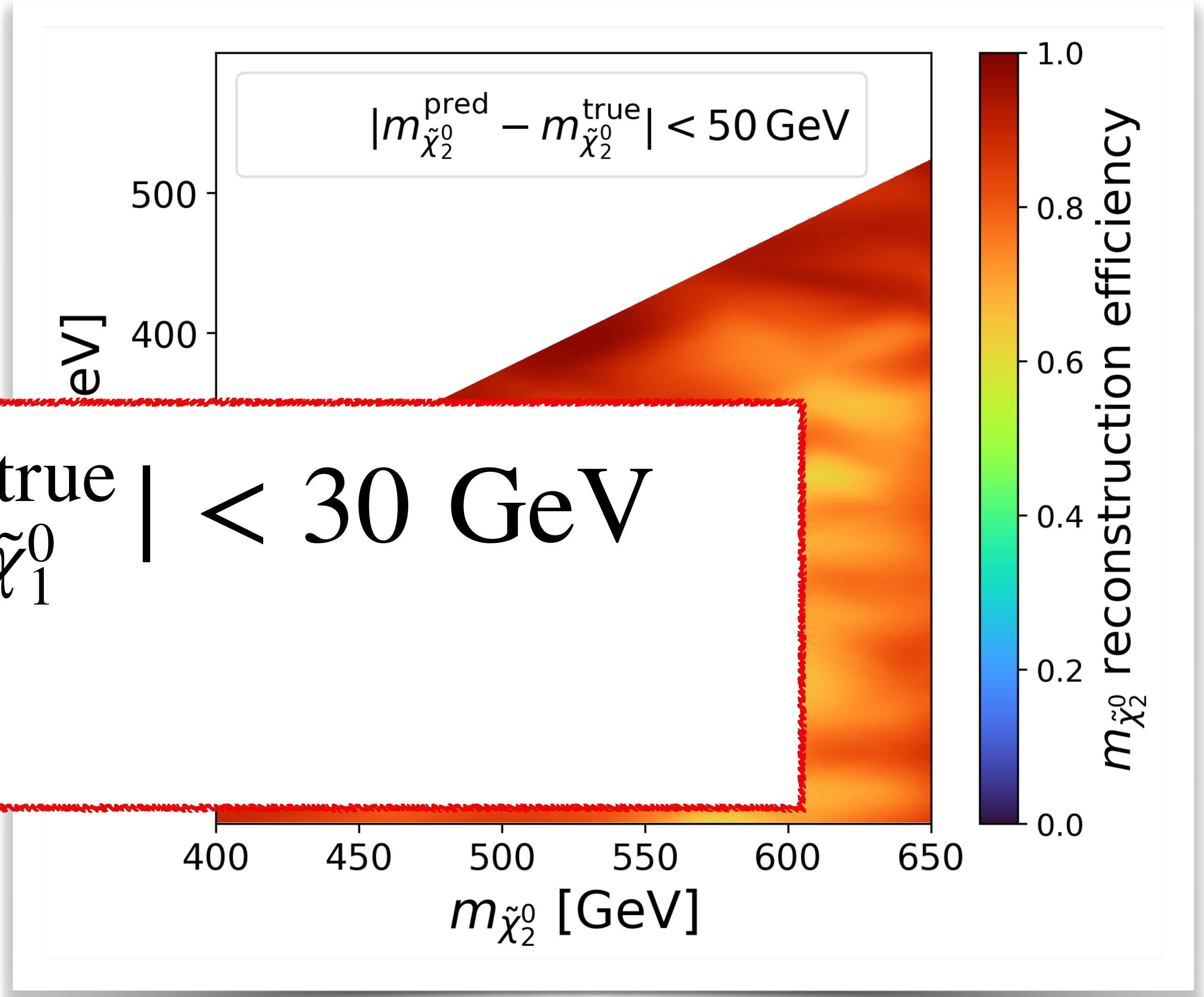
- How well does the network generalize for the full parameter space spacing both $m_{\tilde{\chi}_1^0}$ and $m_{\tilde{\chi}_2^0/\tilde{\chi}_1^\pm}$?

→ Trained on events generated with $m_{\tilde{\chi}_2^0} \in [400,550]$ GeV and $m_{\tilde{\chi}_1^0} \in [50\ \mathrm{GeV}, \mathrm{m}_{\tilde{\chi}_2^0} - \mathrm{m_h}]$









- ◆ Reconstruction efficiency for $m_{\tilde{\chi}_1^0} \gtrsim 70\,\%$ for $\Delta m = |m^{\rm pred}_{\tilde{\chi}_1^0} - m^{\rm true}_{\tilde{\chi}_1^0}| < 30$ GeV
- ◆ Weaker performance for $m_{\tilde{\chi}_2^0}$

Mass reconstruction efficiencies

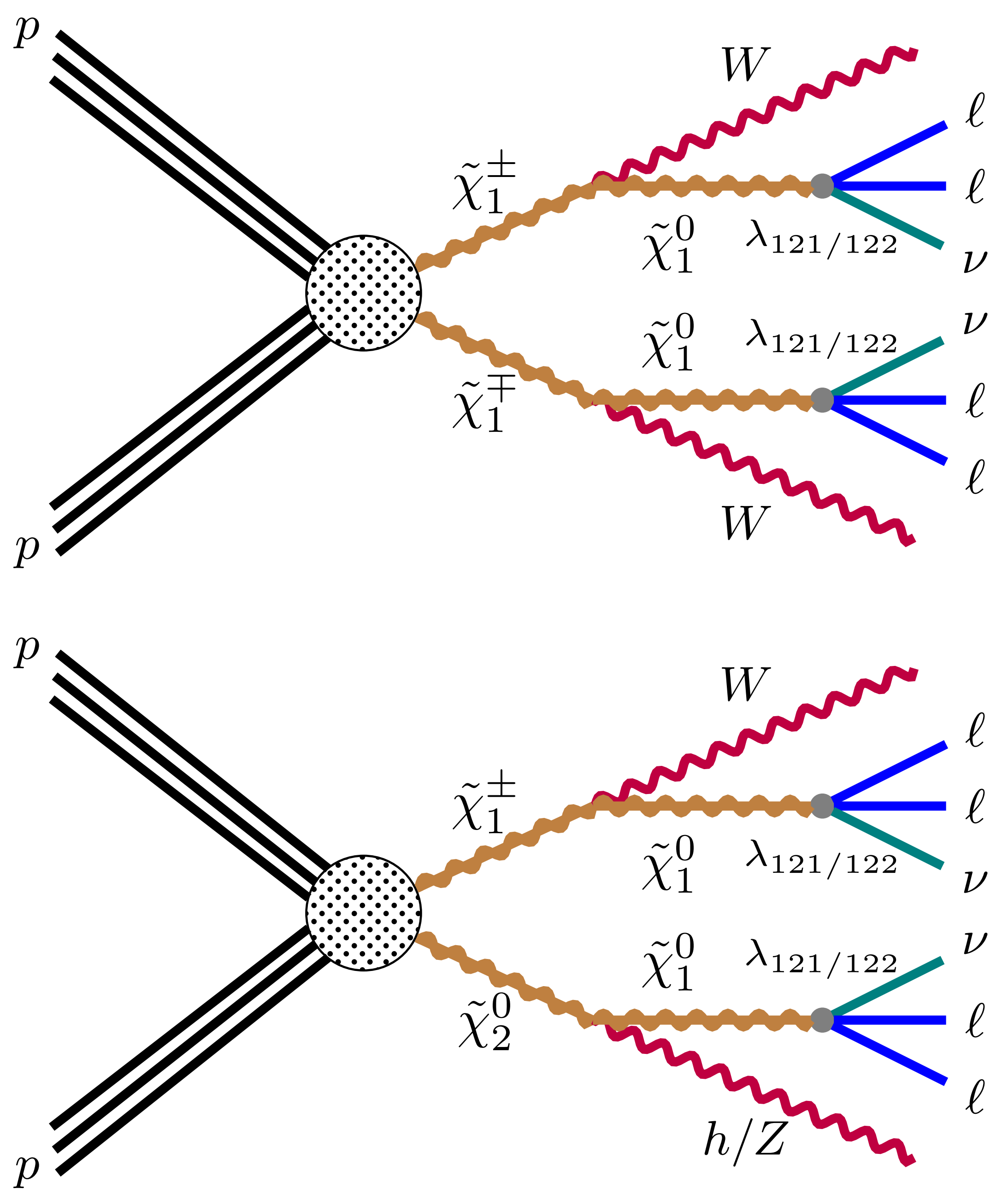
[RKB, A. Choudhury, S. Sarkar, 2507.20869]

A different process with multiple sources of missing energy in the final state:

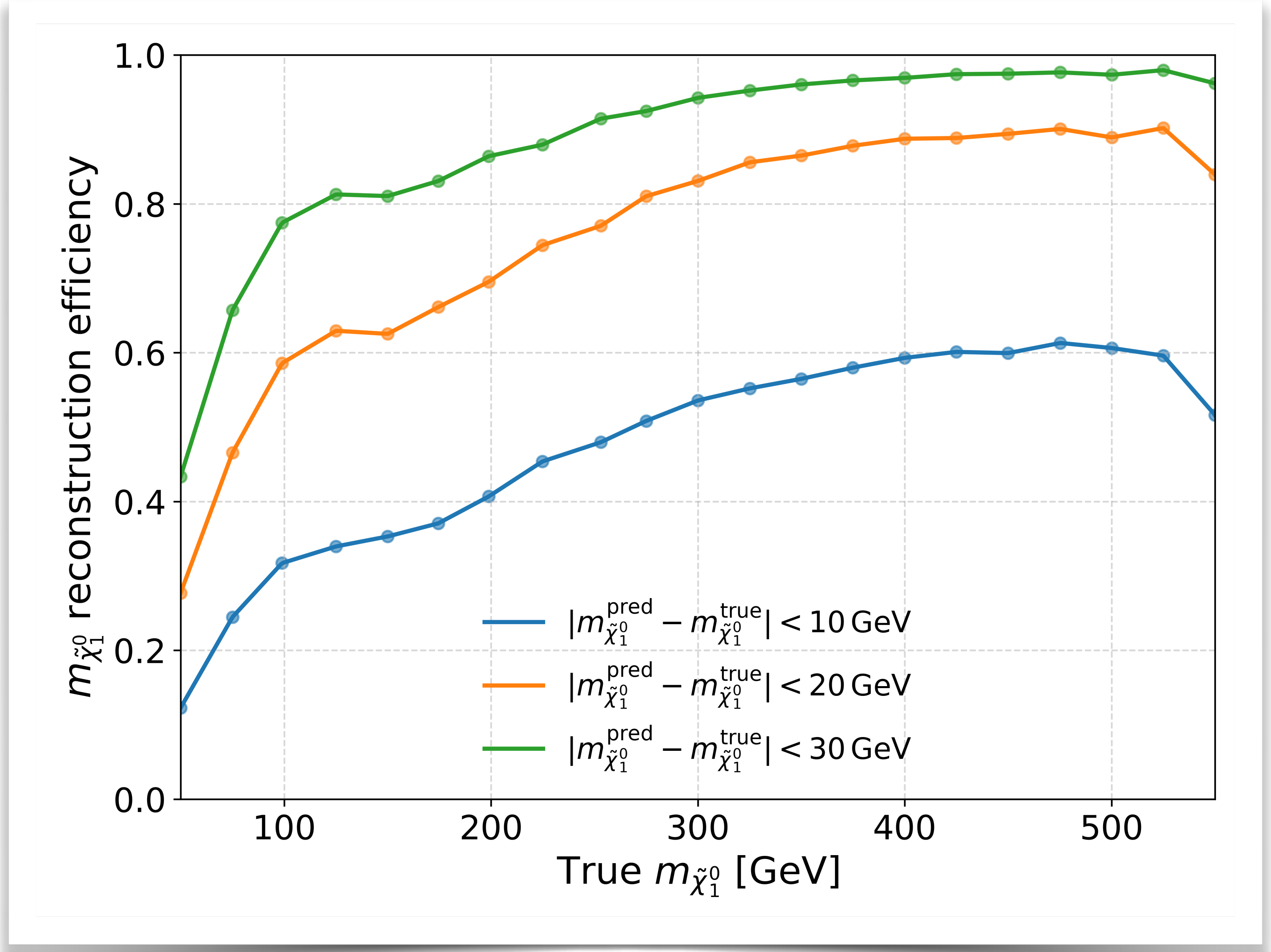
$$pp \to \tilde{\chi}_2^0 \tilde{\chi}_1^\mp + \tilde{\chi}_1^\mp \tilde{\chi}_1^\pm \to (\tilde{\chi}_2^0 \to Z/h + \tilde{\chi}_1^0)\,(\tilde{\chi}_1^\pm \to W^\pm \tilde{\chi}_1^0) + (\tilde{\chi}_1^\mp \to W^\mp \tilde{\chi}_1^0)\,(\tilde{\chi}_1^\pm \to W^\pm \tilde{\chi}_1^0)$$

$$\{p_x, p_y, p_z, E\}_{\ell_i}, \not{p}_x, \not{p}_y, \Delta R(\ell_i, \ell_j), \Delta\phi(\ell_i, \not{E}_T)$$

$\boxed{\text{Detector-level final state: } 4\ell + \not{E}_T}$



- Fix $m_{\tilde{\chi}_2^0} = 1500$ GeV, and interpolate $m_{\tilde{\chi}_1^0}$



◆ Reconstruction efficiency $\gtrsim 80\,\%$ for $\Delta m < 30$ GeV

Summary and outlook

- ML in HEP needs data-efficient models for faster inference and precise high-dimensional unfolding
  $\rightarrow$ made feasible by generative ML, to go beyond simple classifiers.

- The diffusion based framework learns $p(x|y)$ across a wide parameter space with stable training and good mass reconstruction.

- Further calibration and sequential conditional training could help improve the reconstruction efficiency.

- Physics scope: extend from mass reconstruction to full parton-level reconstuction in other BSM scenarios, processes, and include background for end-to-end analyses.

Thank you

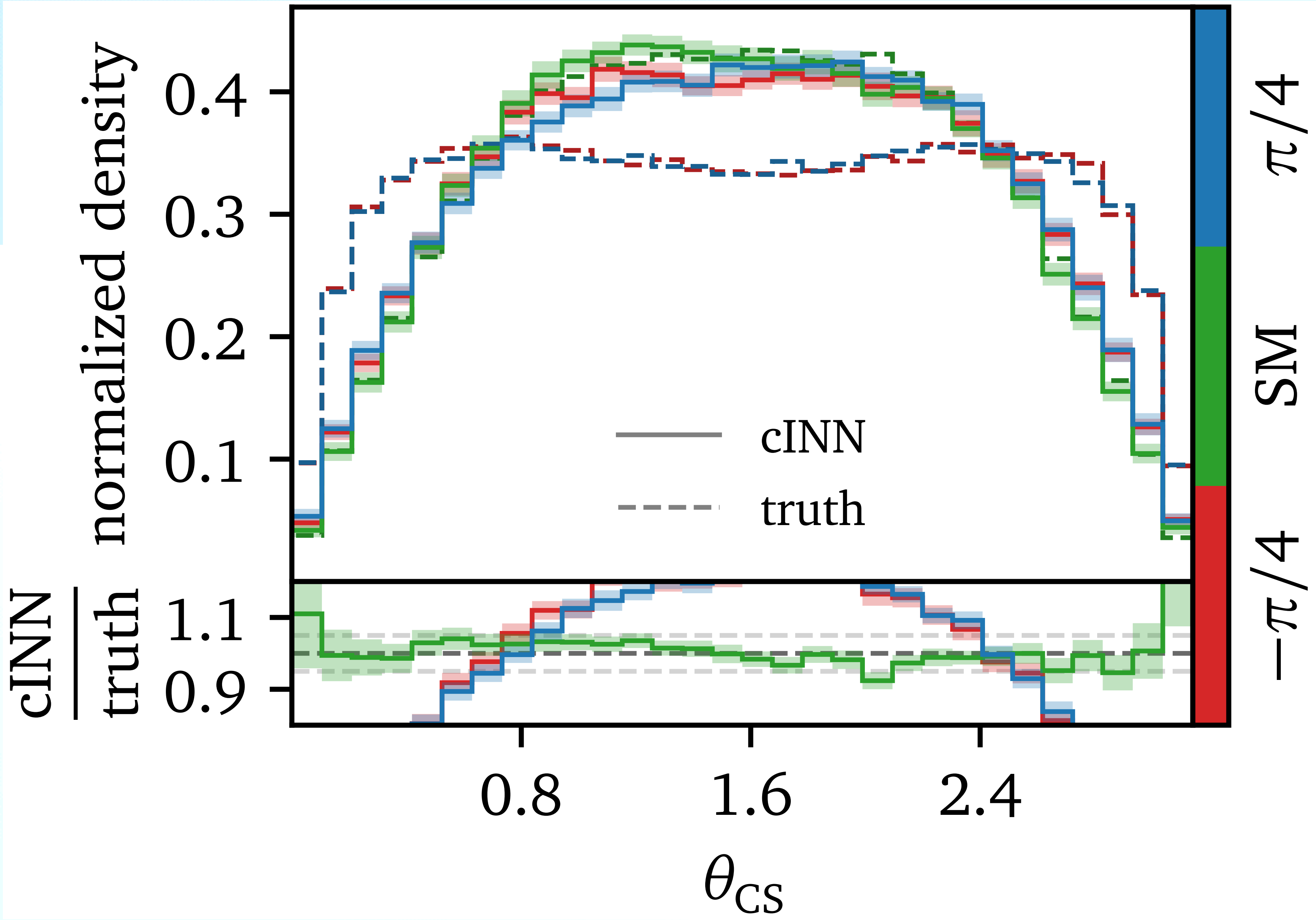
Back-up Slides

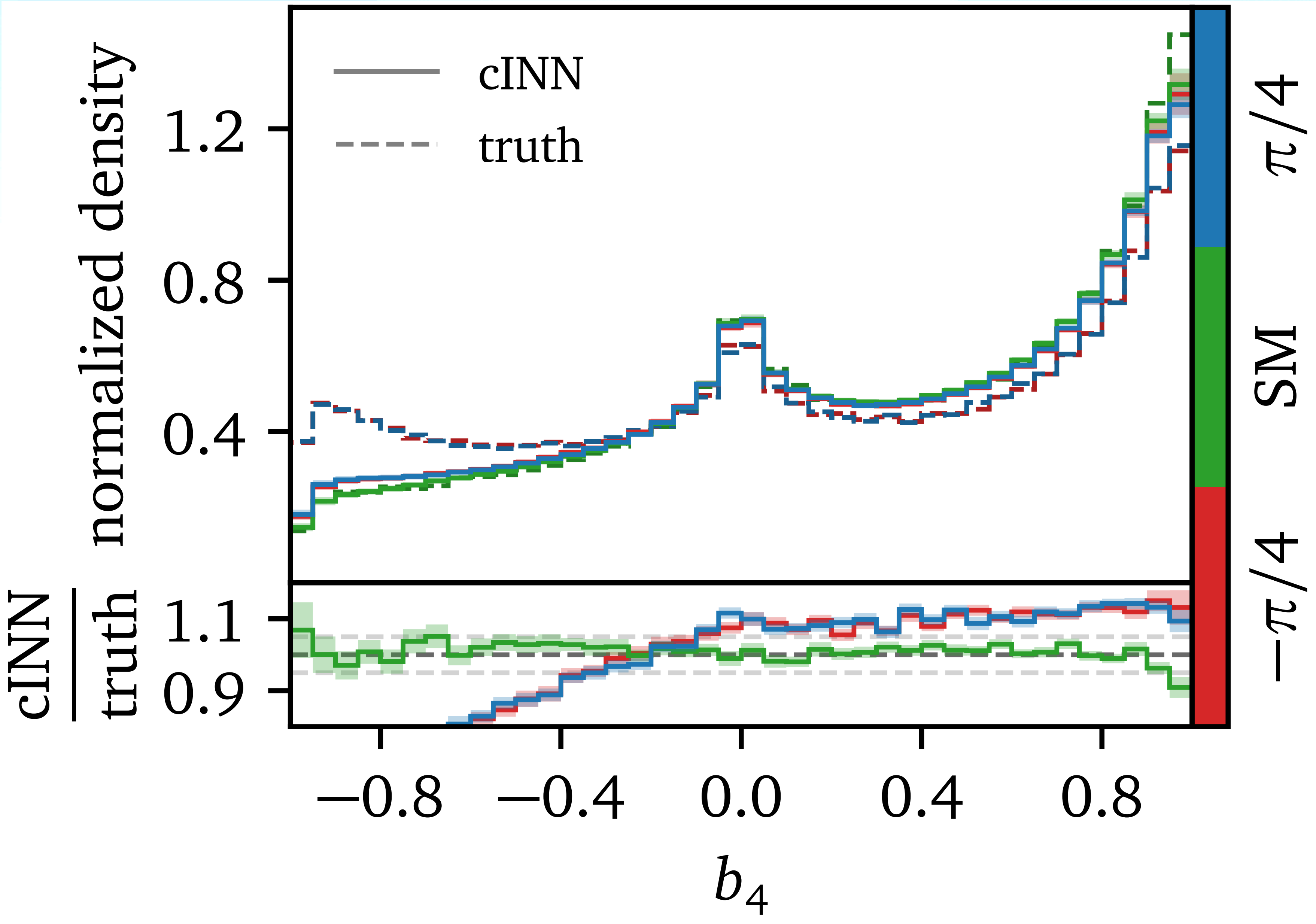
Test Model dependence

[Jona Ackerschott, **RKB**, Dorival Goncalves, Theo Heimel, Tilman Plehn, SciPost Phys. 17 (2024) 1, 001]

Use all 3 models to unfold SM $t\bar{t}h$

- Model 1: cINN trained on $t\bar{t}h$ with Higgs-top CP phase $\alpha = \pi/4$
- Model 2: cINN trained on SM $t\bar{t}h$
- Model 3: cINN trained on $t\bar{t}h$ with Higgs-top CP phase $\alpha = -\pi/4$





☑ Model bias is much smaller compared to variations in parton-level truth when α is varied.