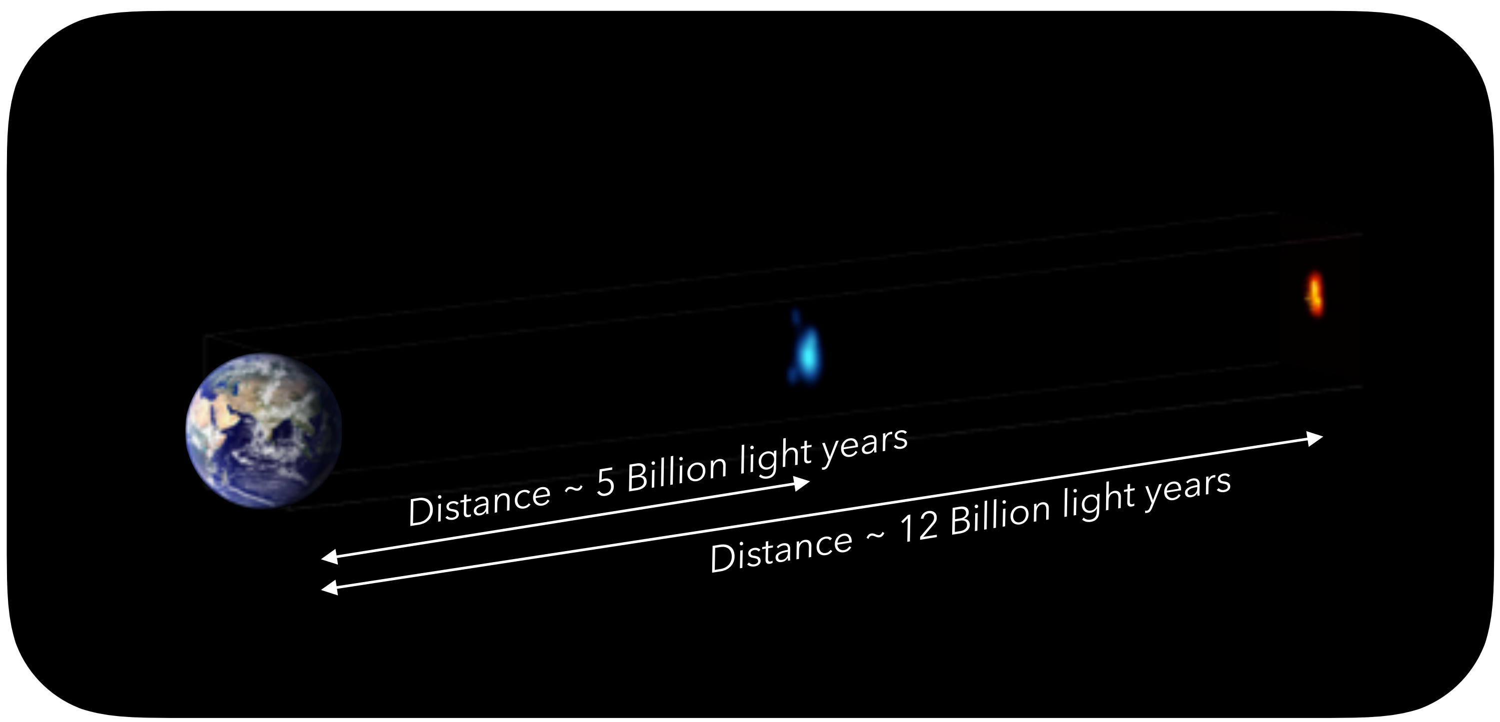
ESTIMATING THE UNCERTAINTIES OF NEURAL NETWORK PREDICTIONS

Laurence Perreault Levasseur
University of Montréal

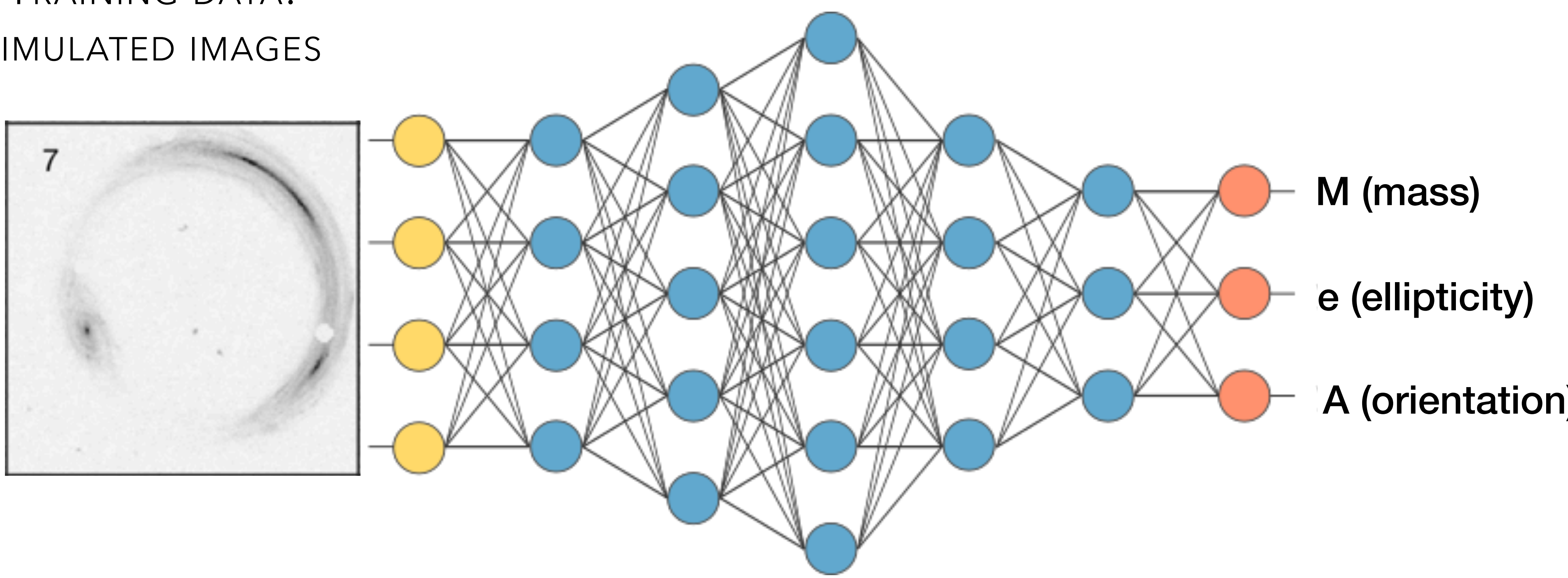
STRONG GRAVITATIONAL LENSING

Formation of **multiple images** of a single distant object due to the **deflection of its light** by the **gravity** of intervening structures.



TRAINING

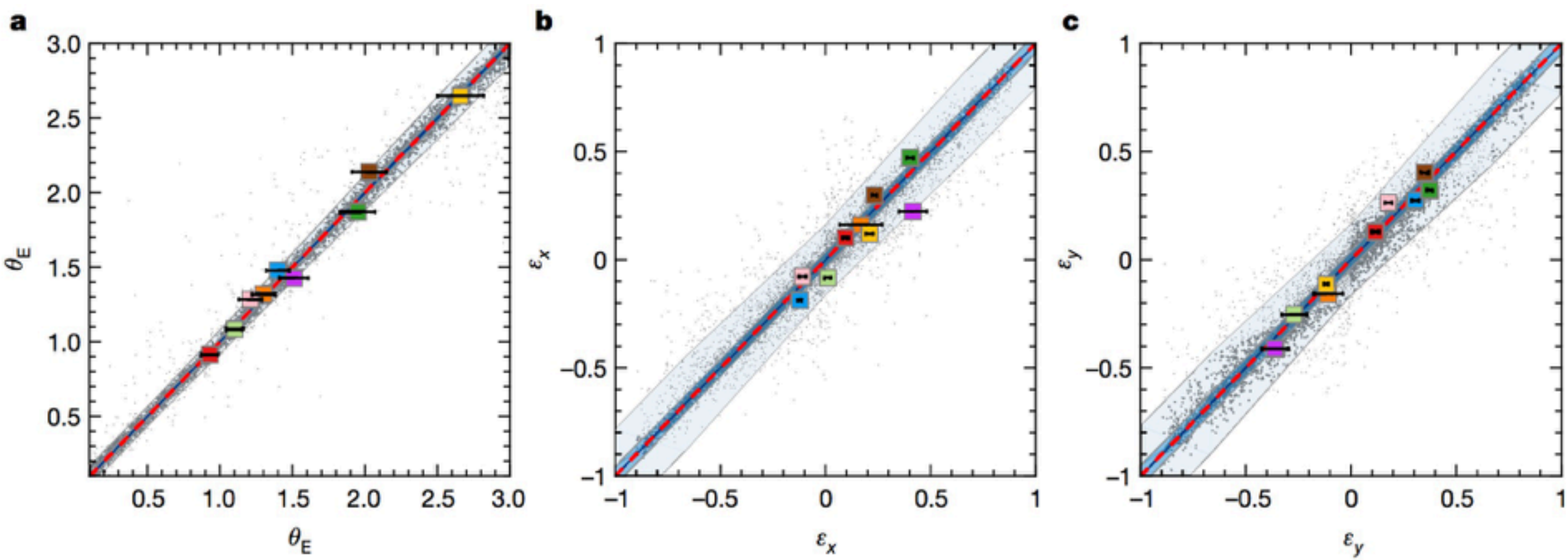
TRAINING DATA:
SIMULATED IMAGES



LOSS: MEAN SQUARE ERROR LOSS

$$MSE = \frac{\sum_{i=1}^{n} (y_i - y_i^p)^2}{n}$$

ESTIMATING THE MATTER DISTRIBUTION PARAMETERS WITH CNNS

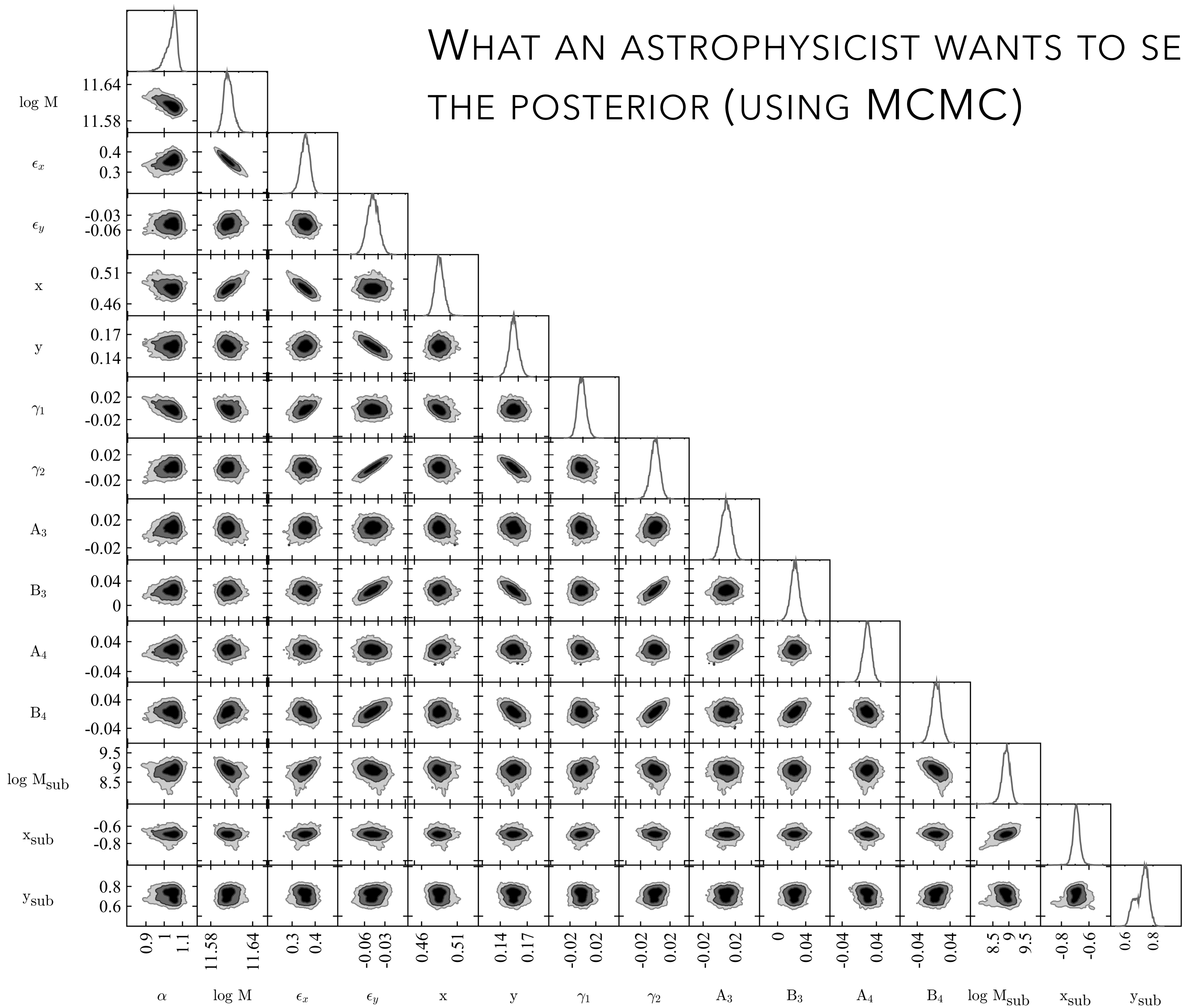


10 million times faster than ML lens modeling.

0.01 seconds on a **single GPU**

Hezaveh, Perreault Levasseur, Marshall, Nature, 2017

WHAT AN ASTROPHYSICIST WANTS TO SEE: THE POSTERIOR (USING MCMC)



Hezaveh, et al. ApJ 2016

What are the uncertainties of the output parameters?

Sources of errors in the predictions:

1 - **Aleatoric**.

Inherent corruptions to the input data: noise, PSF blurring, etc.

2- **Epistemic**.

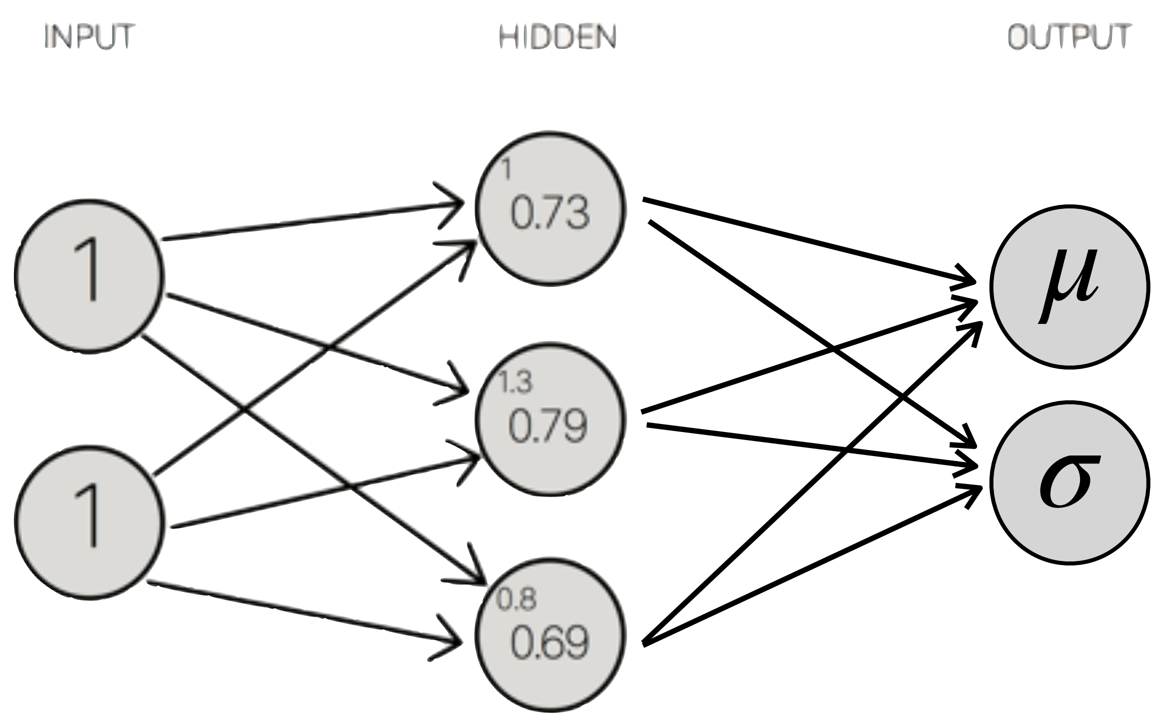
Errors made by the networks: these could be due to insufficient training, network architecture, etc.

WHAT IS THE LOG-LIKELIHOOD OF THE NETWORK OUTPUT, $\mathcal{L}(\mathbf{y}_n, \hat{\mathbf{y}}_n(\mathbf{x}_n, \omega))$?

We approximate the likelihood with an analytic distribution.

Assume Gaussian:

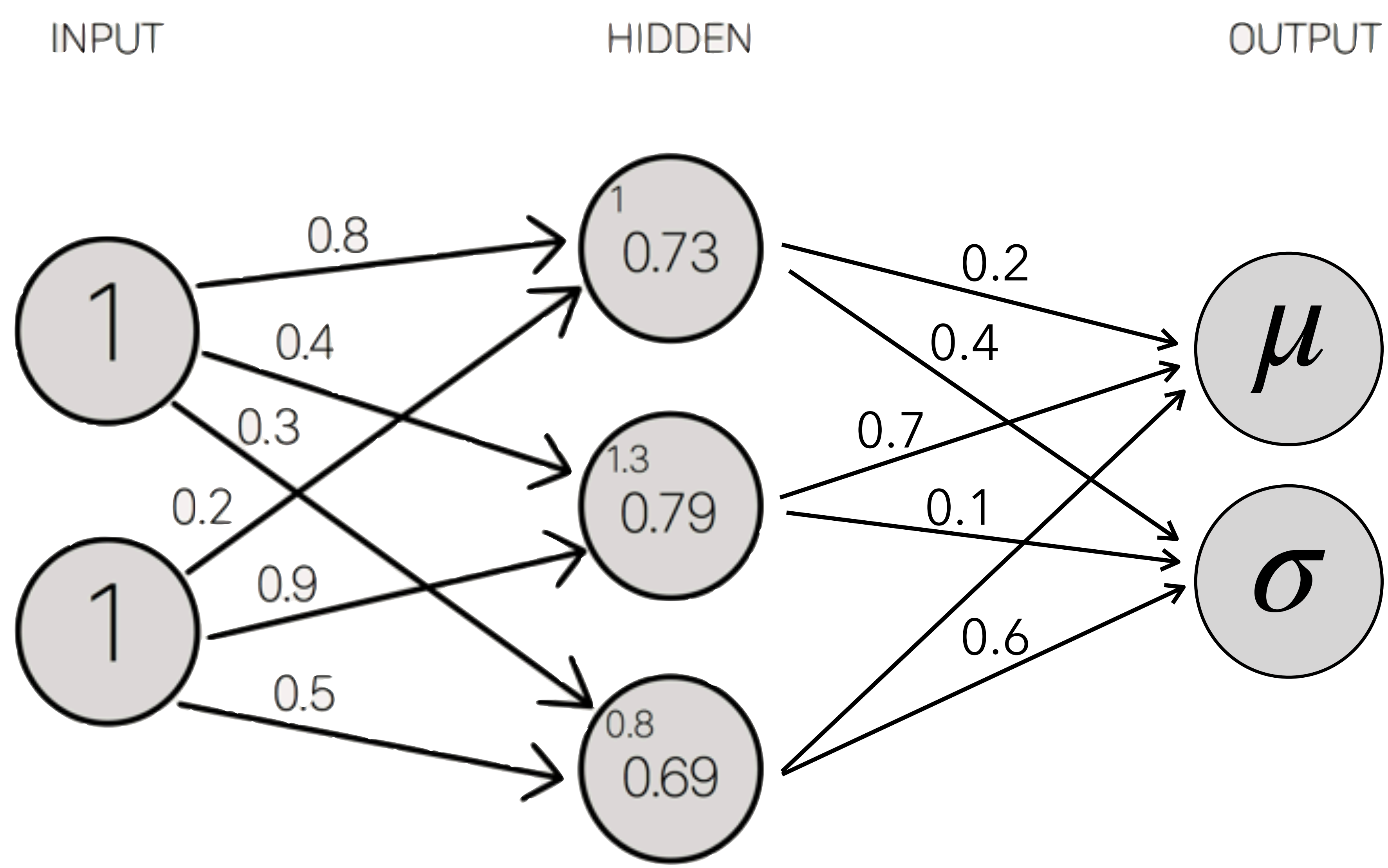
$$\mathcal{L}(\mathbf{y}_n, \hat{\mathbf{y}}_n(\mathbf{x}_n, \omega)) \propto \sum_k \frac{-1}{2\sigma_k^2} ||y_{n,k} - \hat{y}_{n,k}(\mathbf{x}_n, \omega)||^2 - \frac{1}{2} \log \sigma_k^2$$



Epistemic Uncertainties

Standard Neural Networks:

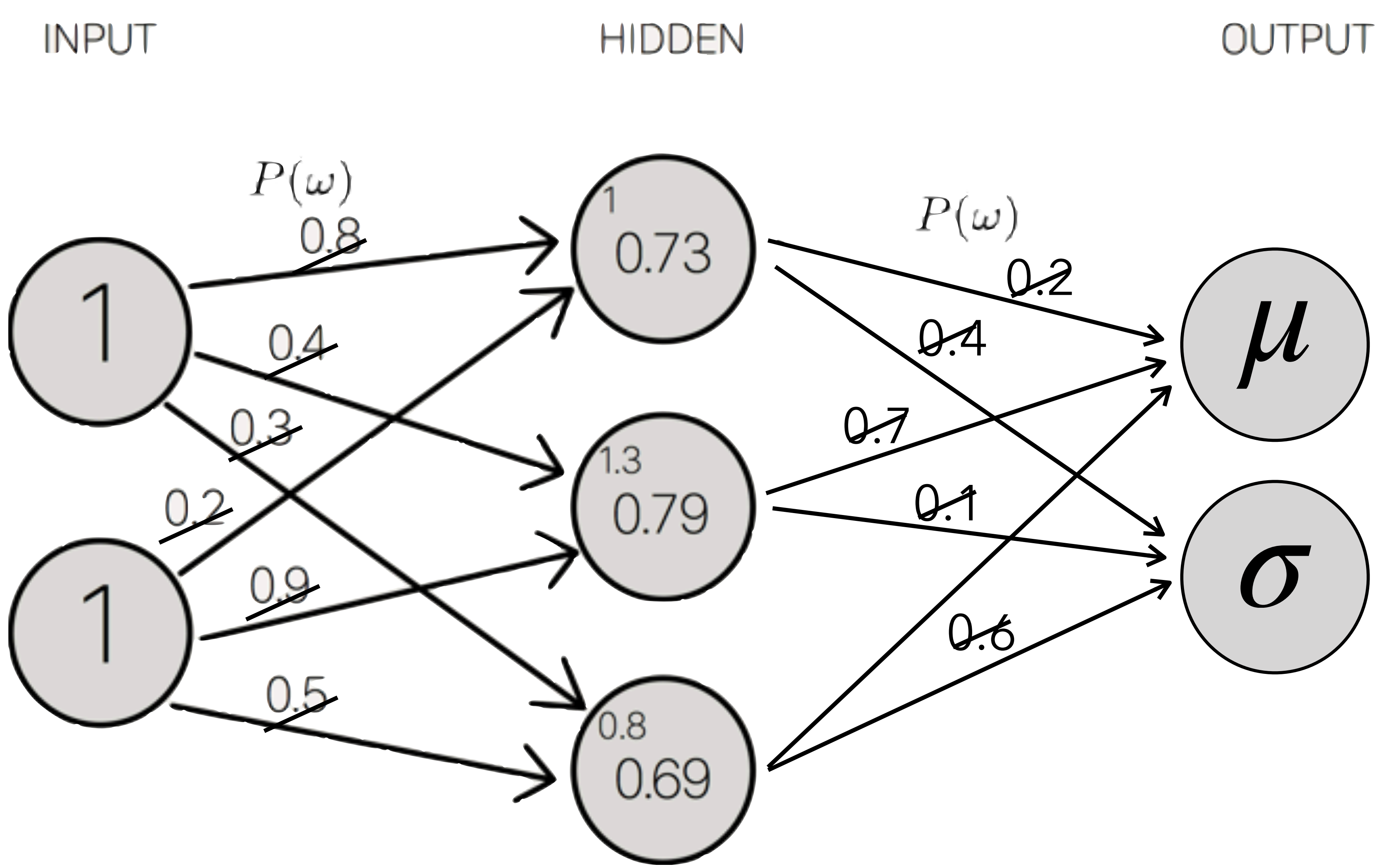
Weight have fixed, deterministic values



Epistemic Uncertainties

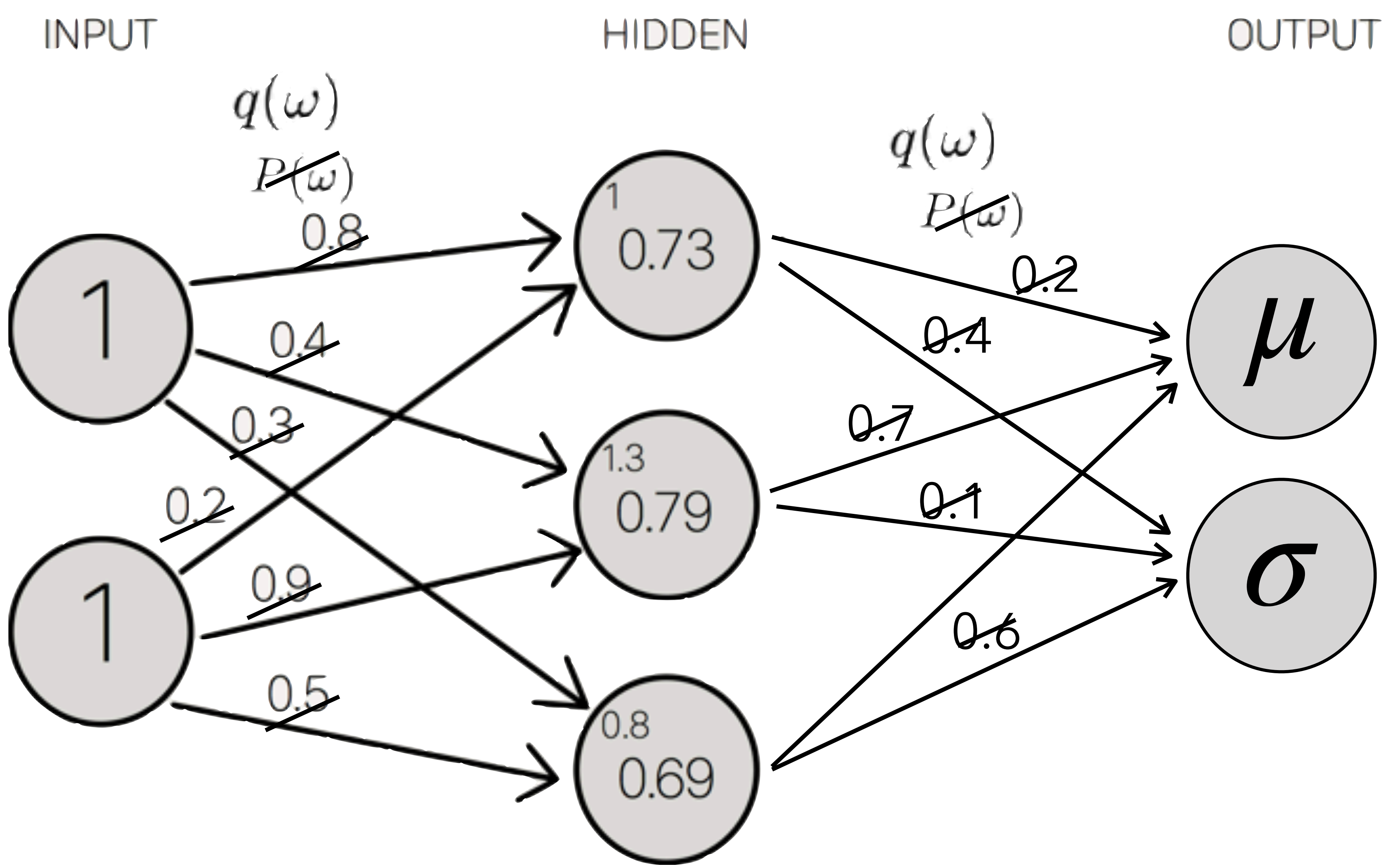
Bayesian Neural Networks:

Instead of fix values, weights are defined by probability distributions



Variational Inference

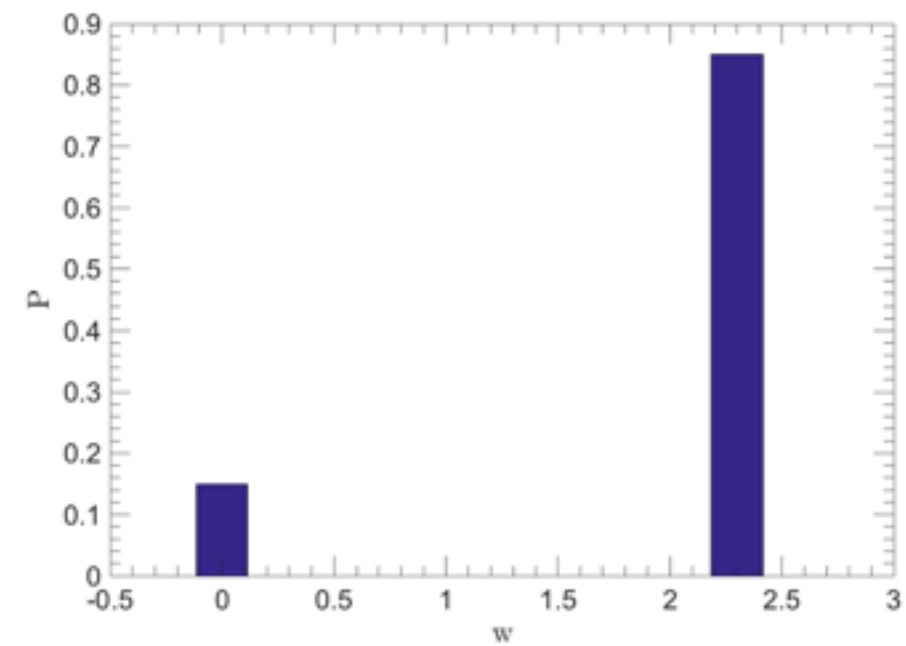
Replace $P(\omega)$ by a distribution with a simple analytic form, $q(\omega)$, (e.g., a Gaussian).



Easiest Variational Distribution

Bernoulli distribution:
Every weight is either zero (with probability p) or some value, w, (with probability $1-p$)

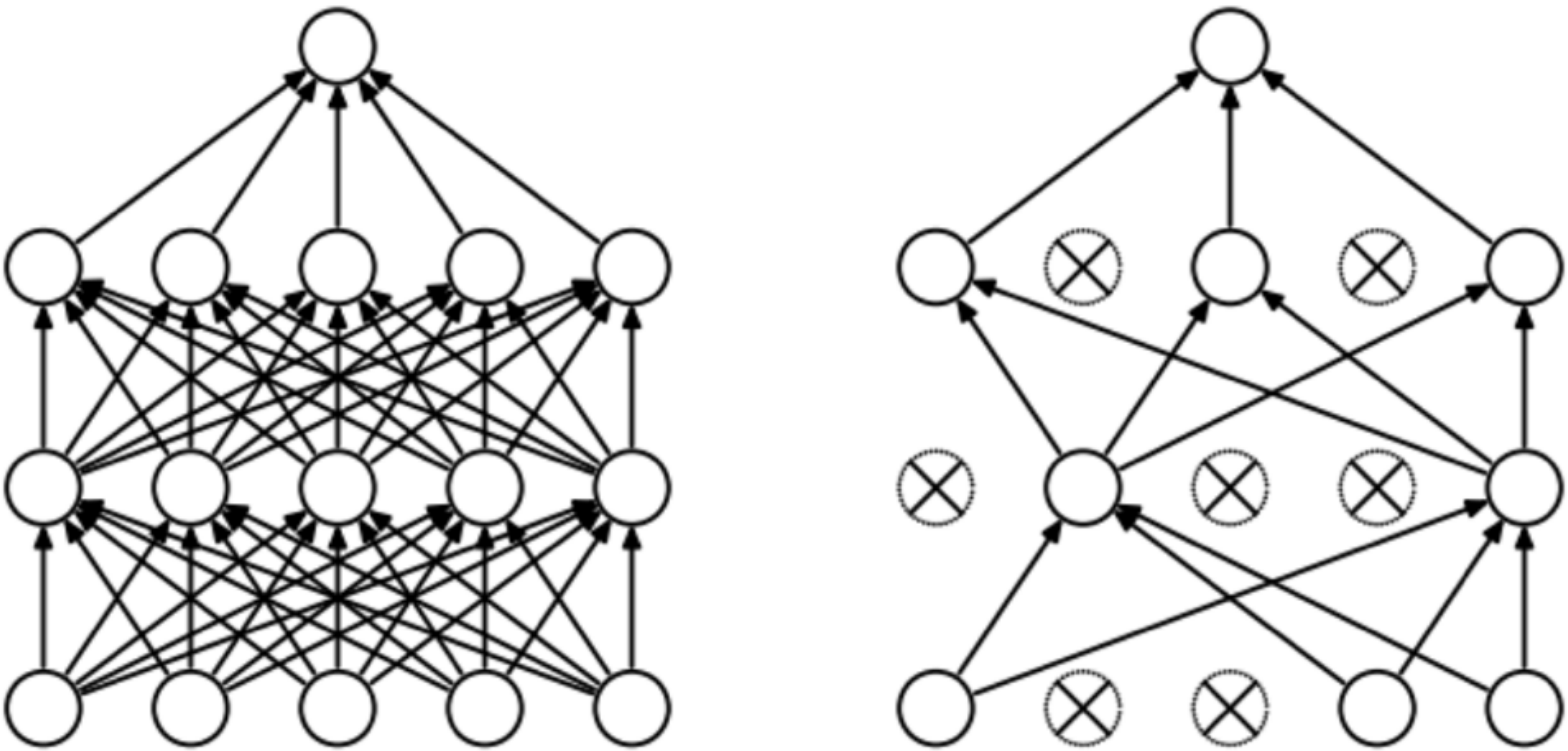
The variational parameter is the value of the weight when it is not zero.



That is DROPOUT

Gal & Ghahraman ICML, 2015

Dropout



(a) Standard Neural Net (b) After applying dropout.

RECAP

1- PLACE DROPOUT BEFORE EVERY WEIGHT LAYER.

2- TRAIN WITH DROPOUT, OPTIMIZING THE LOG LIKELIHOOD

$$\mathcal{L}(\mathbf{y}_n, \hat{\mathbf{y}}_n(\mathbf{x}_n, \omega)) \propto \sum_k \frac{-1}{2\sigma_k^2} ||y_{n,k} - \hat{y}_{n,k}(\mathbf{x}_n, \omega)||^2 - \frac{1}{2} \log \sigma_k^2$$

3- AT TEST TIME, KEEP DROPOUT ON. PERFORM MONTE CARLO DROPOUT: INPUT THE DATA MULTIPLE TIMES, PERFORM DROPOUT AND COLLECT THE OUTPUTS.

4- ADD THE PREDICTED UNCERTAINTY (THE SIGMA ABOVE) TO THE SAMPLE.

Example

Uncertainties on the magnification of lenses

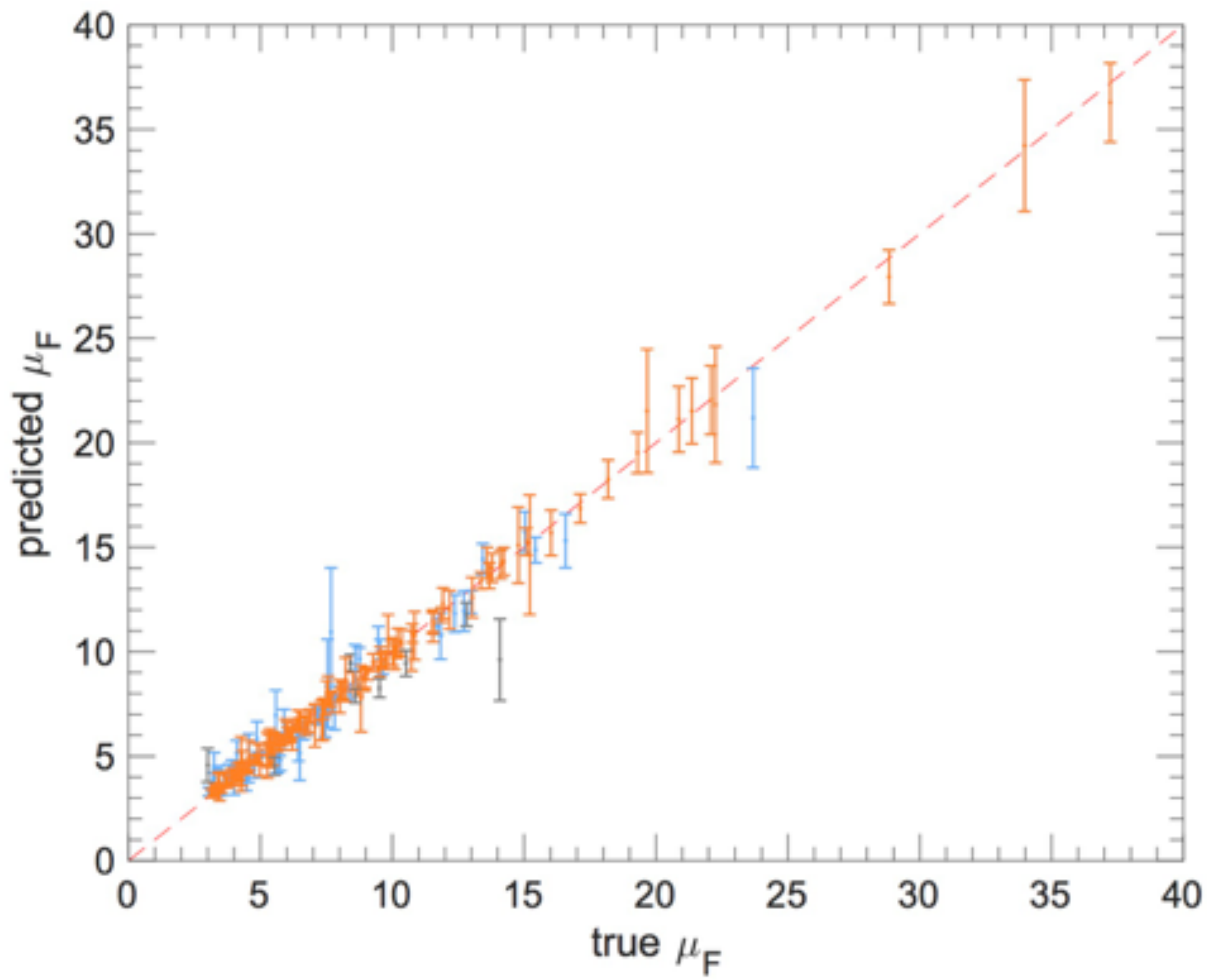
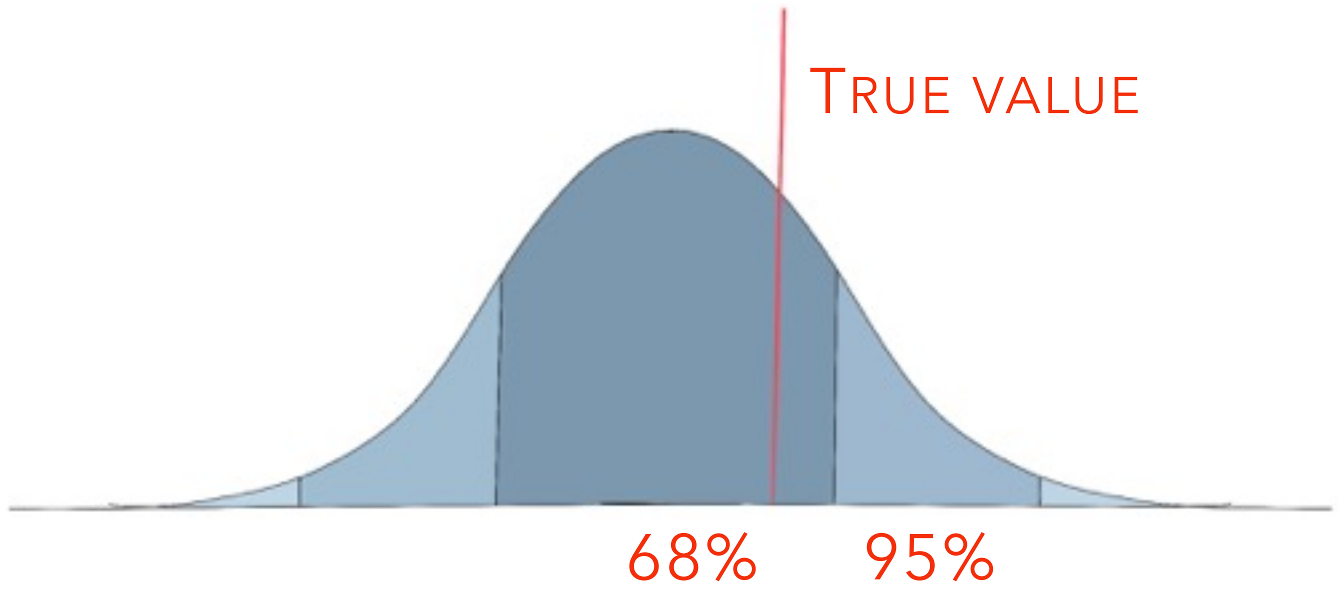


Figure 1. Predicted 68.3% uncertainties for lensing flux magnification, μ_F, as a function of the true value of this parameter. The orange, blue, and black correspond to examples where the true values fall within the 68.3, 95.5, and 99.7% confidence intervals respectively.

Perreault Levasseur, Hezaveh, Wechsler, 2017

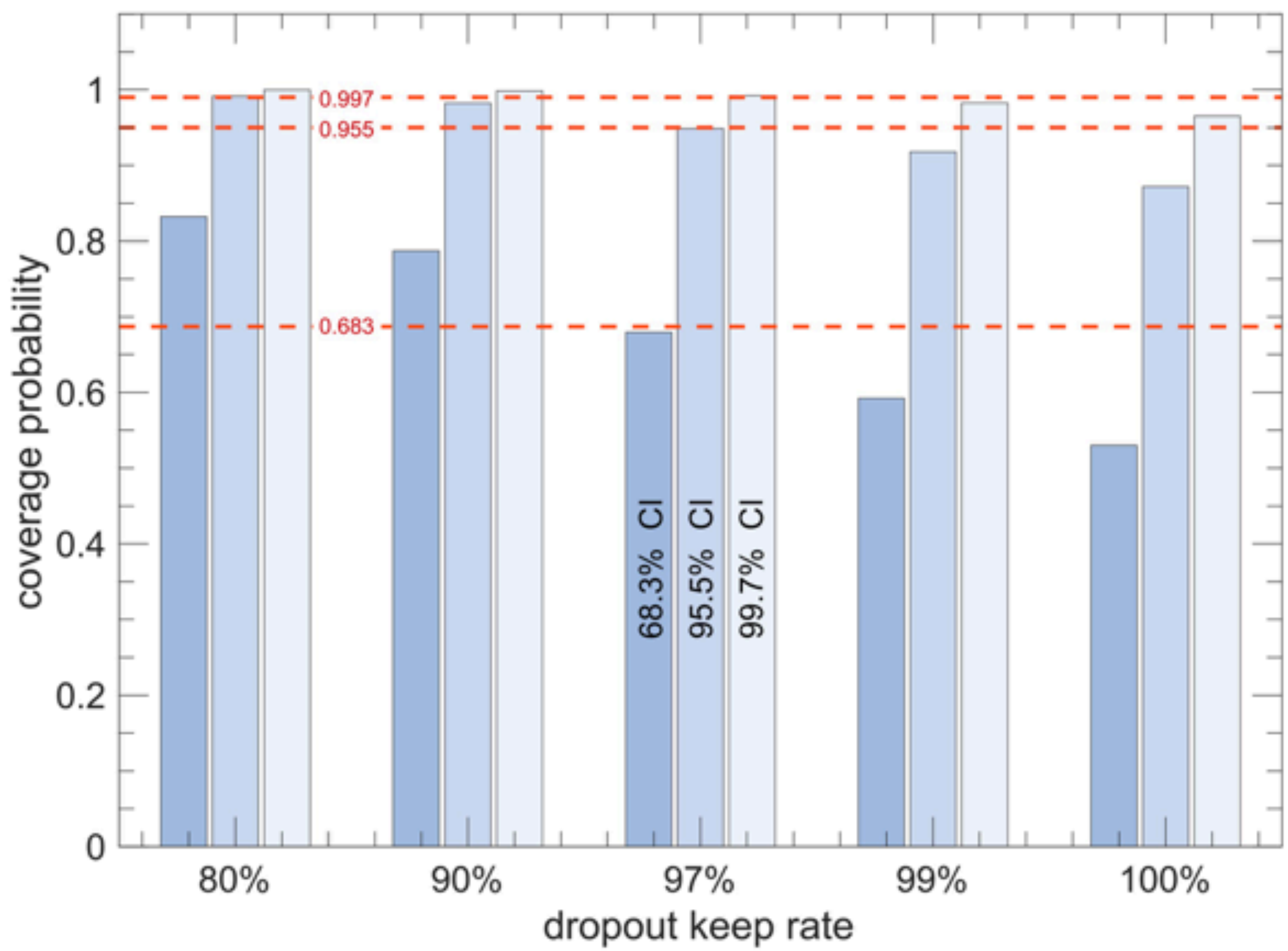
ARE THESE UNCERTAINTIES ACCURATE?

COVERAGE PROBABILITY OF A CONFIDENCE INTERVAL IS THE PROPORTION OF THE TIME THAT THE INTERVAL CONTAINS THE TRUE VALUE OF INTEREST.



FOR AN ACCURATE INTERVAL ESTIMATOR, THE COVERAGE PROBABILITY IS EQUAL TO ITS CONFIDENCE LEVEL

COVERAGE PROBABILITIES



Perreault Levasseur, Hezaveh, Wechsler, 2017

Pros and Cons

- Requires a few hundred forward passes at evaluation time (to collect samples). Still very fast.
- If needed, one could perform importance sampling of the output distribution to get an unbiased posterior.
- Unable to model complex output distributions (e.g., multimodal distributions).

DEALING WITH MORE COMPLEX DISTRIBUTIONS: QUANTILE REGRESSION

$$\mathcal{L}(\xi_i|\alpha) = \begin{cases} \alpha\xi_i & \text{if}\, \xi_i \geq 0, \\ (\alpha-1)\xi_i & \text{if}\, \xi_i < 0. \end{cases}$$

$$\xi_i = y_i - f(\mathbf{x}_i)$$

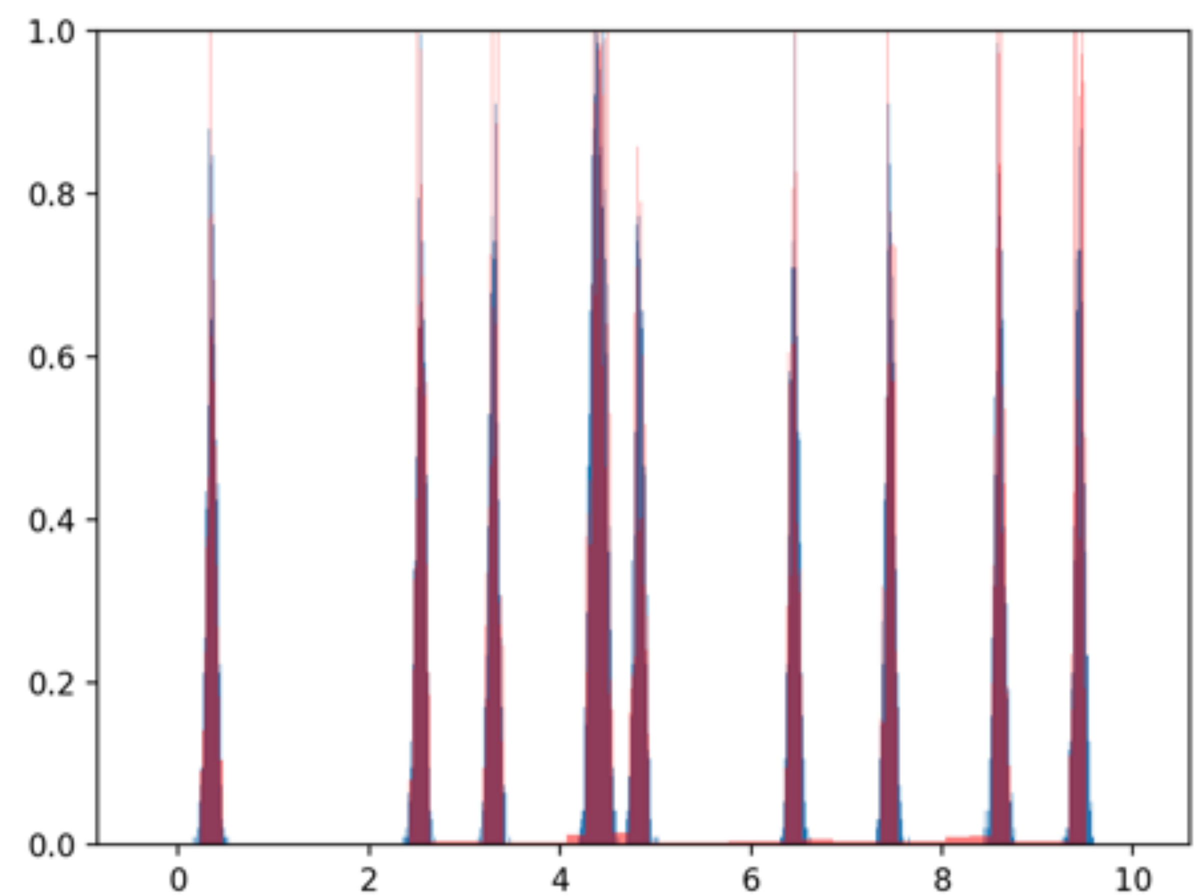


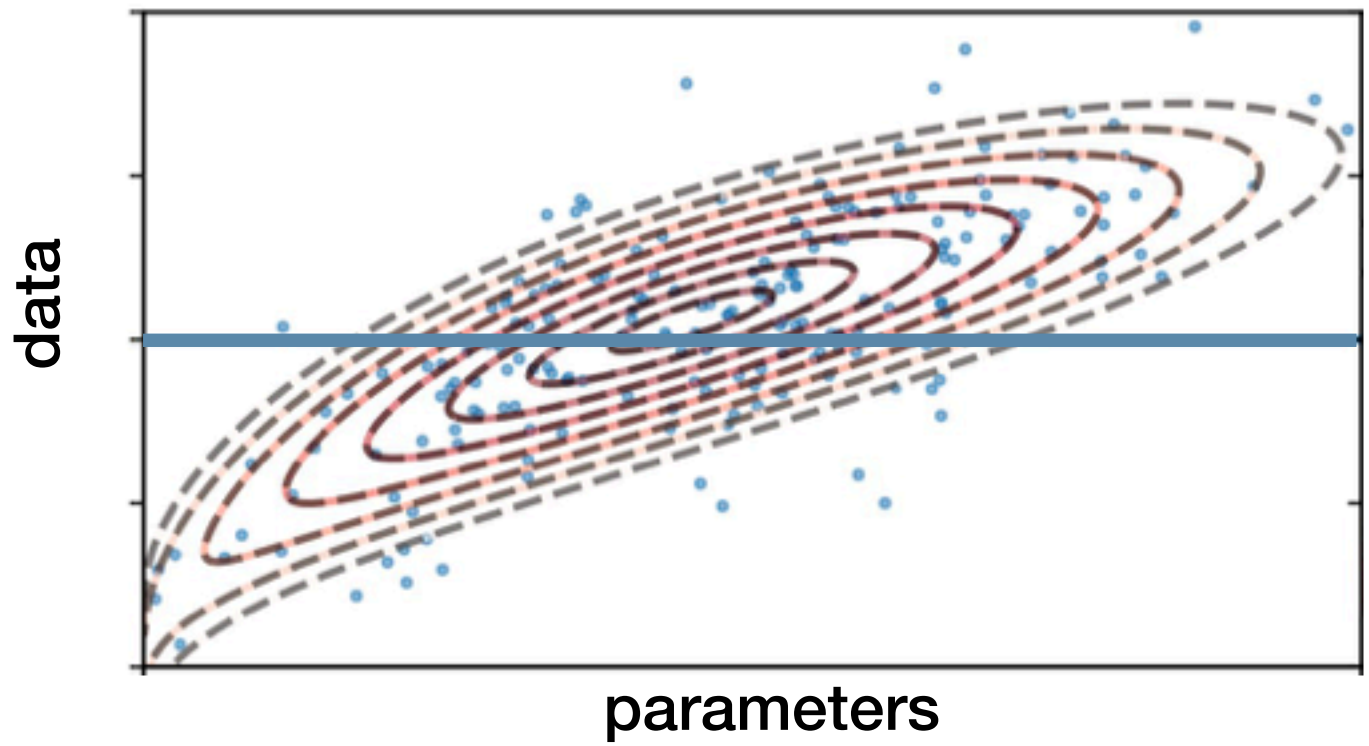
Figure from Miles Cranmer (CCA)

Pros and Cons

- Can easily fit complex and multimodal distributions.
- Implementation is trivial: only change the loss function.
- Only easy for marginal distributions: Modeling the joint distributions is very difficult.

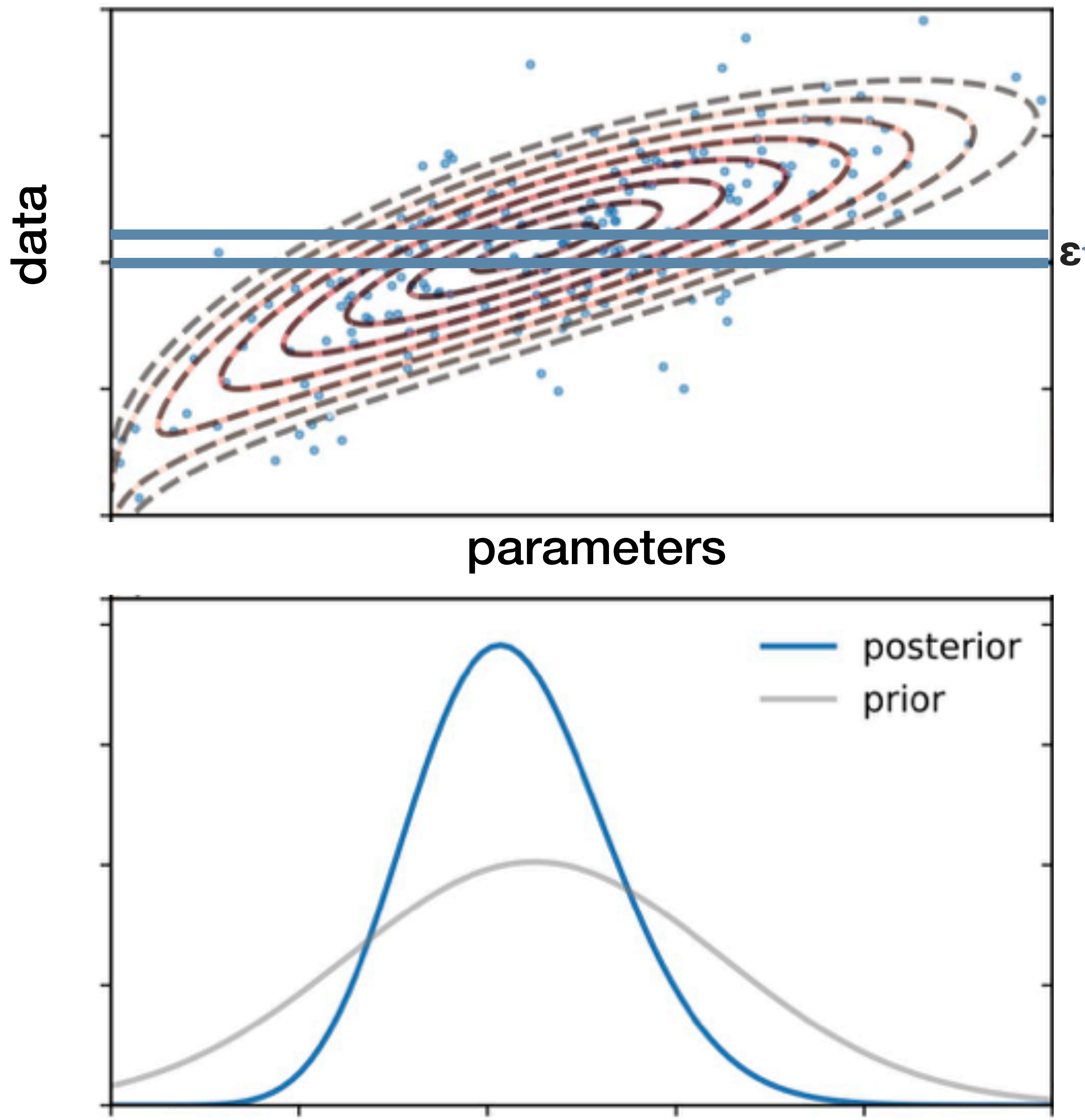
A more Bayesian way to estimate the uncertainties of neural networks: Likelihood-free inference

MONTE CARLO THE DATA



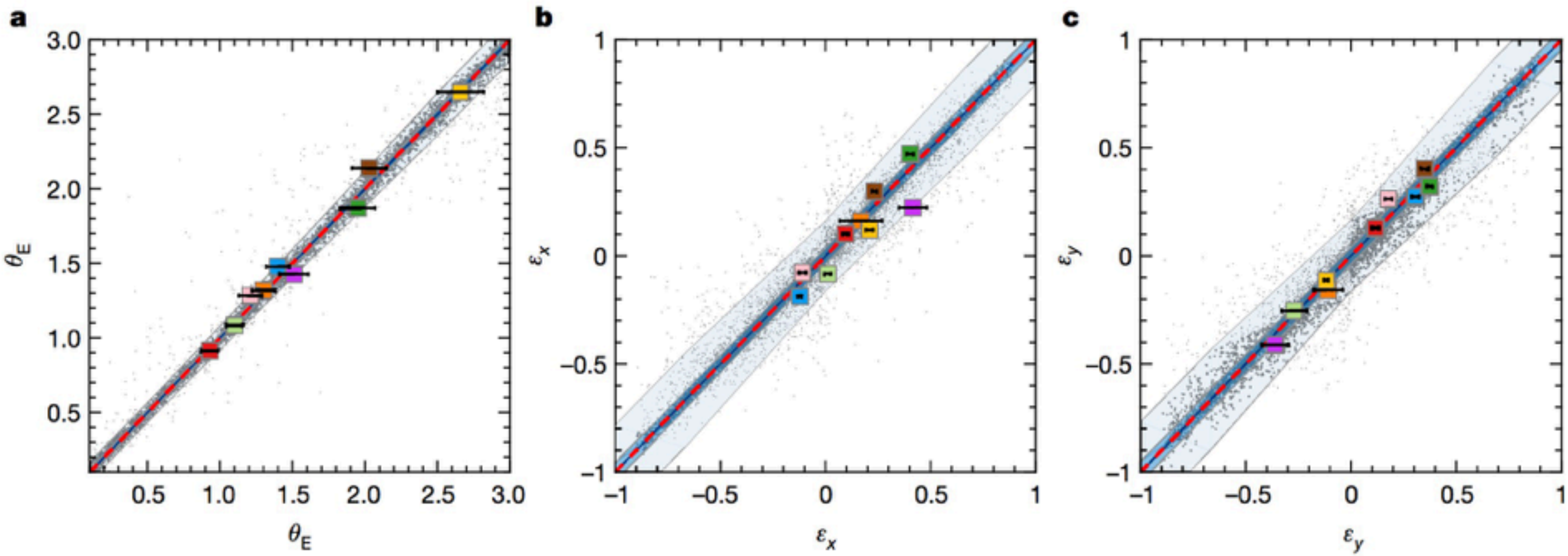
Approximate Bayesian Computation (ABC)

1. Replace data with a compressed statistic



2. Instead of a perfect match, allow an ε range of acceptable match

THE UNCERTAINTIES ARE HORIZONTAL CUTS OF THIS PLOT

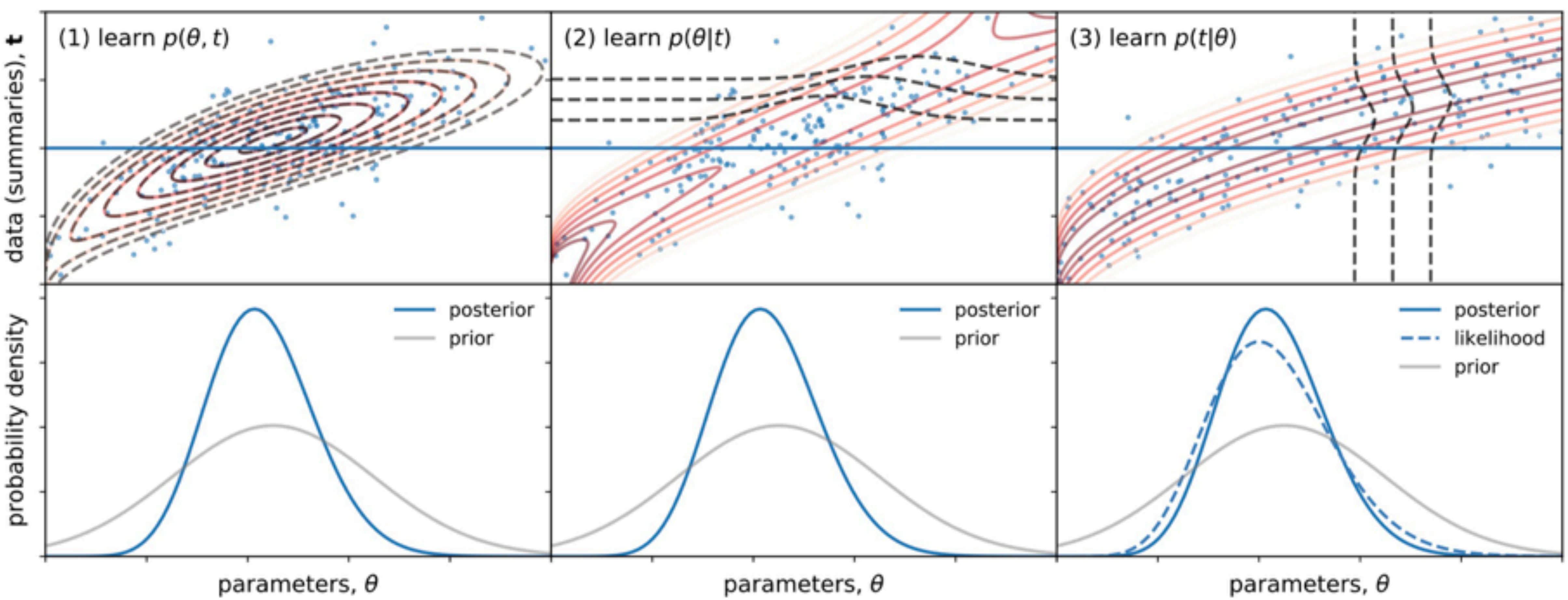


Hezaveh, Perreault Levasseur, Marshall, Nature, 2017

ABC's main caveat:

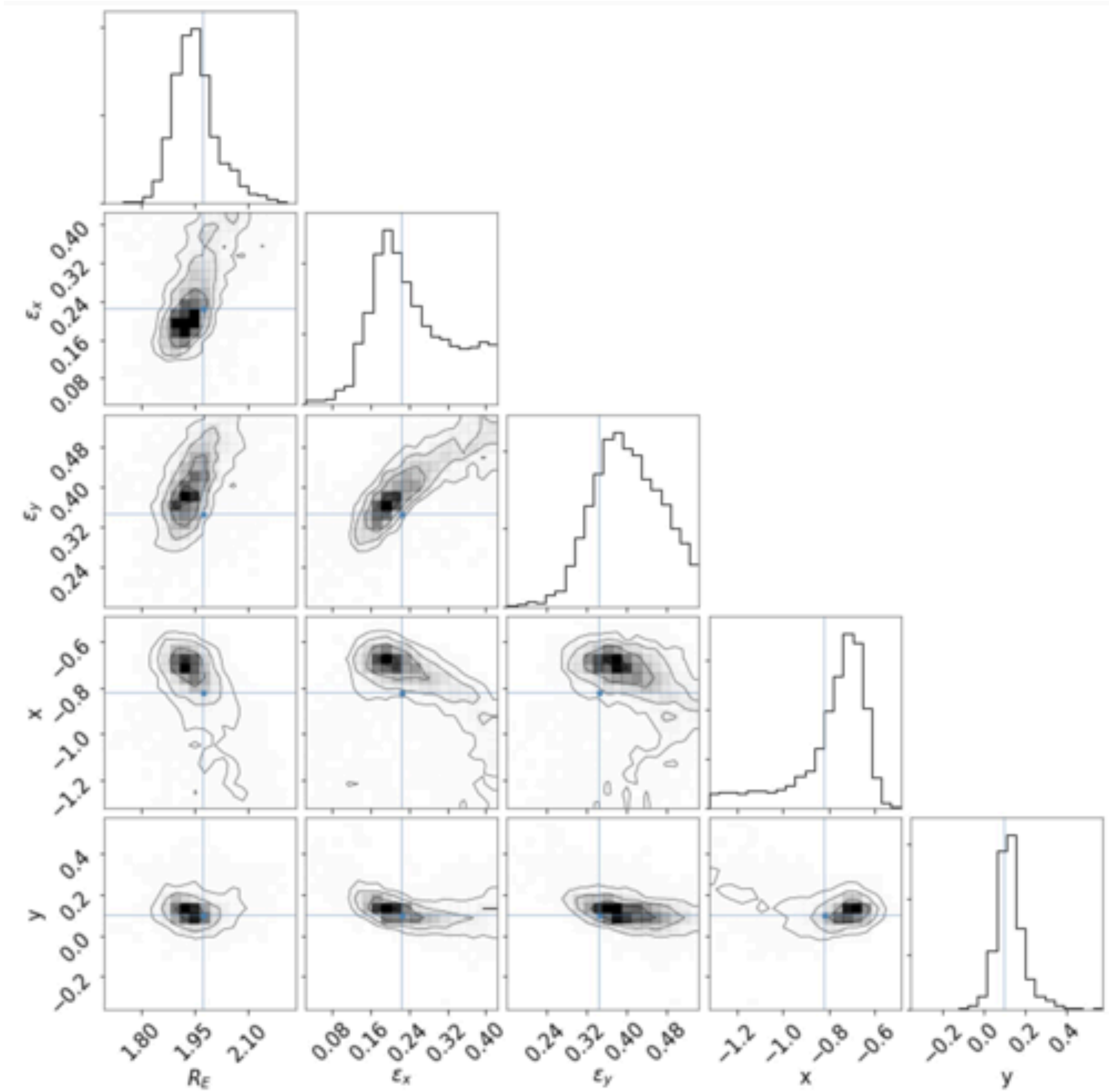
- The choice of epsilon is arbitrary. Too small results in no acceptance and too large results in loss of precision.

A SOLUTIONS:
MODEL THIS DISTRIBUTION WITH ANOTHER NEURAL NETWORK:
A NEURAL DENSITY ESTIMATOR



Alsing, Charnock, Feeney, Wandelt, MNRAS (2019)

Another approach to estimating the uncertainties: Likelihood-free inference



pydelfi:

Alsing, Charnock, Feeney, Wandelt, MNRAS (2019)

Pros and Cons

- A well-defined Bayesian framework, allowing imposing priors.
- Can be relatively fast.
- Can deal with very complex distributions.
- Can model joint posteriors.
- Limited to low-dimensional posteriors (e.g., tens).
- Requires an accurate and realistic simulation pipeline.

Thank You!