Learning Reliable Amplitude Surrogates

Nina Elmer

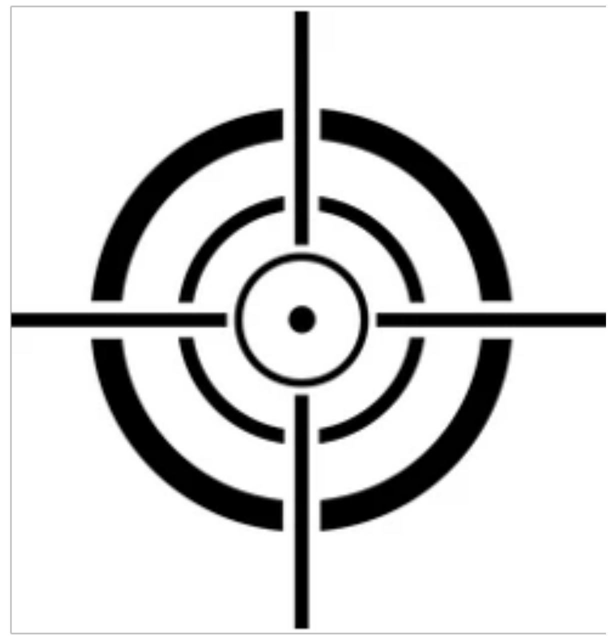
Zurich Phenomenology Workshop 2026

arXiv:2509.00155 and arXiv:2412.12069

with H. Bahl, L. Favaro, R. Winterhalder and T. Plehn

UNIVERSITY OF CAMBRIDGE

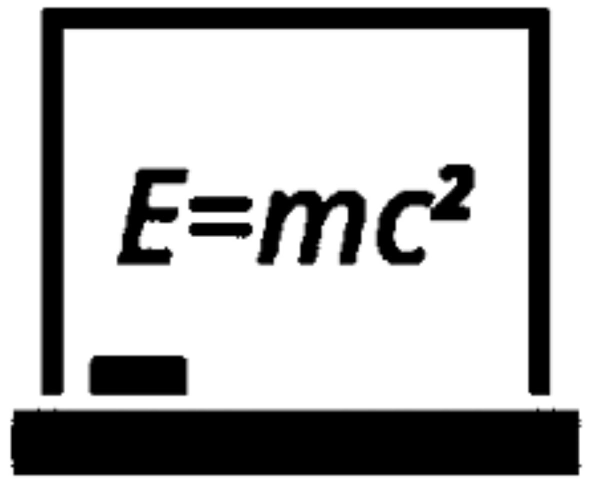
The four horsemen of physics



precision

control

speed



physics knowledge

The four horsemen of physics

Exact physics computations:

✓ Precision

✓ Full control of uncertainties

✓ Physics knowledge

✗ Computational speed

ML approach:

The four horsemen of physics

Exact physics computations:

- ✓ Precision
- ✓ Full control of uncertainties
- ✓ Physics knowledge
- ✗ Computational speed

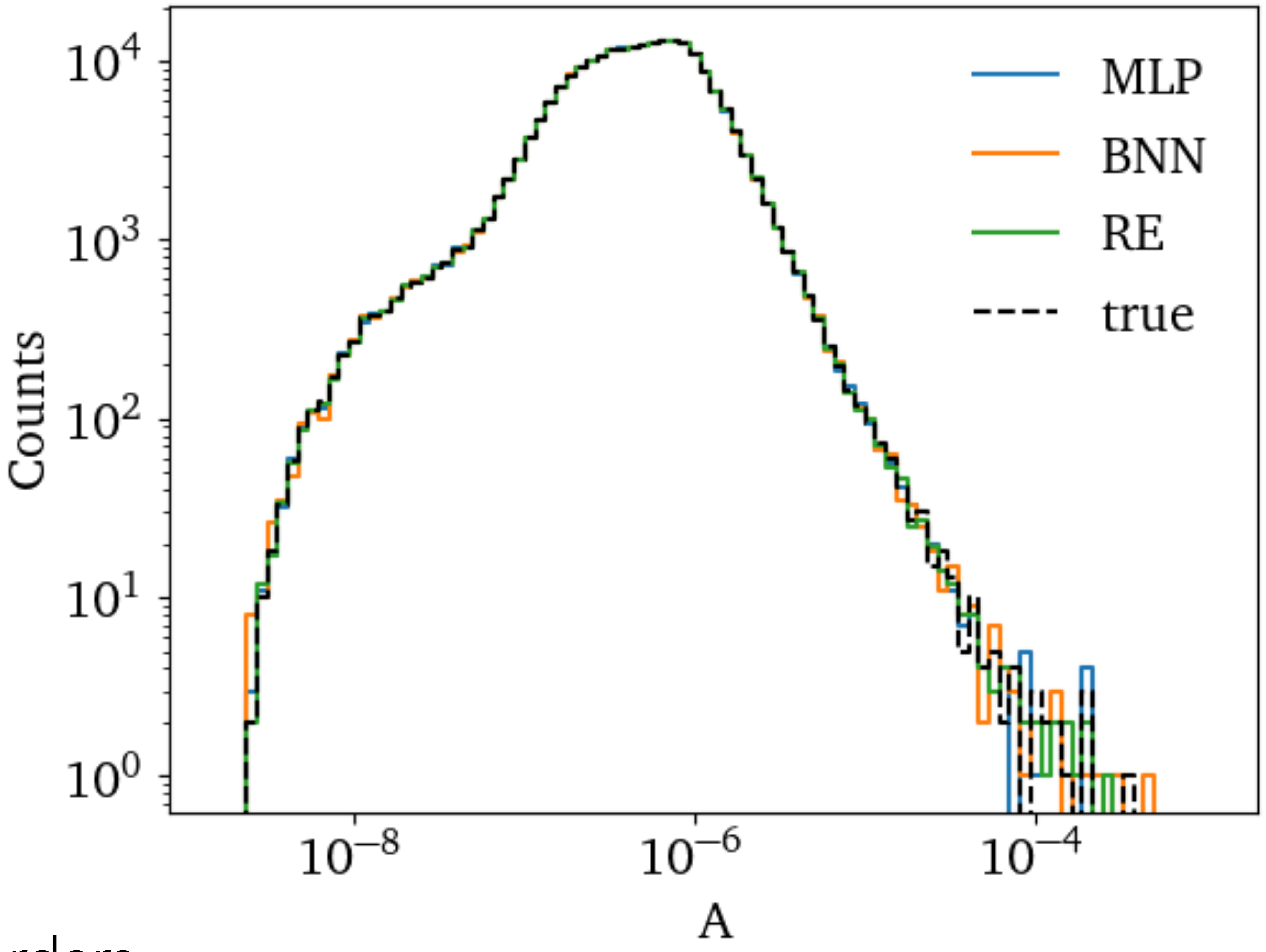
ML approach:

- ✓ Speed
  →(arXiv:2509.07068, 2311.01548 (MadNIS))
- ~ Precision ✓
  → (arXiv:2412.12069 and 2601.00950)
- ~ Physics knowledge ✓
  → Include Invariants, L-GATr
  (arXiv: 2411.00446 and 2412.12069)
- ~ Control of uncertainties → This presentation

Scattering amplitudes as data source

- Data set: 1.1M points of $gg \rightarrow \gamma\gamma g$
- Loop-induced process
- Higher order amplitudes hard to solve analytically

➡Use one-loop for testing, then extend to higher orders



Regression via variational inference

- Fit set of Amplitudes $A(x)$, with training data: $\{x, A(x)\}$
- Prediction in regression:

$$A(x) \equiv \langle A \rangle = \int dA\, A\, p(A\,|\,x) = \int d\theta\, q(\theta)\, \overline{A}(x,\theta) \quad \text{with} \quad p(A\,|\,x) = \int d\theta\, p(A\,|\,\theta, x)\, p(\theta\,|\,x)$$

prior: $q(\theta)$

network output: $\overline{A}(x,\theta)$

Regression via variational inference

- Fit set of Amplitudes $A(x)$, with training data: $\{x, A(x)\}$
- Prediction in regression:

prior

$$A(x) \equiv \langle A \rangle = \int dA\, A\, p(A|x) = \int d\theta\, q(\theta)\, \overline{A}(x,\theta) \quad \text{with} \quad p(A|x) = \int d\theta\, p(A|\theta,x)\, p(\theta|x)$$

network output

$$\sigma_{\text{tot}}^2(x) \equiv \langle (A - \langle A \rangle)^2 \rangle = \int dA\, \left(A - \langle A \rangle\right)^2 p(A|x) = \int d\theta\, q(\theta) \left(\overline{A^2}(x,\theta) - \overline{A}(x,\theta)^2 \right) + \int d\theta\, q(\theta) \left(\overline{A}(x,\theta) - \langle A \rangle \right)^2$$

Regression via variational inference

- Fit set of Amplitudes $A(x)$, with training data: $\{x, A(x)\}$
- Prediction in regression:

prior

$$A(x) \equiv \langle A \rangle = \int \mathrm{d}A\, A\, p(A\,|\,x) = \int \mathrm{d}\theta\, q(\theta)\, \overline{A}(x,\theta) \quad \text{with} \quad p(A\,|\,x) = \int \mathrm{d}\theta\, p(A\,|\,\theta, x)\, p(\theta\,|\,x)$$

network output

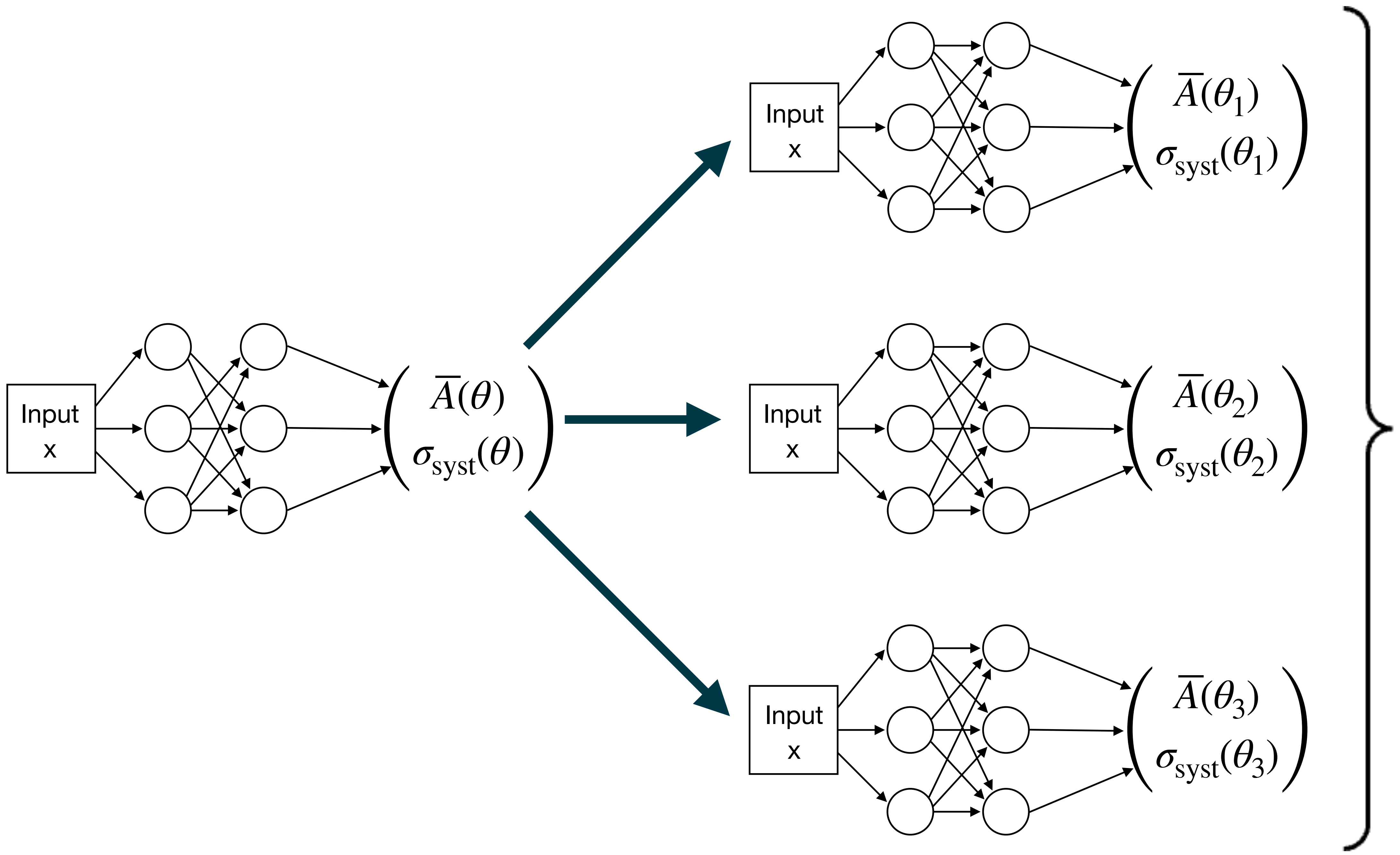
$$\sigma^2_{\text{tot}}(x) \equiv \langle (A - \langle A \rangle)^2 \rangle = \int \mathrm{d}A\, \big(A - \langle A \rangle\big)^2\, p(A\,|\,x) = \int \mathrm{d}\theta\, q(\theta) \Big(\overline{A^2}(x,\theta) - \overline{A}(x,\theta)^2 \Big) + \int \mathrm{d}\theta\, q(\theta) \big(\overline{A}(x,\theta) - \langle A \rangle\big)^2$$

$$\equiv \sigma^2_\theta(x)$$

$$\mathscr{L}_{\text{heteroscedastic}} = \sum_i \frac{|f(x_i) - f_\theta(x_i)|^2}{2\sigma_\theta(x_i)^2} + \log \sigma_\theta(x_i) + \dots$$

Ensemble of networks

Ensemble of networks



Output

$$\langle A \rangle = \frac{1}{N}\sum_{i}^{N} \overline{A}\left(\theta_i\right)$$

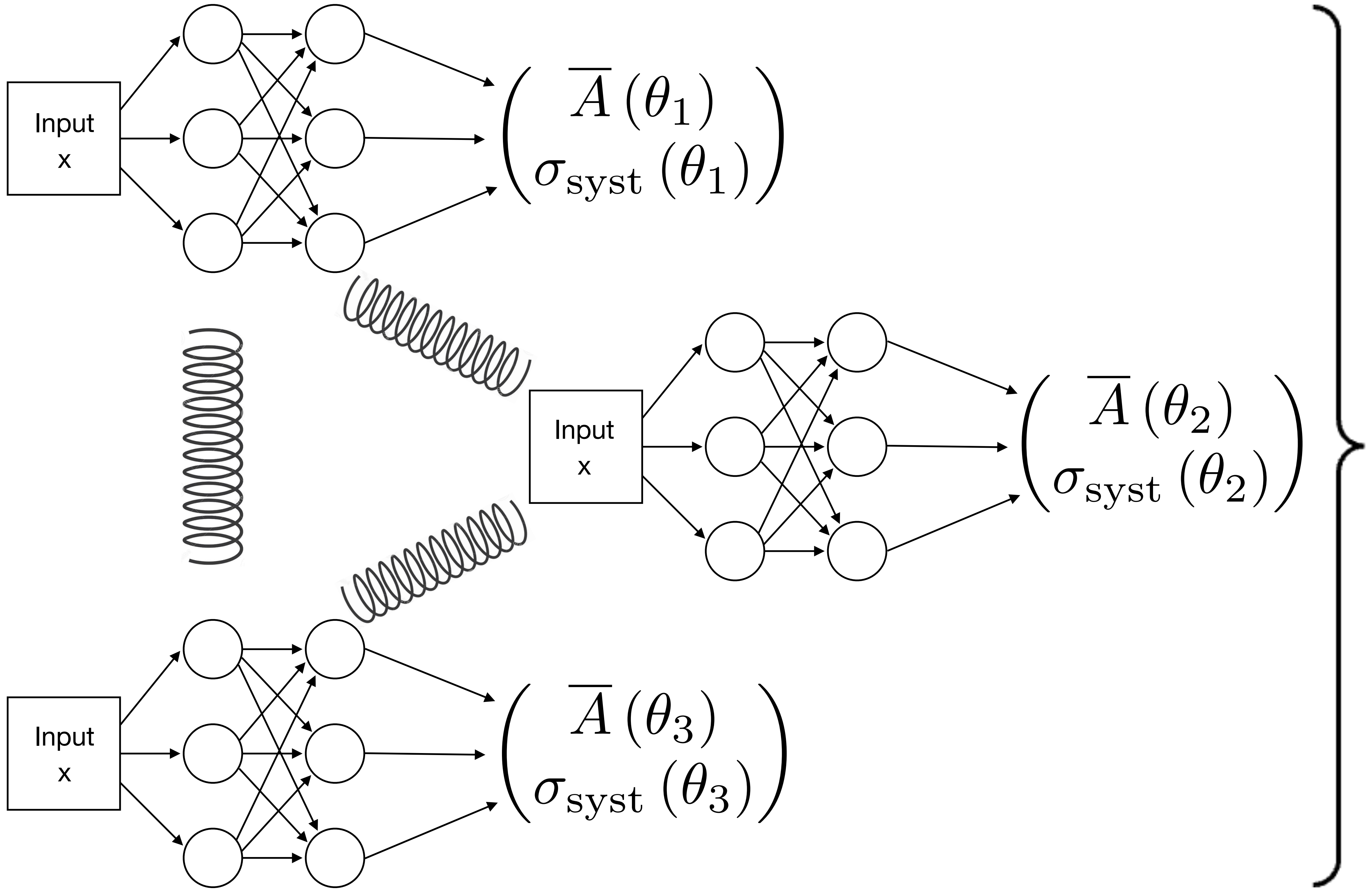
$$\sigma^2_{\text{syst}} = \frac{1}{N}\sum_{i}^{N} \sigma^2_{\text{syst}}\left(\theta_i\right)$$

$$\sigma^2_{\text{stat}} = \frac{1}{N}\sum_{i}^{N} \left(\langle A \rangle - \overline{A}\left(\theta_i\right)\right)^2$$

- Ensemble members **trained simultaneously**
- Systematic uncertainty from **heteroscedastic loss** part
- Statsistical uncertainty from ensemble nature

Repulsive ensemble of networks

Ensemble of networks | **Output**



$$\langle A \rangle = \frac{1}{N} \sum_{i}^{N} \overline{A}(\theta_i)$$

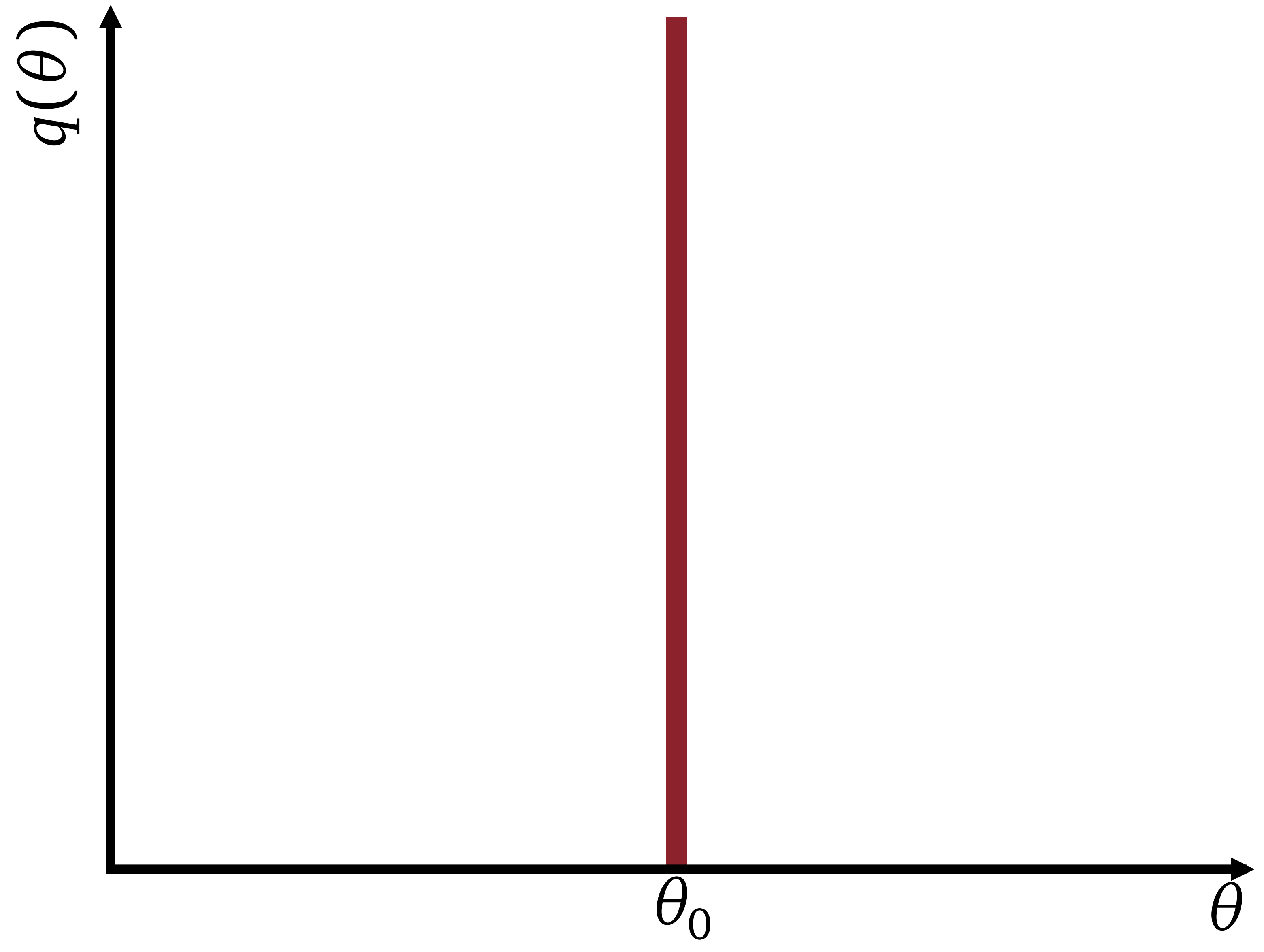
$$\sigma_{\text{syst}}^2 = \frac{1}{N} \sum_{i}^{N} \sigma_{\text{syst}}^2(\theta_i)$$

$$\sigma_{\text{stat}}^2 = \frac{1}{N} \sum_{i}^{N} \left(\langle A \rangle - \overline{A}(\theta_i) \right)^2$$

- Repulsive term: Cover full posterior distribution
- Ensemble members **trained simultaneously**

Bayesian neural network (BNN)

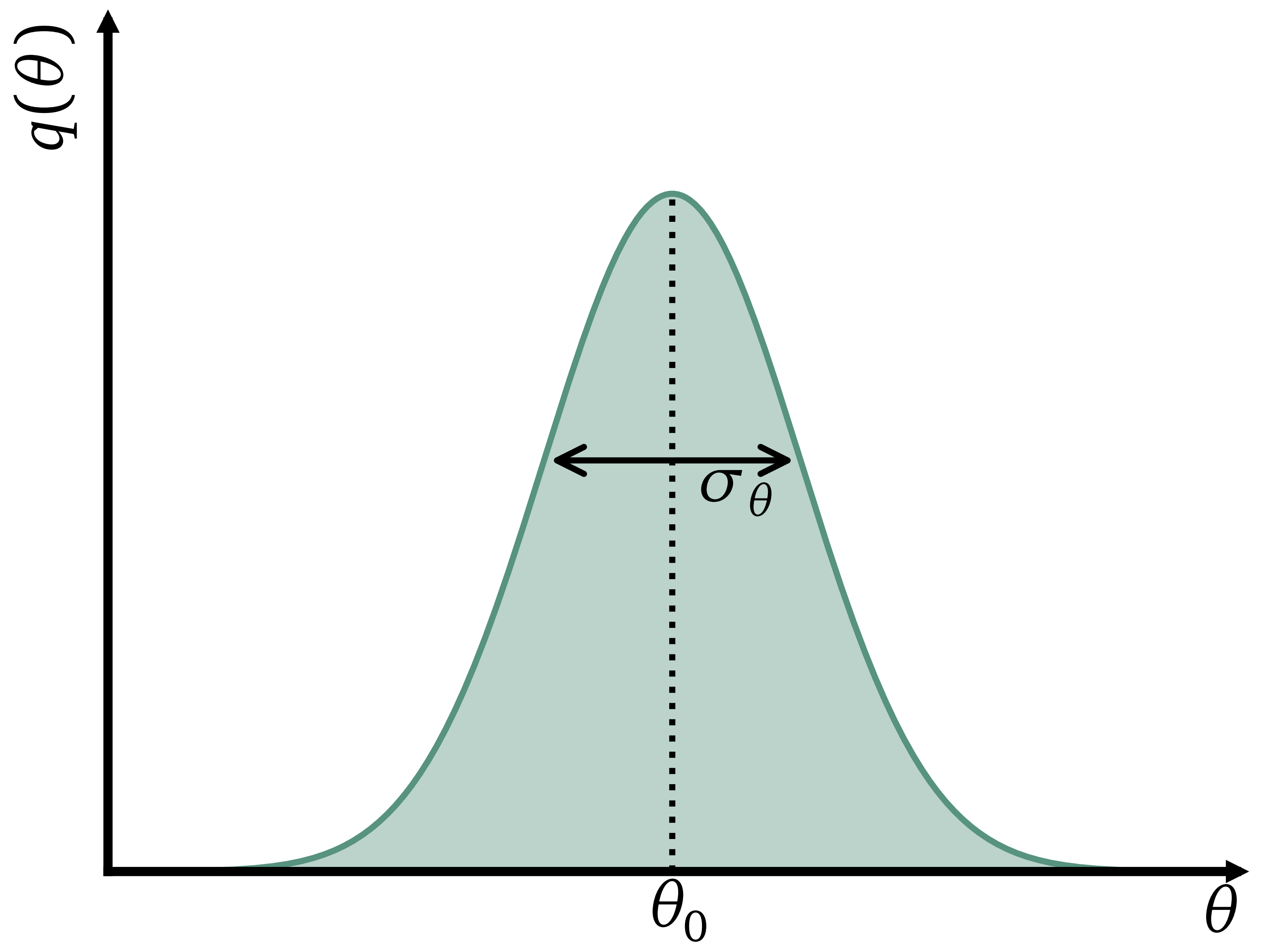
Neural network



Network weights are deterministic

$$\theta = \theta_0$$

Bayesian neural network

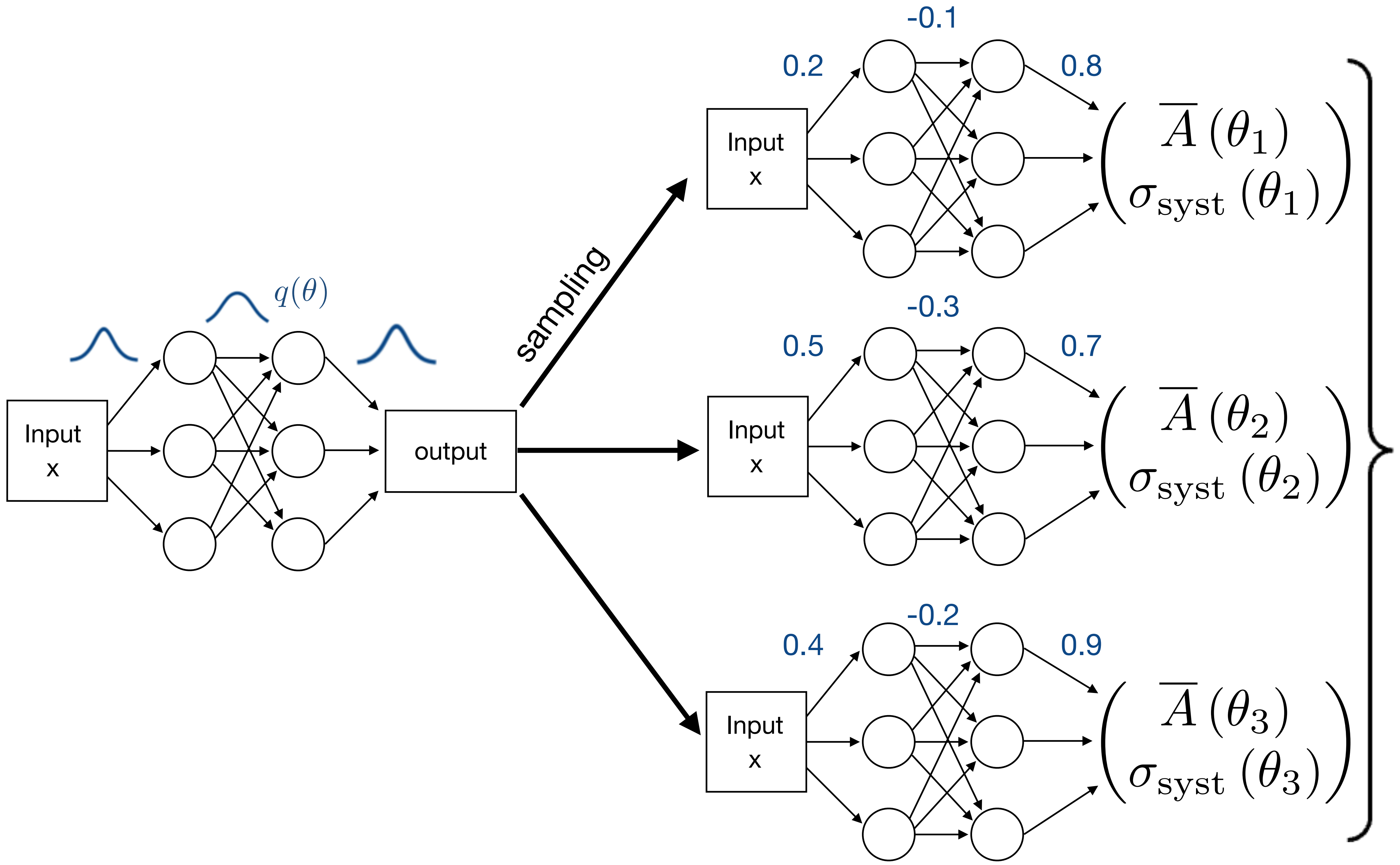


Network weights are drawn from distribution

$$\theta \sim q(\theta)$$

Bayesian neural network (BNN)

BNN **Ensemble of networks** **Output**



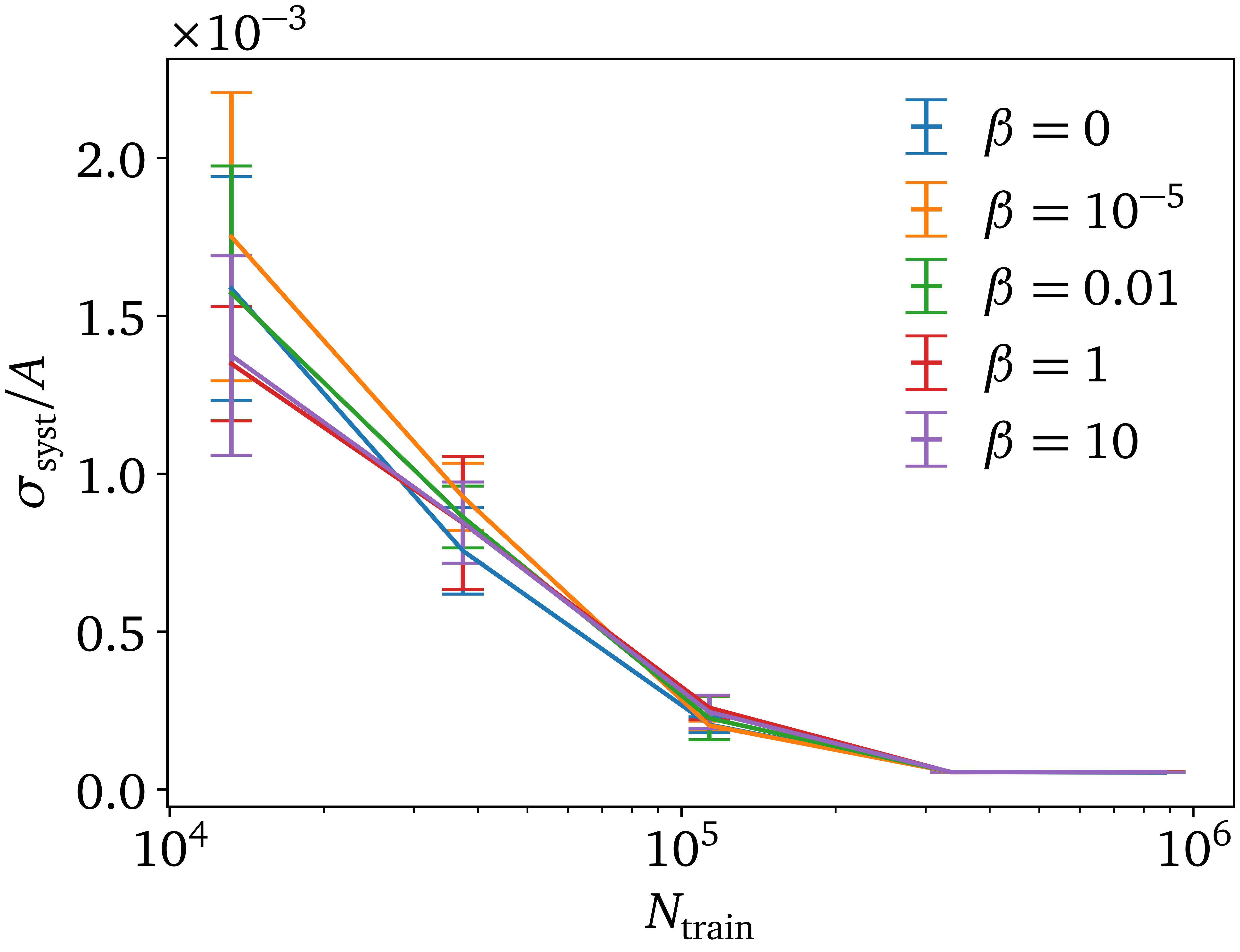
$$\langle A \rangle = \frac{1}{N} \sum_i^N \overline{A}(\theta_i)$$

$$\sigma_{\text{syst}}^2 = \frac{1}{N} \sum_i^N \sigma_{\text{syst}}^2(\theta_i)$$

$$\sigma_{\text{stat}}^2 = \frac{1}{N} \sum_i^N \left(\langle A \rangle - \overline{A}(\theta_i)\right)^2$$

- Parameters: **Network weights $q(\theta)$**
- $q(\theta)$ params of a Gaussian distribution
- Ensemble: Sample from weight distribution

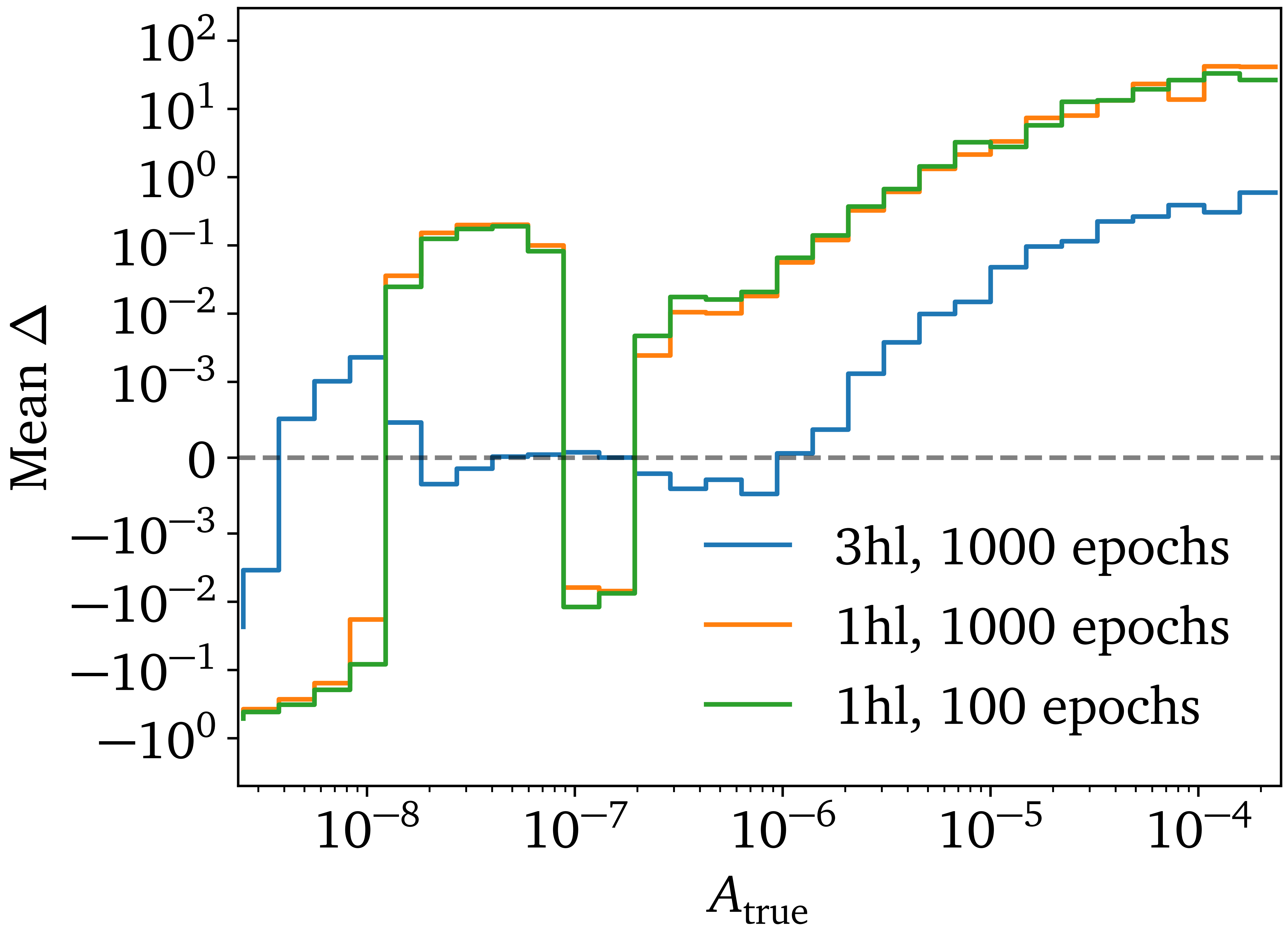
Kernel dependencies in the RE



- Vary repulsive prefactor β and N_{train}
- Spread vanishes with more training data
- Similar behaviour for statistical uncertainty

➡ Repulsive kernel impactful for **small training data** sets

Intrinsic bias of networks

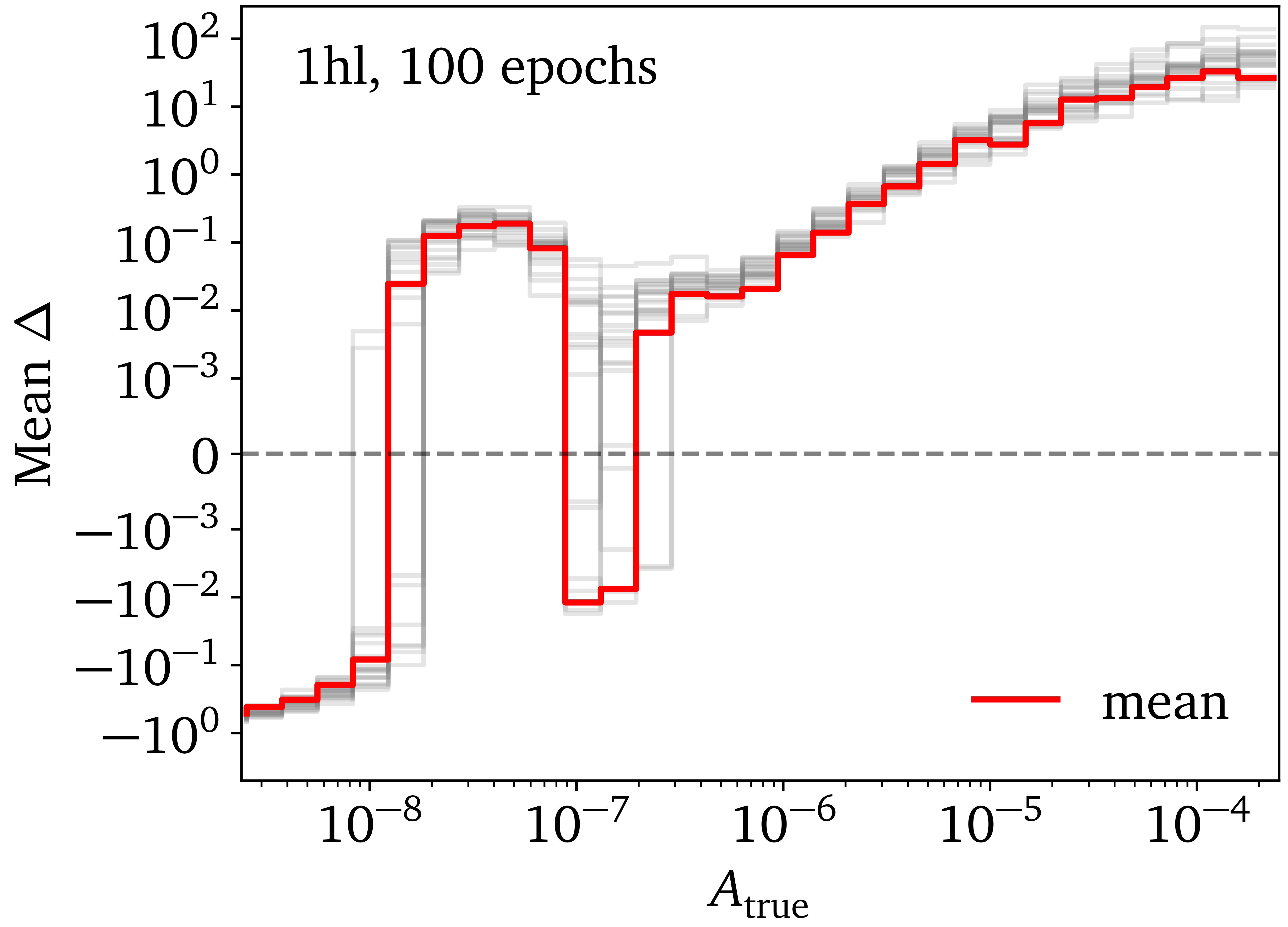


- Networks have inherent bias
- Independent of training duration
- Depend on network expressivity
- Do ensembles help?

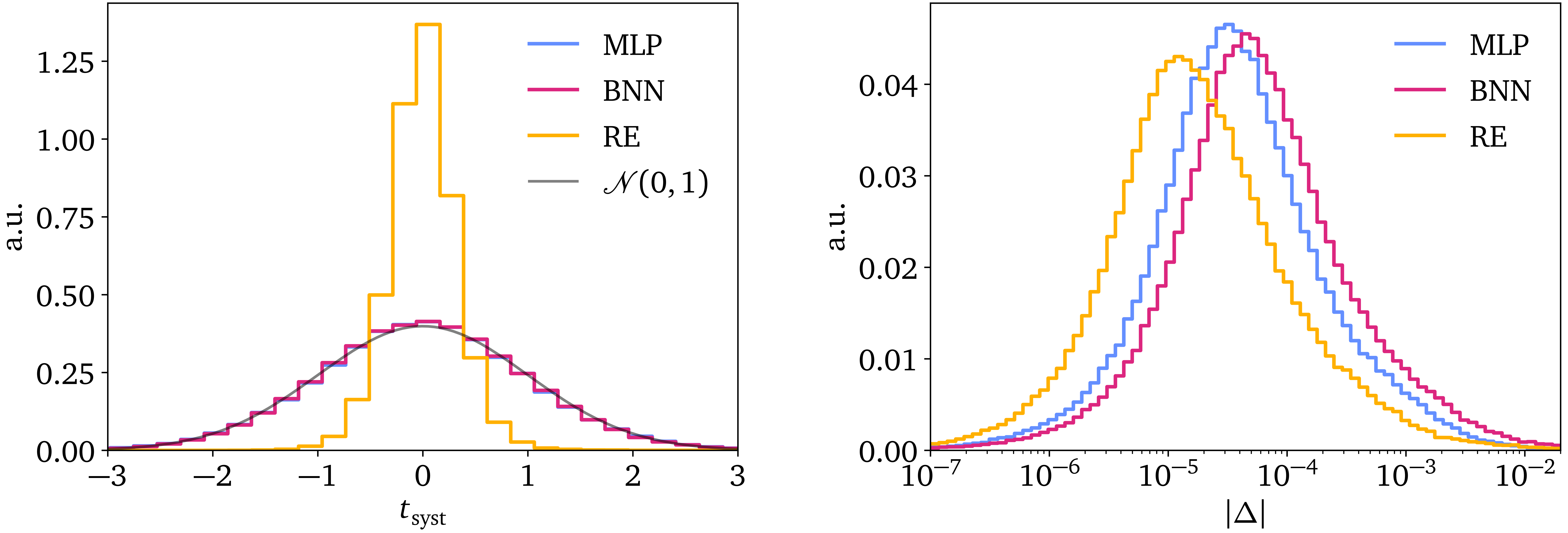
Intrinsic bias of ensembles

- Compare bias between full ensemble ↔ members

- Only small improvement for ensemble



Prediction and uncertainty accuracy

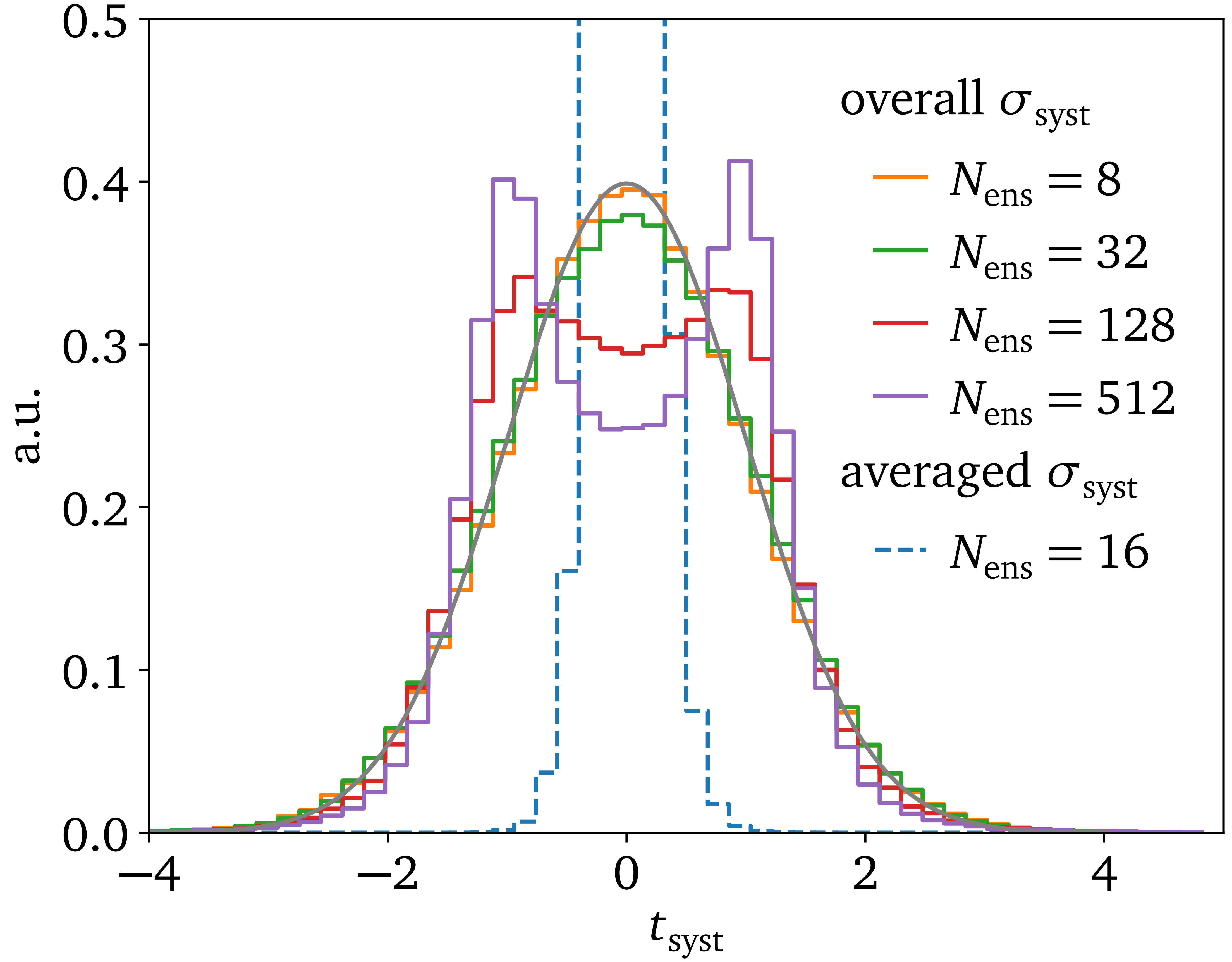


- Uncertainties perfectly calibrated for MLP and BNN → RE overestimates σ_{syst}
- Surrogate precision of $\mathcal{O} \sim 10^{-5}$

A more global approach to uncertainties

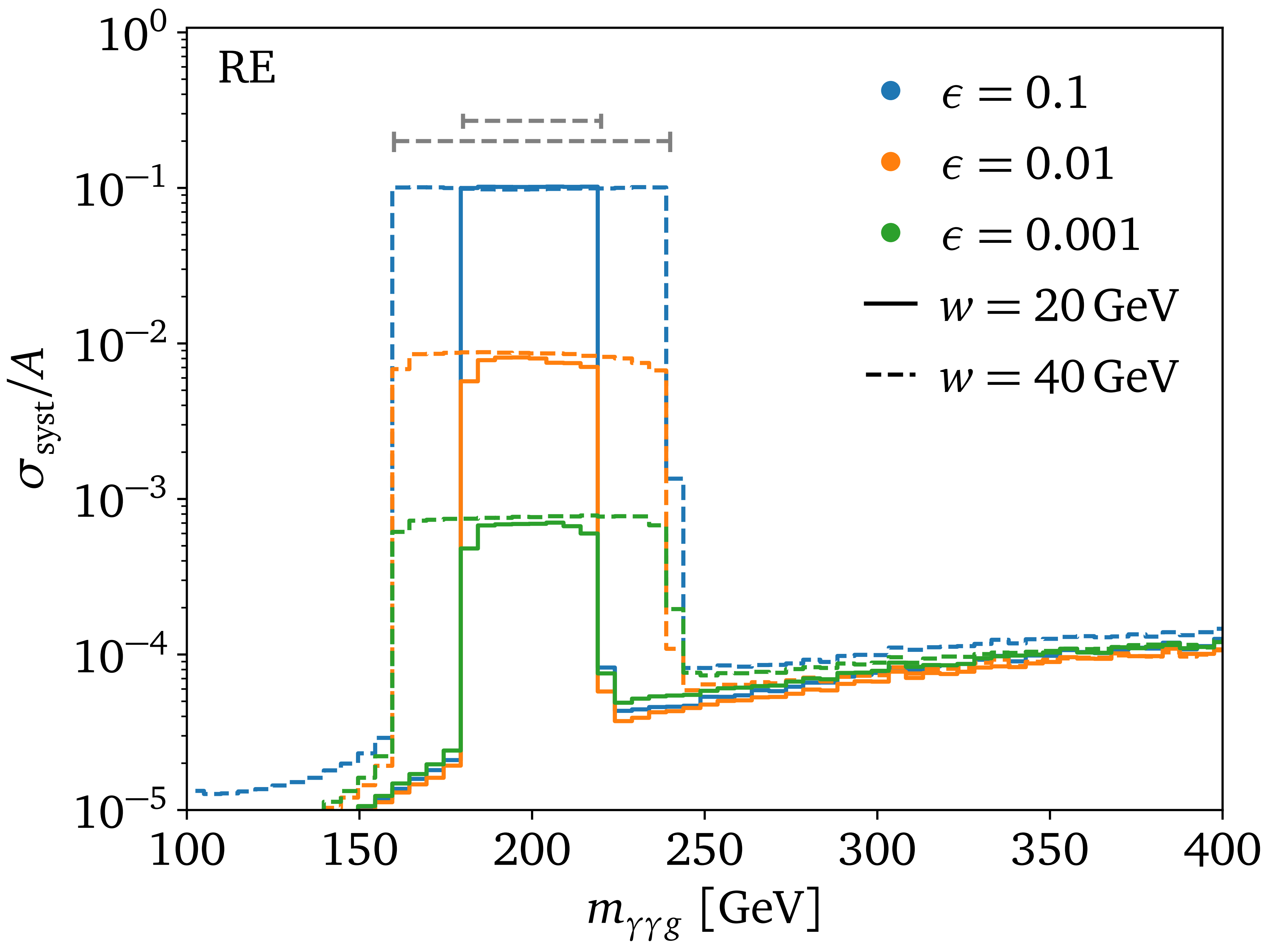
- Individually σ_{syst} per member: Underconfident
- Introduce **global systematic unc**:
  Train additional NN to predict global σ_{syst}
- Large N_{ens}: Reduced noise, discover bias

➡Two-mode structure



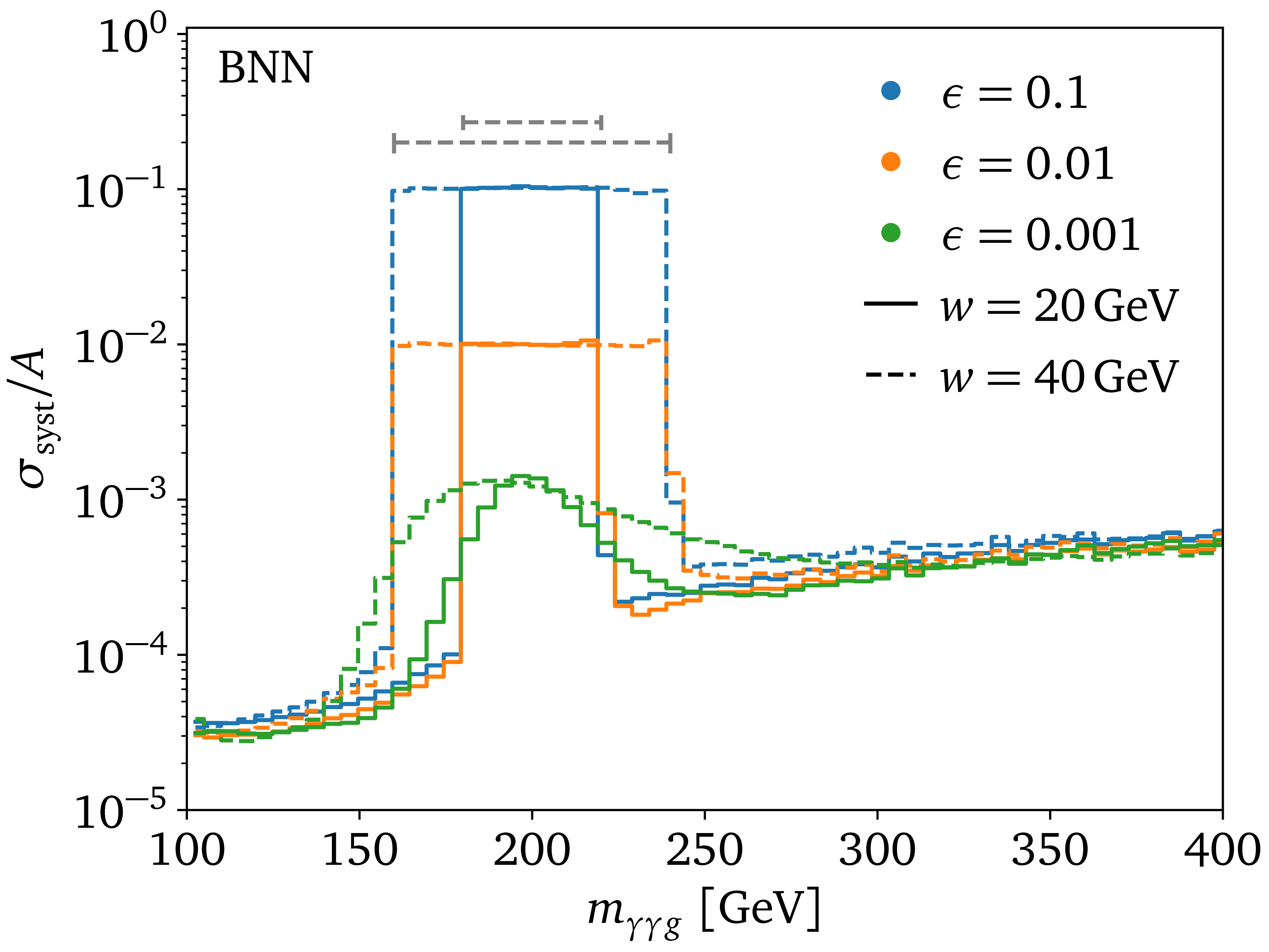
Flat box threshold smearing

Repulsive ensemble



- Captures noise perfectly
- Drops outside noise region

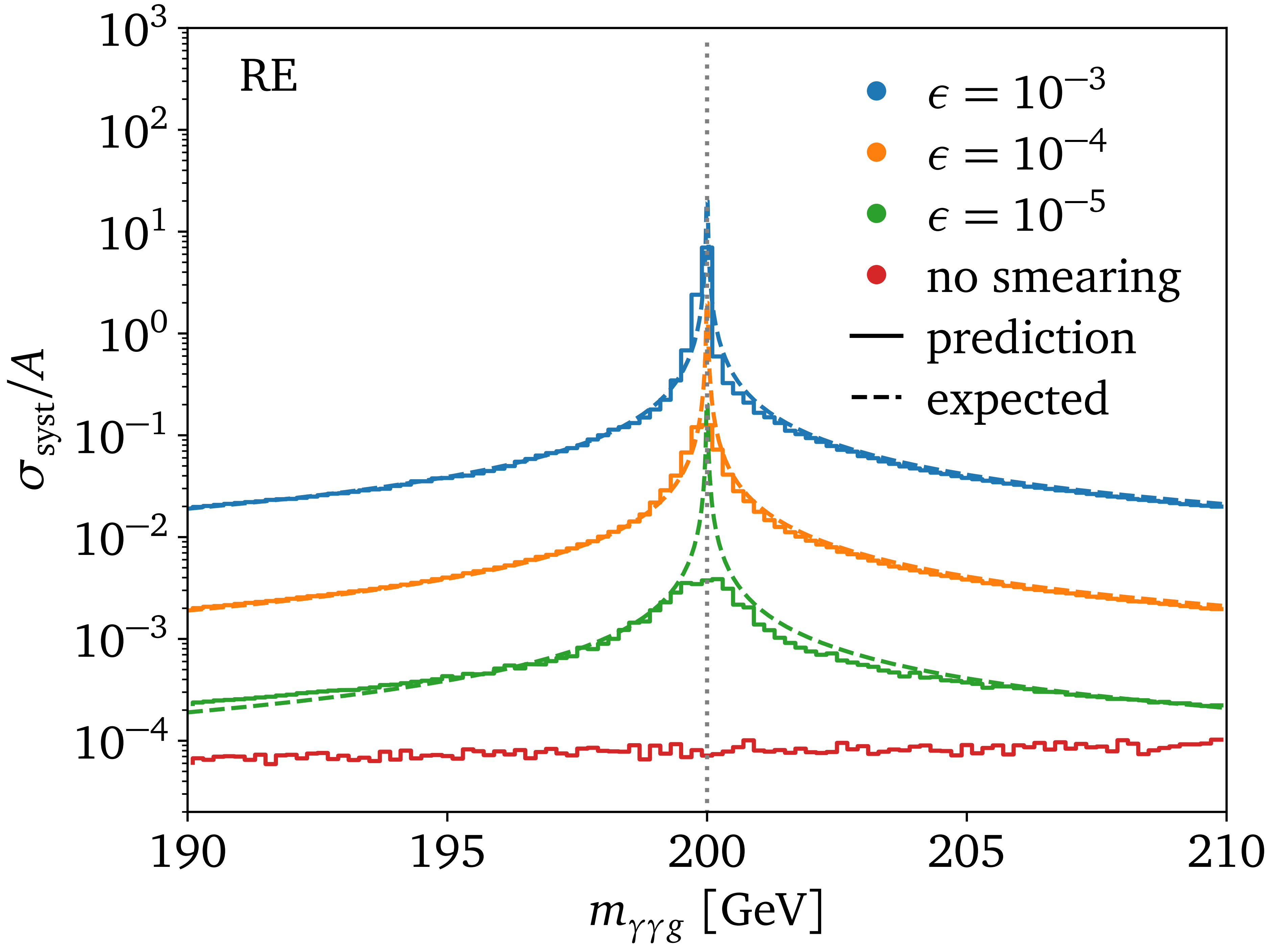
BNN



- Struggle for small ϵ
- Drops outside noise region

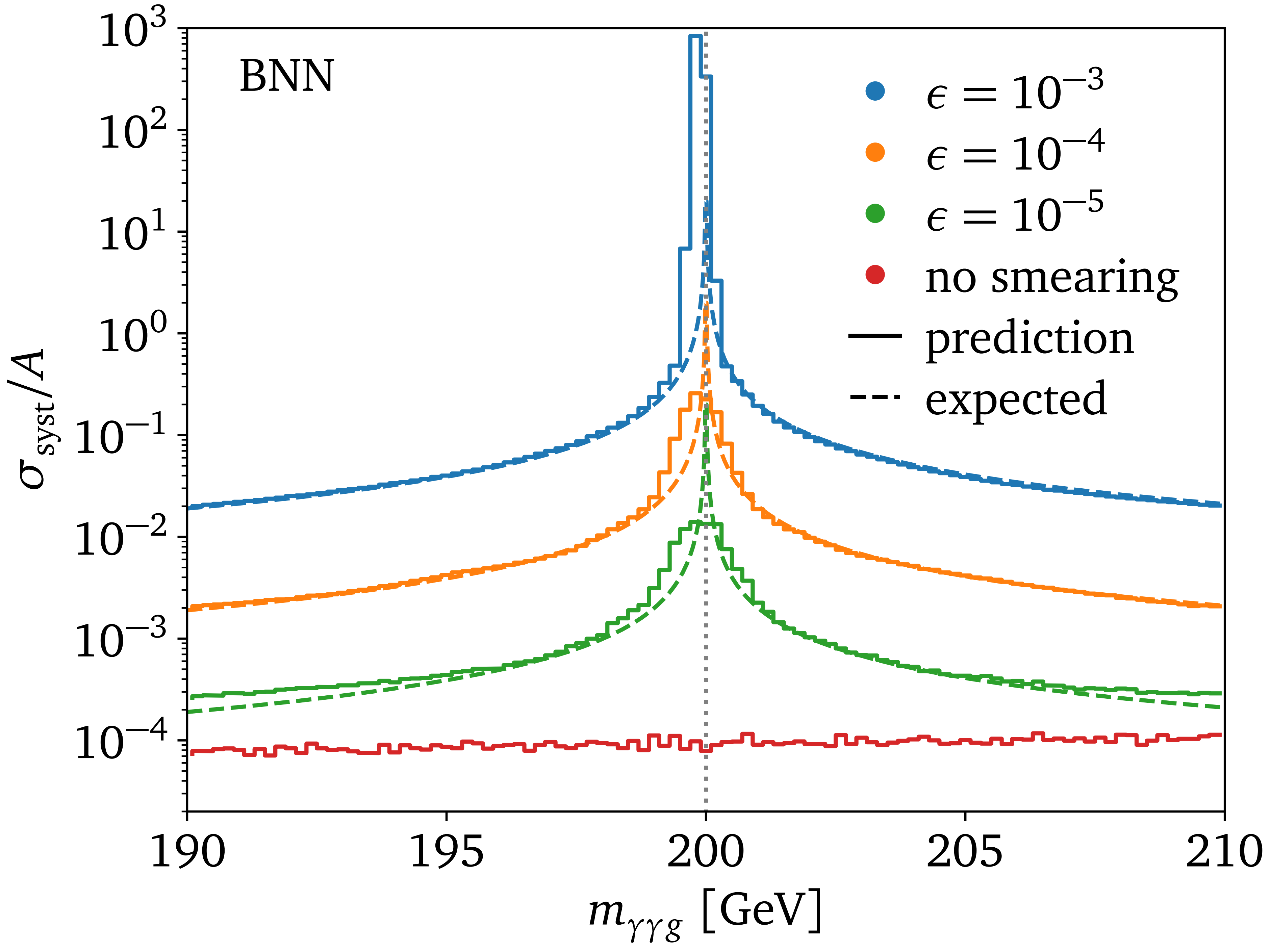
Peaked threshold smearing

Repulsive ensemble - systematic



- Captures noise perfectly
- Underestimation for small ϵ

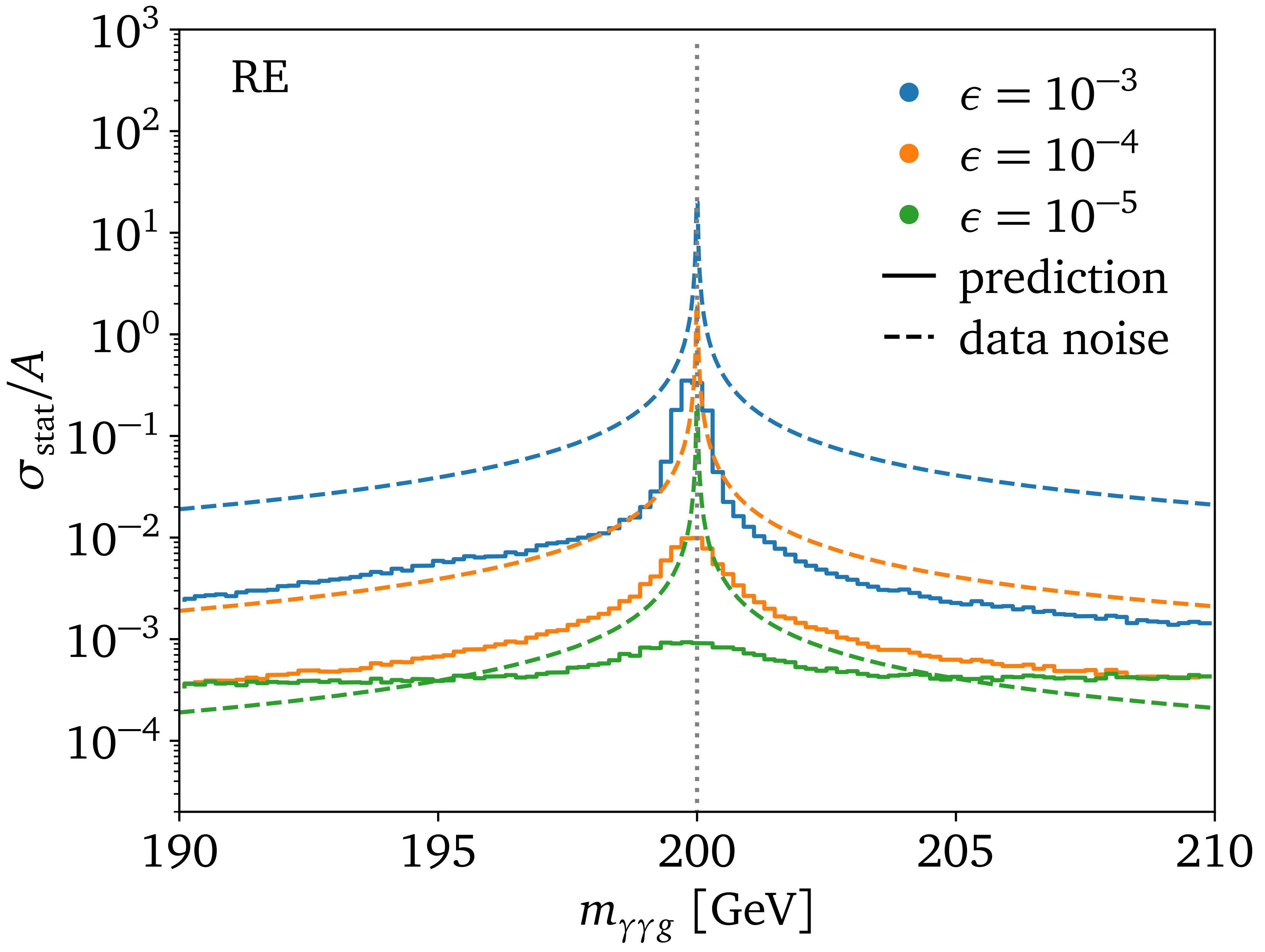
BNN - systematic



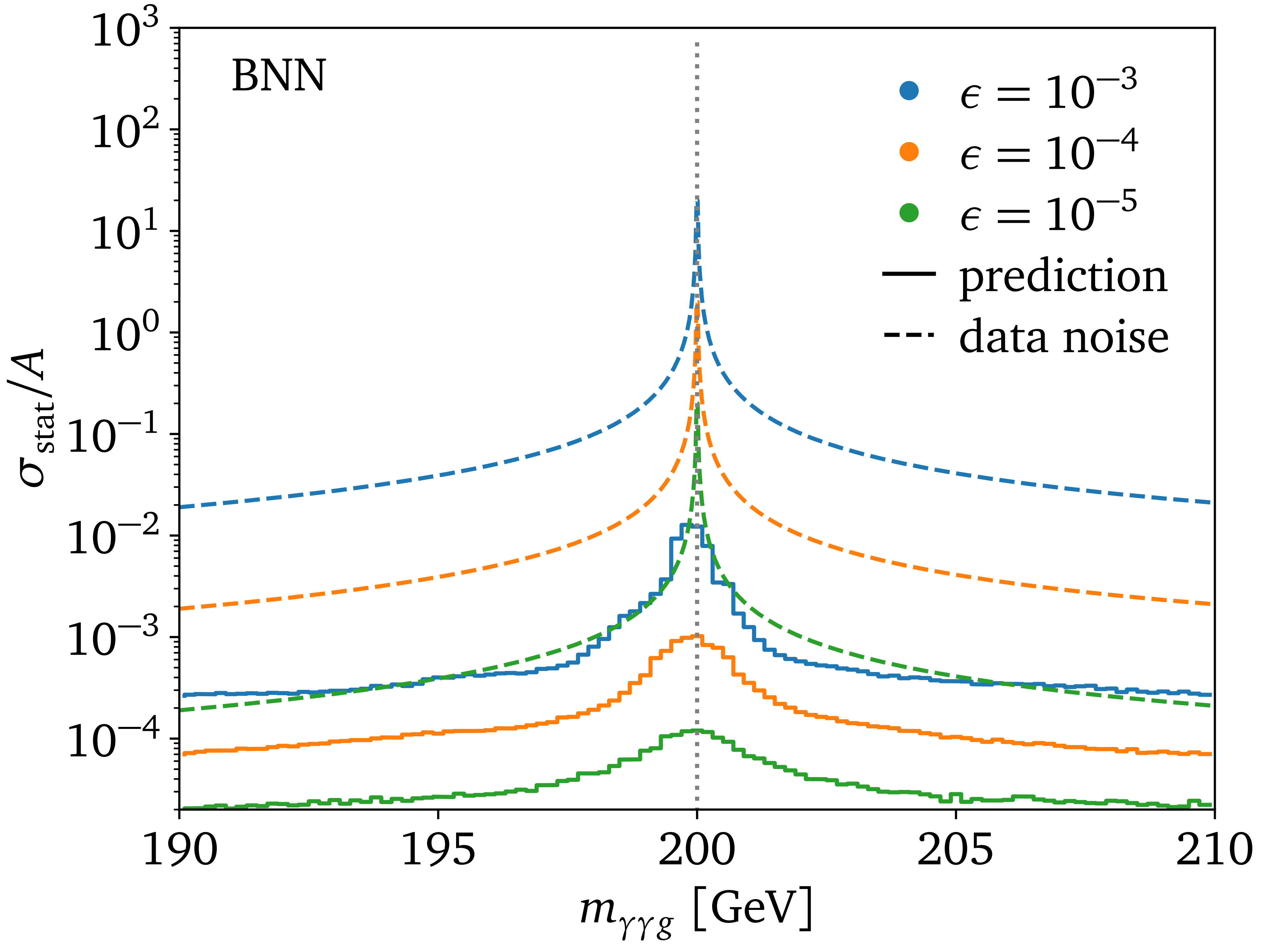
- Overestimates unc. for large ϵ
- Struggles close to threshold

Peaked threshold smearing

Repulsive ensemble - statistic



BNN - statistic

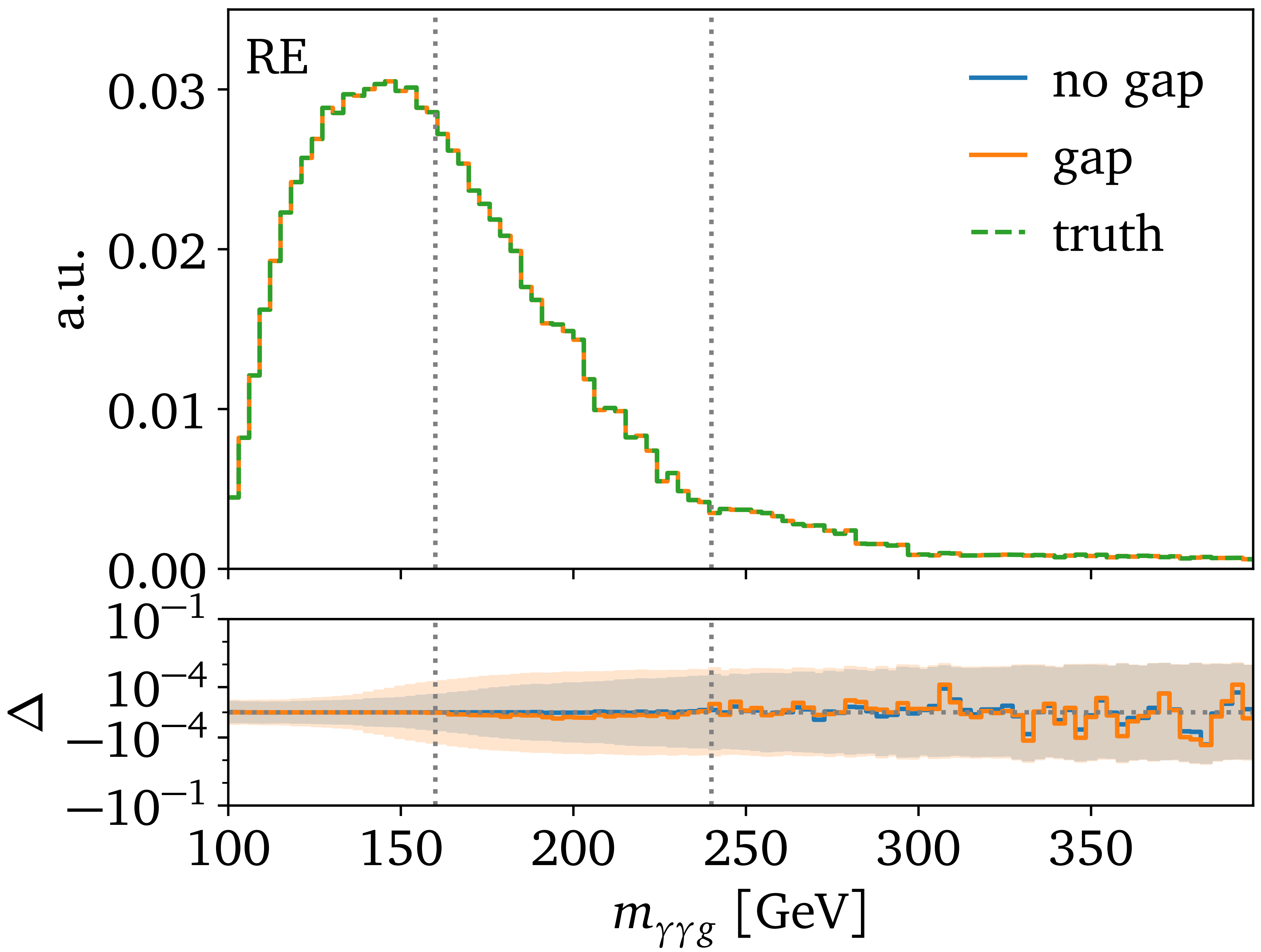


➡Statistical uncertainty smaller than induced noise

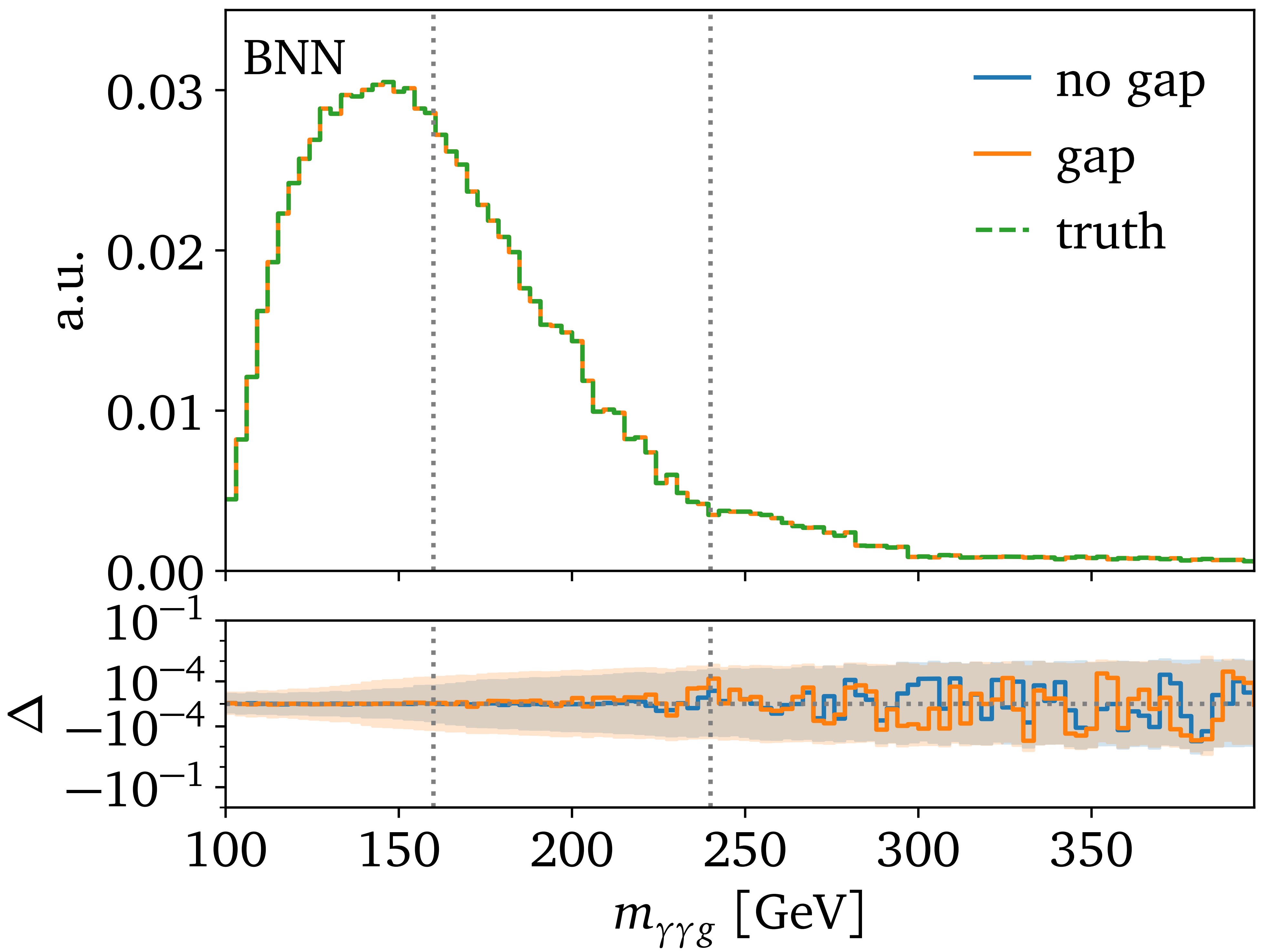
➡Learn accurate prediction in presence of strong local smearing

Gap in the data

Repulsive ensemble



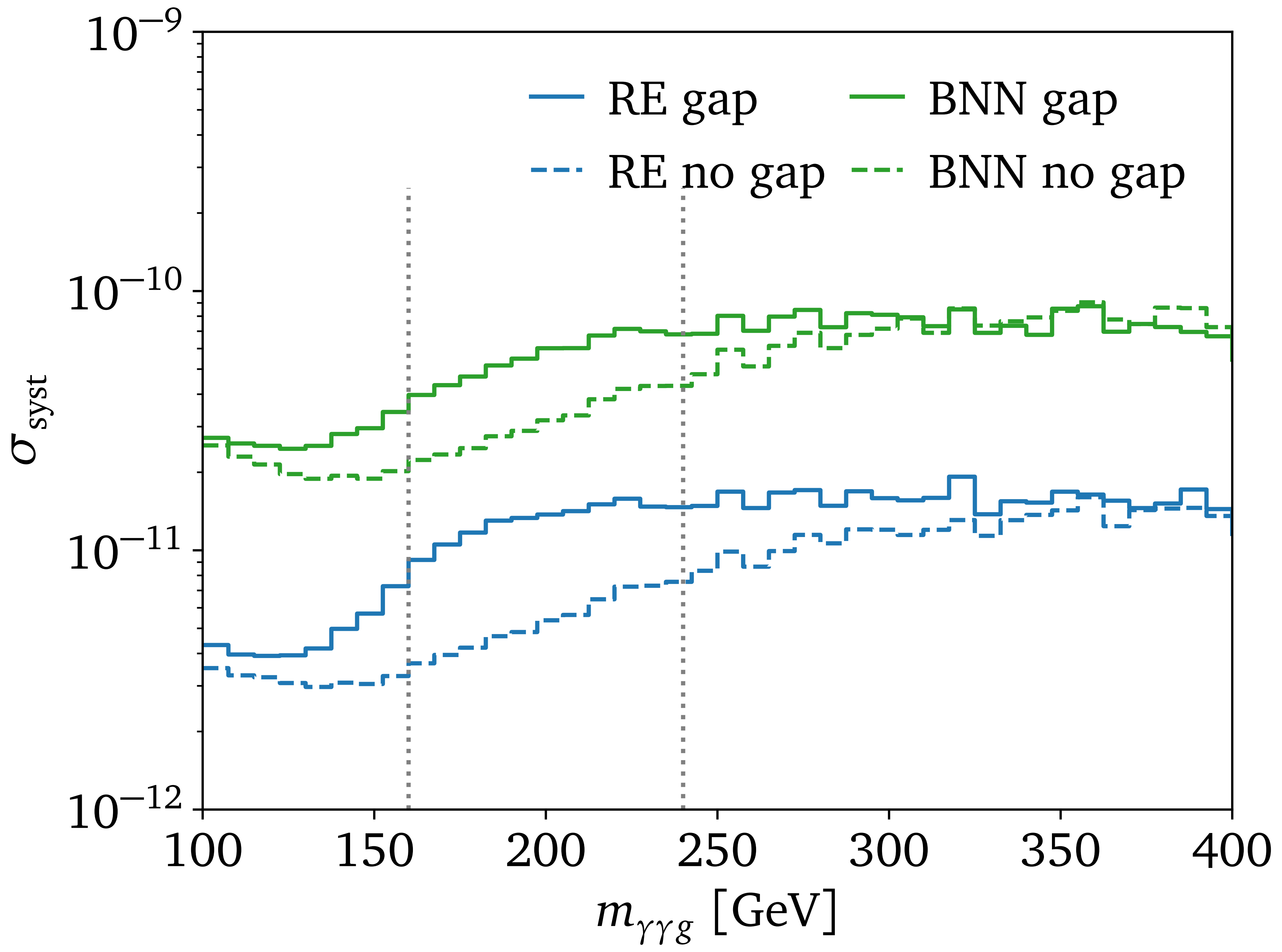
BNN



➡ Perfect extrapolation in gap region for both architectures

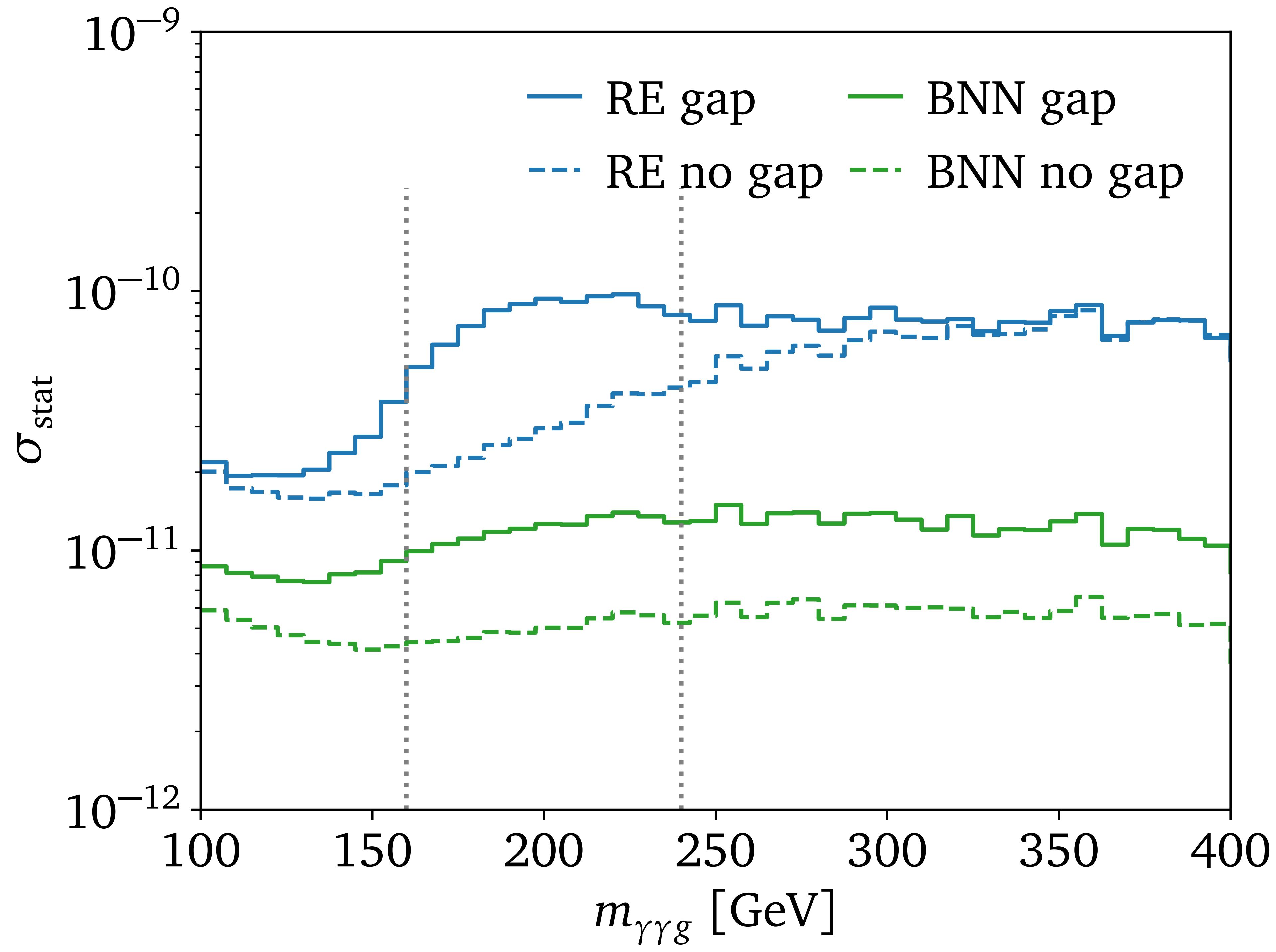
Gap in the data - Uncertainty

Systematic



- Increased uncertainty inside gap
- Expected uncertainty outside gap

Statistic



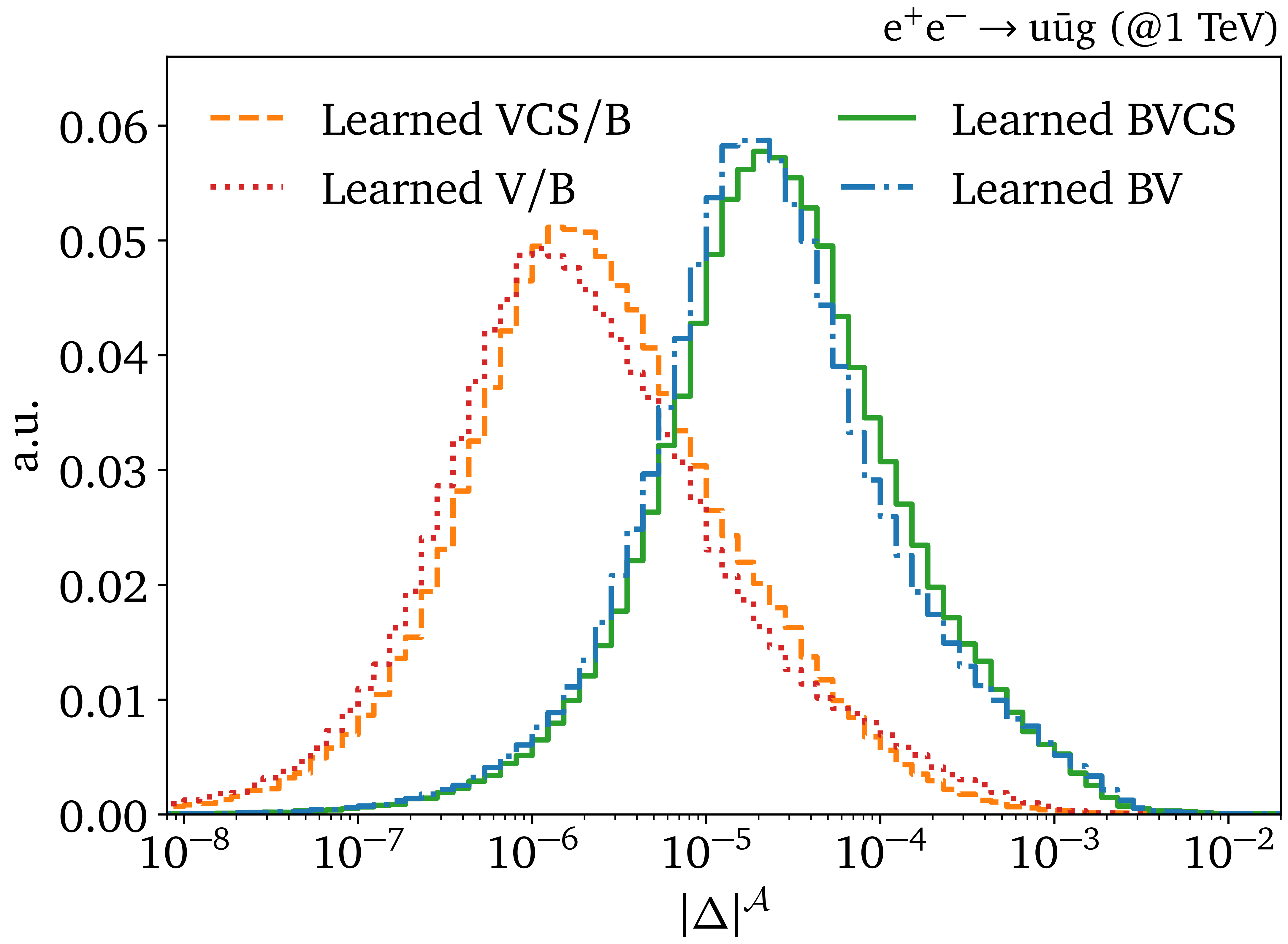
- RE: Expected behaviour
- BNN: Increased uncertainty outside gap

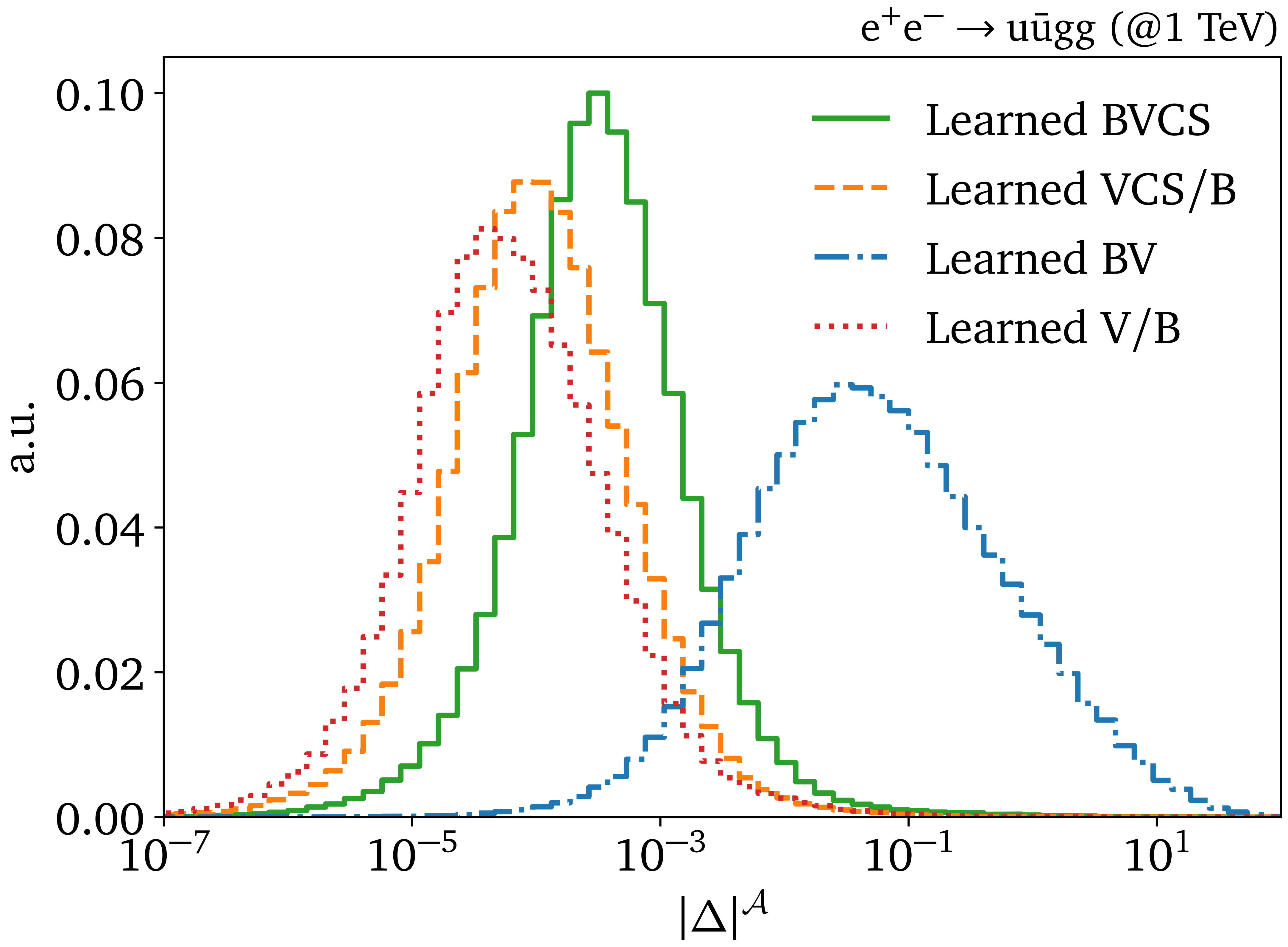
Other applications - MadGraph @ NLO

with: G. De Crescenzo, T. Heimel,
J. Marino Villadamigo, T. Plehn,
R. Winterhalder, M. Zaro

- Split up $A \rightarrow$ **B**orn, **V**irtual, **C**ollinear and **S**oft contributions

- Different ansatz to predict B+V+C+S contribution



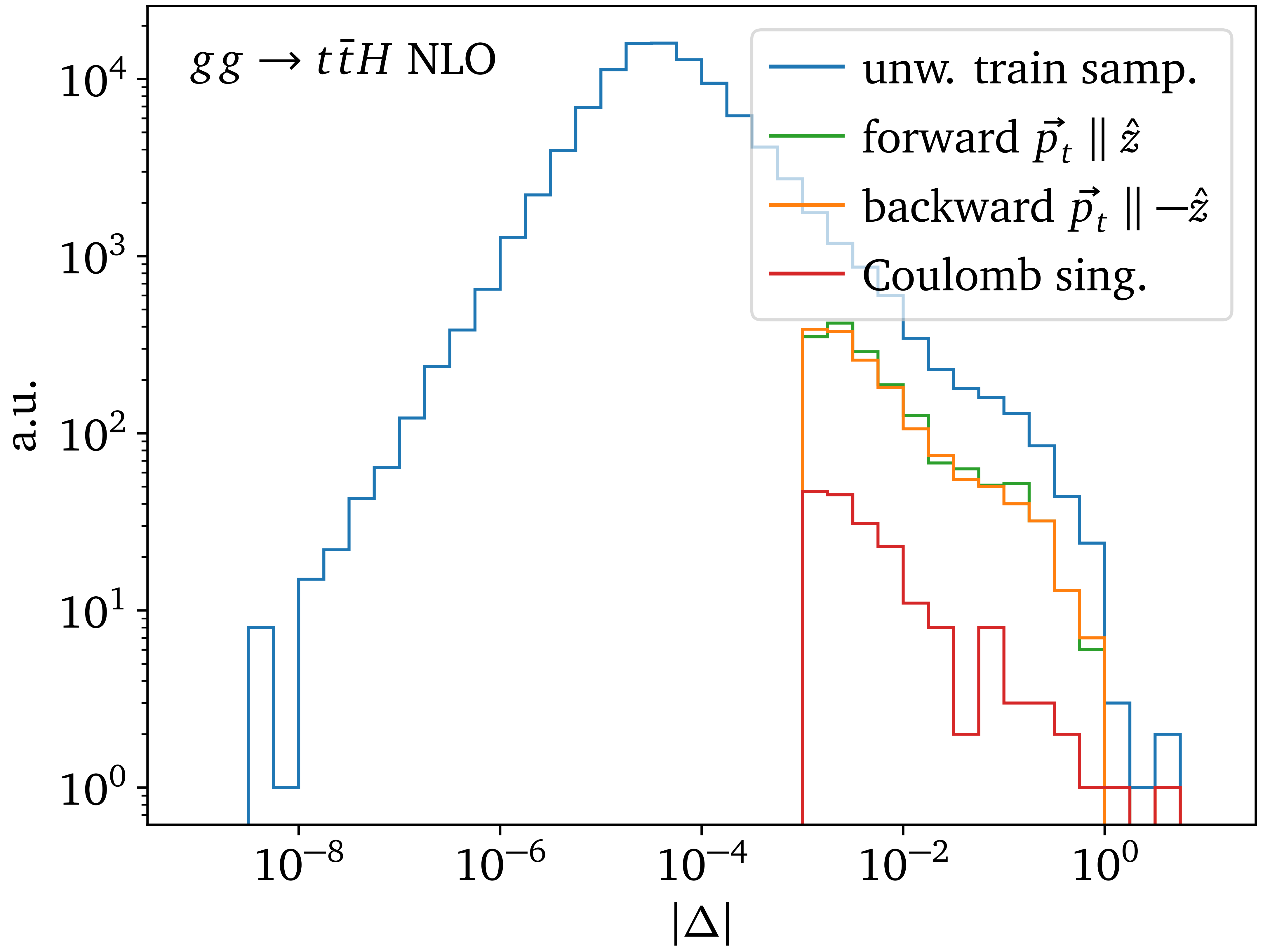


Preliminary

Other applications - $t\bar{t}H$ @ NLO production

- $gg \rightarrow t\bar{t}H$ @ NLO production
- Include soft and collinear singularities
- Precision enhancement: Adding events in singular regions
- Remark: In this case need Student's t-likelihood

arXiv: 2601.00950



work from: H. Bahl, J. Braun, G. Heinrich, T. Plehn, R. Revelli

Conclusion and Outlook

1. We can learn calibrated uncertainties
2. BNNs and (repulsive) ensembles are reliable uncertainty estimators
3. Global uncertainty approach for RE fixes calibration
4. Transferable to other use-cases (see MadGraph@NLO)

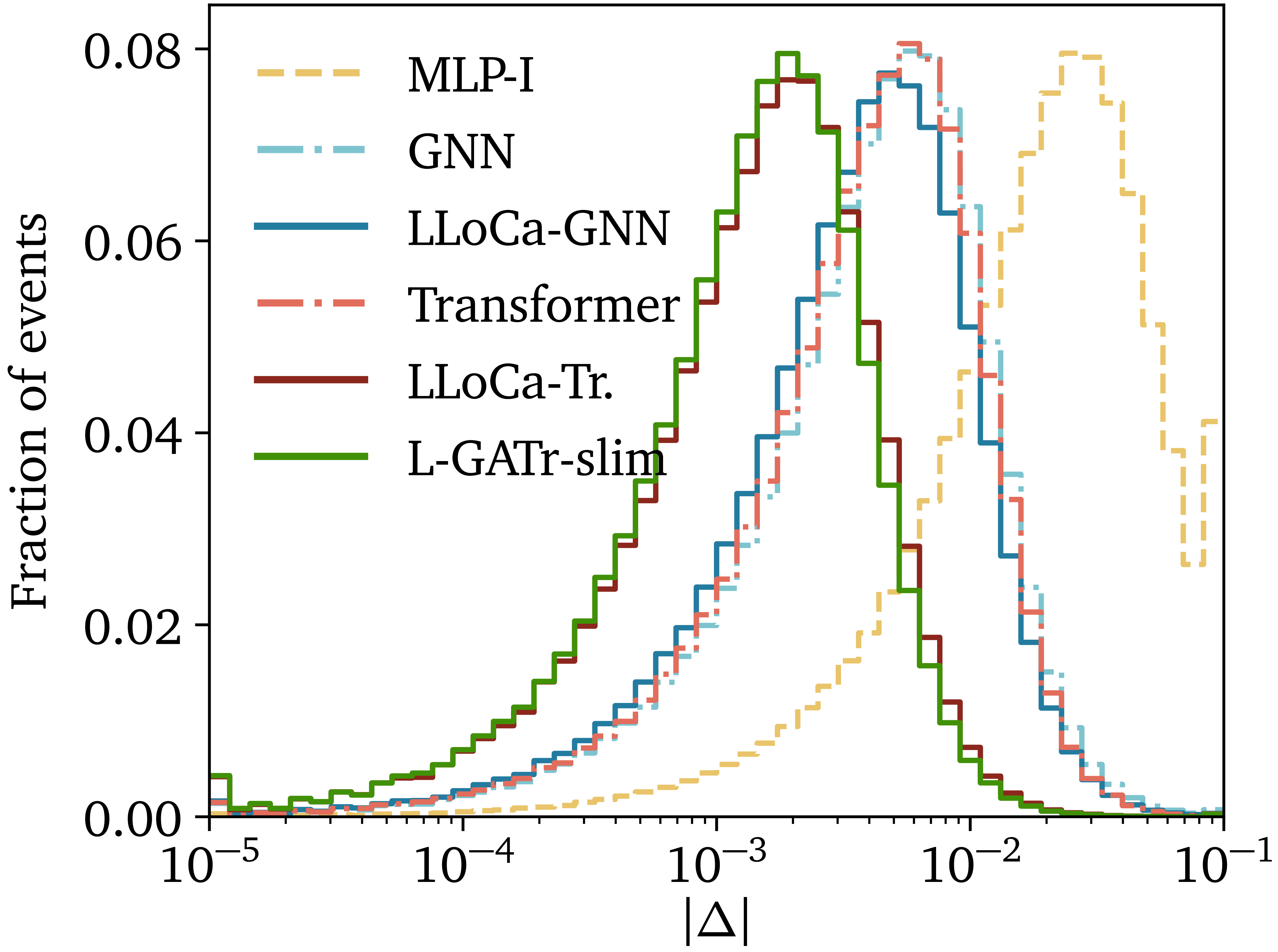
Outlook: Can we propagate these uncertainties into analyses and simulations?

Thank you for your attention!

Additional Material

Taking it to higher multiplicities

- Look at $q\bar{q} \to Zgggg$ process, 10M datapoints
- Use Lorentz-equivariant architectures
- Either transformers (L-GATr, LLoCa-Tr.) or graph neural networks (GNN, LLoCa-GNN)
- Need additional information encoded for higher multiplicities



Results from: A. Petitjean and J. Spinner
arXiv:

How to: evidential regression (ER)

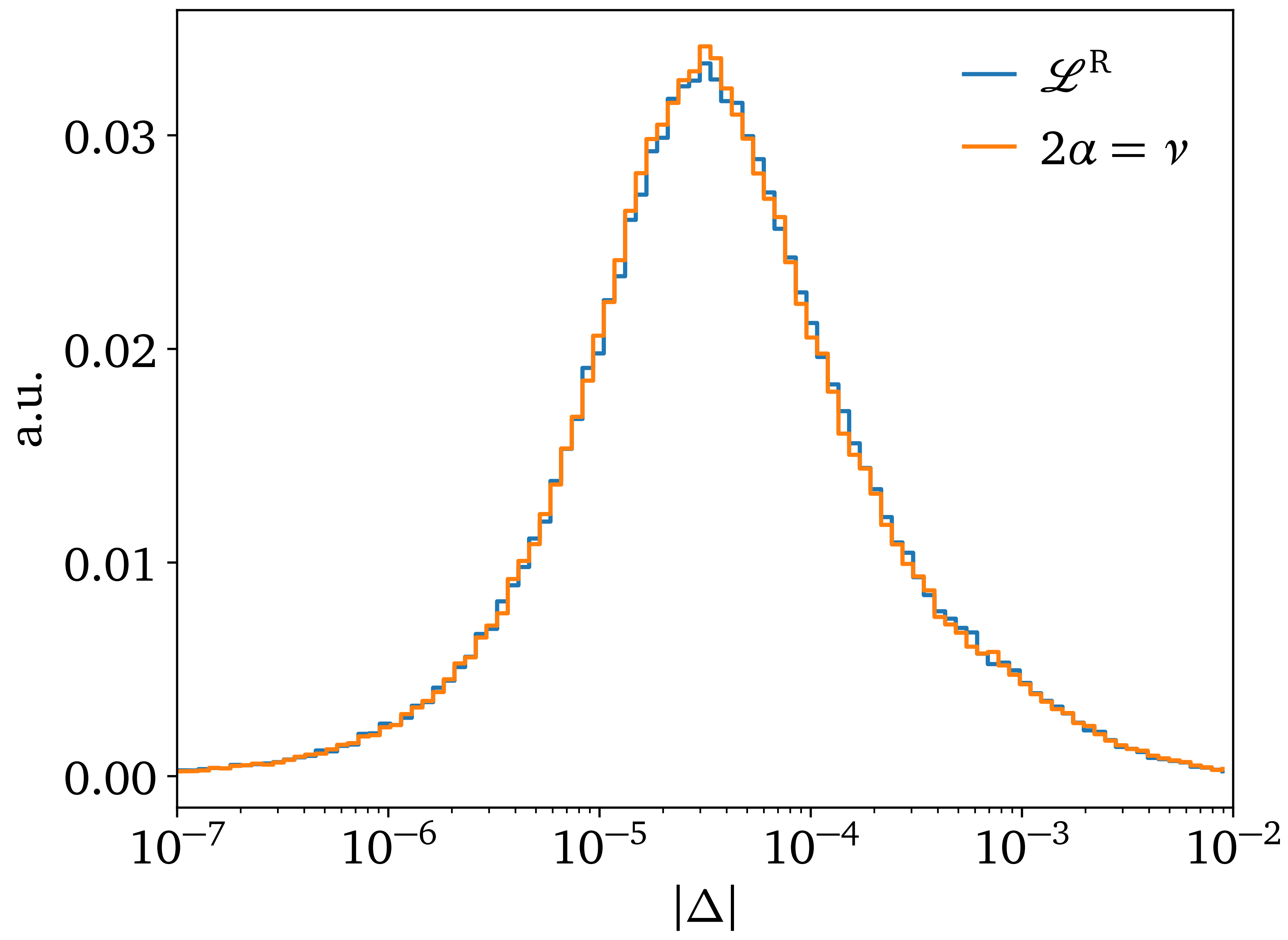
- Previously: Systematic uncertainty from loss: $\mathscr{L}_{\text{heteroscedastic}} = \sum_i \frac{|f(x_i) - f_\theta(x_i)|^2}{2\sigma(x_i)^2} + \log \sigma(x_i) + \ldots$
- Different approach: Introduce conjugate prior $p(\lambda \,|\, m)$, λ likelihood parameter
  - Likelihood $\lambda = (f_\theta(x_i), \sigma)$ and evidential parameters $m(x, \theta) = \{\gamma, v, \alpha, \beta\}(x, \theta)$
- New likelihood: $p(A|x) \approx \int \mathrm{d}\lambda \, p(A|x, \lambda) p(\lambda|m)$

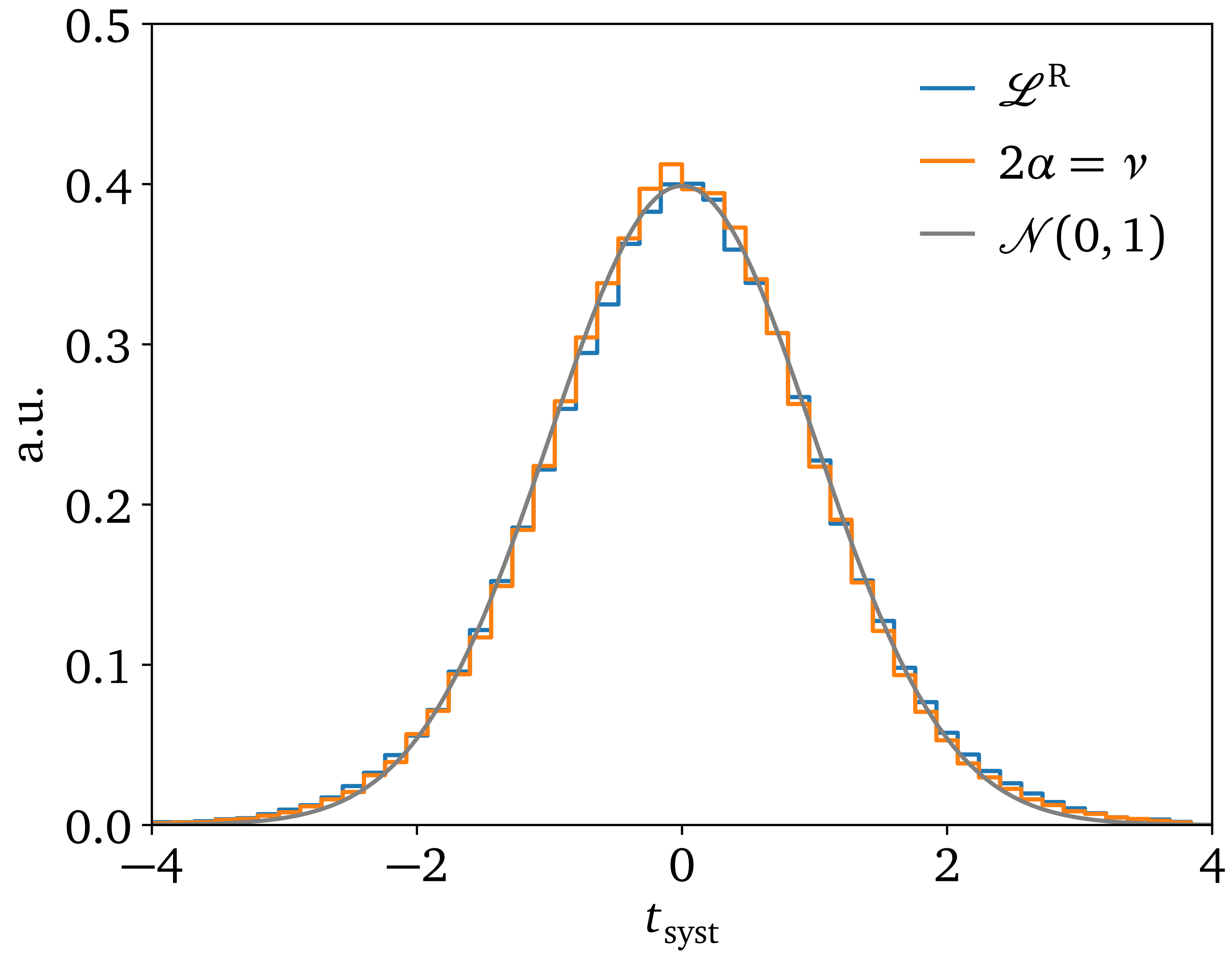
follows Normal-Inverse-Gamma distribution

- Analytic likelihood: $p(A|x, m) = \mathrm{St}\left(A \middle| \gamma, \frac{\beta(1+v)}{v\alpha}, 2\alpha\right)$

Evidential regression

- Mean and uncertainties: Averaging over $p(\lambda\,|\,m)$ → Analytical **closed form**
- No MC sampling necessary
- Integrals **directly solvable** → Uncertainties and mean without sampling accessible

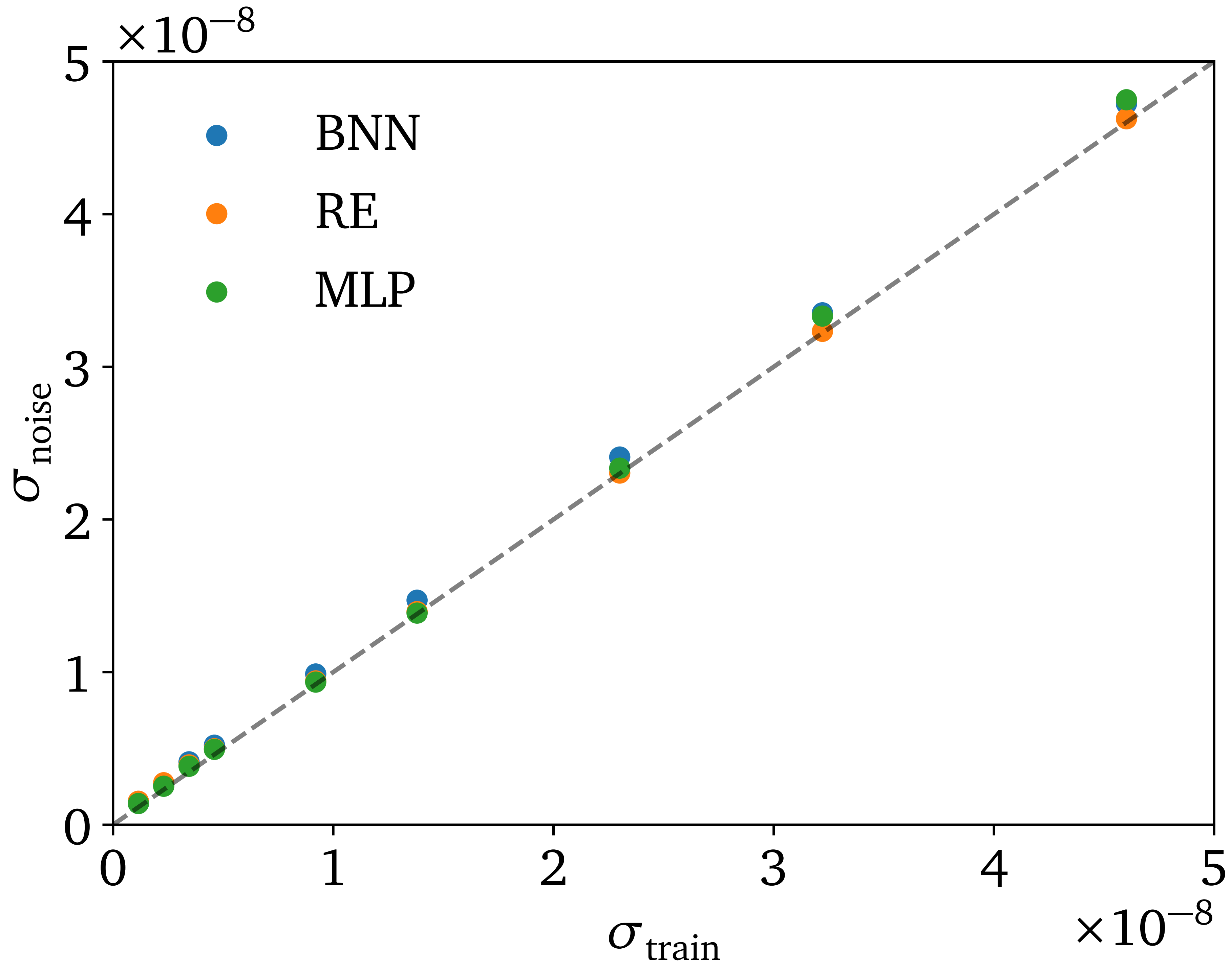


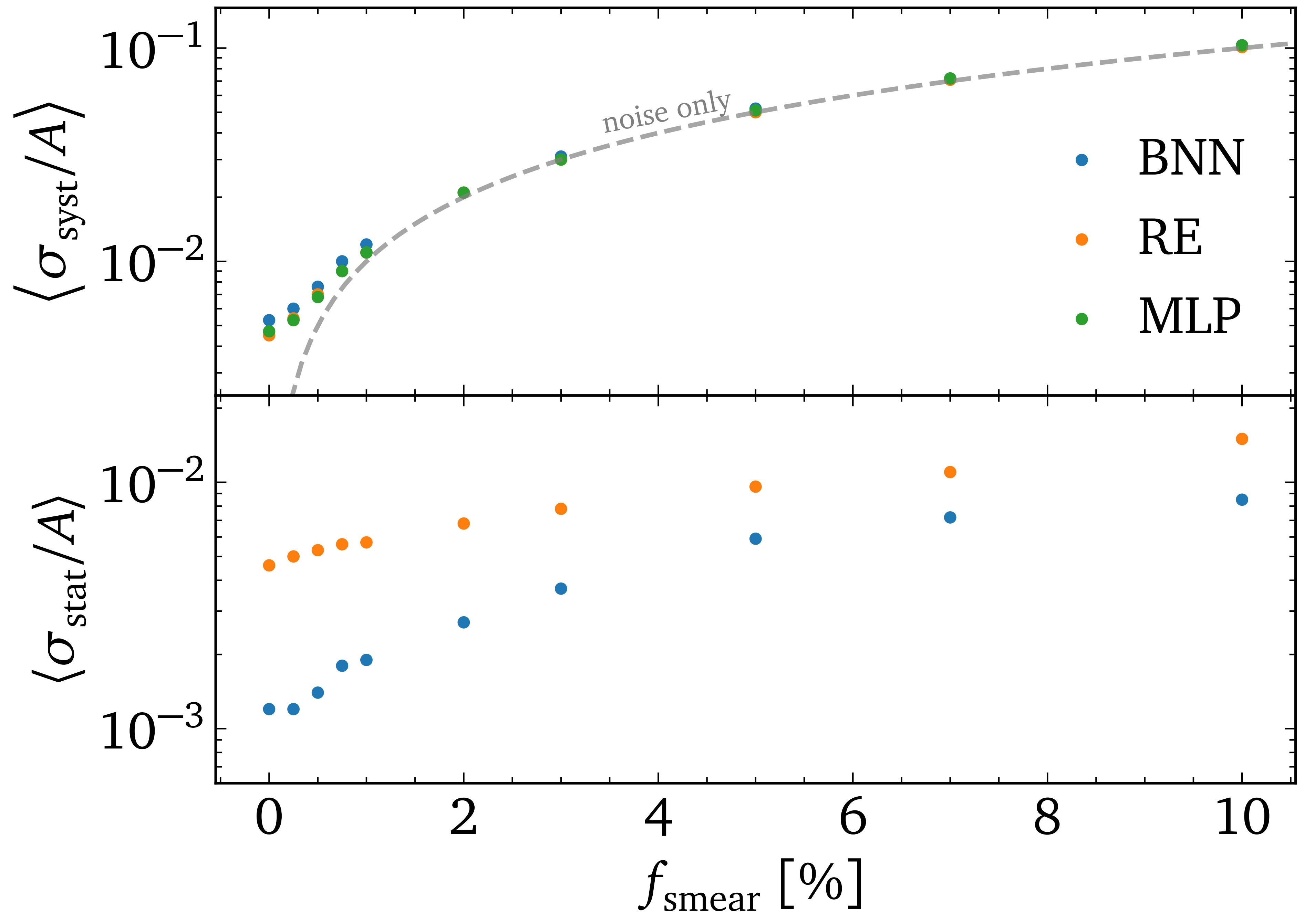


Learning Gaussian noise

$$\sigma^2_{\text{tot}} = \sigma^2_{\text{syst},0} + \sigma^2_{\text{noise}} + \sigma^2_{\text{stat}}$$

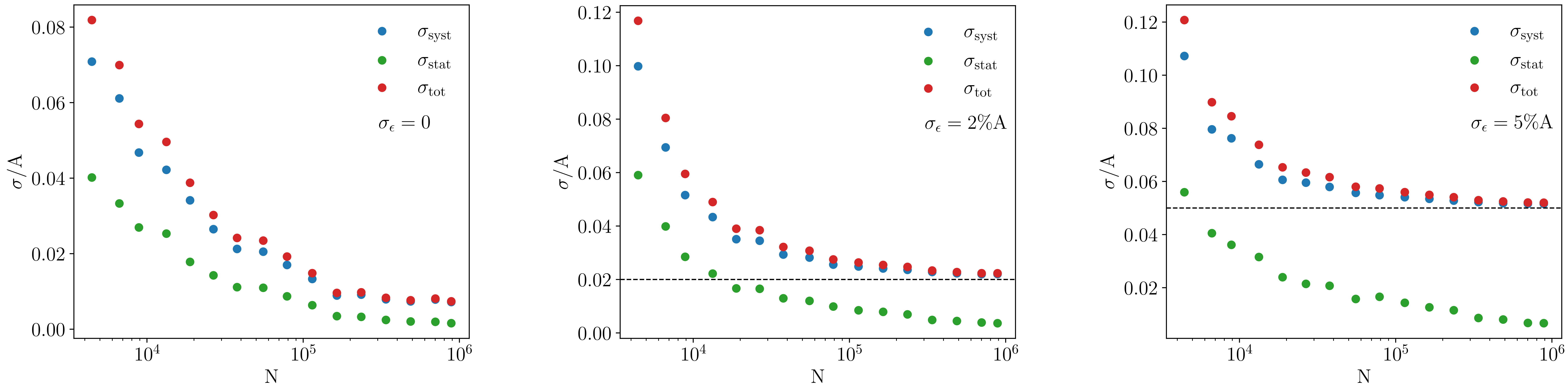
$$\sigma_{\text{train}} = f_{\text{smear}}\, A_{\text{true}}$$





➡ Networks learn noise as **systematic** uncertainties

Learning Gaussian noise

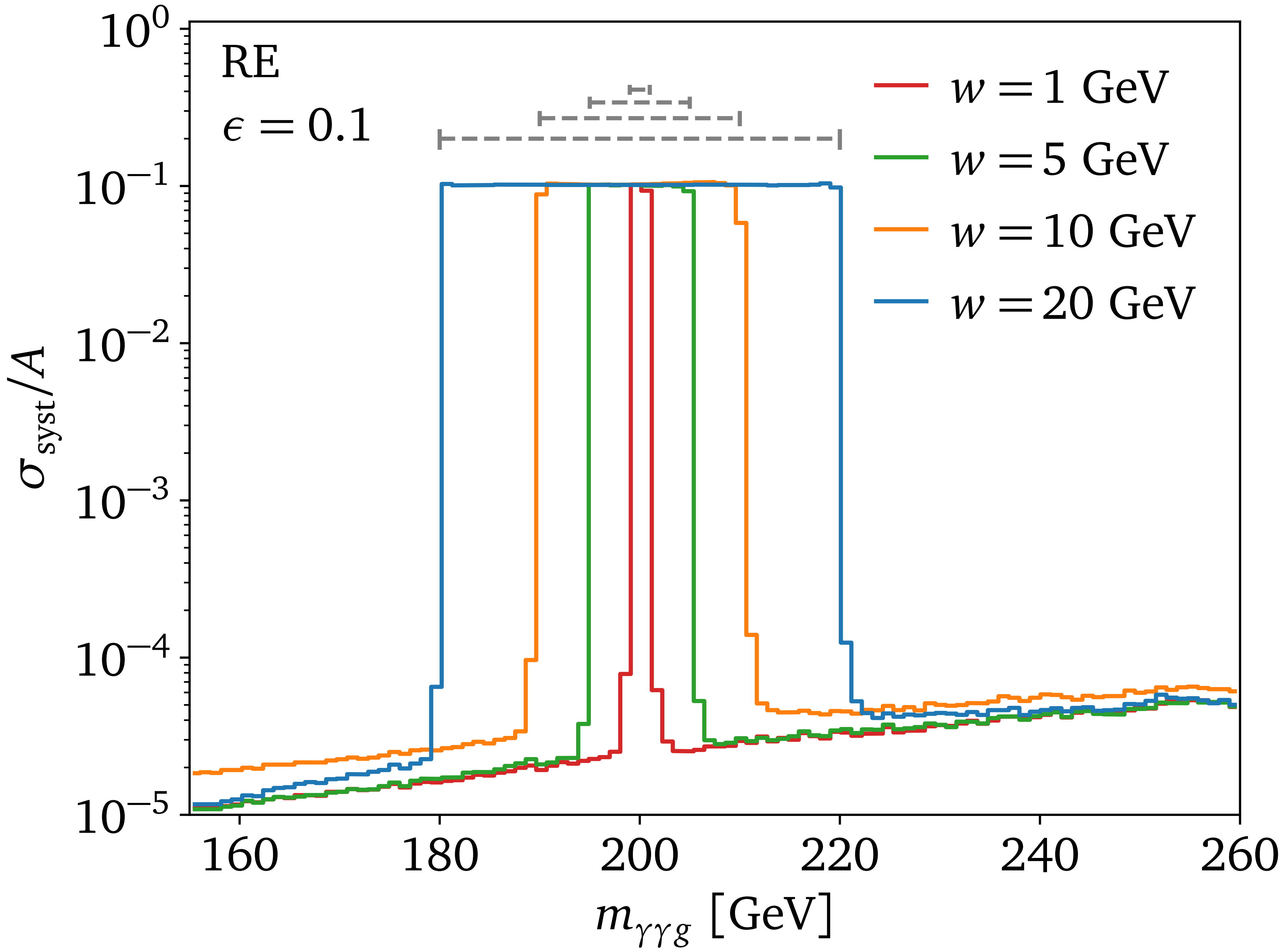


BNN only

1. Statistical uncertainty **independent** of noise
2. Systematic uncertainty **plateaus** on noise level

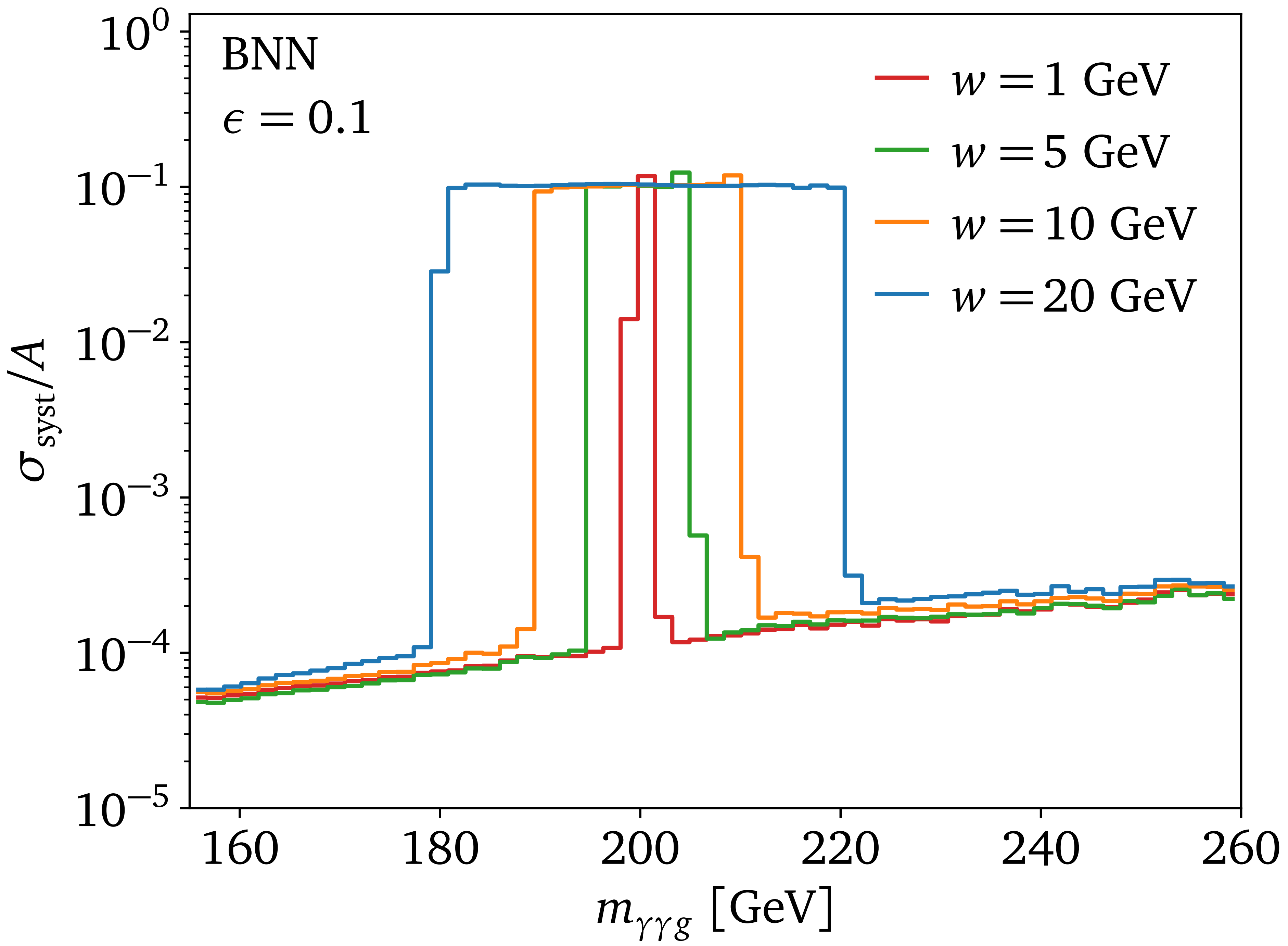
Flat box threshold smearing

Repulsive ensemble



- Captures noise perfectly
- Expected behaviour

BNN



- Box shape not as accurate
- Expected behaviour