Physikalisch-Technische Bundesanstalt
Braunschweig und Berlin
Nationales Metrologieinstitut

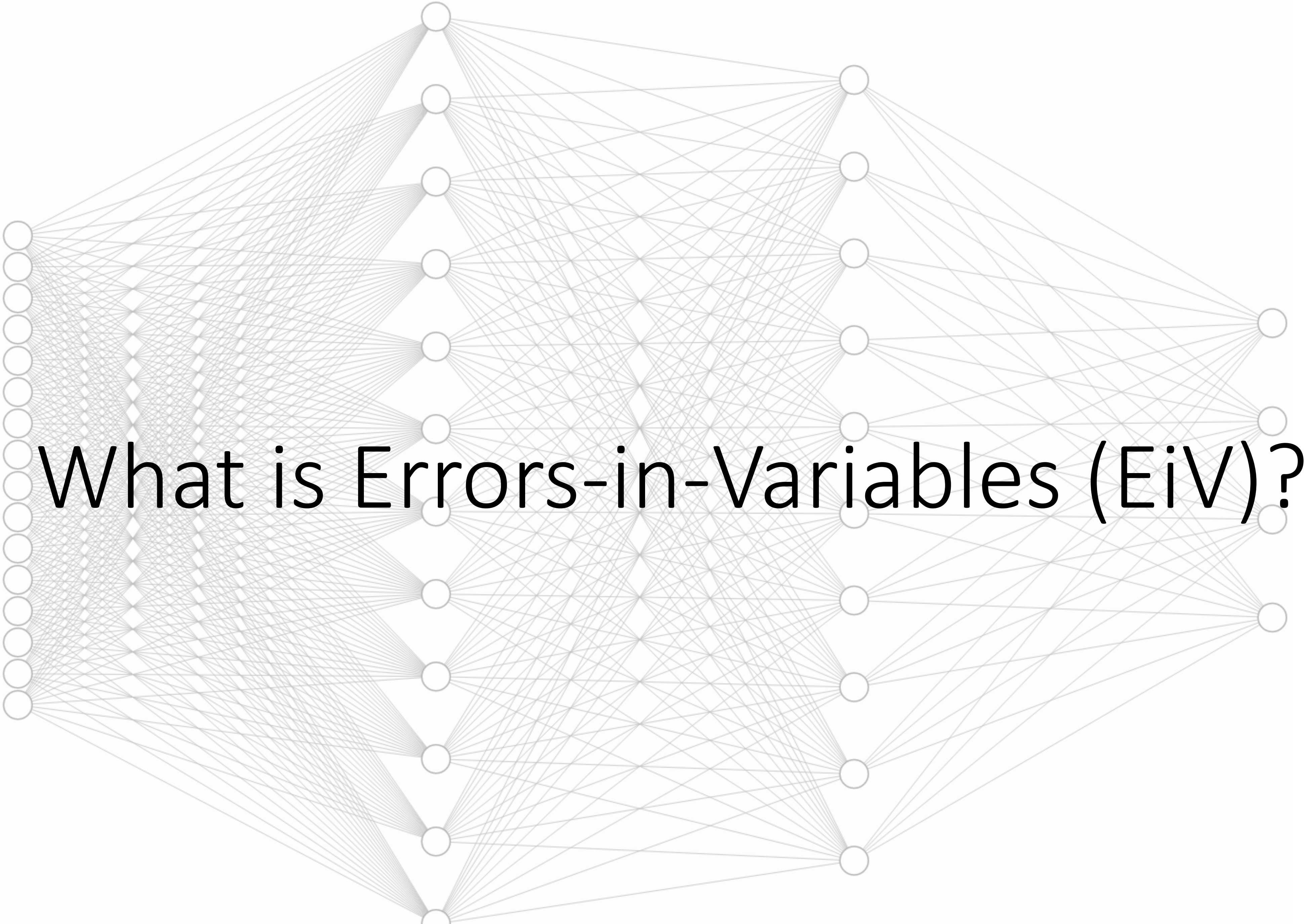
Errors-in-Variables models in deep regression

Jörg Martin (PTB)

joint with Clemens Elster

MATHMET 2022
ENBIS session
3rd November 2022

What is Errors-in-Variables (EiV)?



Deep regression & EiV

Classical model (non-EiV)

$$y = f_\theta(x) + \varepsilon_y$$

Neural net

$\varepsilon_y \sim N(0, \sigma_y^2 I_{n_y \times n_y})$

Deep regression & EiV

Classical model (non-EiV)

$$y = f_\theta(x) + \varepsilon_y$$

Neural net

$$\varepsilon_y \sim N(0, \sigma_y^2 I_{n_y \times n_y})$$

Errors-in-Variables (EiV)

$$y = f_\theta(\zeta) + \varepsilon_y$$

$$x = \zeta + \varepsilon_x$$

Deep regression & EiV

Classical model (non-EiV)

$$y = f_\theta(x) + \varepsilon_y$$

Neural net

$$\varepsilon_y \sim N(0, \sigma_y^2 I_{n_y \times n_y})$$

Errors-in-Variables (EiV)

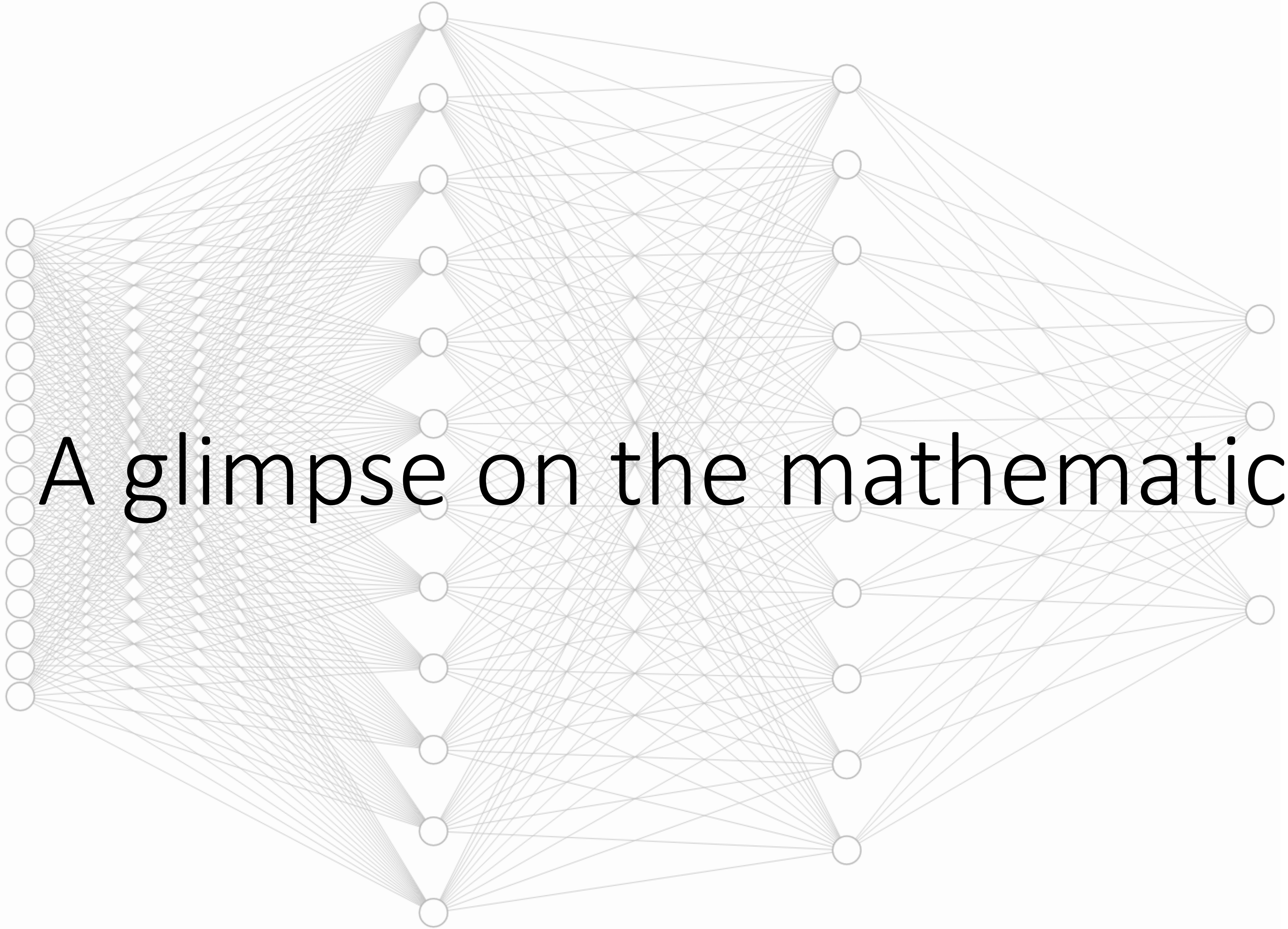
$$y = f_\theta(\zeta) + \varepsilon_y$$

$$x = \zeta + \varepsilon_x$$

(e.g. Gorp et al. 1998, Pavone 2018, Xie at al. 2020)

Our work: **Bayesian Neural Networks** → **Uncertainty**

A glimpse on the mathematics



Recap on Bayesian NNs

Fix prior, e.g. $\pi(\theta) = N(0, \lambda^{-1} I_{n_\theta \times n_\theta})$

(Unfeasible) posterior given $D = \{(x_i, y_i) \mid \; i = 1, \ldots, N\}$

$$\pi(\theta|D) \propto \pi(\theta) \cdot p(D|\theta)$$

Approximate via feasible $q_\phi(\theta) \approx \pi(\theta|D)$ (Variational Inference)

Recap on Bayesian NNs

Kullback-Leibler loss:

$$L(\phi) = KL\left(q_\phi(\theta)||\pi(\theta|D)\right) \quad \text{``=``} \quad \text{L2 Loss} + \text{regularization}$$

Key result of Gal et al. (2016):

- Dropout gives naturally rise to some $q_\phi(\theta)$
- Minimizing $L(\phi)$ „equivalent“ to standard training under dropout

  $\rightarrow$ scalable way to do Variational Inference for NNs

Additional ingredients for EiV

Additional prior $\pi(\zeta) = N\left(0, \sigma_\zeta^2 I_{n_\zeta \times n_\zeta}\right)$ (Uninformative: $\sigma_\zeta \to \infty$)

Additional posterior for input

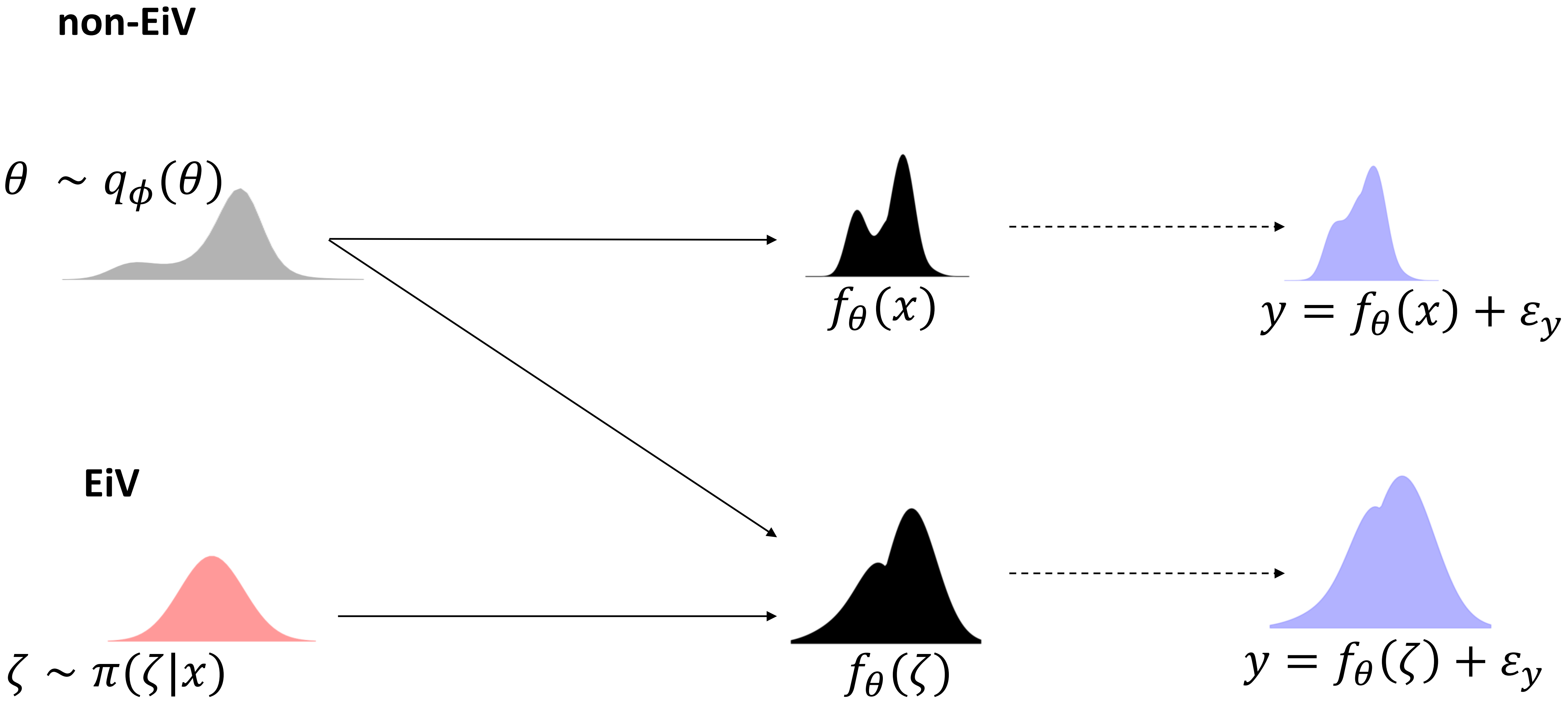
$$\pi(\zeta|x) = N\left(x, \left(1 + \frac{\sigma_x^2}{\sigma_\zeta^2}\right)^{-1} \sigma_x^2\right)$$

$q_\phi(\theta)$ again based on Dropout

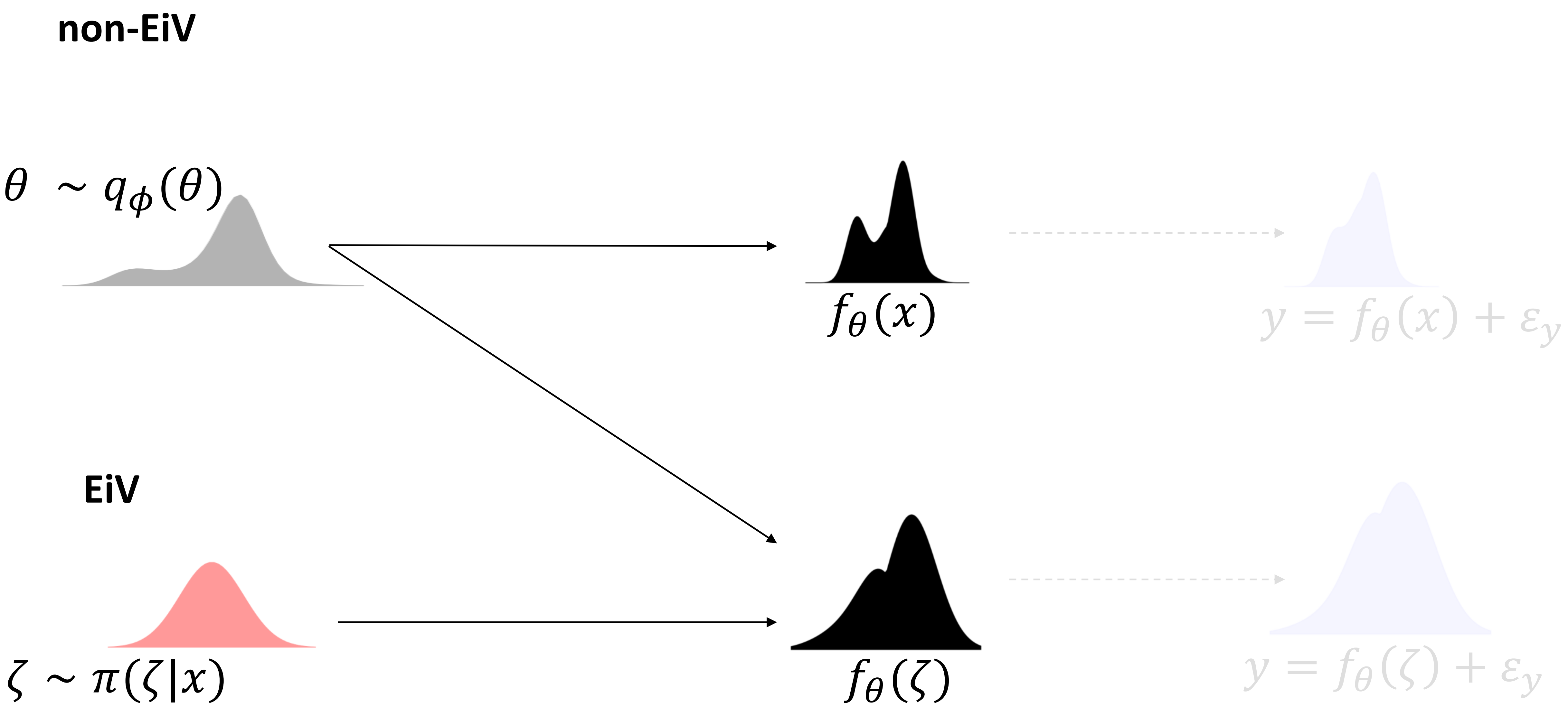
Kullback-Leibler loss

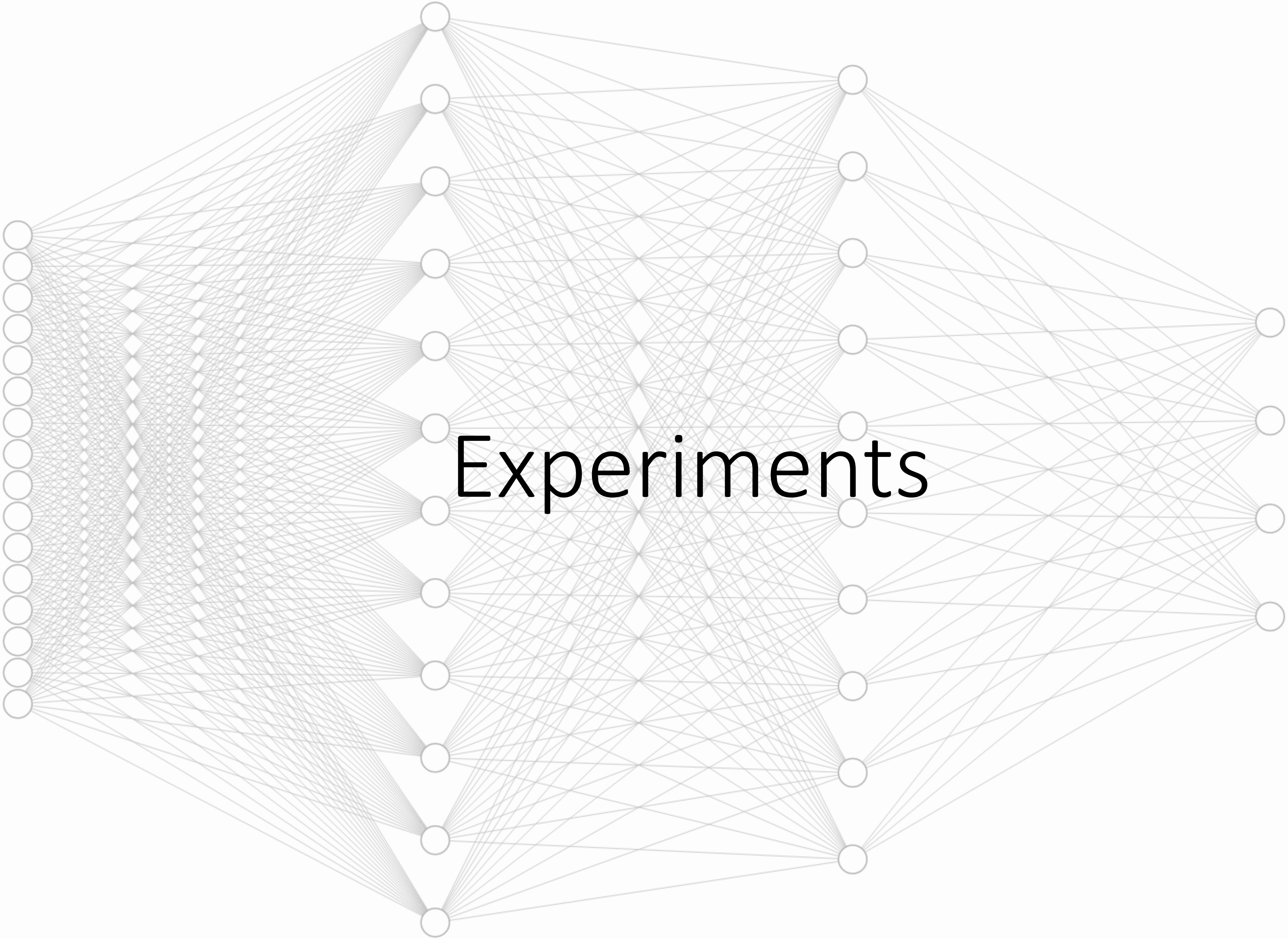
$$L(\phi\ ) = \sum_i \int d\theta q_\phi(\theta) \cdot \underbrace{\log\left[\int d\zeta_i\, \pi(\zeta_i|x_i) \cdot p(y_i|\zeta_i, \theta)\right]}_{\text{„L2 Loss“}} - \underbrace{D_{KL}(q_\phi(\theta)||\pi(\theta))}_{\text{„Regularization“}}$$

Uncertainty



Uncertainty

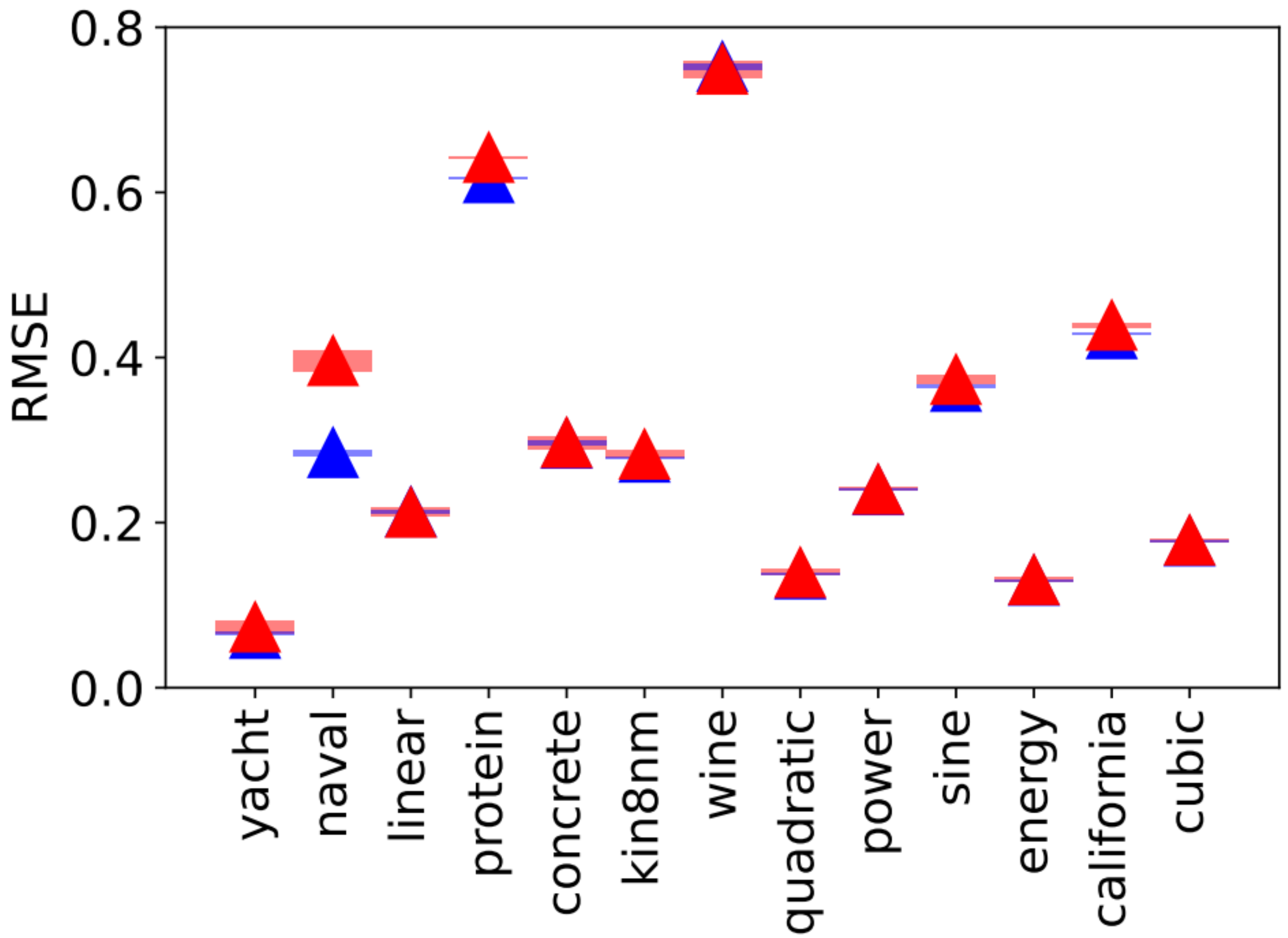




Experiments

Root-mean-squared-error (RMSE)

RMSE similar for both,
EiV and non-EiV



Simulated models give deeper insight

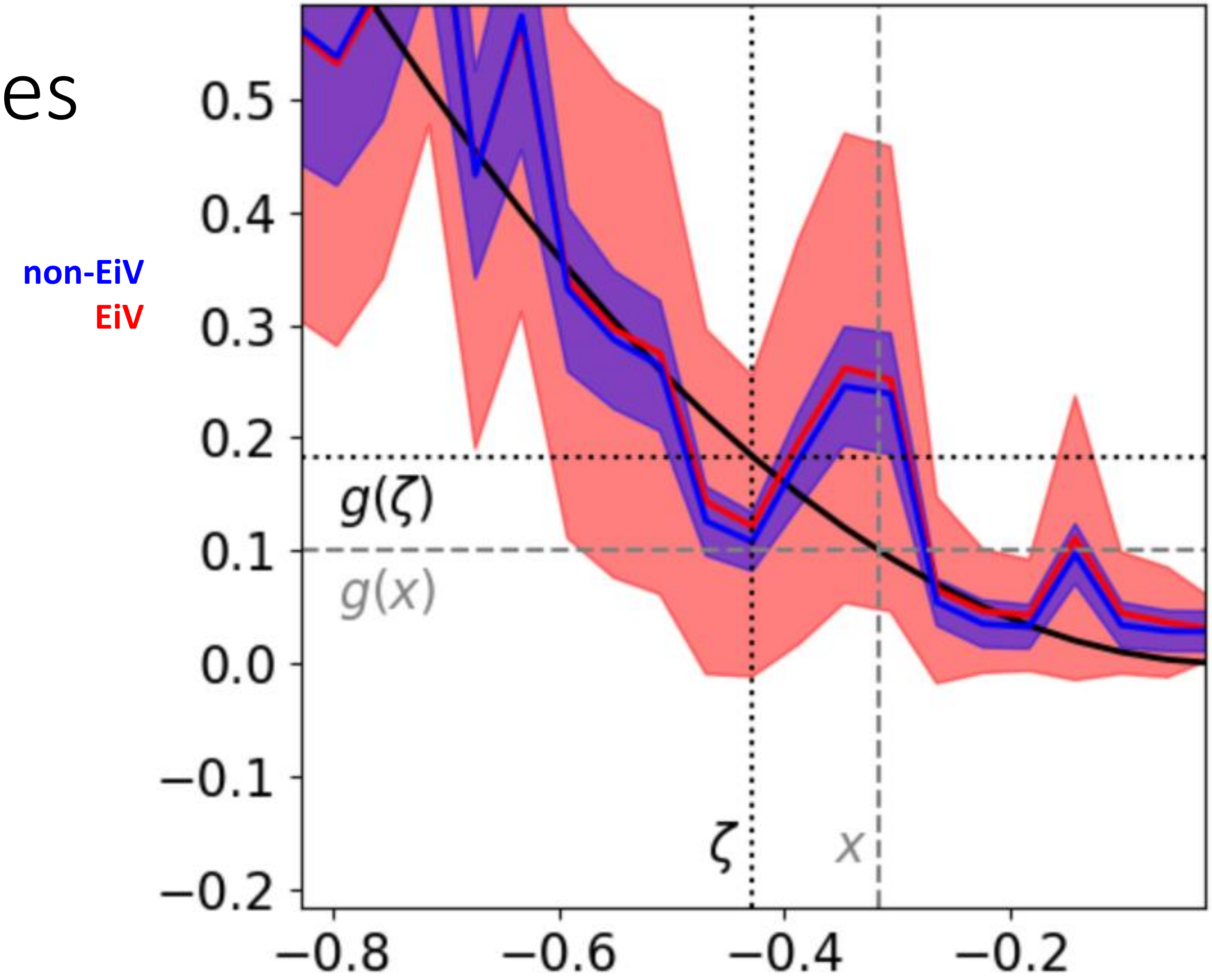
Check for examples with known ground truth (i.e. simulated examples)

$g: \zeta \mapsto g(\zeta)$ models ground truth

Generate training and testing data via

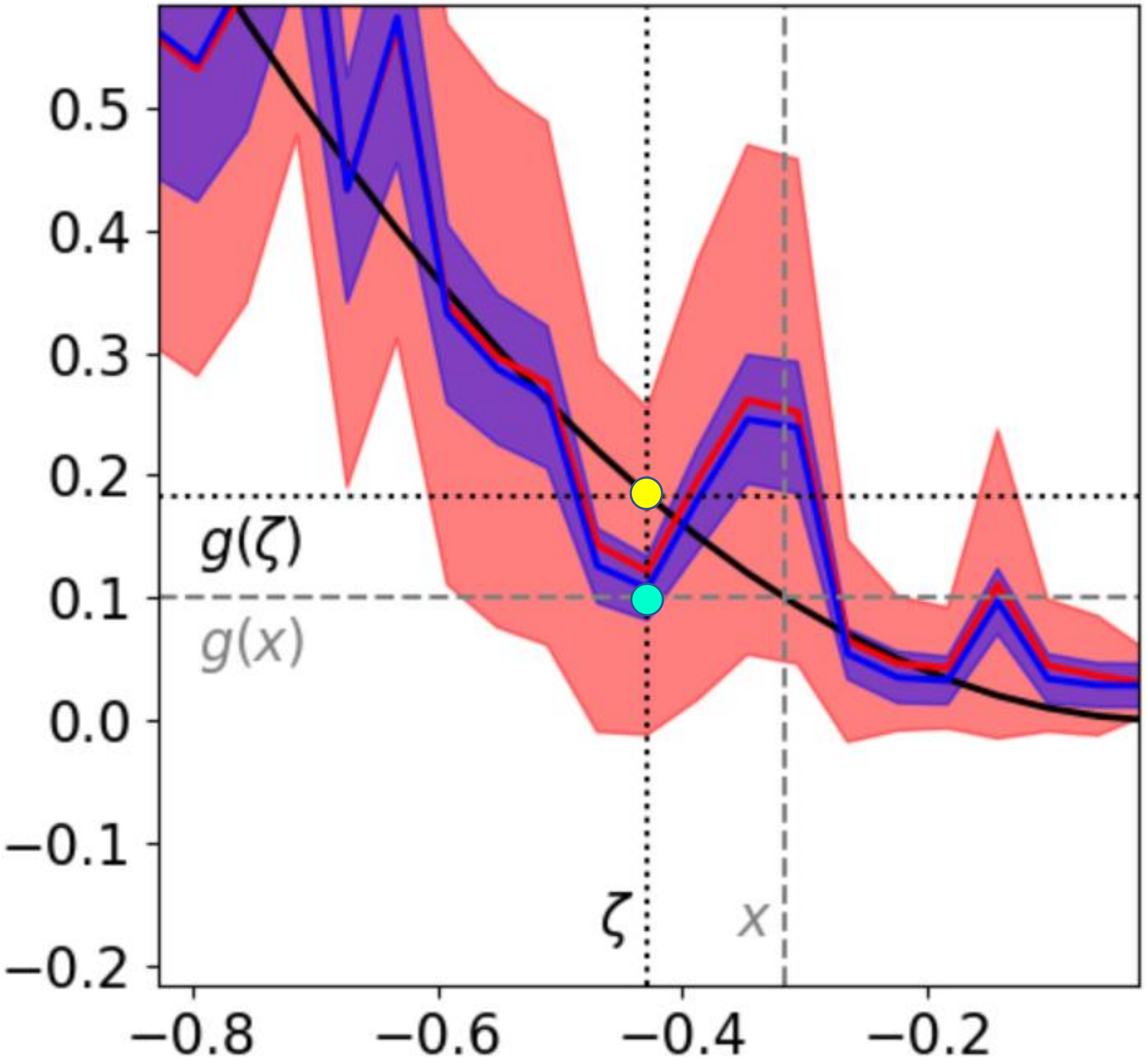
$$x = \zeta + \varepsilon_x$$
$$y = g(\zeta) + \varepsilon_y$$

Simulated examples



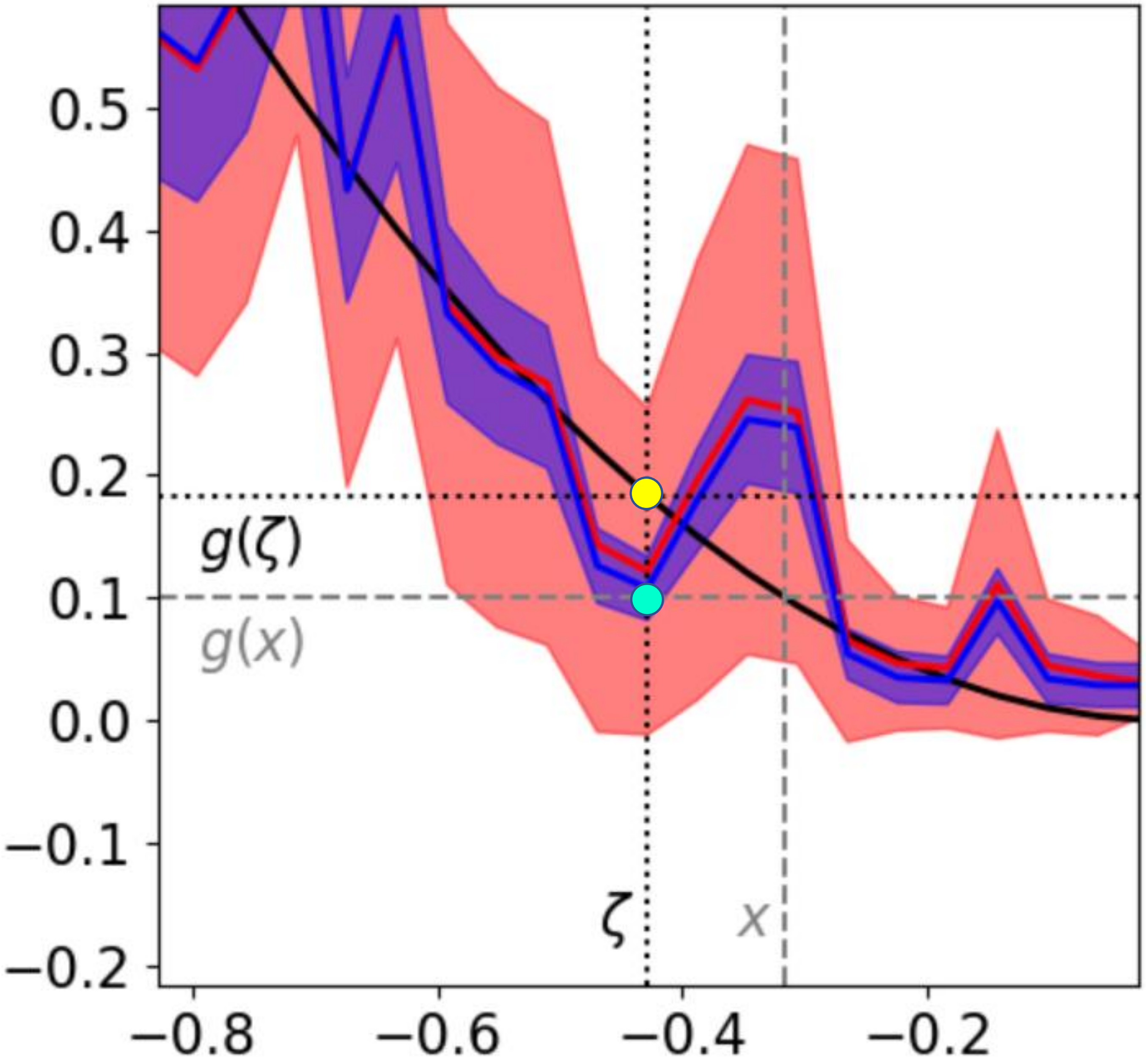
Simulated examples

non-EiV
EiV



Simulated examples

non-EiV
EiV



Predictions closer to $g(x)$ than to $g(\zeta)$

→ Error

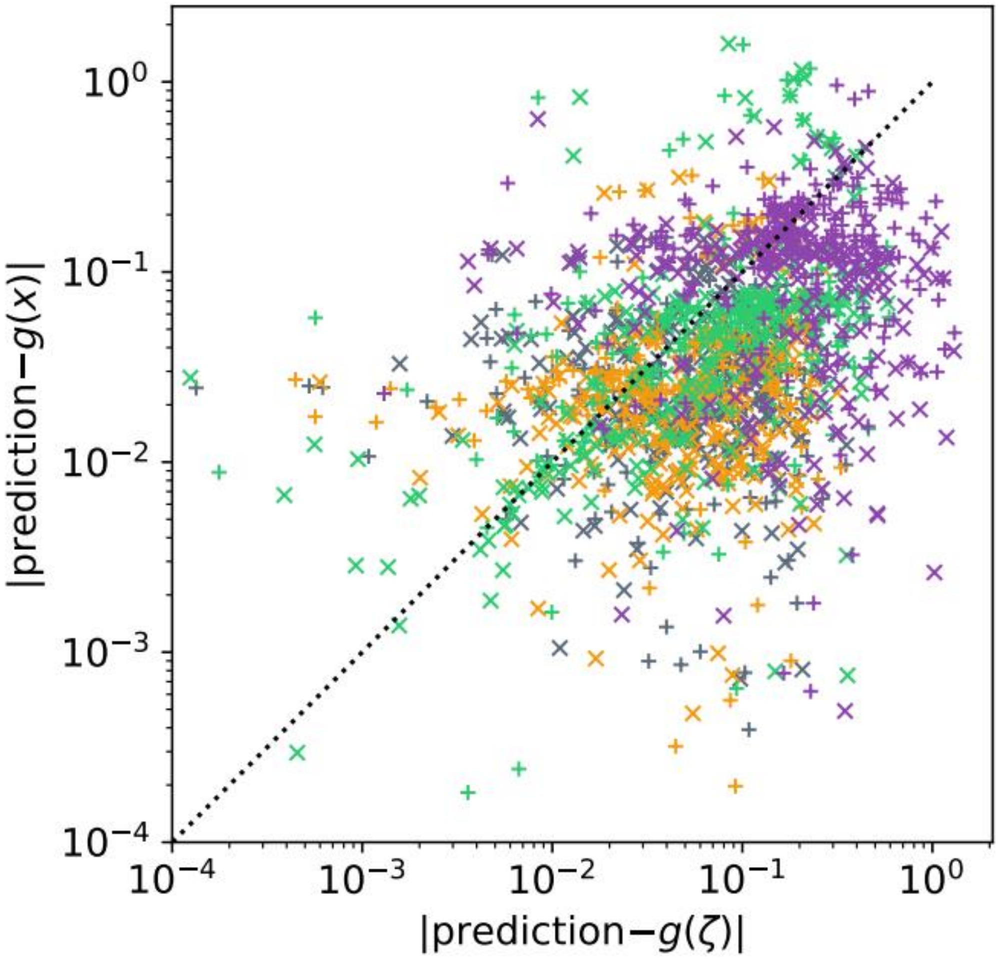
This behavior is systematic

Study on various simulated datasets

The behavior

$$|prediction\ - g(x)| < |prediction\ - g(\zeta)|$$

seems to be a typical behavior in problems with input noise.

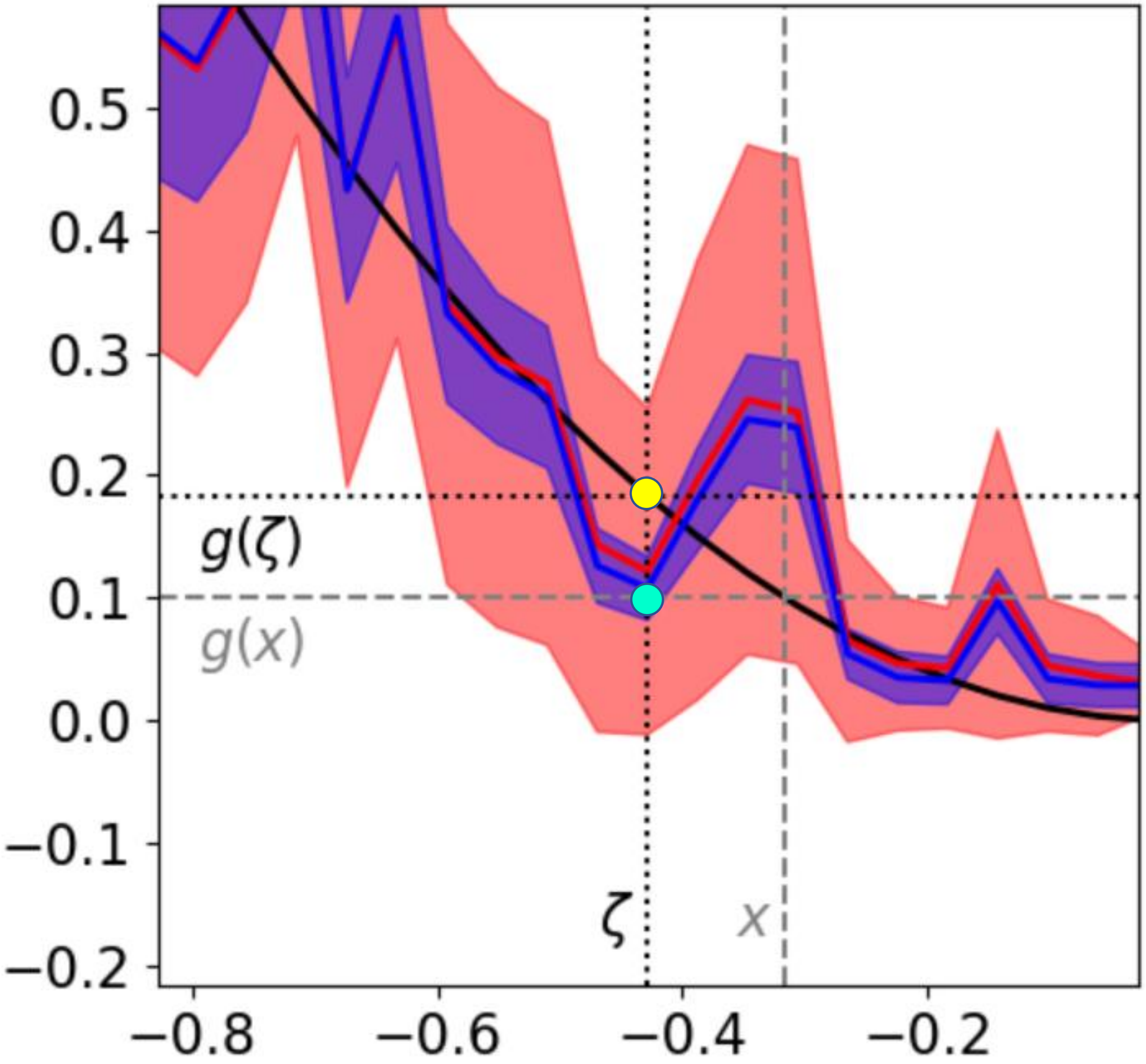


Simulated examples

non-EiV
EiV

Predictions closer to $g(x)$ than to $g(\zeta)$

→ Error



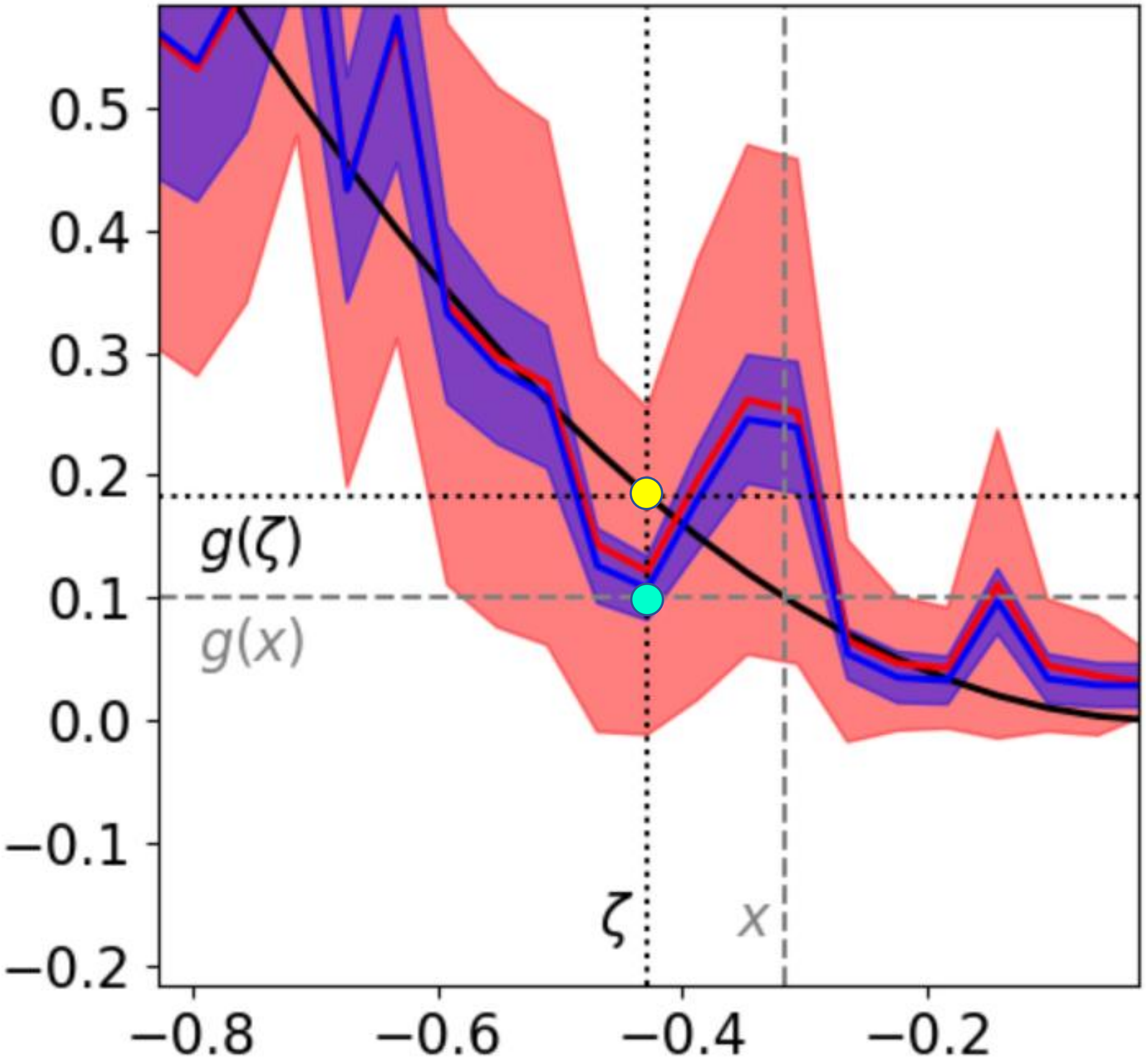
Simulated examples

non-EiV
EiV

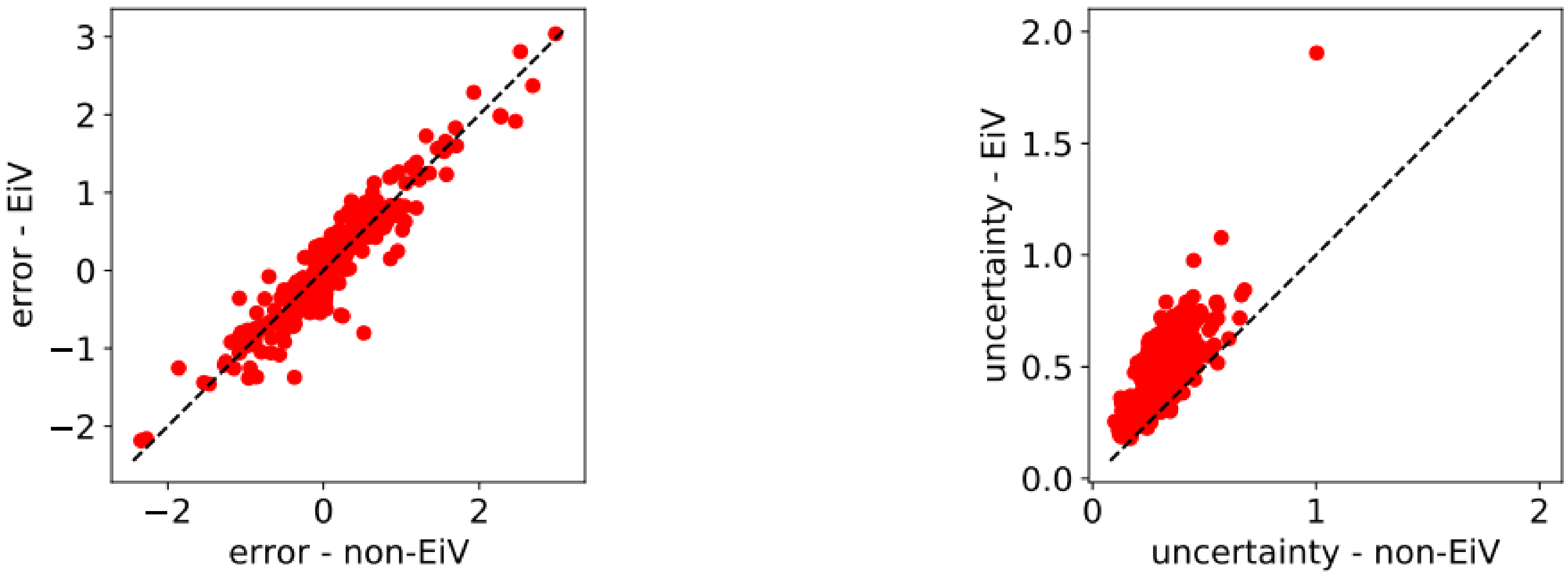
Predictions closer to $g(x)$ than to $g(\zeta)$

→ Error

→ better covered EiV uncertainty



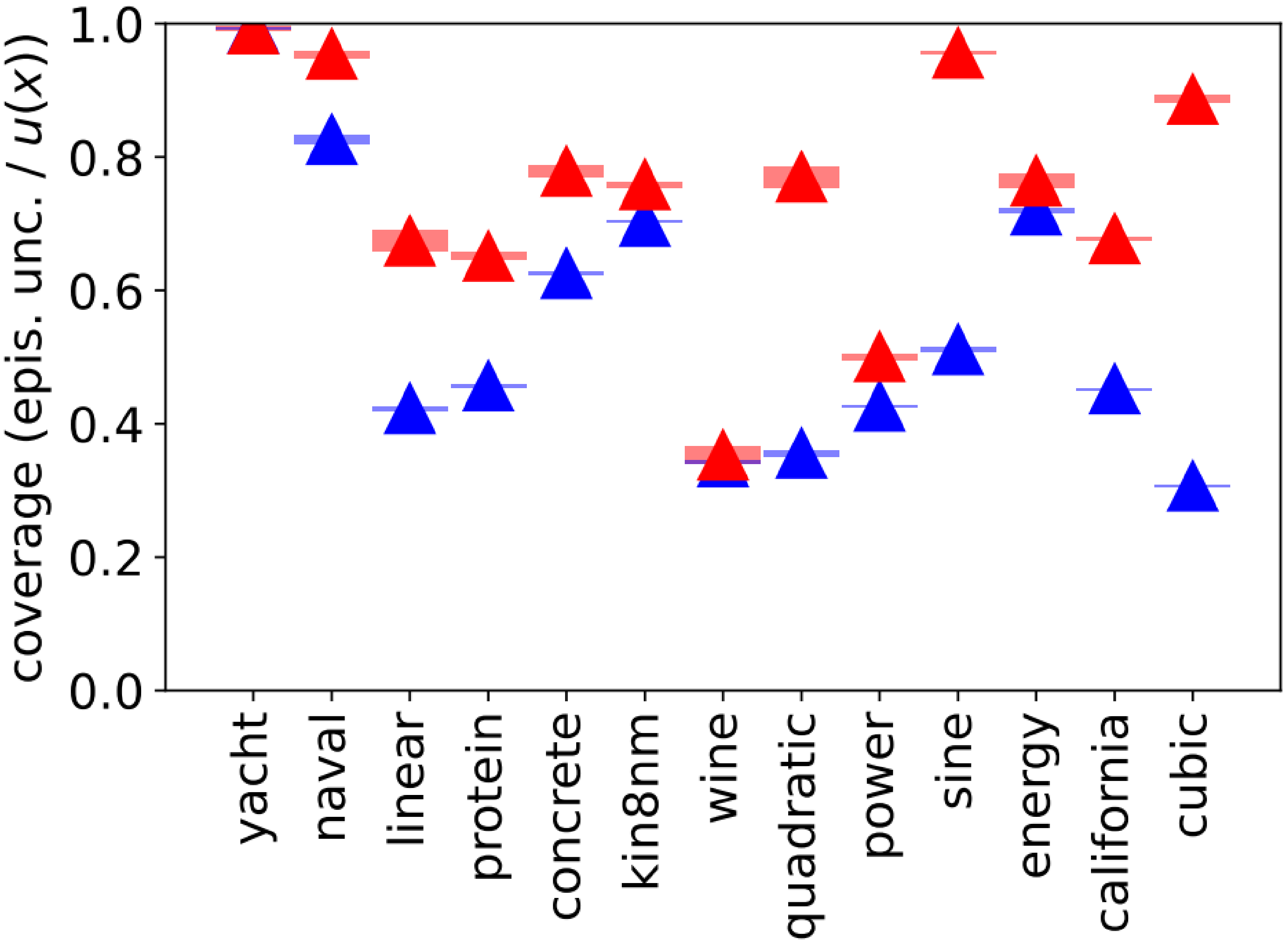
Example: wine quality dataset



Similar error, but *higher uncertainty*

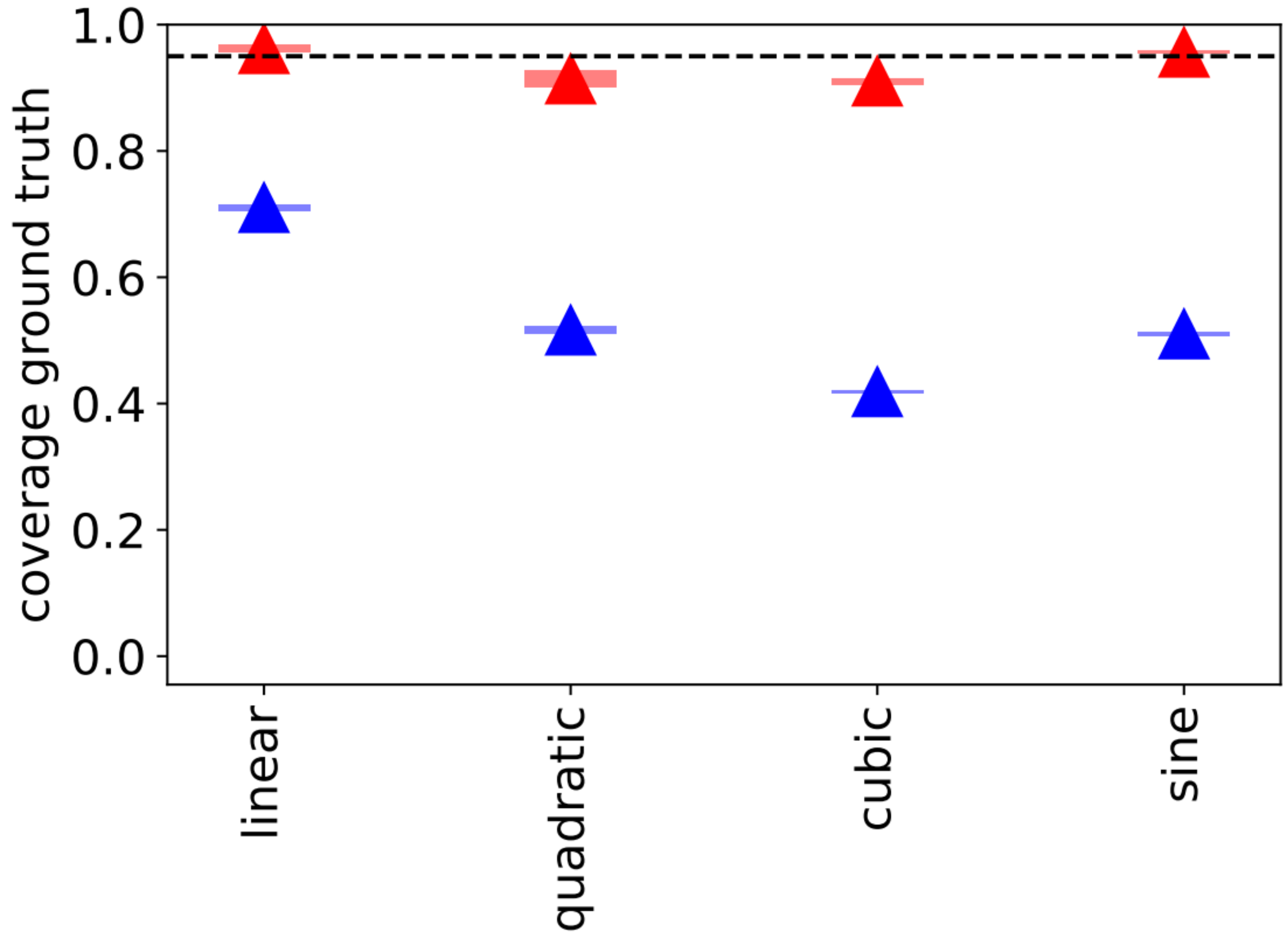
Coverage by uncertainty

Coverage substantially increased when considering EiV instead of non-EiV



Coverage: examples with known ground truth

Coverage of EiV model matches far better theoretical expectations



Conclusion

- EiV allows to account for the input uncertainty
- We propose an approach based on variational inference
- Our approach has the same prediction performance but increased uncertainties
- For examples with known ground truth:

  Increased uncertainty matches better theoretical expectations

Talk **based on the article**

Aleatoric uncertainty for Errors-in-Variables models in deep regression. Jörg Martin and Clemens Elster. To appear in *Neural Processing Letters.*

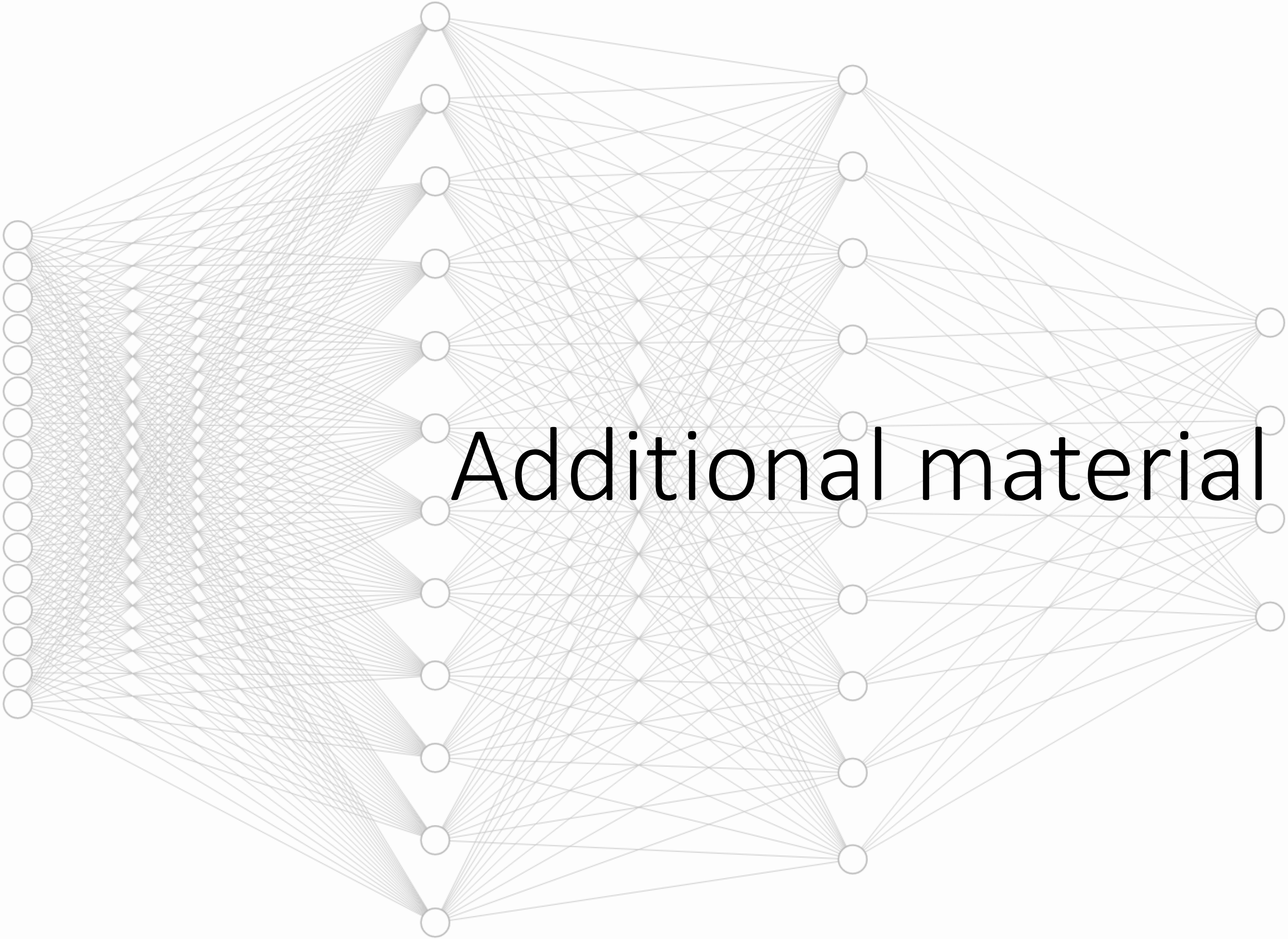
*There is also a (simplified) preprint available (**different title**!)*

Errors-in-Variables for deep learning: rethinking aleatoric uncertainty. Jörg Martin and Clemens Elster. arXiv preprint arXiv:2105.09095, 2021.

Other references

- *The errors-in-variables cost function for learning neural networks with noisy inputs*. J. Van Gorp, J. Schoukens, and R. Pintelon. Intelligent Engineering through Artificial Neural Networks, 8:141–146, 1998.
- *Bayesian uncertainty calculation in neural network inference of ion and electron temperature profiles at w7-x*. Pavone, J. Svensson, A. Langenberg, N. Pablant, U. Hoefel, S. Kwak, R. Wolf, and W. .-X. Team. Review of Scientific Instruments, 89(10):10K102, 2018.
- *Input modeling and uncertainty quantification for improving volatile residential load forecasting*. G. Xie, X. Chen, and Y. Weng. Energy, 211:119007, 2020.
- *Dropout as a bayesian approximation: Representing model uncertainty in deep learning*. Y. Gal, and Z. Ghahramani. *International conference on machine learning*. PMLR, 2016.

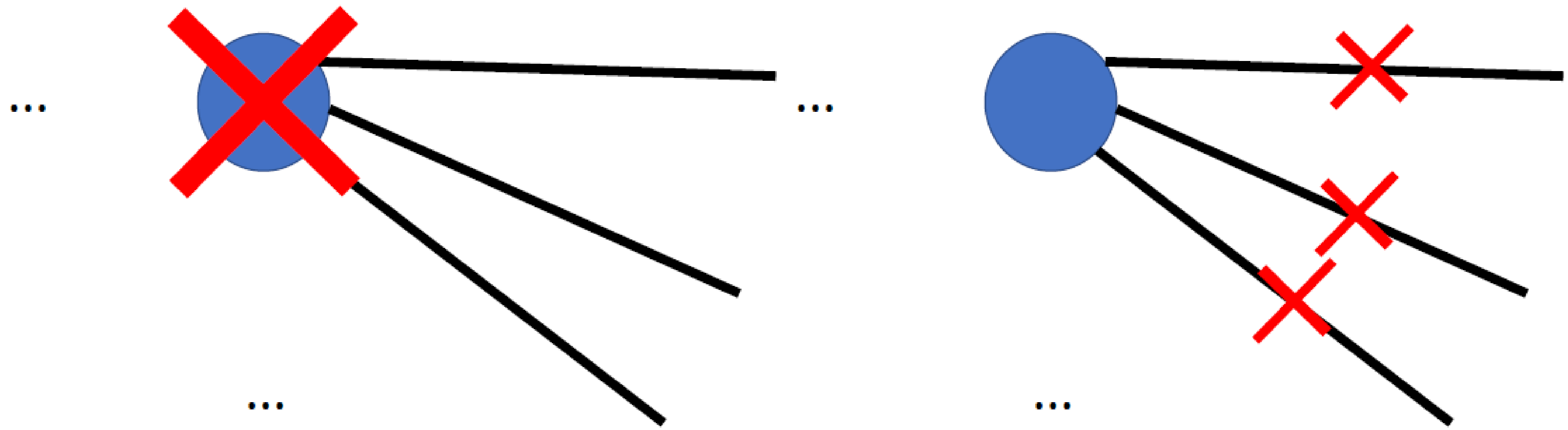
Thank you for your attention!



Additional material

Dropout and variational inference

The following are equivalent



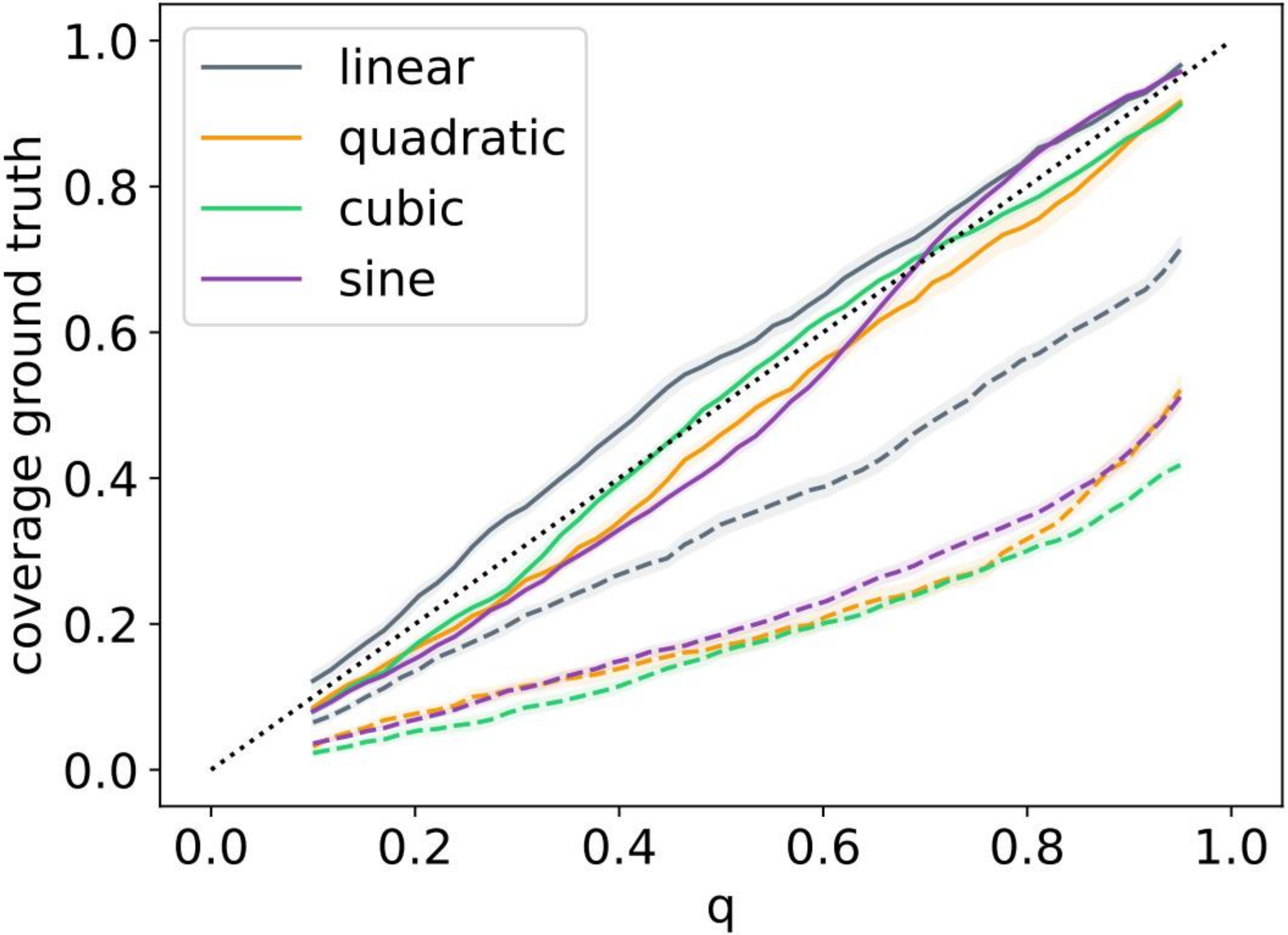
⇒ Dropout can be read as randomly dropping **groups of weights**

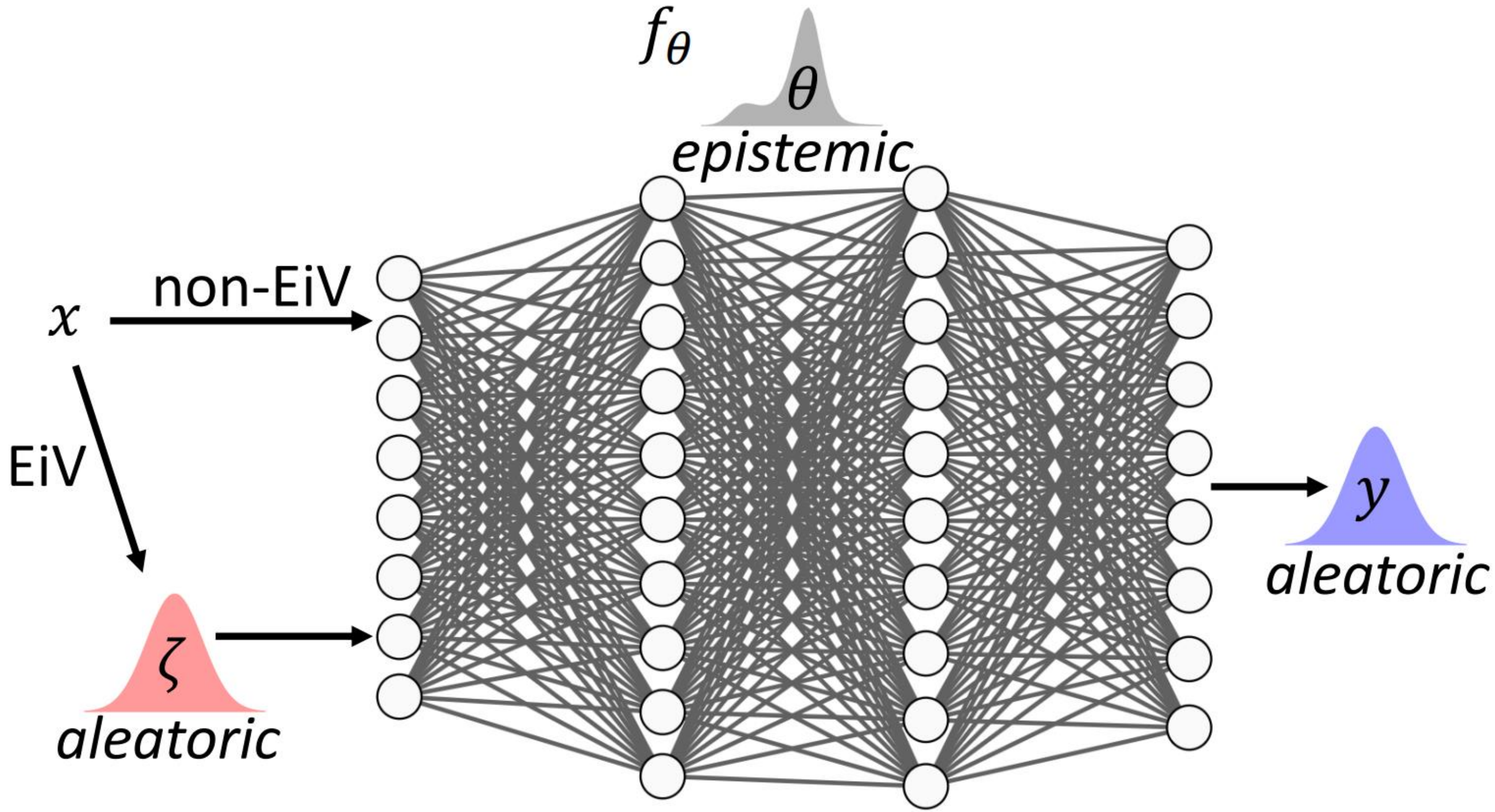
M.C.-Loss fucntion from article

is measured via the Kullback-Leibler divergence $D_{\mathrm{KL}}(q_\phi(\theta)\|\pi(\theta|\mathcal{D},\boldsymbol{\sigma}^2))$. As we argue in Appendix A, to find a ϕ that minimizes this divergence, we can use backpropagation on the following loss function

$$\mathcal{L}^{\mathrm{M.C.}}(\phi) := -\frac{1}{M}\sum_{m=1}^{M}\log\left(\frac{1}{L}\sum_{l=1}^{L}p(y_{i_m}|\theta_m,\zeta_{i_m,l},\sigma_y^2)\right)+\frac{1}{N}D_{\mathrm{KL}}(q_\phi(\theta)\|\pi(\theta)) \tag{9}$$

where the minibatches $\{(x_{i_1},y_{i_1}),\ldots,(x_{i_M},y_{i_M}))\}$, the M samples $\theta_m \sim q_\phi(\theta_m)$ and $M\cdot L$ samples $\zeta_{i_m,l}\sim\pi(\zeta_{i_m,l}|x_i,\sigma_x^2 I_{n_x\times n_x})$ are re-drawn in each optimization step. We used throughout this work $M=1$ for training, $M=100$ for evaluation and $L=5$ for both, training and evaluation.


linear
quadratic
cubic
sine
coverage ground truth
q
0.0
0.2
0.4
0.6
0.8
1.0



f_θ
θ
epistemic
non-EiV
x
EiV
ζ
aleatoric
y
aleatoric