Calomplification: The Power of Generative Calorimeter Models

Sebastian Bieringer[1], Anja Butter, Sascha Diefenbacher, Engin Enren, Frank Gaede, Daniel Hundshausen, Gregor Kasieczka, Benjamin Nachman, Tilman Plehn, Mathias Trabs

[1]Institut für Experimentalphysik, Universität Hamburg, Germany
sebastian.guido.bieringer@uni-hamburg.de

CERN-IML Workshop 2022

Introduction

Need to speed up MC
- Event generation
- Calorimeter simulation

→

Use generative machine learning models like
- Generative Adversarial Networks (GANs)
- or Variational Autoencoders (VAEs)

$$\text{simulation speed} = \frac{\#\text{ samples}}{\text{time}}$$

What about # samples?

A. Butter et al. *GANplifying Event Samples*. 2021. arXiv: 2008.06545 [hep-ph]

S. Bieringer et al. *Calomplification -- The Power of Generative Calorimeter Models*. 2022. arXiv: 2202.07352 [hep-ph]

Introduction

train

Limited data

Generative machine learning model

- Generative Adversarial Network (GANs)
- or Variational Autoencoder (VAEs)

sample

Unlimited data?

- Slow simulation e.g. calorimeter simulation

- Fast simulation?

Scaling properties with number of sampled points

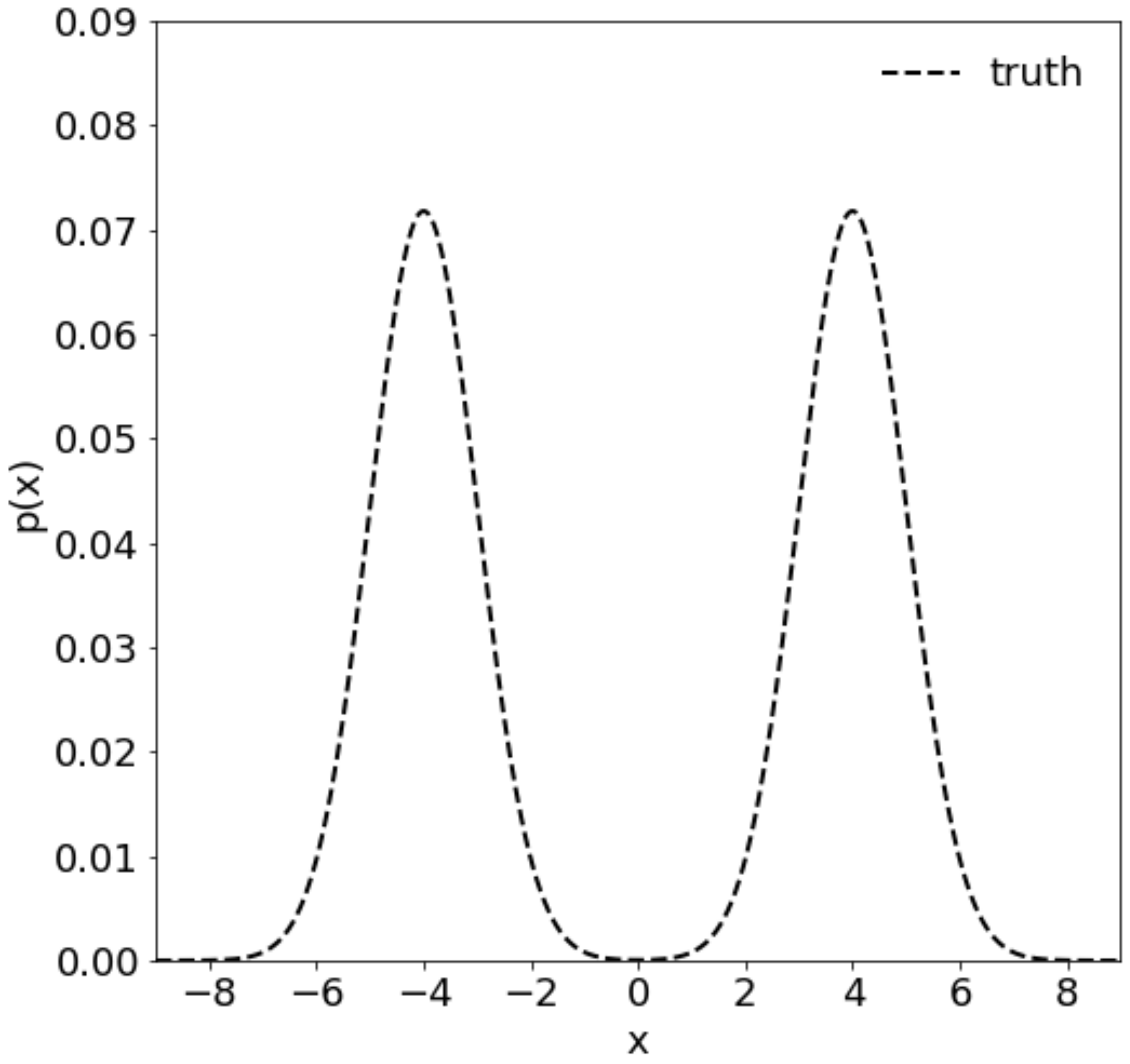
A. Butter et al. *GANplifying Event Samples*. 2021. arXiv: 2008.06545 [hep-ph]

S. Bieringer et al. *Calomplification -- The Power of Generative Calorimeter Models*. 2022. arXiv: 2202.07352 [hep-ph]

Toy Model: Setup

- Underlying function:

$$P(x) = \frac{1}{2}\left(\mathcal{N}_{-4,1}(x) + \mathcal{N}_{4,1}(x)\right)$$

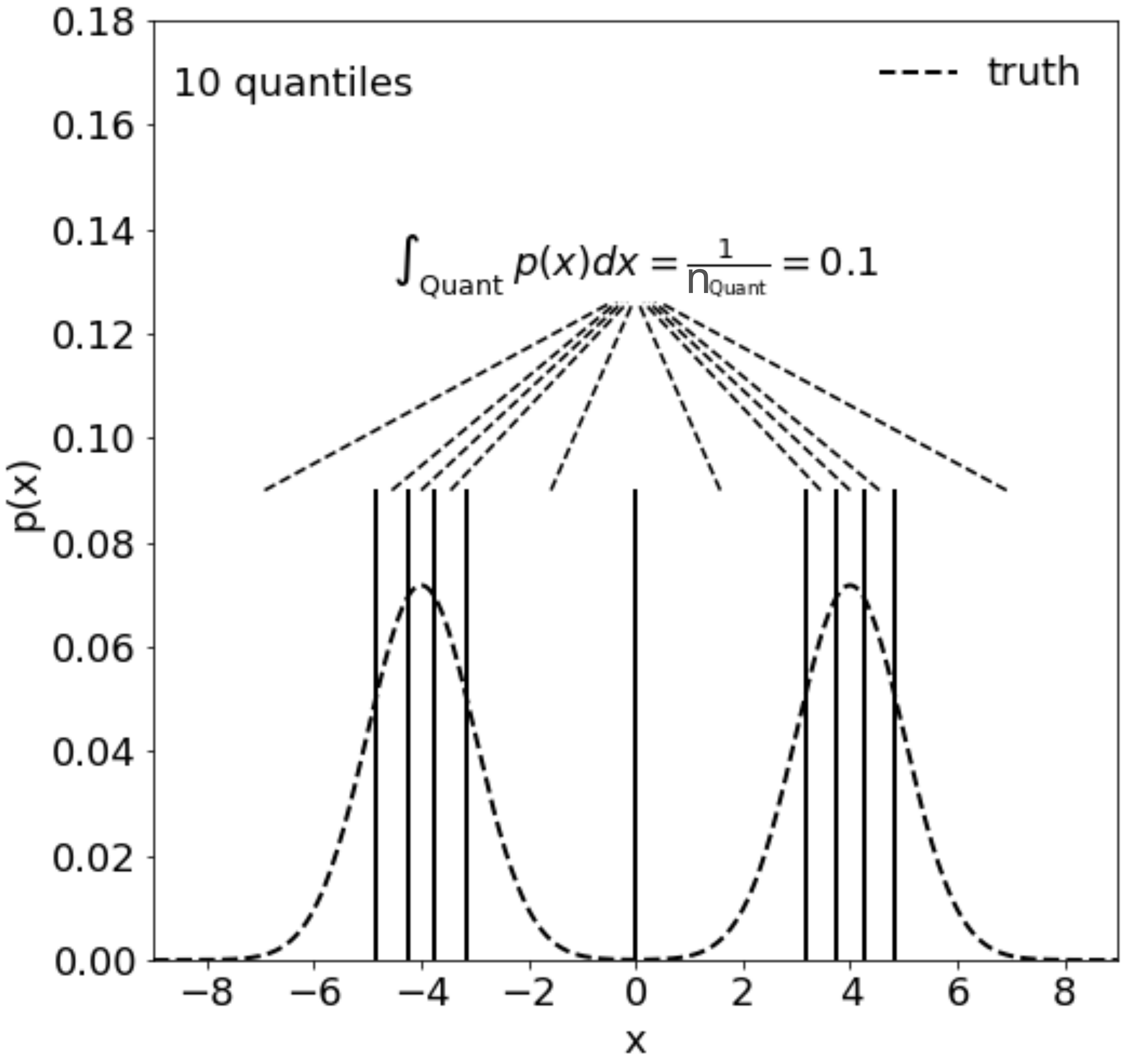


Toy Model: Setup

- Underlying function:

$$P(x) = \frac{1}{2}\left(\mathcal{N}_{-4,1}(x) + \mathcal{N}_{4,1}(x)\right)$$

- *"Pearson χ^2-test"*:
  - Introduce equal probability quantiles



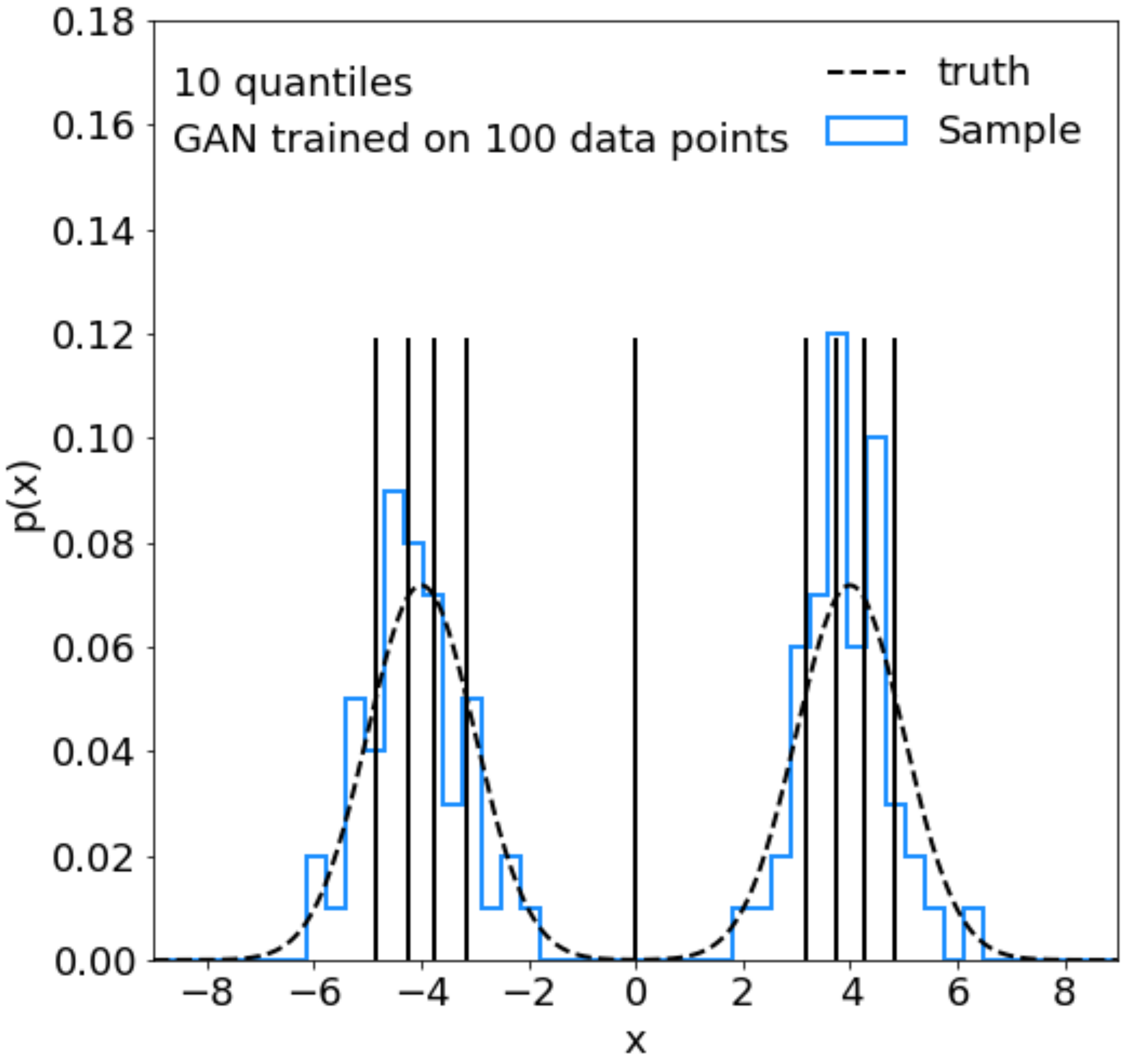
Toy Model: Setup

- Underlying function:

$$P(x) = \frac{1}{2}\left(\mathcal{N}_{-4,1}(x) + \mathcal{N}_{4,1}(x)\right)$$

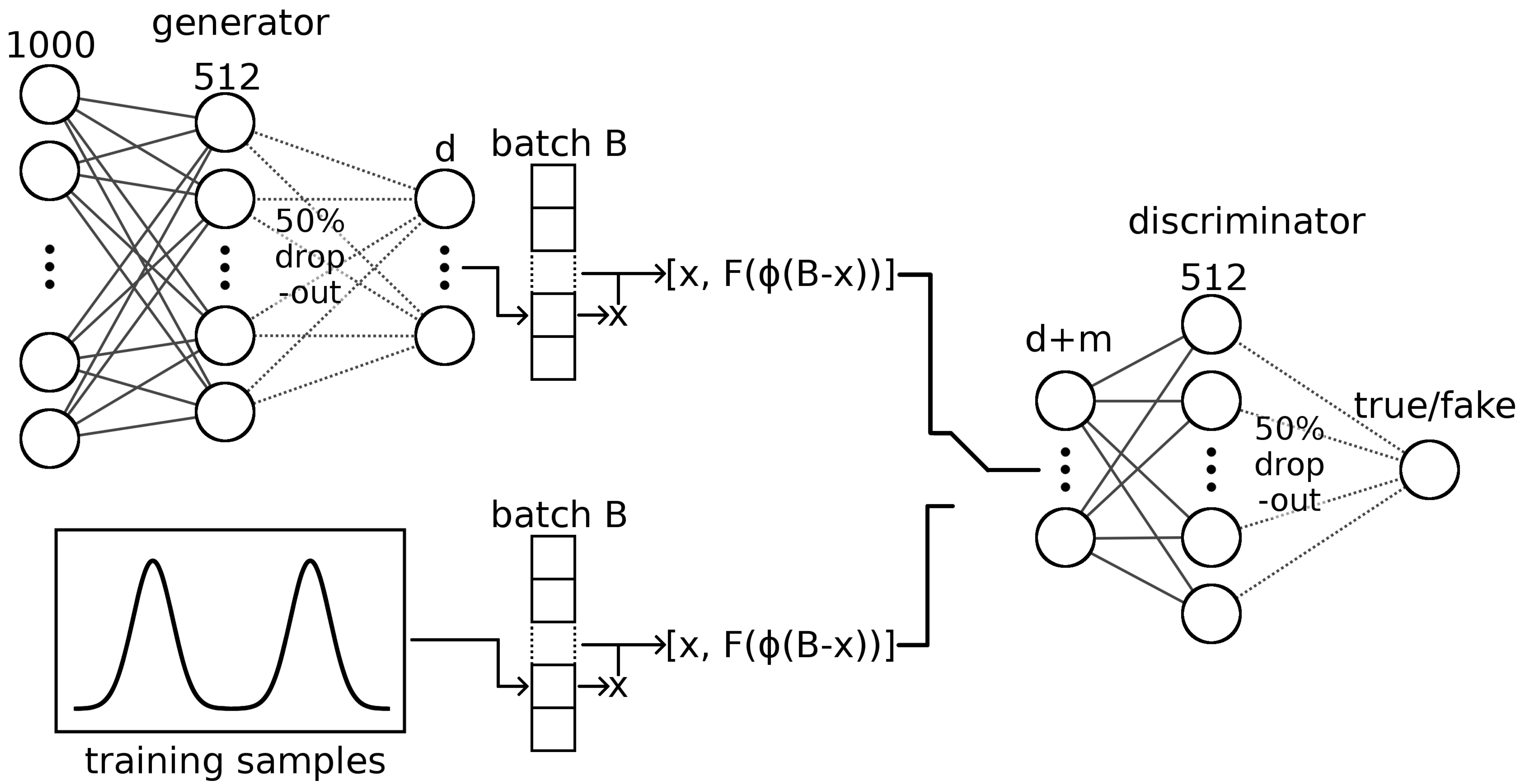
- *"Pearson χ^2-test"*:
  - Introduce equal probability quantiles
  - Generate data
  - Calculate deviation metric

$$\hat{\chi}^2_{n_\text{quant}} = n_\text{quant} \sum_{j=0}^{n_\text{quant}} \left(x_j - \frac{1}{n_\text{quant}}\right)^2$$



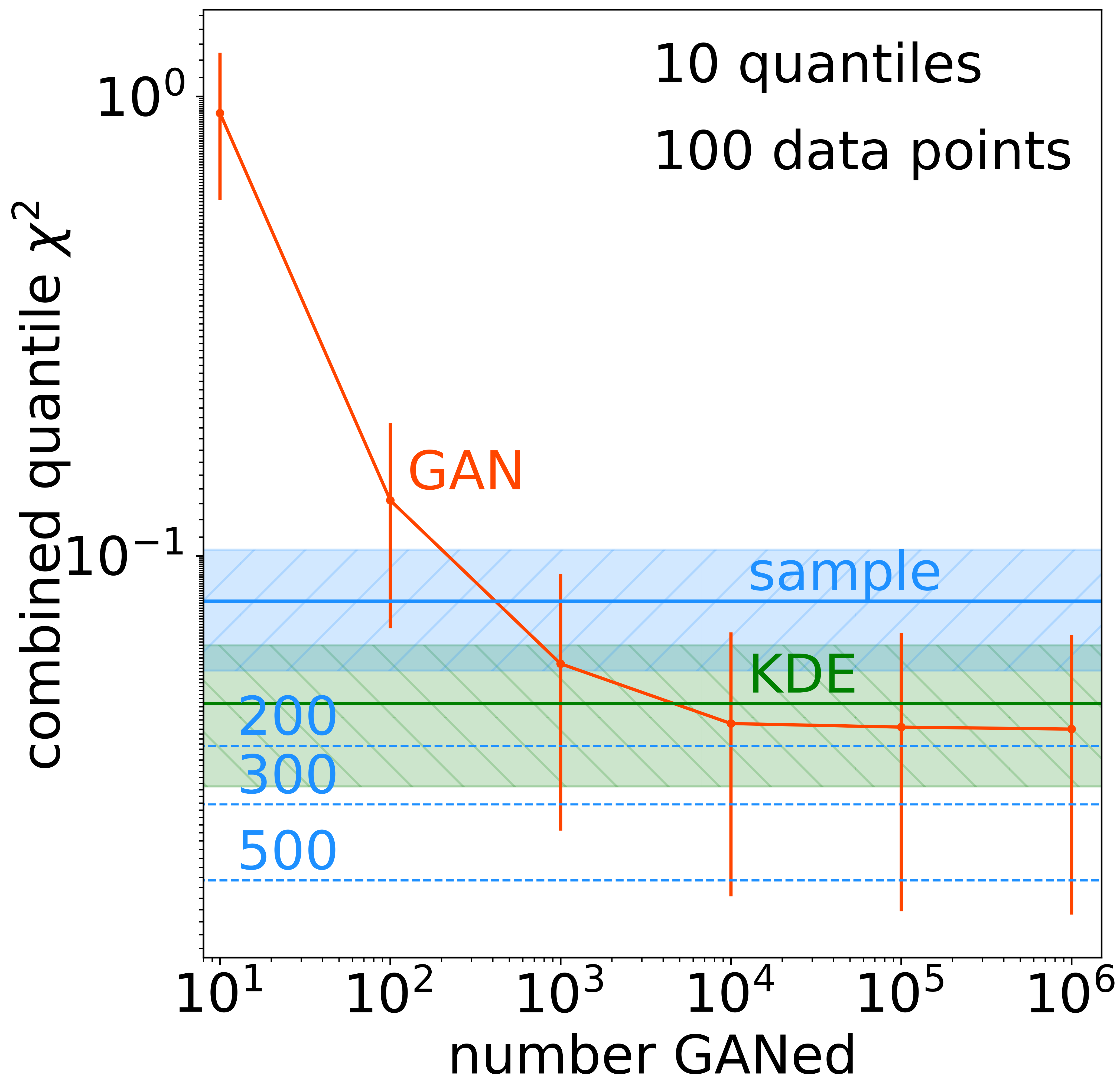
Toy Model: Generative Network

- Train on $n_{\text{data}} = 100$ data points generated from P(x)
- Prone to mode-collapse and overfitting:
  - Dropout
  - Noise augmentation
  - Batch-statistics
- Generate high amounts of data from Network



Toy Model: 1D

- GAN (red) and KDE (green) reach higher value than training data
  - sample: only data points
  - KDE: data + smooth, continuous function
  - GAN: data + smooth, continuous function
- 10.000 GANed points match 180 true ones
- Statistical uncertainty of training data becomes systematic uncertainty of the model



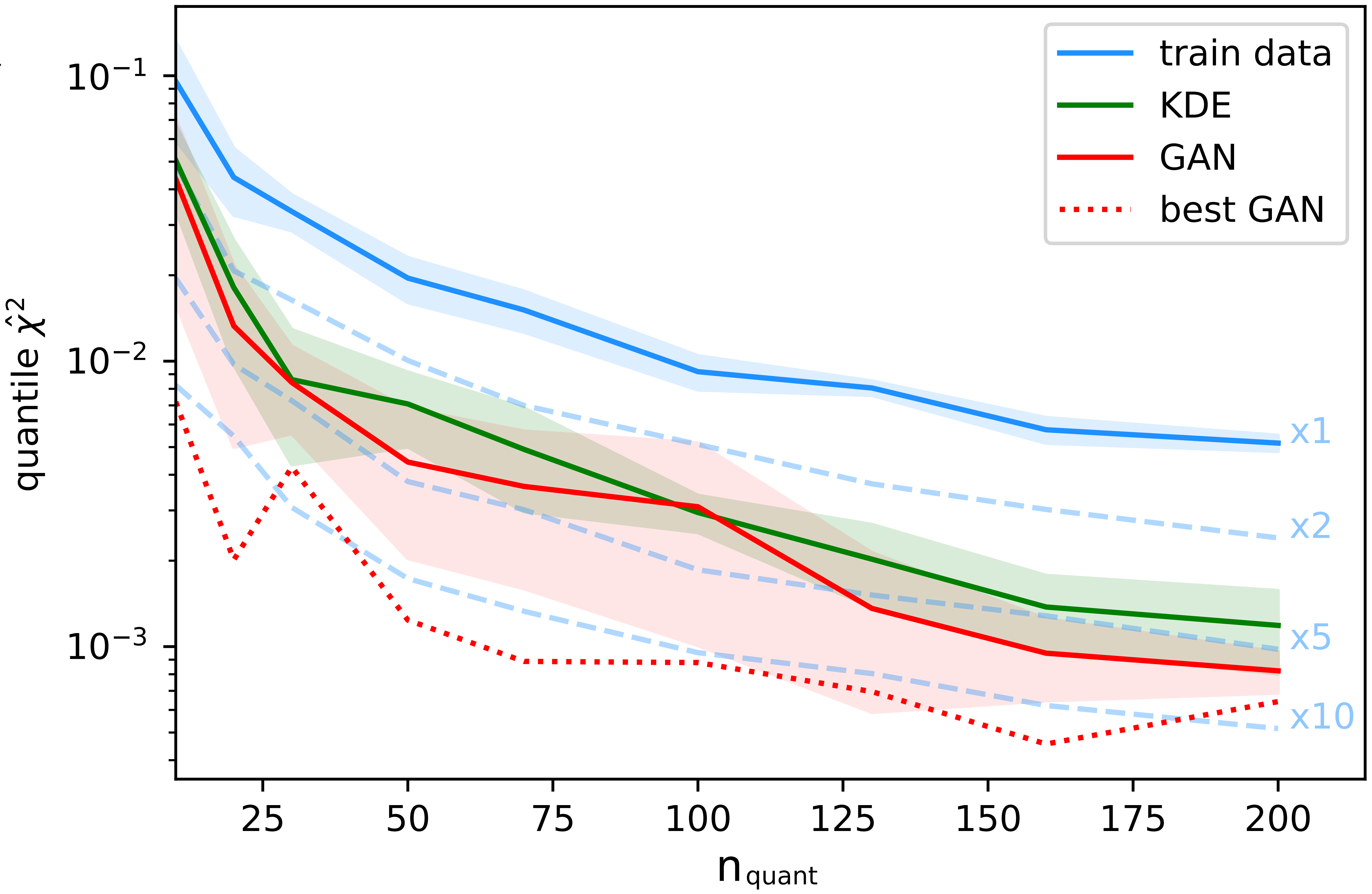
Toy Model: 1D

- large $n_{\mathrm{quant}} \longrightarrow$ global properties of the fit

- **However,** quantile measure breaks down for sparse data

Toy Model: 1D

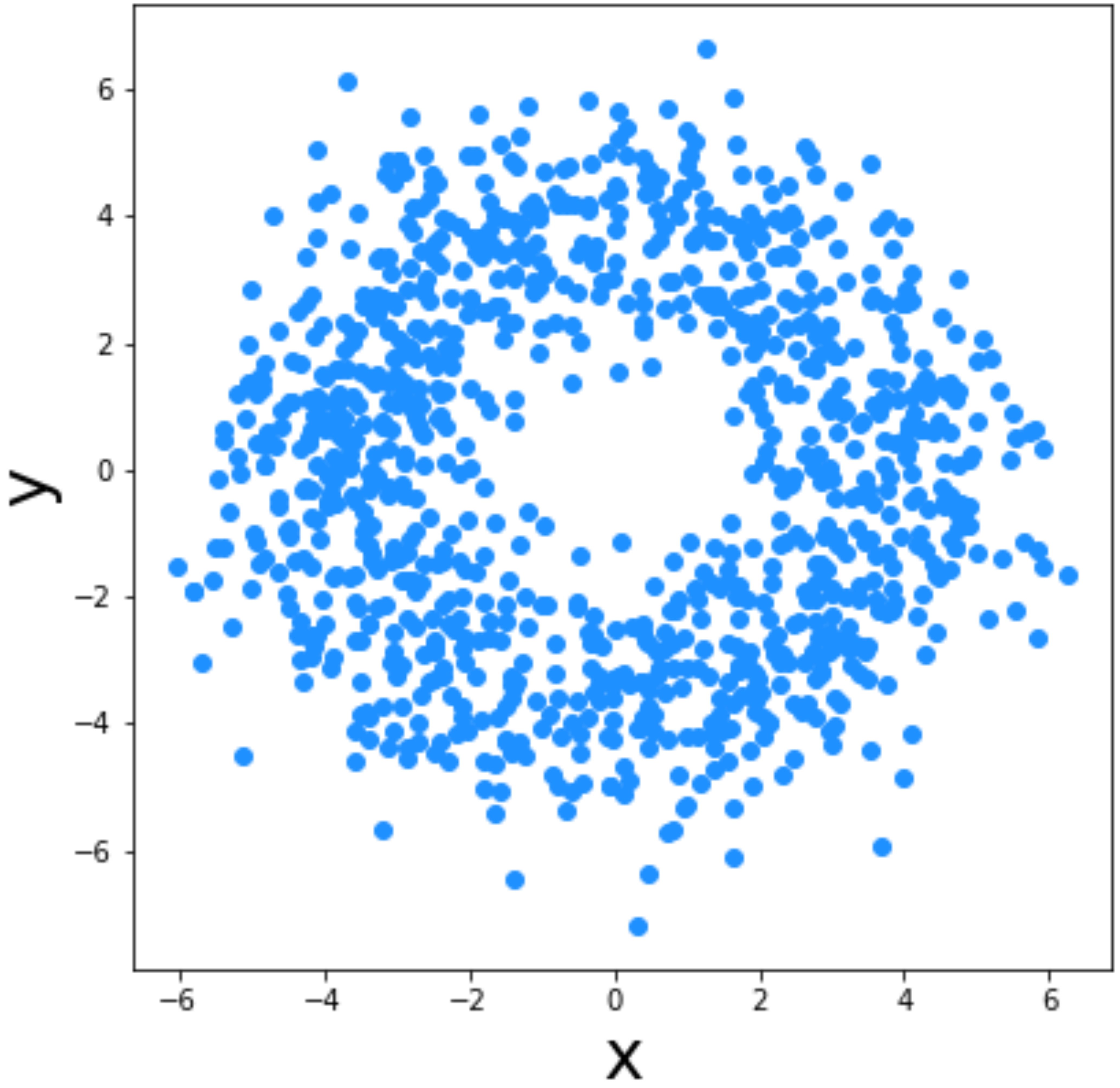
- Examine high n_{quant} and high n_{data}
  - Train on $n_{\text{data}} = n_{\text{quant}}^2$
  - Generate $100 \cdot n_{\text{data}}$
- Examine which data converges to 0 fastest
- GAN amplifies data by a factor ~5



Toy Model: 2D

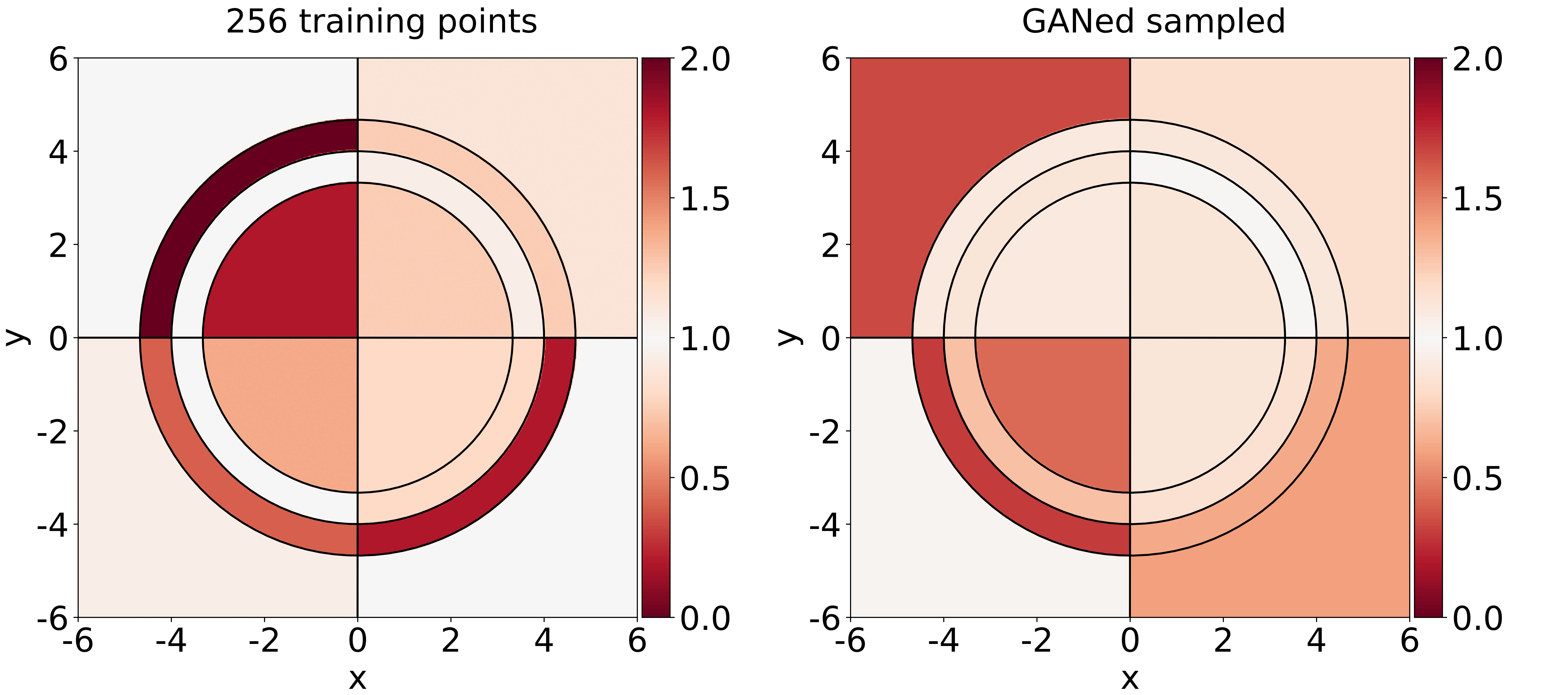
- Ring with gaussian radius
- GAN is trained on cartesian coordinates
- Quantiles are calculated on polar coordinates

GAN has to learn correlations



Toy Model: 2D

- Quantiles in radial and angular direction

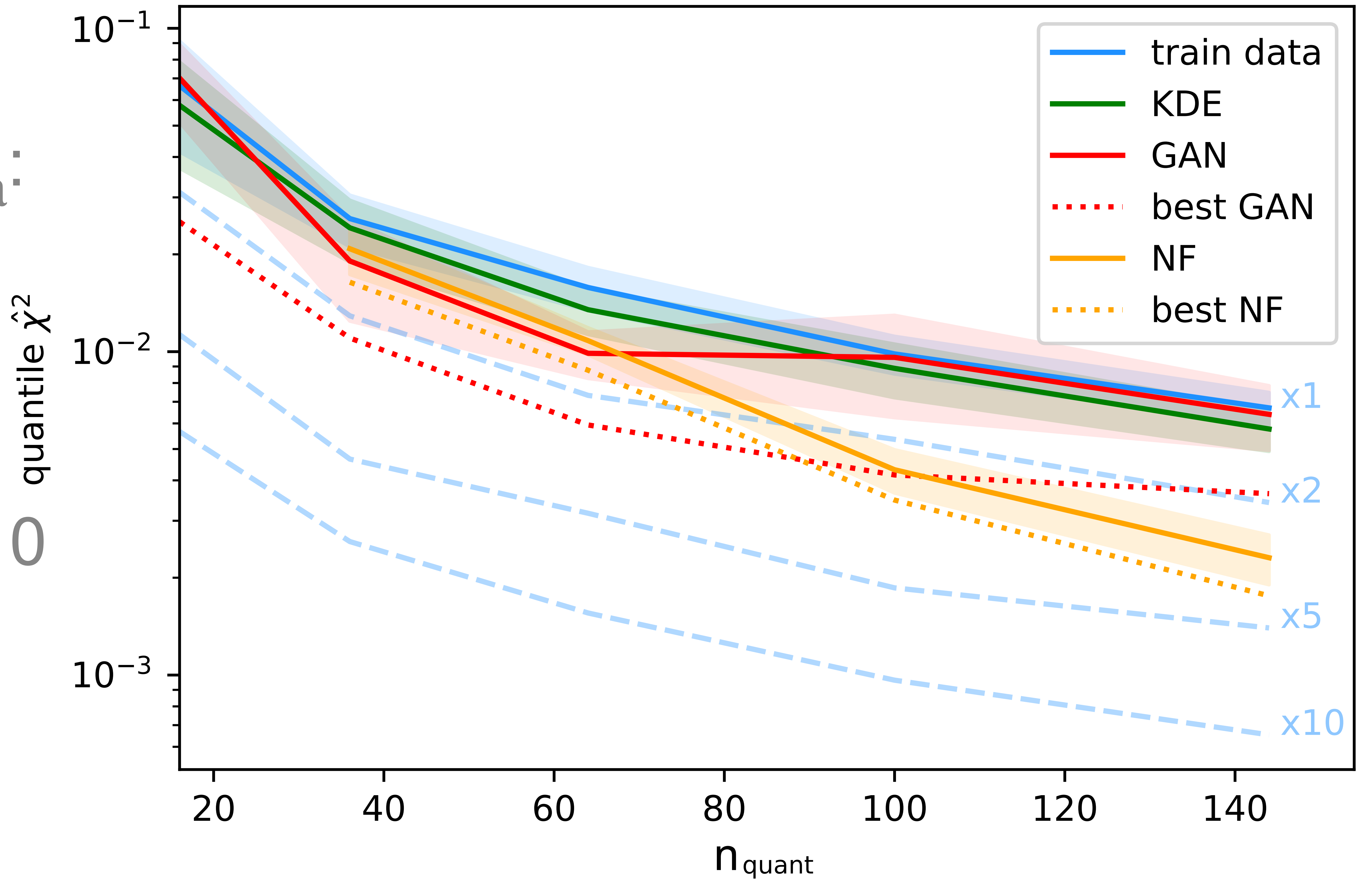


Toy Model: 2D

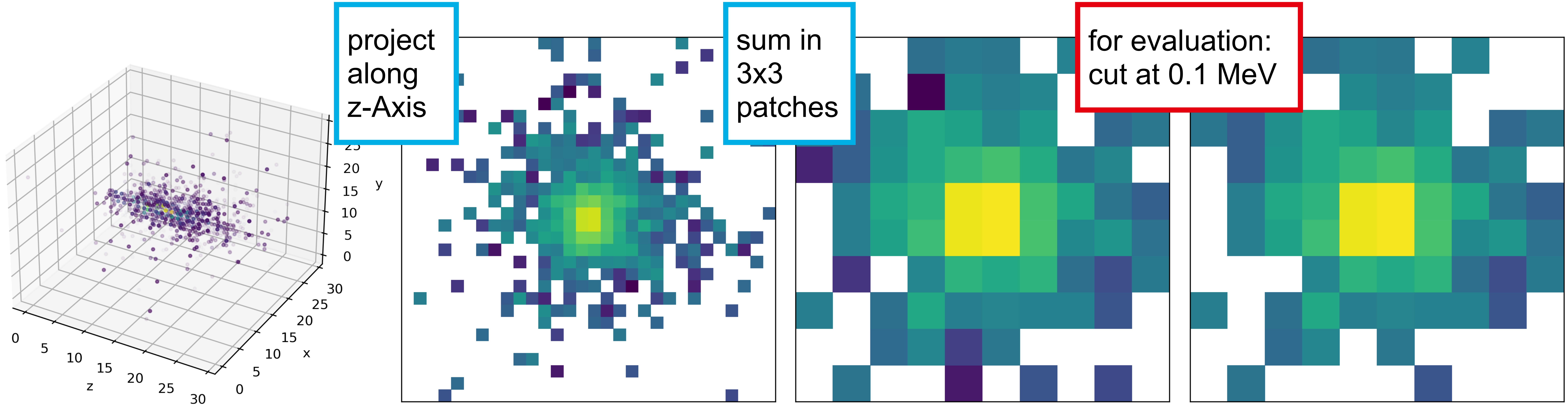
DASHH

Do the same thing again:

- Examine high n_{quant} and high n_{data}:
  - Train on n_{quant}^2 data points
  - Generate $100 \cdot n_{\text{data}}$
- Examine which data converges to 0 (fastest)



Calorimeter Simulations: Data



- 269k photon showers at 50 GeV in International Large Detector [1]

Image-shaped data

Unknown true distribution, limited data

Harder learning task → training on multiple training set sizes unfeasible

Calorimeter Simulations: Architecture

- Change to location-aware VAE-GAN architecture → 2202.07352 [hep-ph]

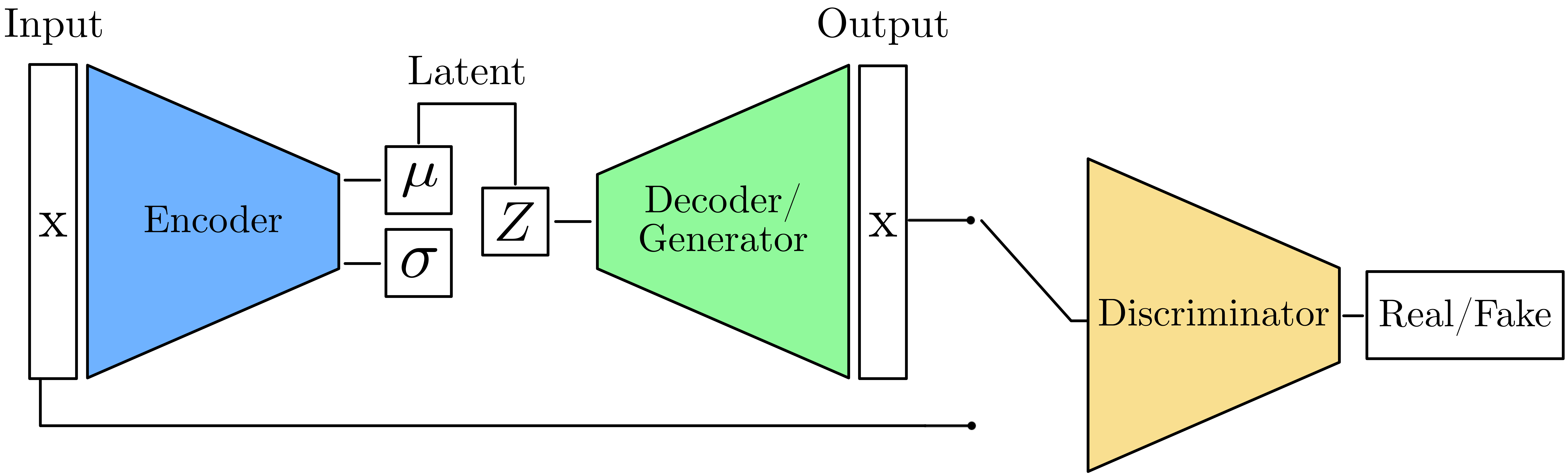
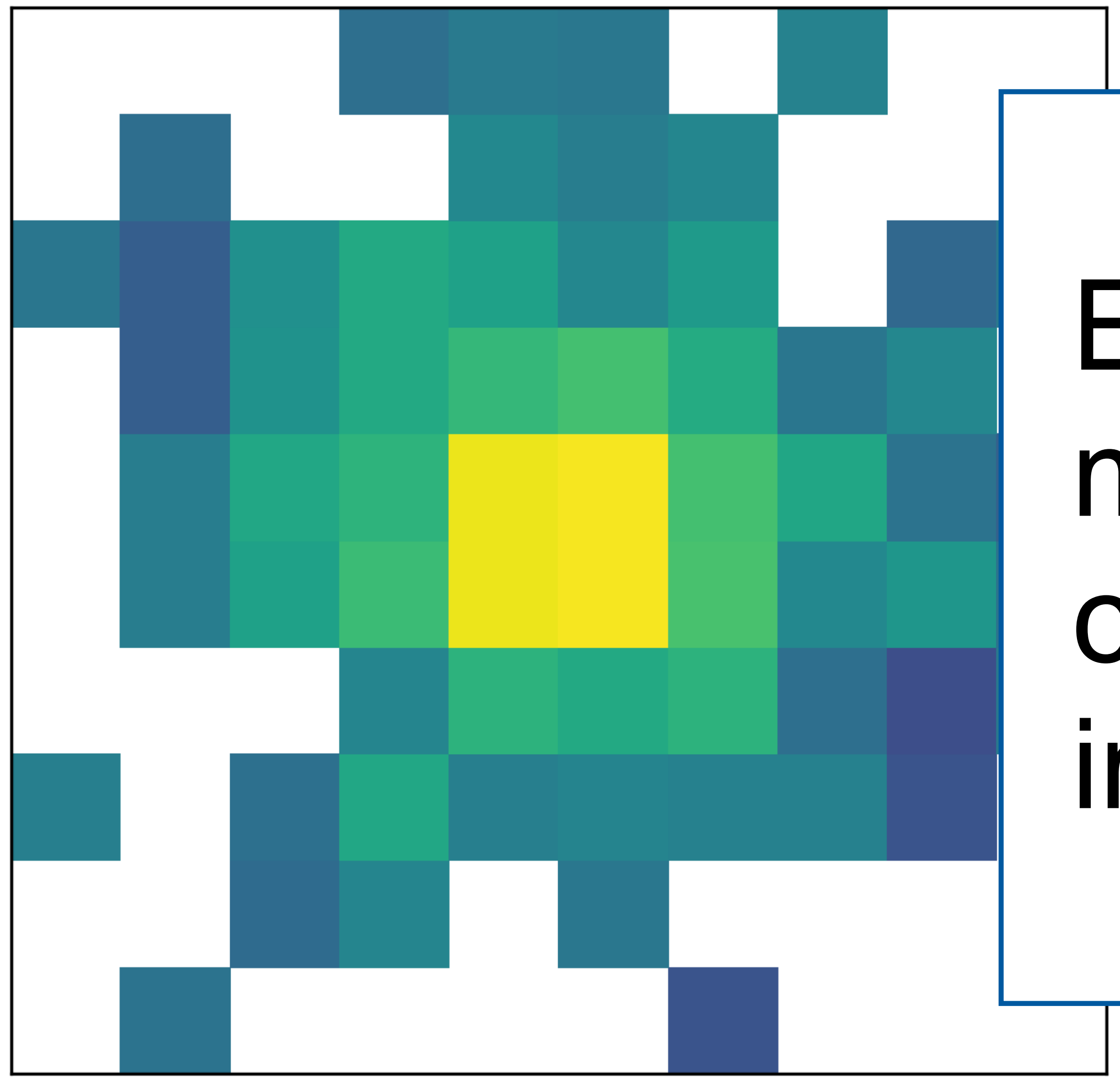


Image-shaped data

Unknown true distribution, limited data

Harder learning task → training on multiple training set sizes unfeasible

Calorimeter Simulations: Setup



Evaluate 1D metrics, calculated on images

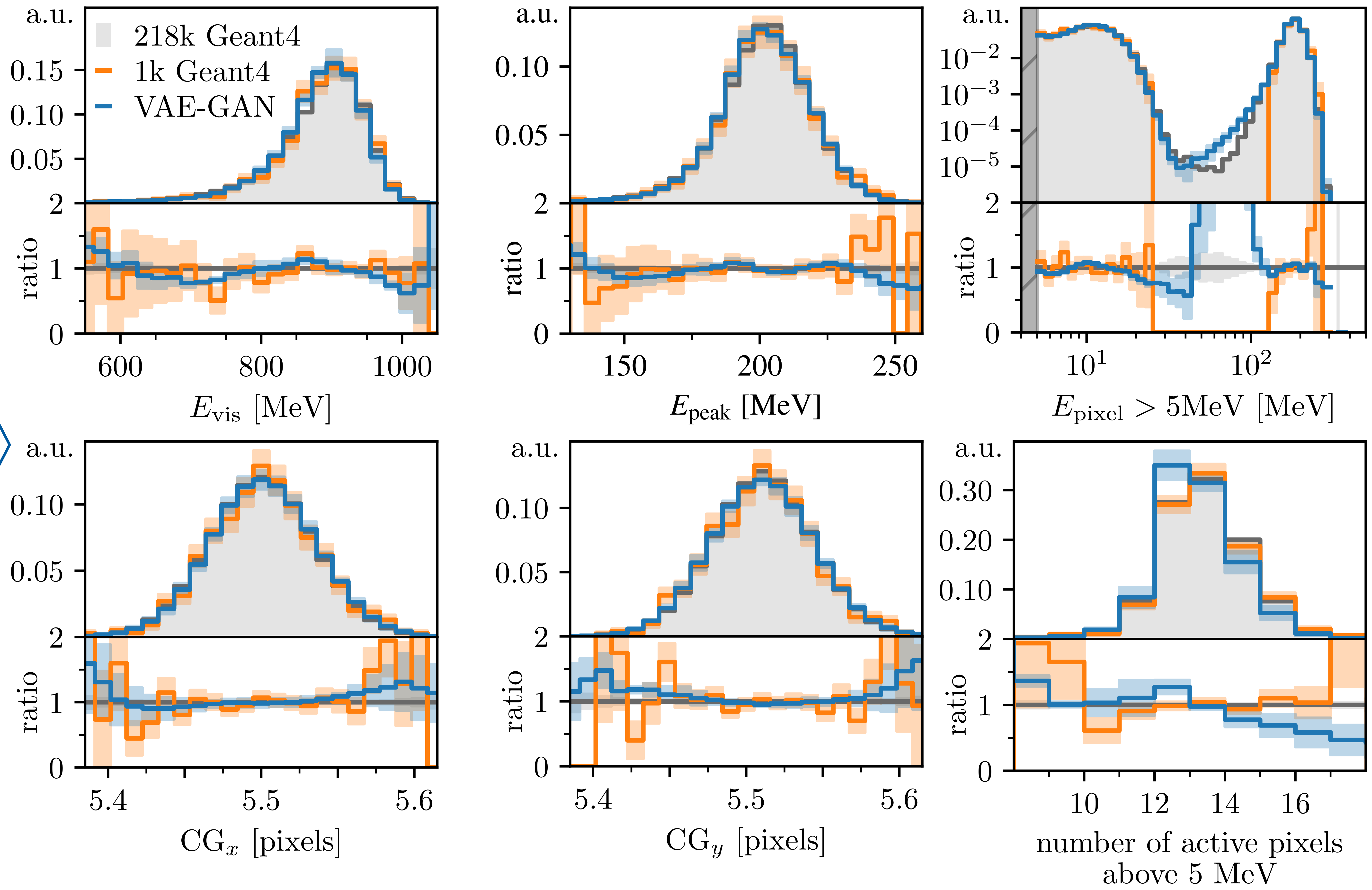


Image-shaped data

Unknown true distribution, limited data

Harder learning task → training on multiple training set sizes unfeasible

Calorimeter Simulations: Setup

- Split into **218k validation data points** and **50k evaluation data points**
- Generate quantiles by dividing the validation set into equally populated parts

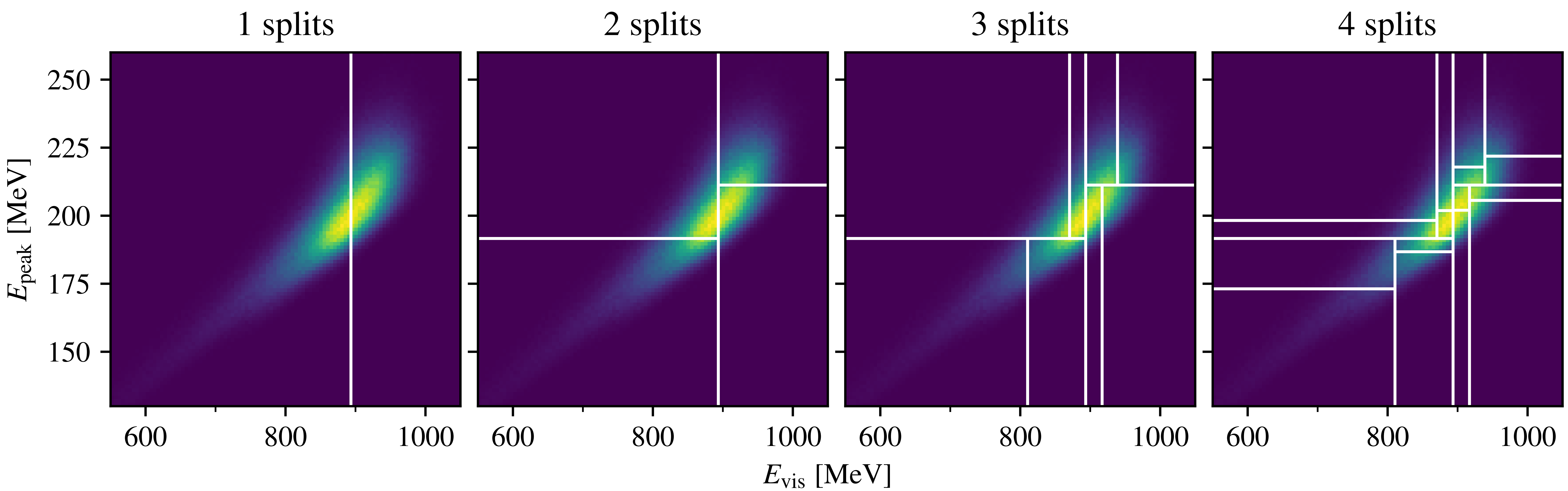


Image-shaped data

Unknown true distribution, limited data

Harder learning task → training on multiple training set sizes unfeasible

Calorimeter Simulations: Setup

Calculate deviation metric $\overline{D_{\mathrm{JS}}}(g\,||\,p) = \frac{1}{2} \sum_{Q_i \in \mathbf{Q}} \left(g_i \log \frac{g_i}{\frac{1}{2}(g_i + p_i)} + p_i \log \frac{p_i}{\frac{1}{2}(g_i + p_i)} \right).$

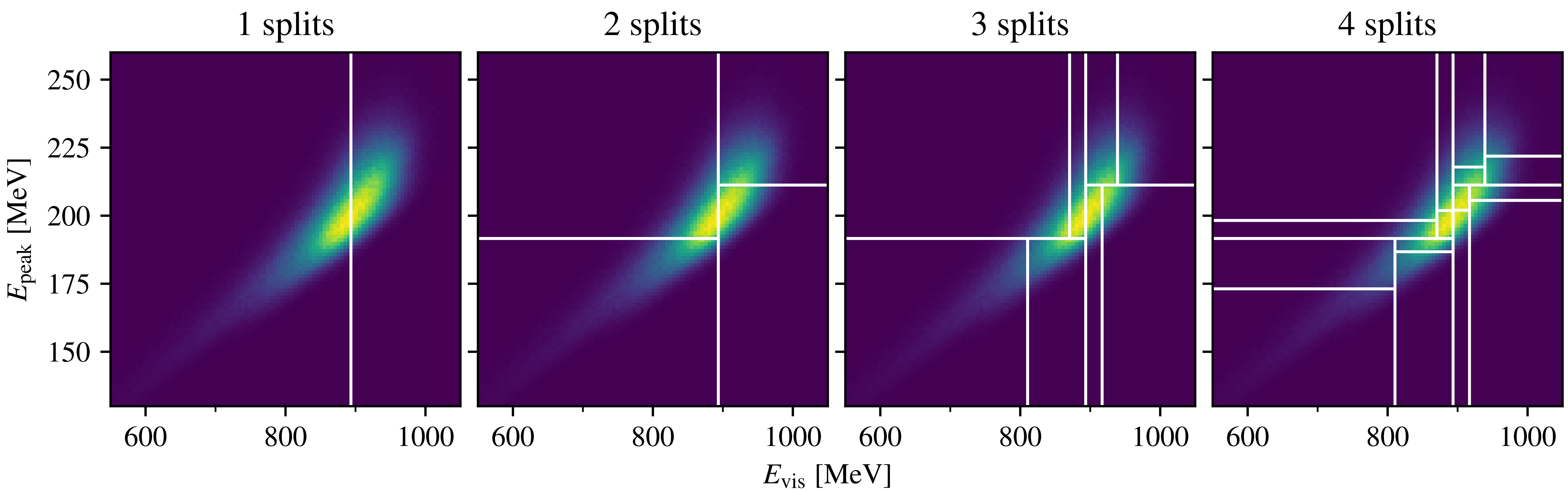


Image-shaped data

Unknown true distribution, limited data

Harder learning task → training on multiple training set sizes unfeasible

Calorimeter Simulations: Results

- Evaluate for fixed training (1k) and evaluation set sizes (5k, 10k, 50k)
- Use less than $n_{\text{data}}/10$ bins

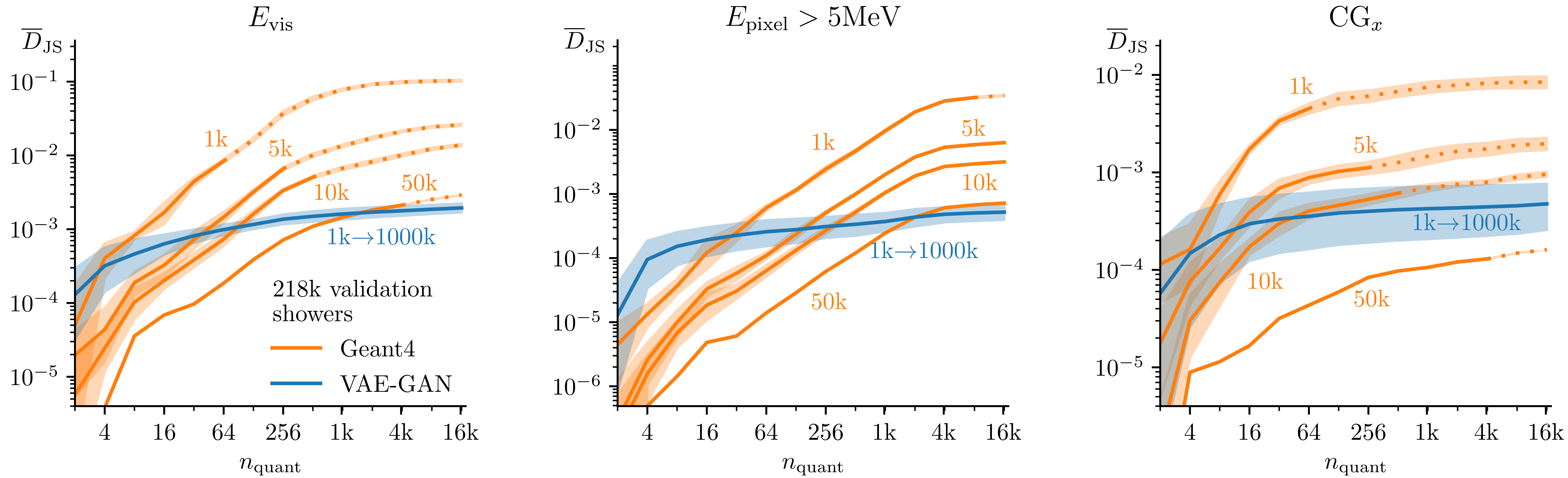


Image-shaped data

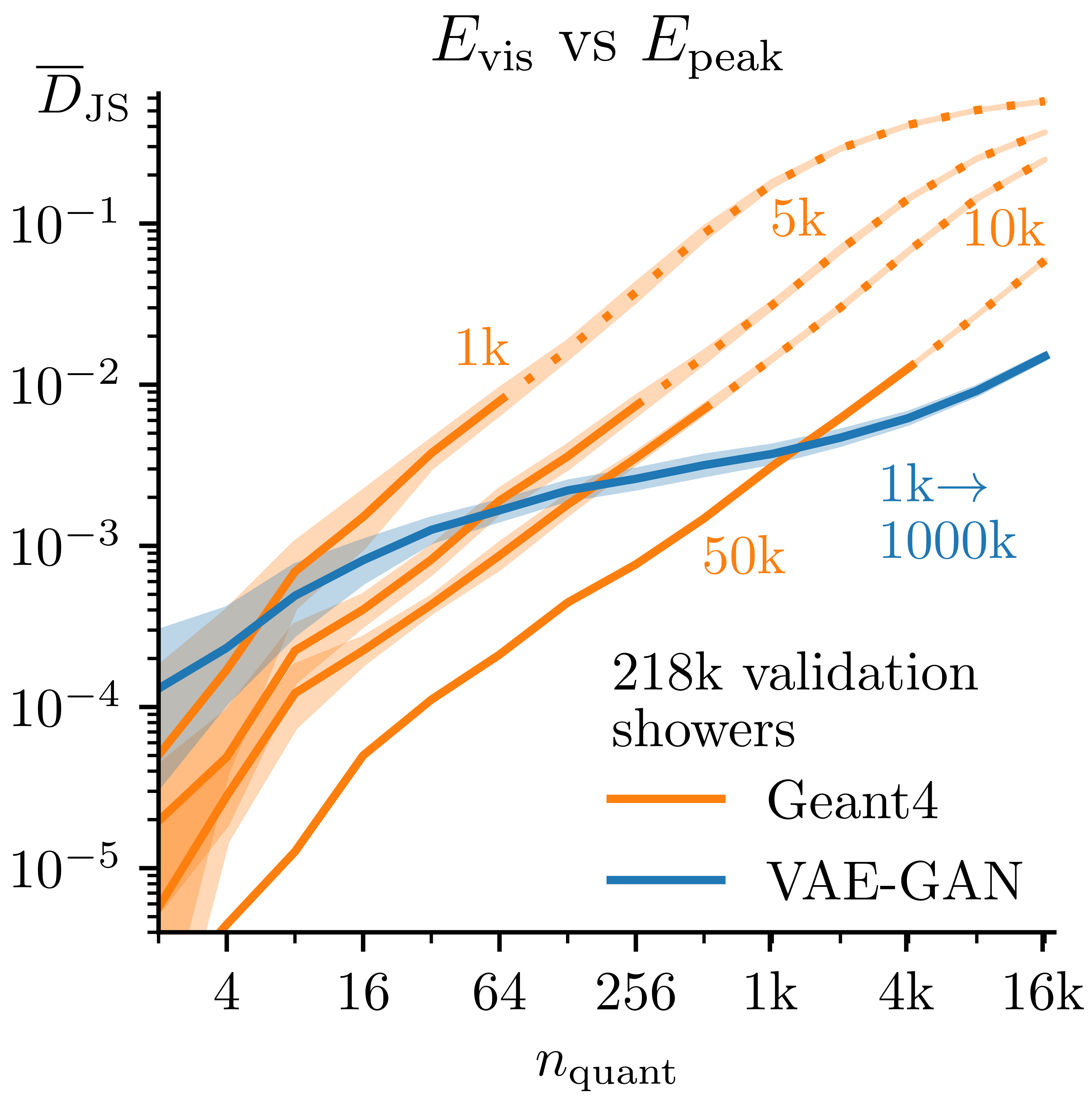
Unknown true distribution, limited data

Harder learning task → training on multiple training set sizes unfeasible

Calorimeter Simulations: Results

- Evaluate for fixed training (1k) and evaluation set sizes (5k, 10k, 50k)
- Use less than $n_{\text{data}}/10$ bins

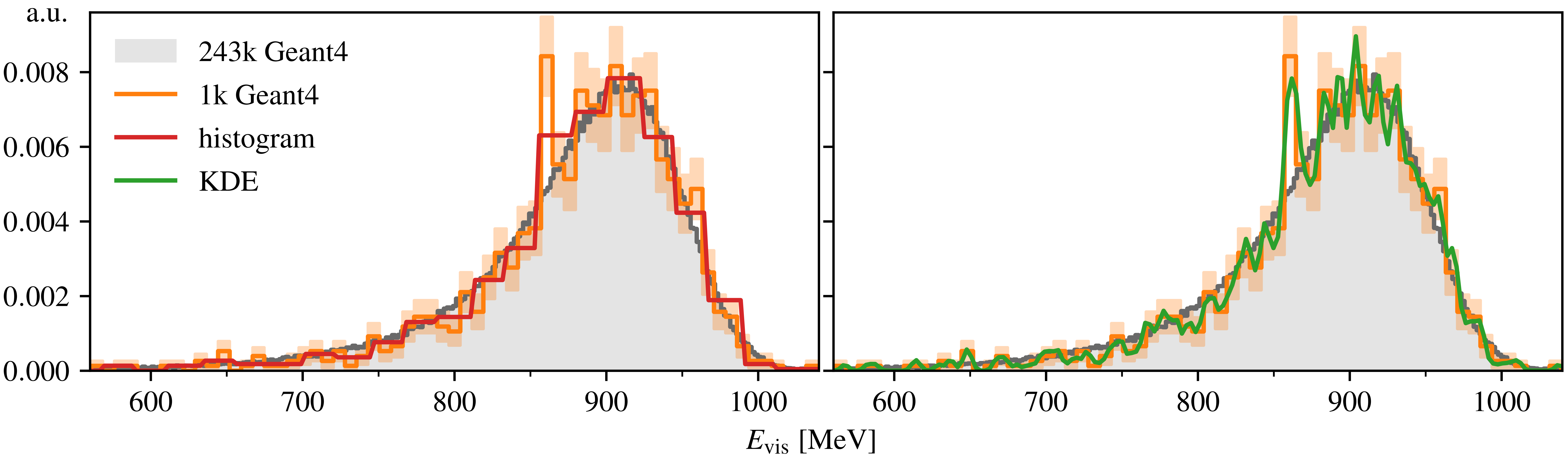
- High-scale features: limited by amount of training data
- Low-scale features: GAN estimation can not be matched by adding more data



Calorimeter Simulations: Results

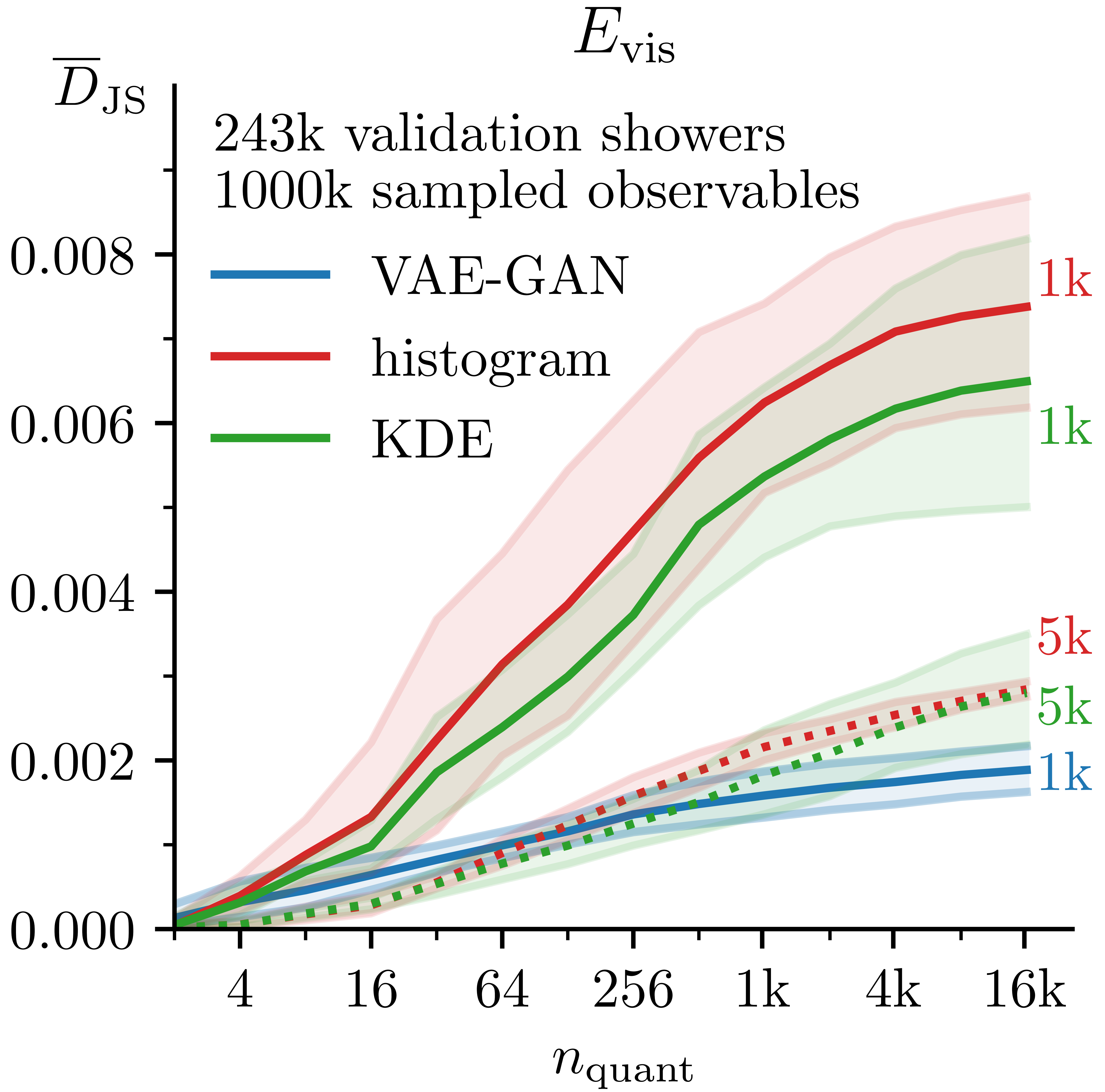
How good is the density estimation actually?

- Compare to KDE and histogram estimators (maximizing log-likelihood of cross-validation sets)



Calorimeter Simulations: Results

- Generate 10^6 samples from every density estimator



- GAN outperforms standard density estimators

Conclusion

- What about # samples? How many new points should we generate from a generative model?
  - Depends on GAN setup and problem

- For high-scale observables (e.g. *mean, standard deviation, low moments*) generative network limited to the amount of training data
- For a smooth interpolation (e.g. *segments of the distribution, integrated quantities*) a generative networks outperform even higher numbers of data

References

DASHH

[0]: P. Calafiura, J. Catmore, D. Costanzo, and A. Di Girolamo, “ATLAS HLLHC Computing Conceptual Design Report,” CERN, Geneva, Tech. Rep., Sep 2020. [Online]. Available: https://cds.cern.ch/record/2729668

[1]: ILD Concept Group, H. Abramowicz et al., *International Large Detector: Interim Design Report*, 3, 2020.

[2]: L. de Oliveira, M. Paganini, and B. Nachman, “Learning particle physics by example: Location-aware generative adversarial networks for physics synthesis,” *Computing and Software for Big Science*, vol. 1, no. 1, Sep 2017. [Online]. Available: http://dx.doi.org/10.1007/s4178101700046

[3]: M. Paganini, L. de Oliveira, and B. Nachman, “Calogan: Simulating 3d high energy particle showers in multilayer electromagnetic calorimeters with generative adversarial networks,” *Physical Review D*, vol. 97, no. 1, Jan 2018. [Online]. Available: http://dx.doi.org/10.1103/PhysRevD.97.014021

[4]: A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, “Autoencoding beyond pixels using a learned similarity metric,” in *Proceedings of the 33rd International Conference on International Conference on Machine Learning Volume 48*. JMLR.org, 2016, p. 1558–1566.