Whatever

GAN for Inference, Uncertainty Quantification, and Others

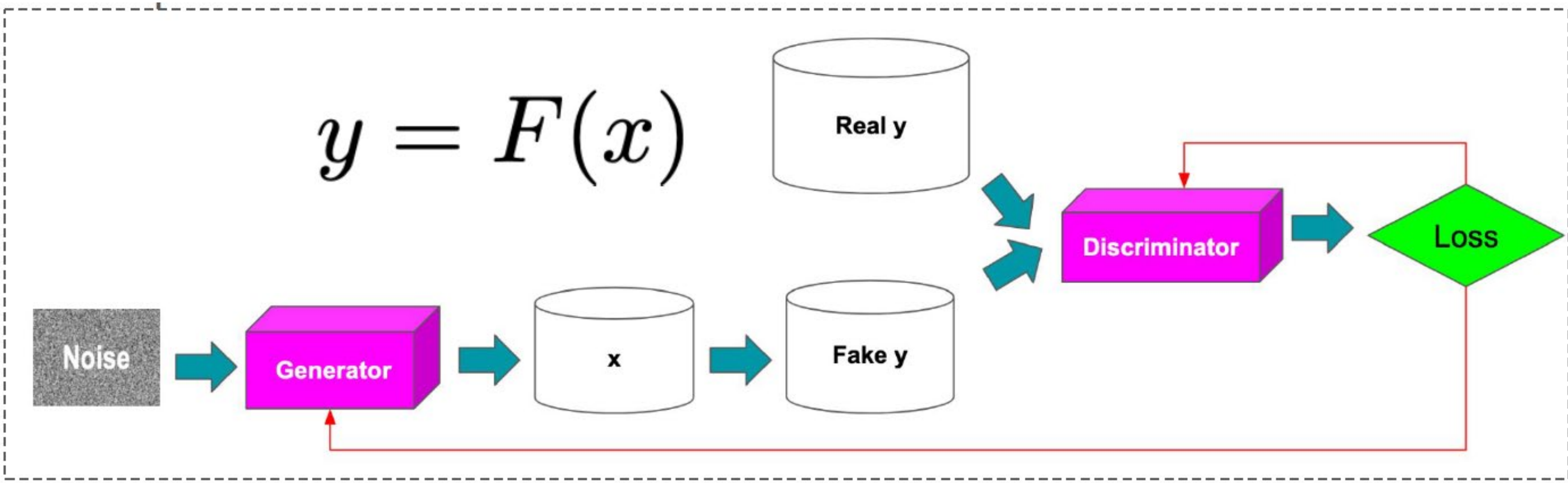
Yaohang Li

Department of Computer Science
Old Dominion University
Norfolk, Virginia 23529
yaohang@cs.odu.edu

6/25/2024 @ INT Workshop

Some figures are from Nobuo's "Thoughts on GANs for inference" Slides

GAN for inference



- Two Questions
  - Can $P_g(x|\theta)$ represent the posterior distribution?
  - Can $P_g(x|\theta)$ display the central limit behavior?
    - $O(N^{-1/2})$

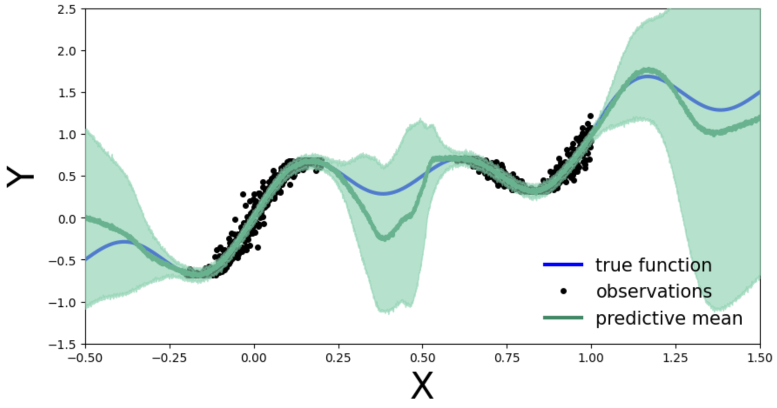
Central Limit Behavior

- Aleatory Uncertainty and Epistemic Uncertainty
- Bayesian Neural Network
- Case Study in a Simple Regression Problem
- Case Study in the Napkin Problem
    - GAN Inferencing
    - BNN-GAN Inferencing

Uncertainty Quantification

- ML Challenges
  - When to say “I don’t know”
  - Limited Amount of Experimental Data
  - Large Problem Space
  - Large Number of Model Parameters
  - ML models are difficult to explain

- The Key
  - Uncertainty Quantification



Aleatory Uncertainty and Epistemic Uncertainty

- **Aleatory Uncertainty (Stochastic Uncertainty)**
  - Inherent random effects
  - Not related to the number of data samples
  - Not reducible with increasing number of data samples
- **Epistemic Uncertainty (Systematic Uncertainty)**
  - Uncertainty due to lack of knowledge
  - Reducible with more data samples
- Aleatoric and epistemic uncertainty can occur simultaneously in a single term

Bayesian Neural Networks

Sampling the model parameters, BNN can create an ensemble of models that generate a distribution over the output

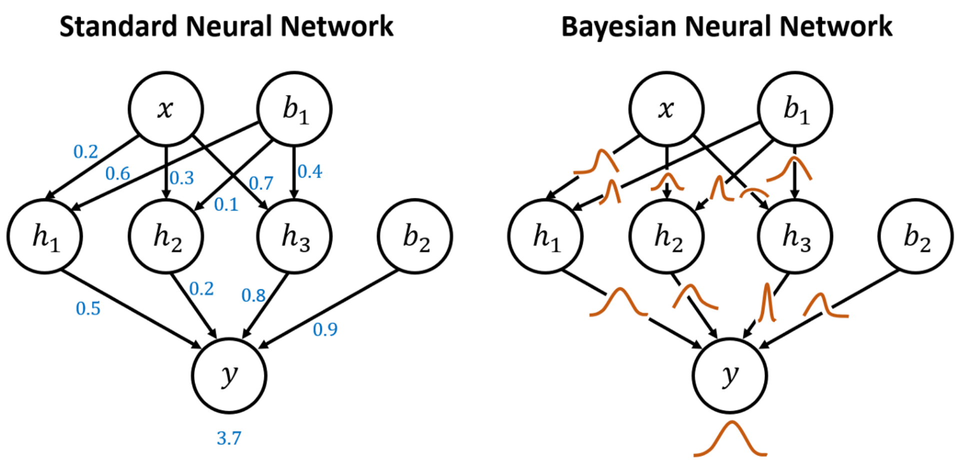


Image from Neptune.ai

NN and BNN

	NN	BNN
Weights	Deterministic	Stochastic
Training	D -> θ	Prior p(θ) + D -> Posterior p(θ|D)
Prediction	$\boldsymbol{p(y\|x,\theta)}$	$\boldsymbol{p(y\|x,D) = \int p(y\|x,\theta)p(\theta\|D)d\theta}$

Bayesian Neural Networks

$$\underbrace{}_{}$$

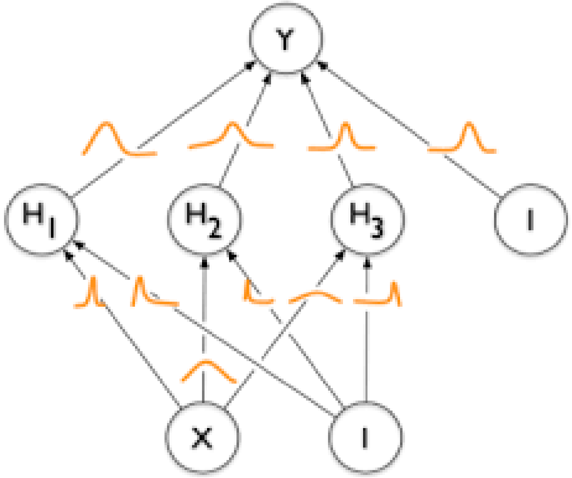
Aleatory Uncertainty — **Epistemic Uncertainty**

$$p(y|x,D) = \int p(y|x,\theta)p(\theta|D)d\theta$$

BNN can be used to estimate aleatory uncertainty and epistemic uncertainty.

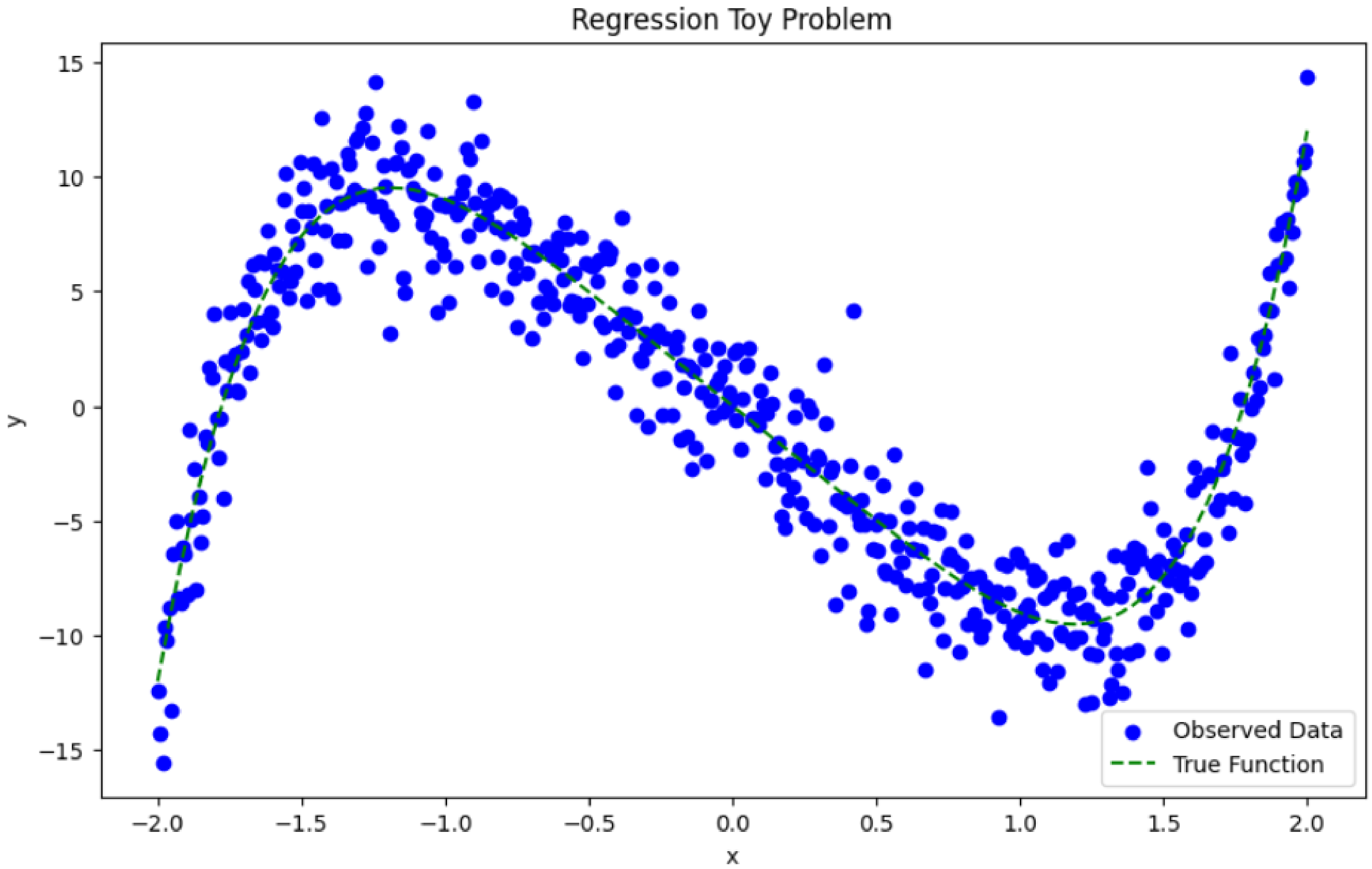
Sampling in Bayesian Neural Networks

- **MCMC**
- **Variational Inferencing**



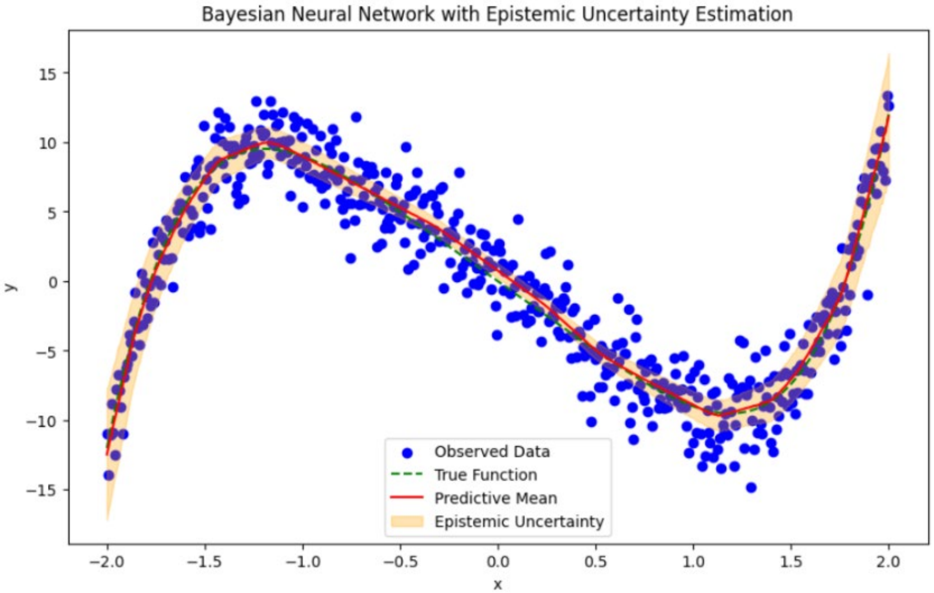
Case Study 1: BNN on A Simple Regression Problem

- **A simple 1D regression problem**



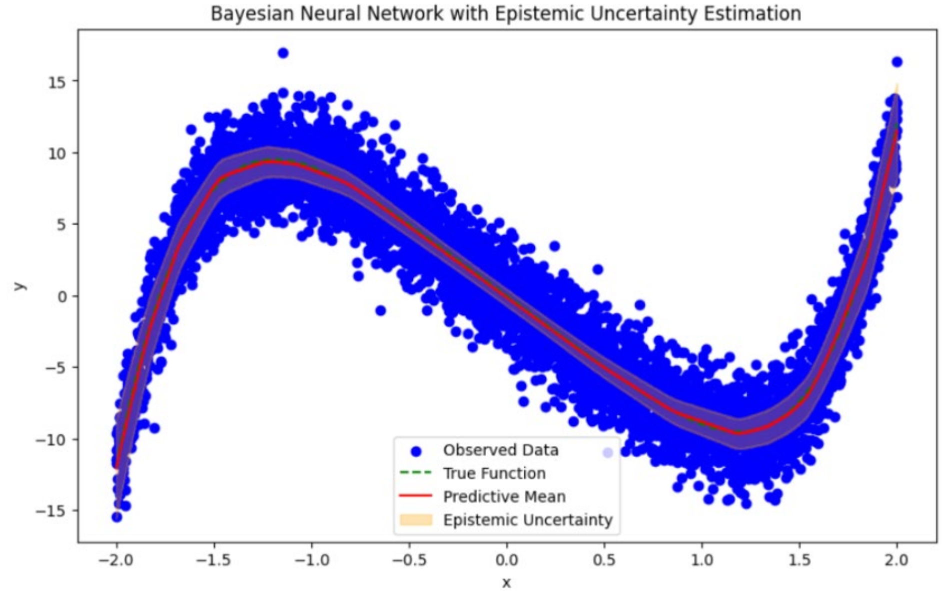
Epistemic Uncertainty

- **BNN captures the epistemic uncertainty**
  - Decreased as more data involved




```
y_pred_std.mean()
```

```
0.45432025
```

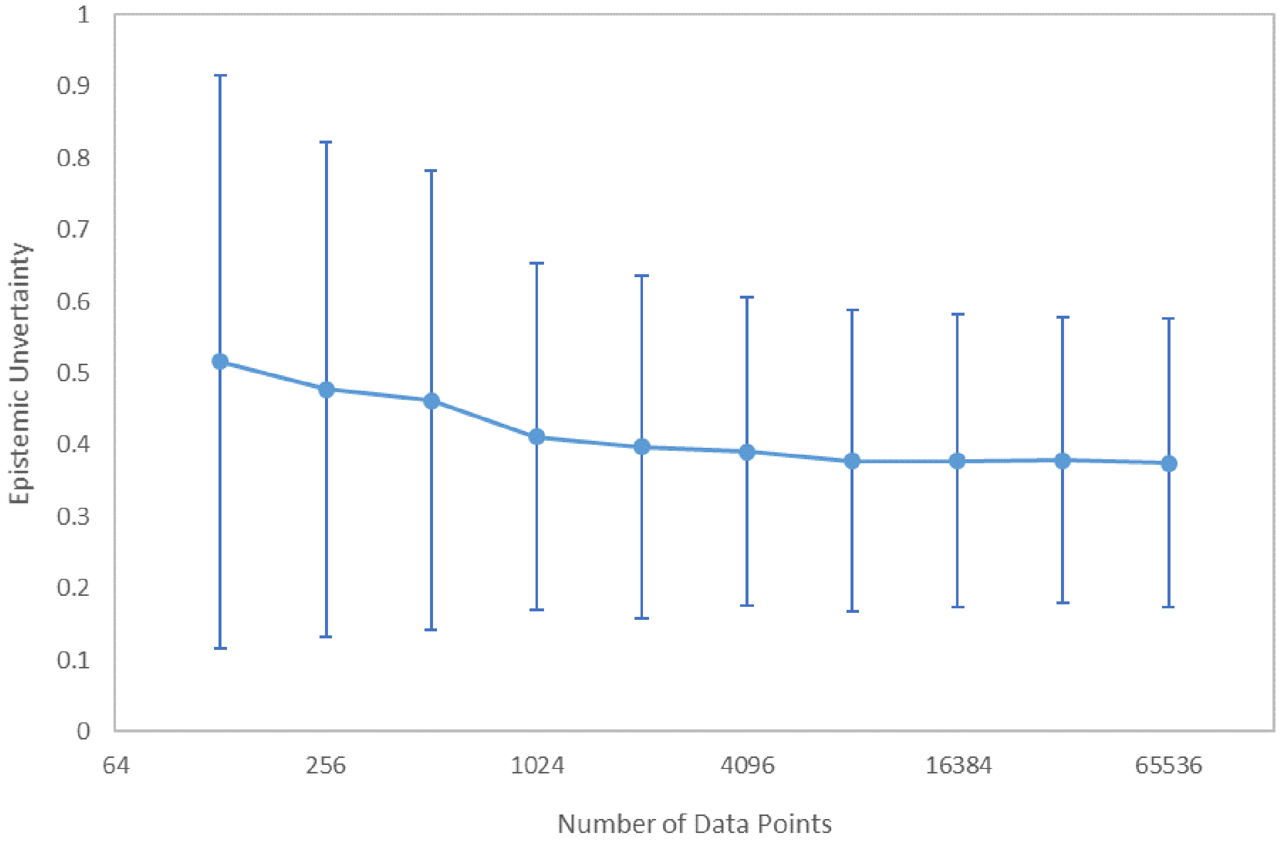



```
y_pred_std.mean()
```

```
0.38451898
```

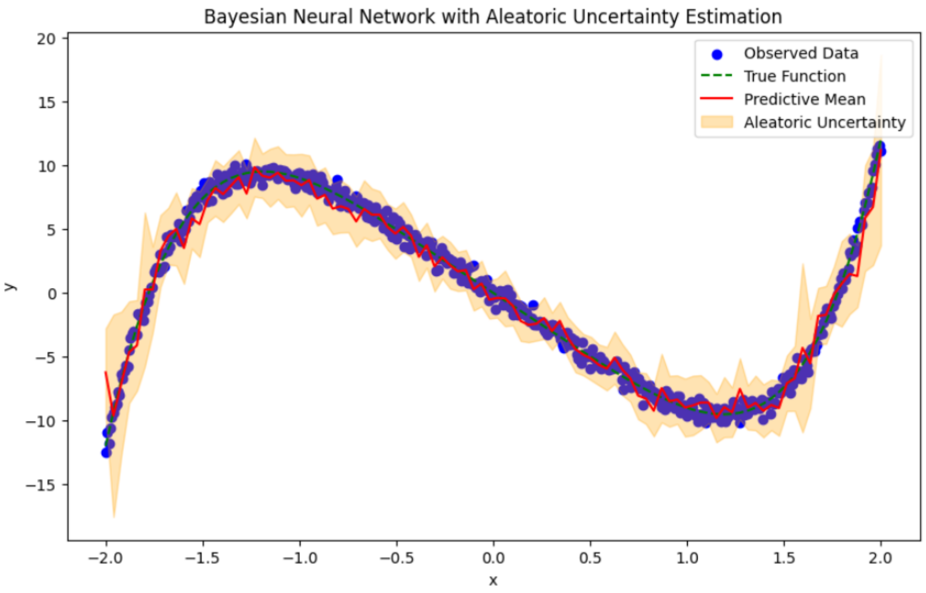
Epistemic Uncertainty

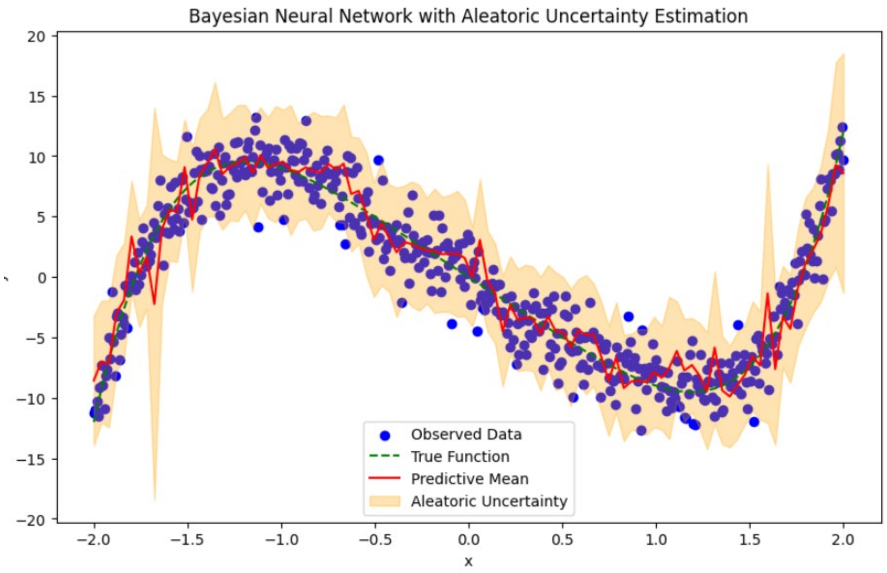
- **Not decreasing with $O(N^{-1/2})$**



Aleatoric Uncertainty

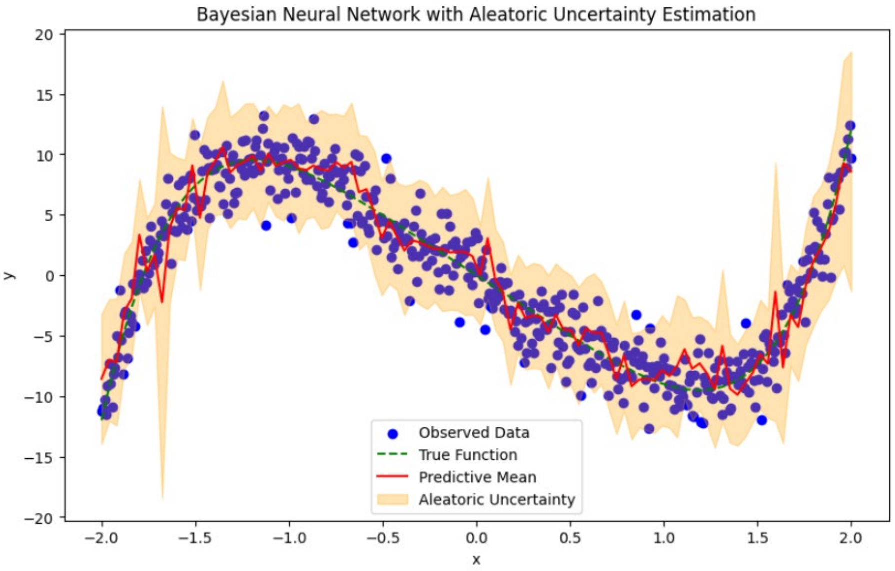
- **PBNN captures the inherent uncertainty in data**

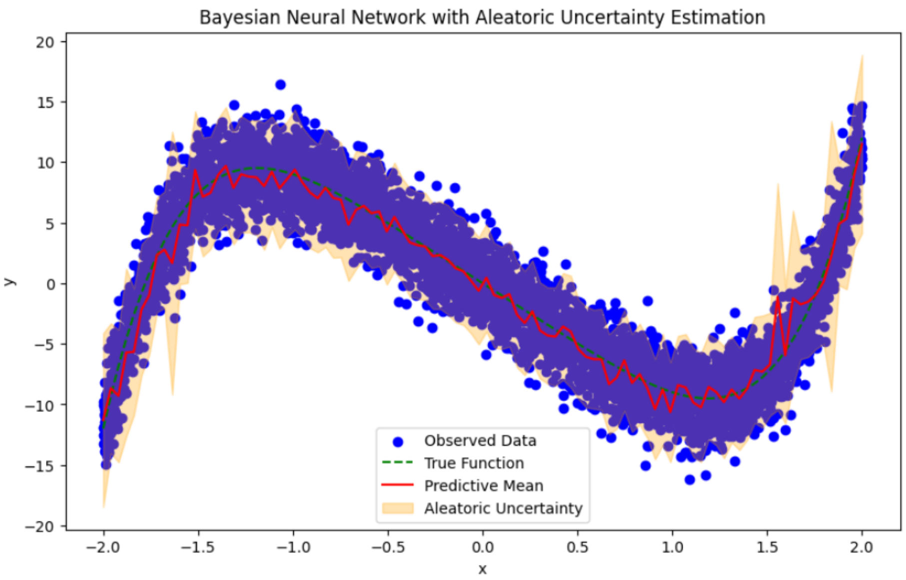




Aleatoric Uncertainty

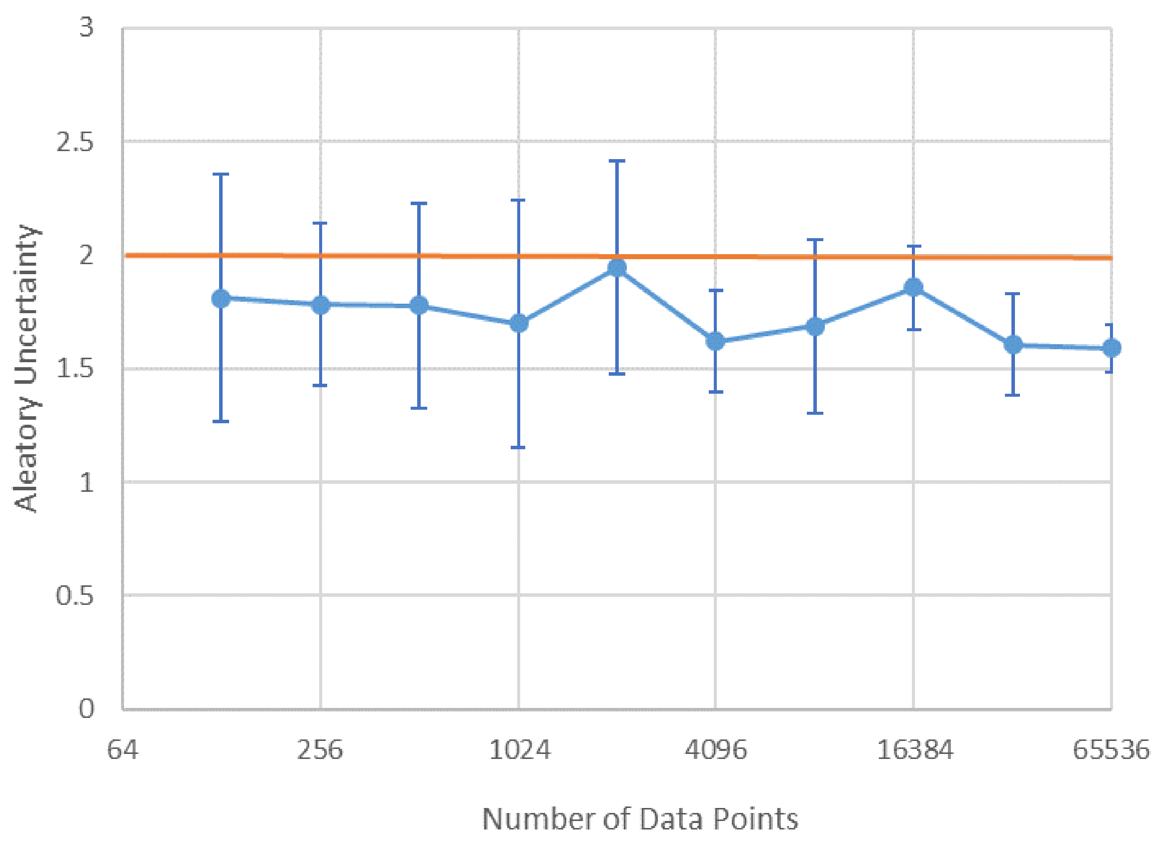
- **Aleatoric Uncertainty does not change with # of samples**



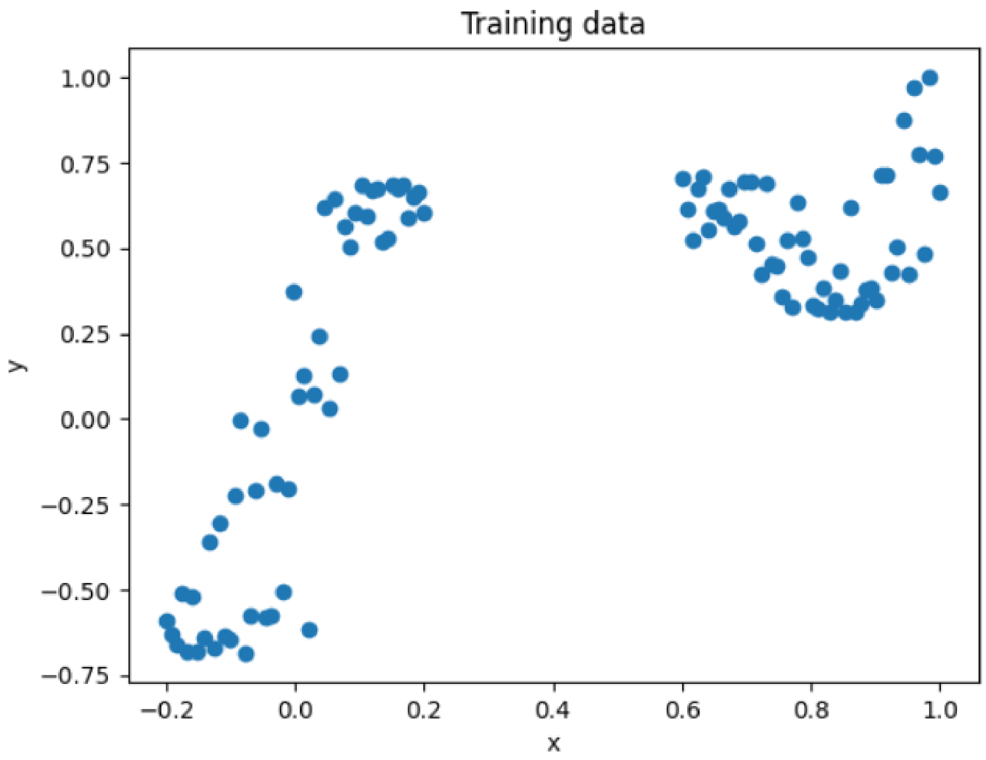


Aleatoric Uncertainty

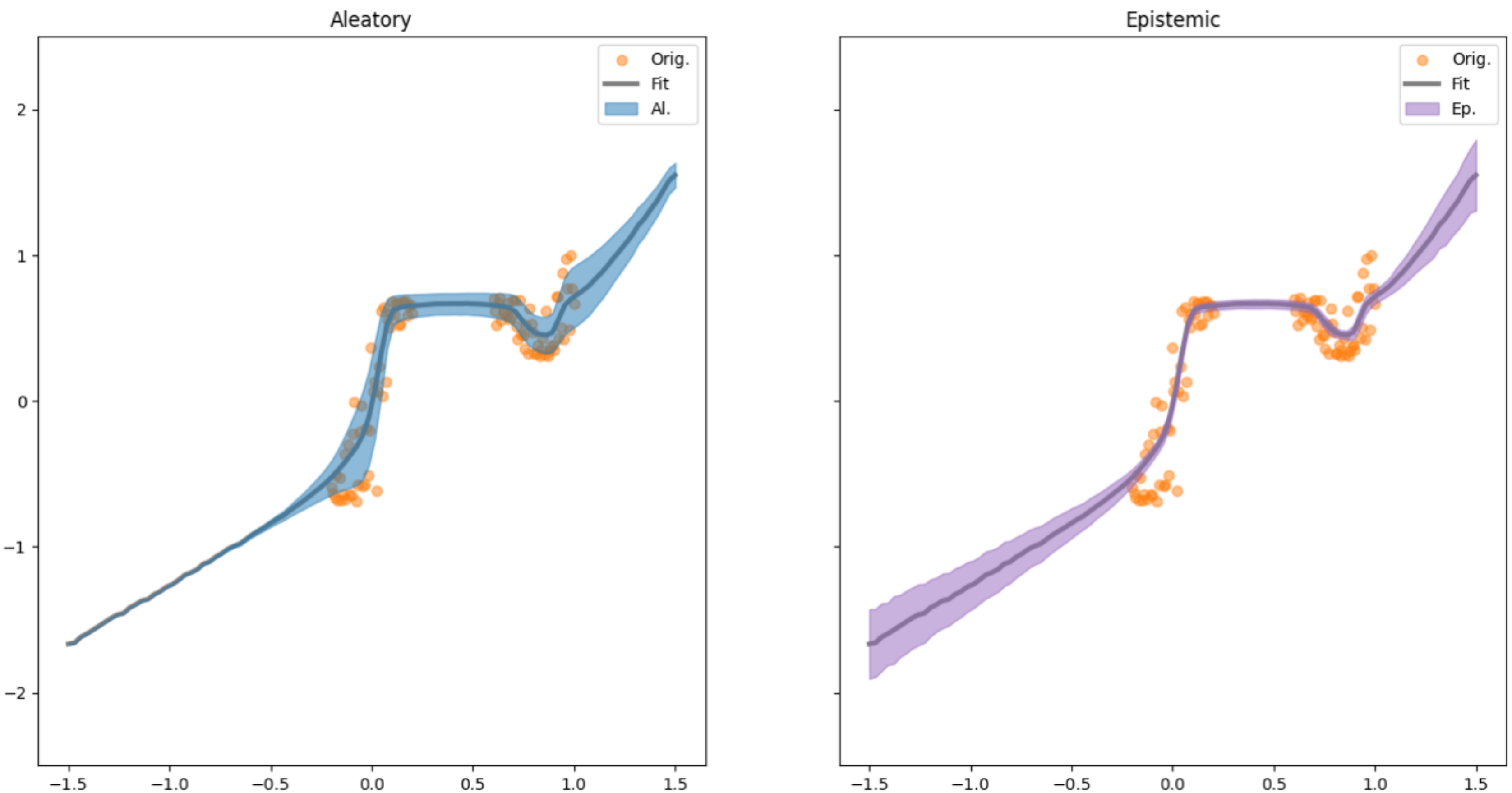
- **Trend**



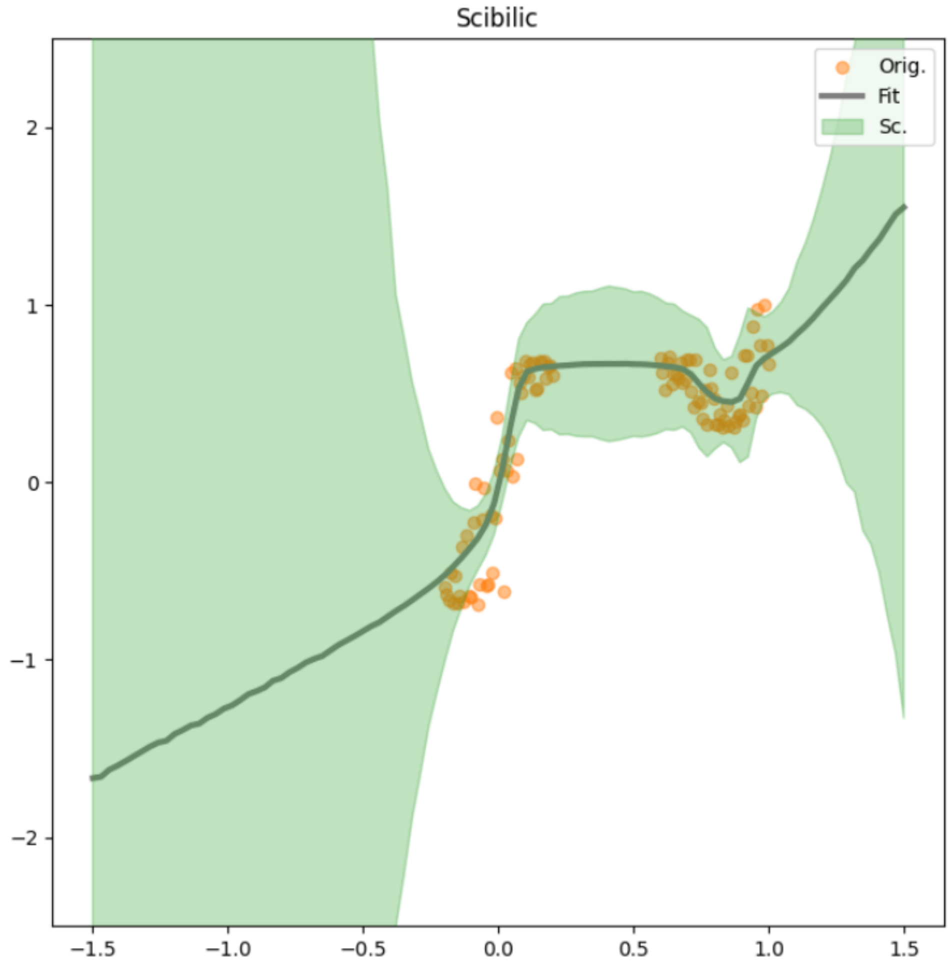
Case Study 2: Scibilic Uncertainty



Case Study 2: Aleatory Uncertainty and Epistemic Uncertainty



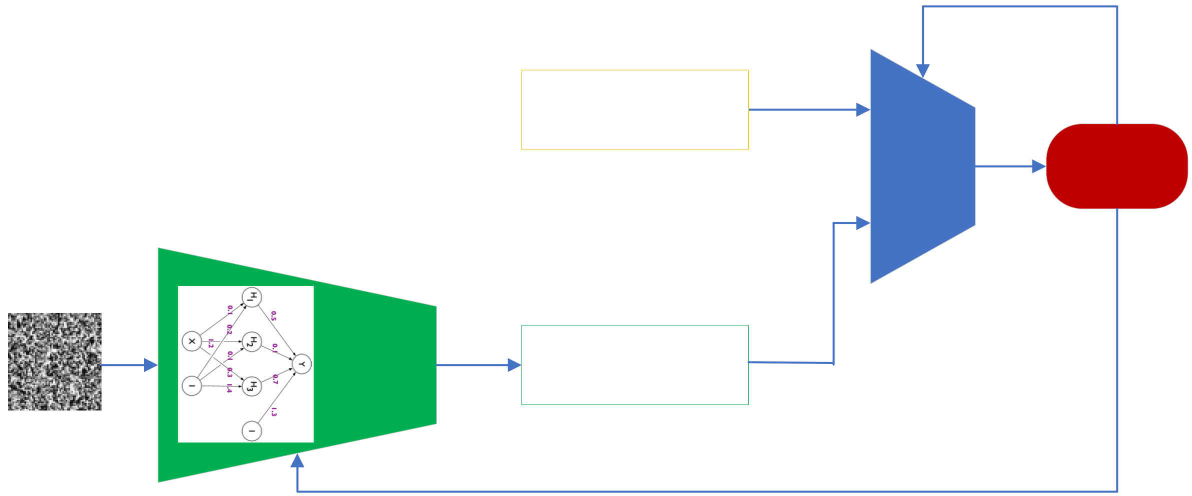
Case Study 2: Scibilic Uncertainty



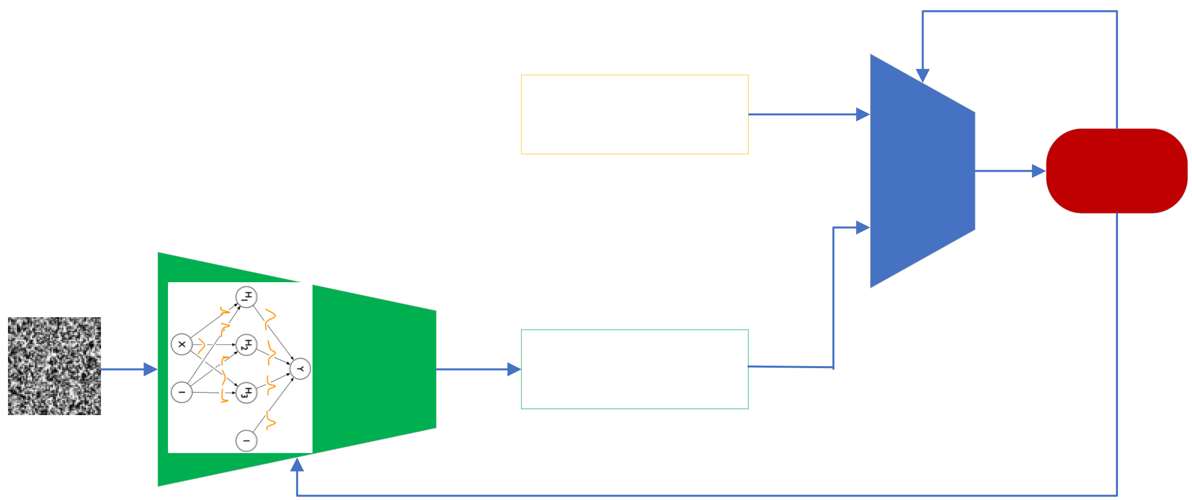
$$\textbf{scibilic uncertainty} = \frac{\textbf{epistemic uncertainty}}{\textbf{alleatory uncertainty} + \varepsilon}$$

BNN-GAN

GAN



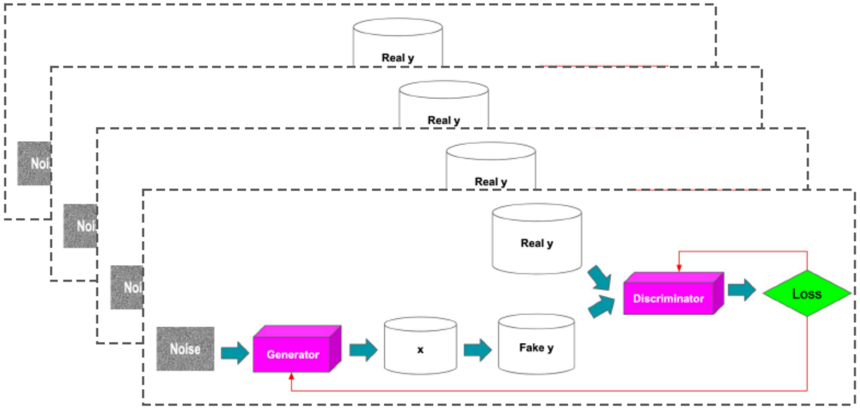
BNN-GAN



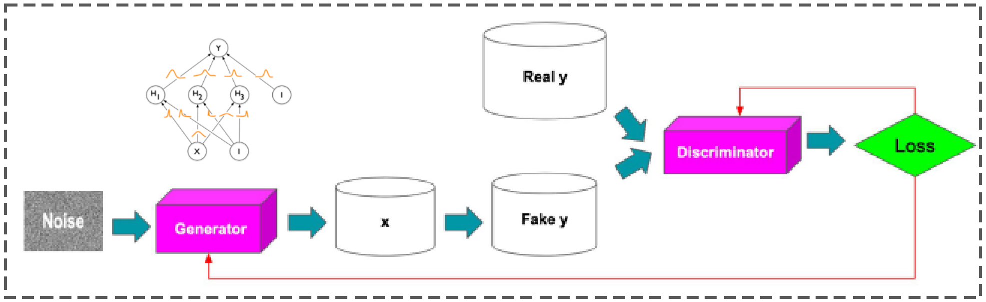
Rationality: It trains an ensemble of models

The Fundamental Idea of BNN-GAN

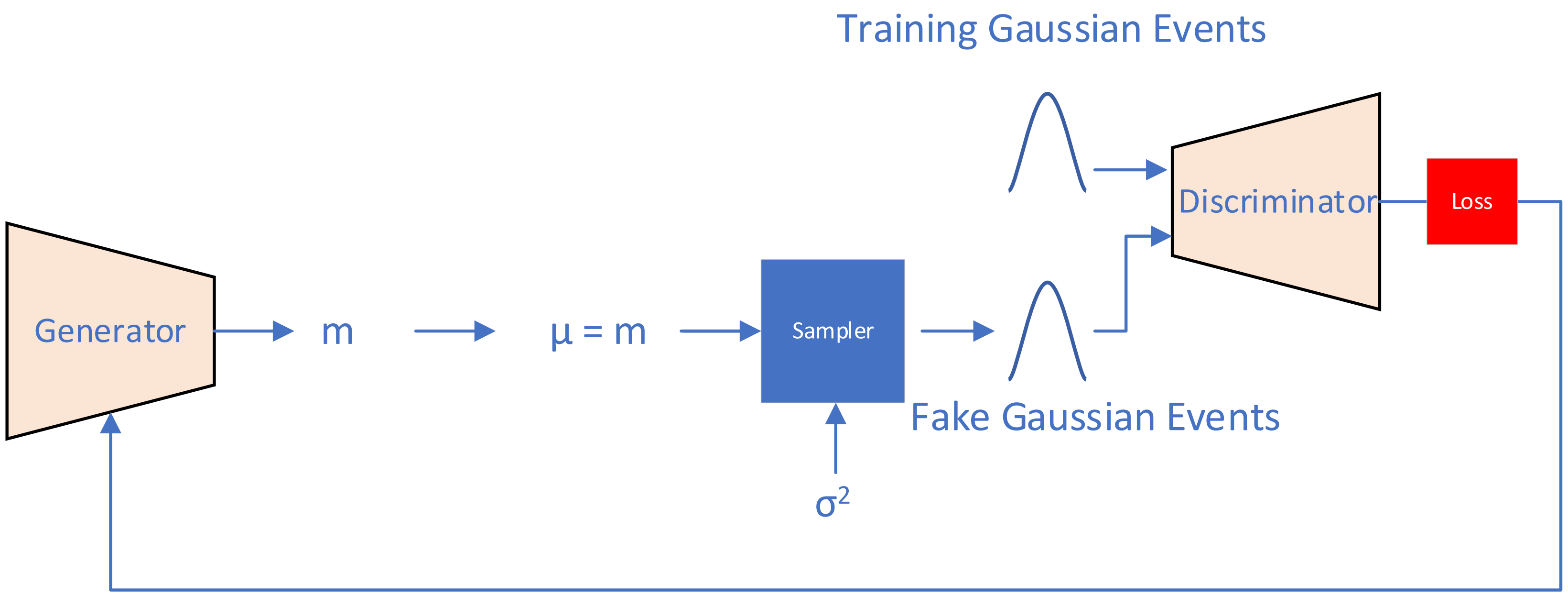
- **Bootstrap Approach with GANs (from Nobuo's idea notes)**
  - Train many GANs
    - An ensemble of GANs
  - Each GAN learns from the real data
  - The collection of GAN predictions constitute the posterior distribution samples



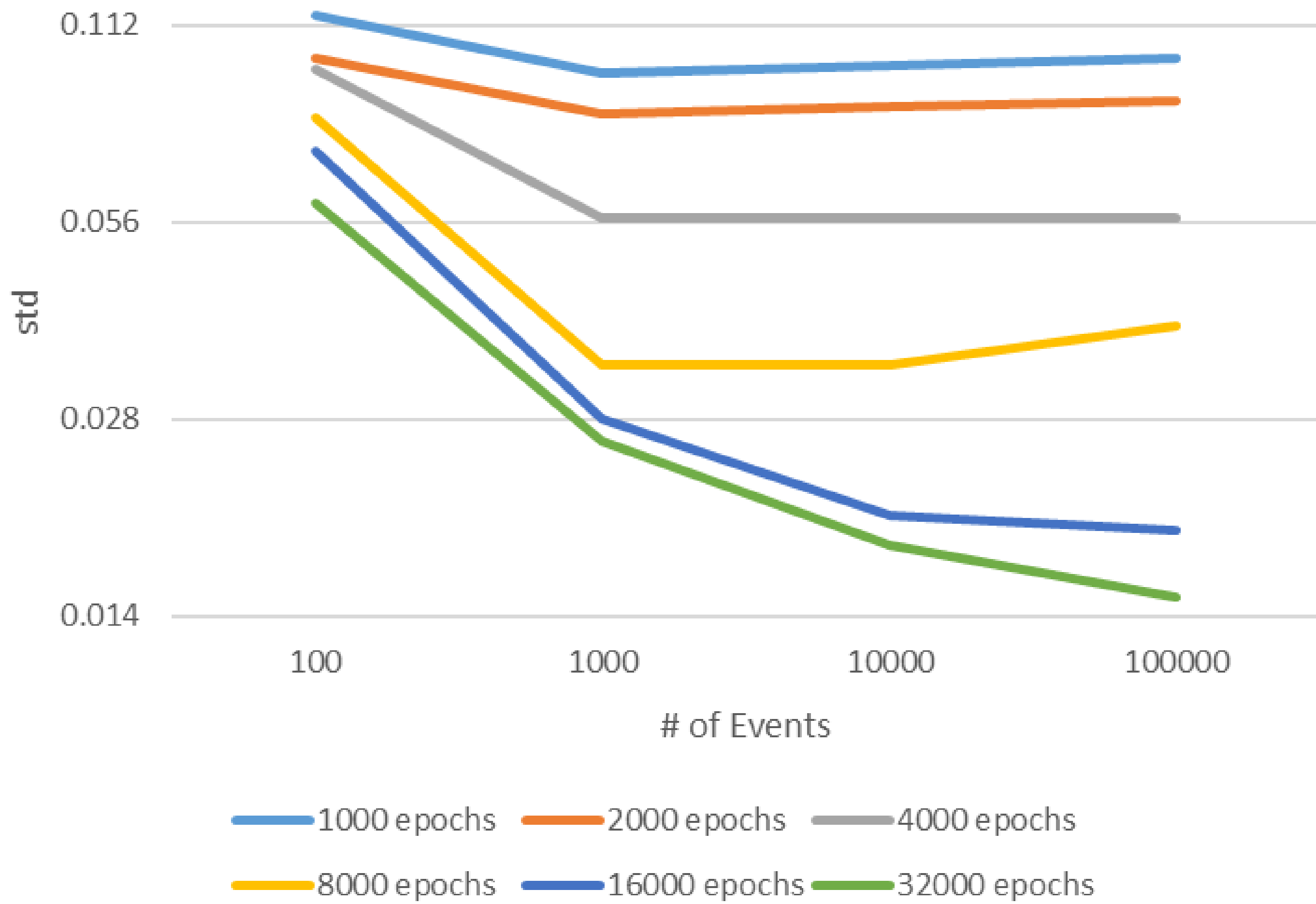
- **BNN-GAN**
  - Use BNN to replace NN in the generator
  - Learn an ensemble of GANs in one training
  - Sampling the weight distribution to generate the posterior distribution



Napkin Problem

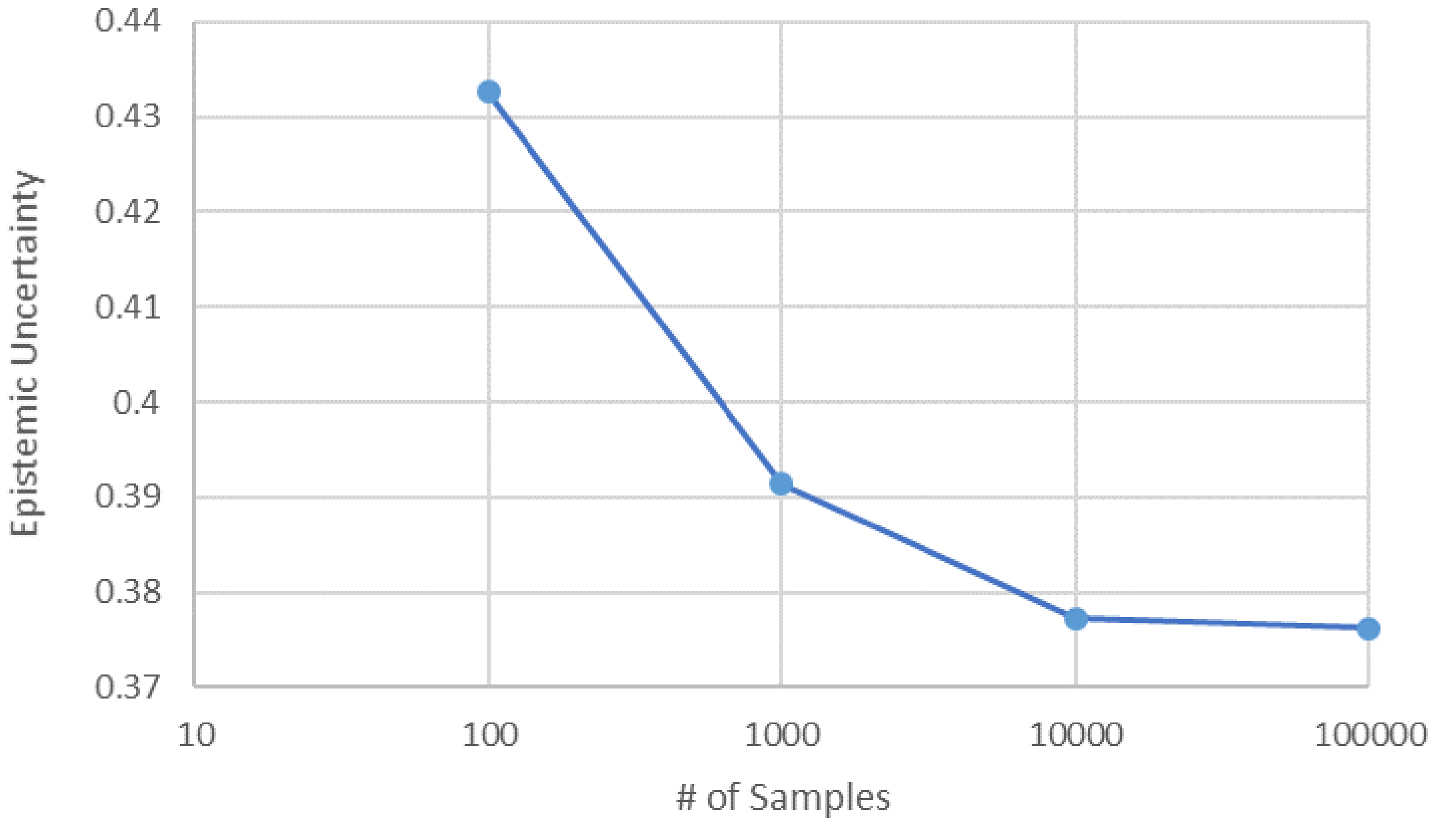


Napkin Problem



It is a log-log plot of std and # of events for GANs trained after certain number of epochs. There are two kinds of uncertainty here: the machine learning modeling uncertainty by the GAN and statistical uncertainty. At the beginning, without sufficient training, the modeling uncertainty is dominating. The curves across 100 to 100,000 events are almost flat -- it looks like there is no difference between training with a lot of events or a few events. However, with more epochs of training being carried out, the modeling uncertainty continues to reduce and the effect of statistical uncertainty starts to appear. The nearly $O(N^{-1/2})$ degrading starts to gradually appear (see the curve with 32K epochs).

BNN-GAN on the Napkin Problem



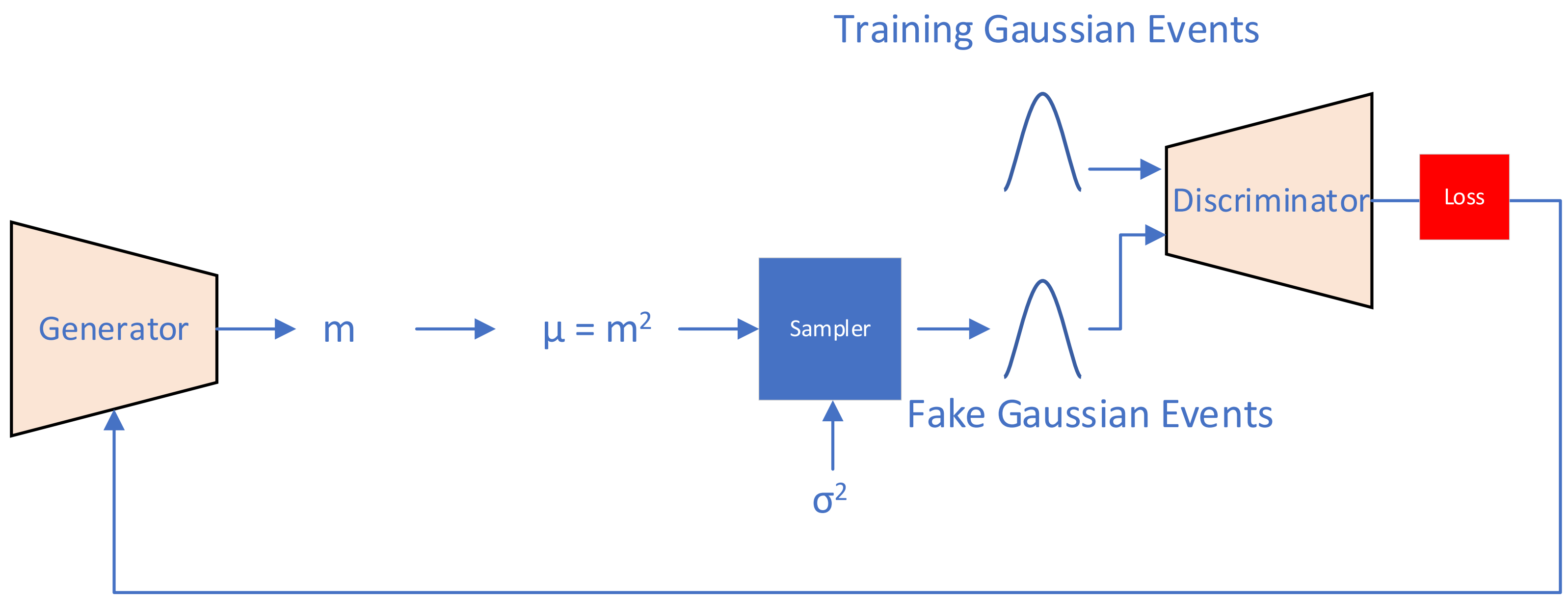
Why not $N^{-1/2}$?

- **Epistemic Uncertainty includes**
  - Model uncertainty
    - Measure how much the ML model can wiggle (biase?)
    - Decreases as N increases
    - But not necessarily $N^{-1/2}$
  - MC uncertainty
    - $O(N^{-1/2})$

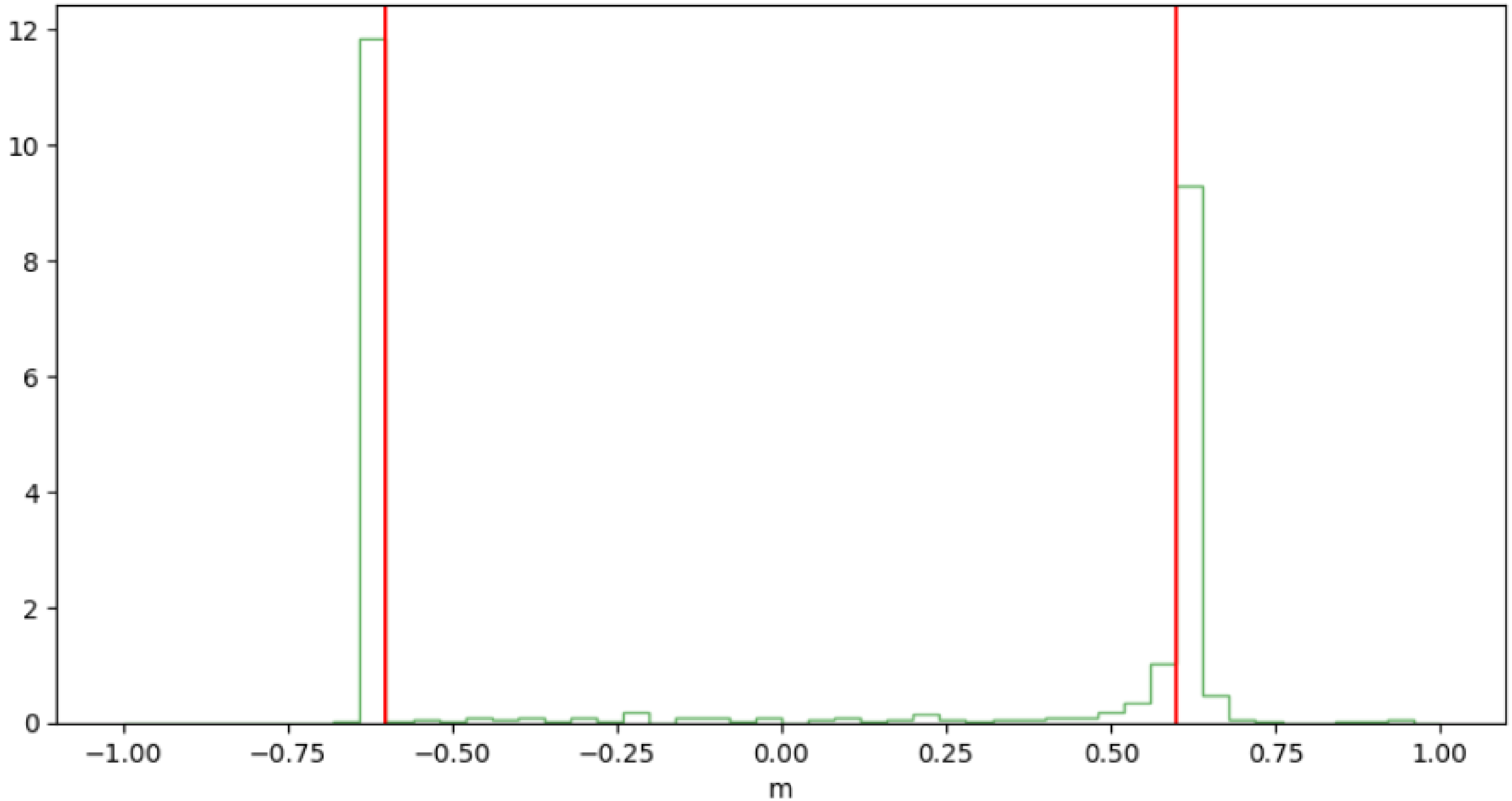
Can $P_g(x|\theta)$ represent the posterior distribution?

- 1. If F is identity [OK]
- 2. If F is invertible [OK]
- 3. If F is not invertible?
  - Ill-posedness
    - There can be multiple solutions
  - Inverse problem

Napkin Problem 2



BNN-GAN Results on the Napkin Problem 2

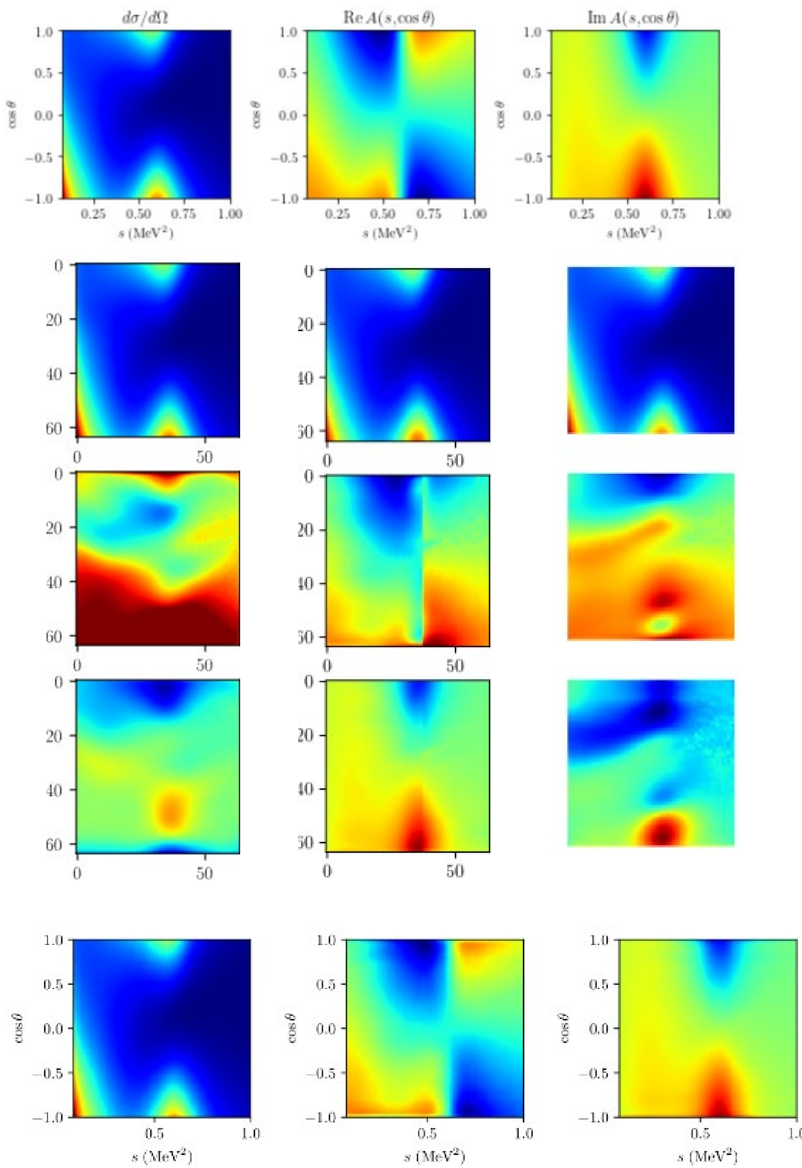


$\mu = m^2 = 0.36$

Reconstruction of Amplitude from Cross Section

Truth

Results from Different Runs



Images Generated from Gloria’s program

Occasionally, we get it right.

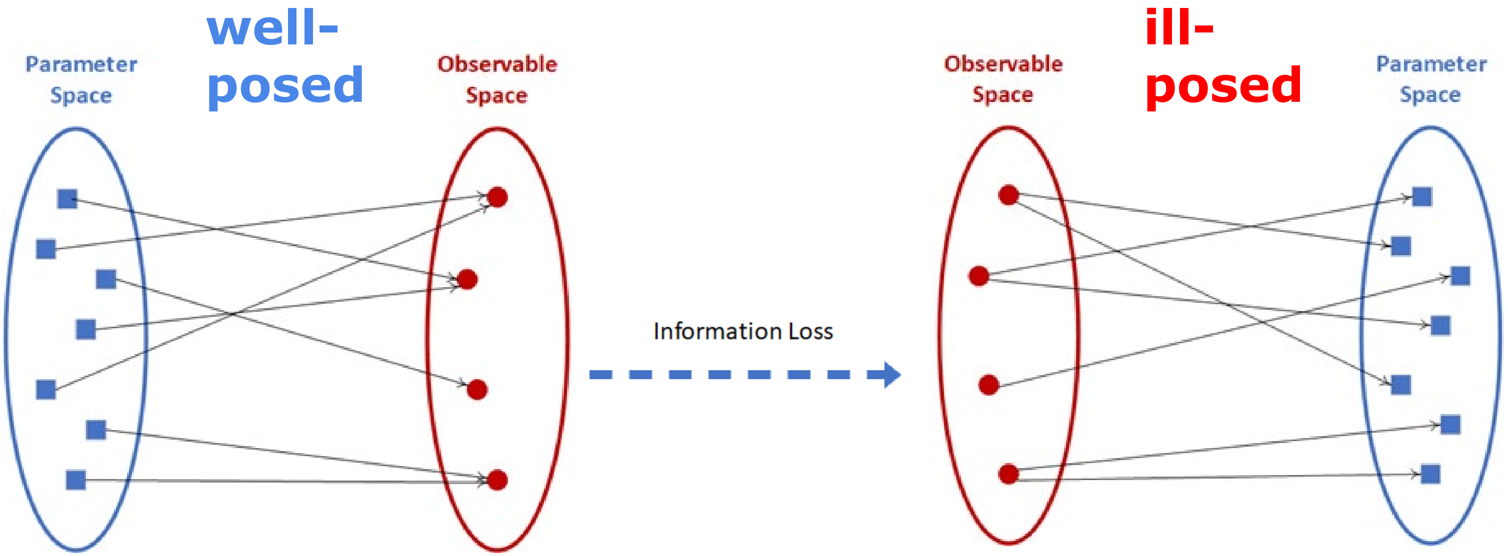
BNN on Inverse Problem

- **BNN on Variational Autoencoder Inverse Mapper (VAIM)**
  - Estimation of each individual solution in an inverse problem
  - Provide “I don’t know”

Ill-posedness of Inverse Problems

Forward Mapper

Backward Mapper

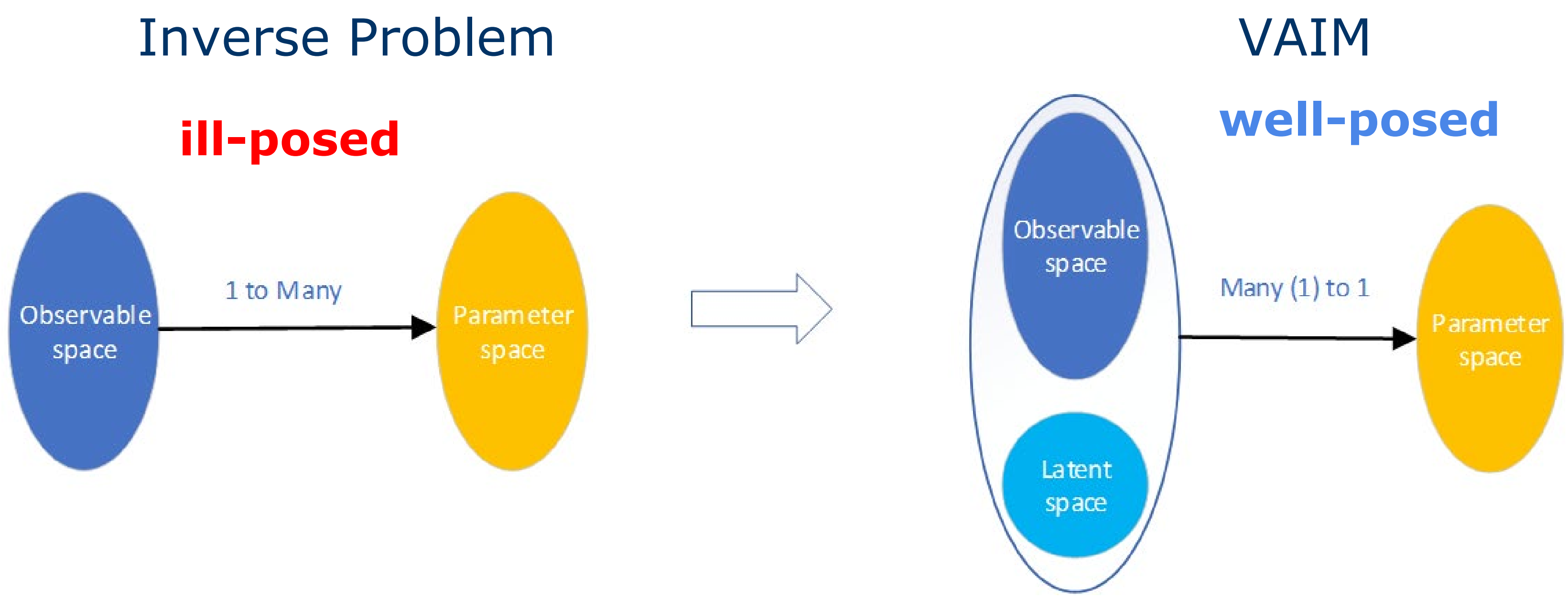


$\mathbf{y} = f(\mathbf{x})$

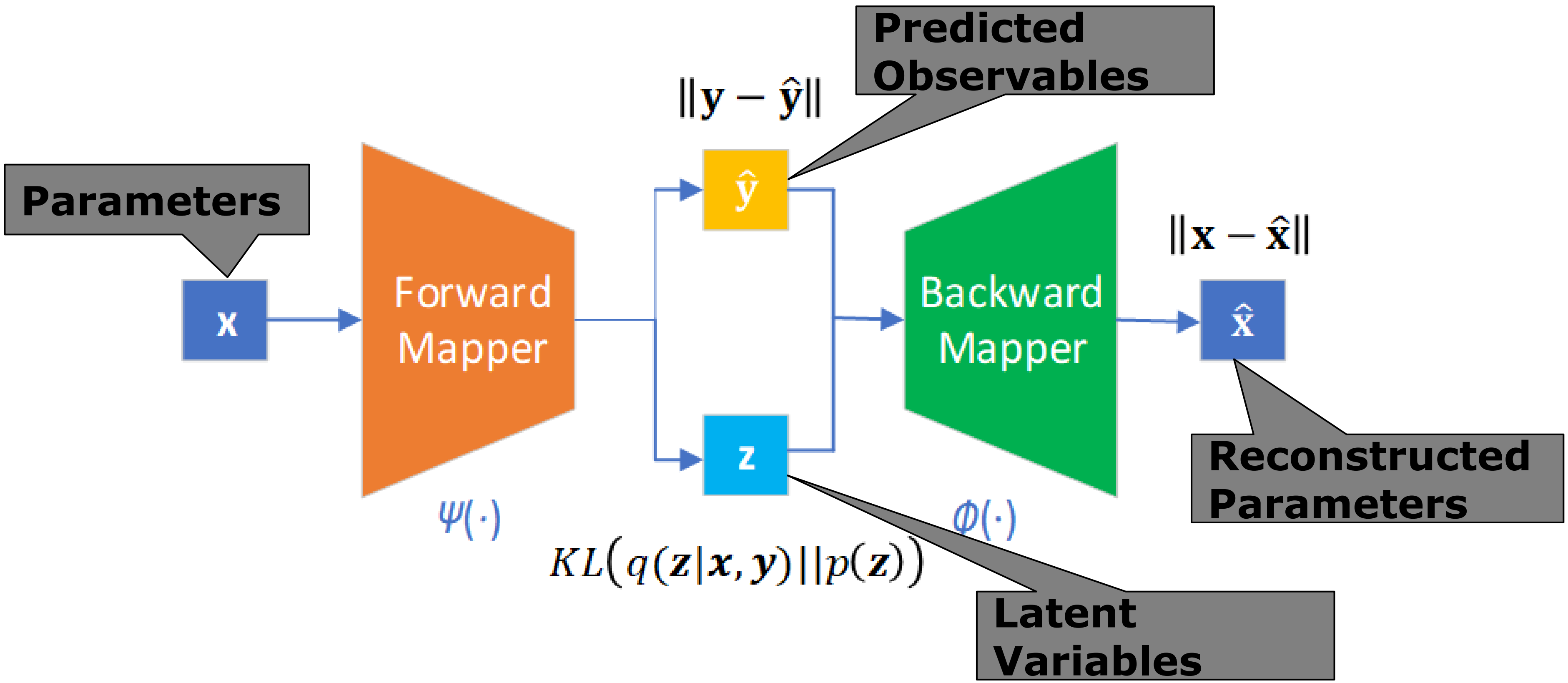
$\mathbf{x} = f^{-1}(\mathbf{y})$

Fundamental Idea

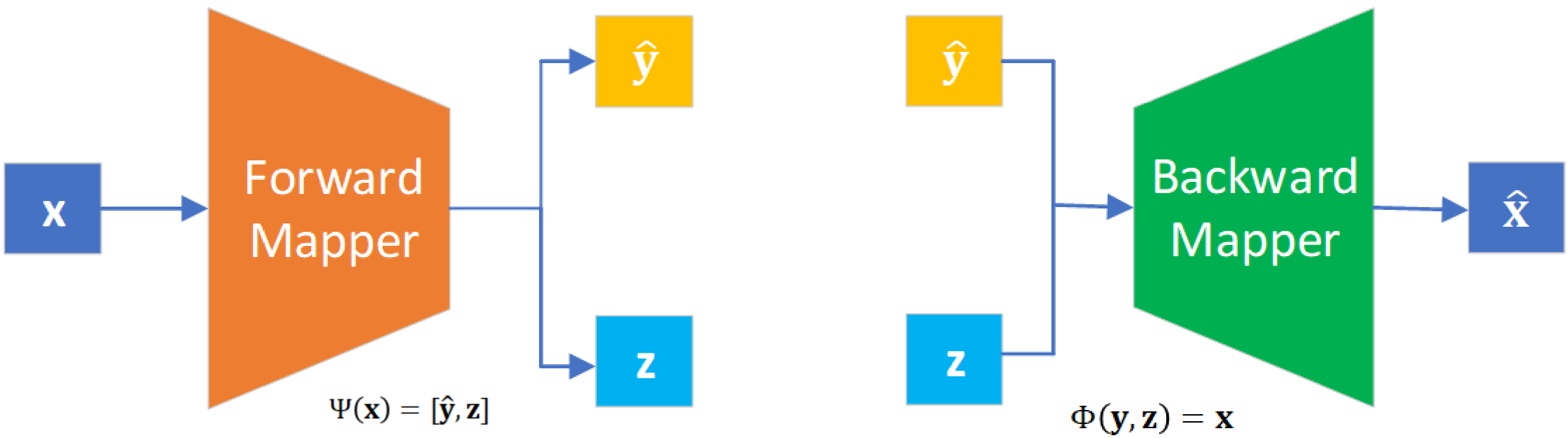
Variational Autoencoder Inverse Mapper



Variational Autoencoder Inverse Mapper



Forward Mapper and Backward Mapper



Learn posterior distribution
$p(\mathbf{z}|\mathbf{x}, \mathbf{y})$

Learn likelihood distribution
$p(\mathbf{x}, \mathbf{y}|\mathbf{z})$

Math behind Variational Autoencoder Inverse Mapper

- Approximate
  - True posterior distribution $p(\mathbf{z}|\mathbf{x},\mathbf{y})$
- Variational Inference
  - Learn an approximate distribution $q(\mathbf{z}|\mathbf{x},\mathbf{y})$ such that

    $q(\mathbf{z}|\mathbf{x},\mathbf{y}) \sim p(\mathbf{z}|\mathbf{x},\mathbf{y})$
  - Minimize the Kullback-Leibler (KL) divergence

    $\min KL(q(\mathbf{z}|\mathbf{x},\mathbf{y})||p(\mathbf{z}|\mathbf{x},\mathbf{y}))$

- Variational Autoencoder Theory

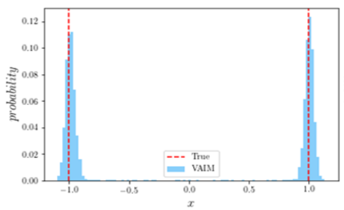
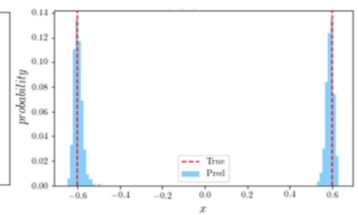
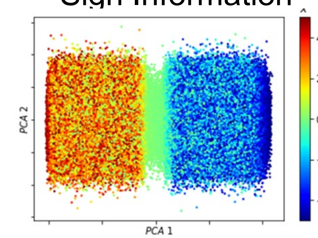
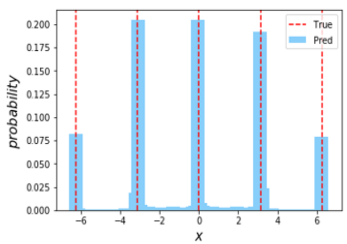
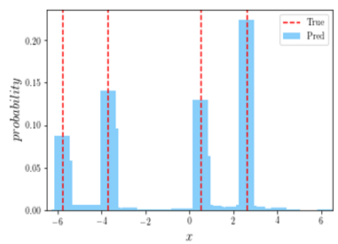
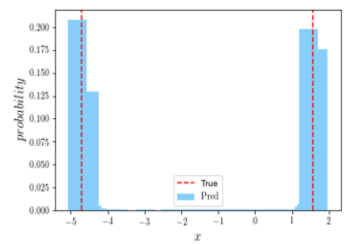
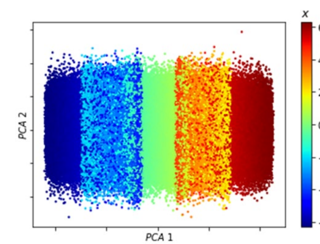
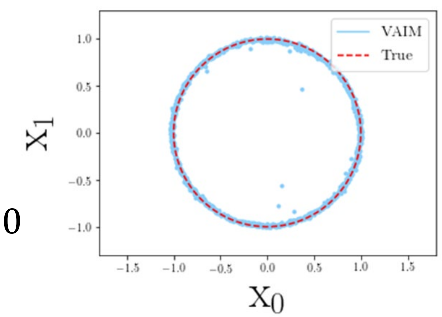
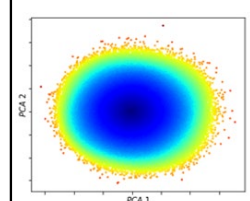
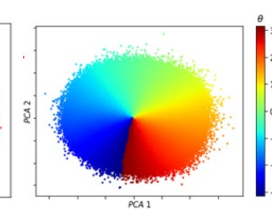
  $\min KL(q(\mathbf{z}|\mathbf{x},\mathbf{y})||p(\mathbf{z}|\mathbf{x},\mathbf{y}))$

  equivalent to

$$\min \|\mathbf{y}-\hat{\mathbf{y}}\|_2^2 + \|\mathbf{x}-\hat{\mathbf{x}}\|_2^2 + KL(q(\mathbf{z} \mid \mathbf{x},\mathbf{y}) \,||\, p(\mathbf{z}))$$

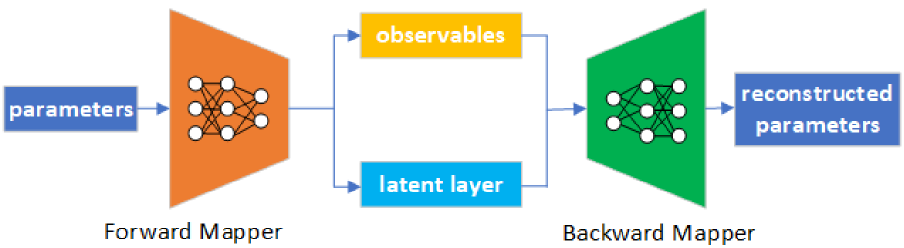
  - True prior distribution $p(\mathbf{z})$
    - Select tractable distribution easy to generate
      - Gaussian
      - Uniform

VAIM on Toy Inverse Problems

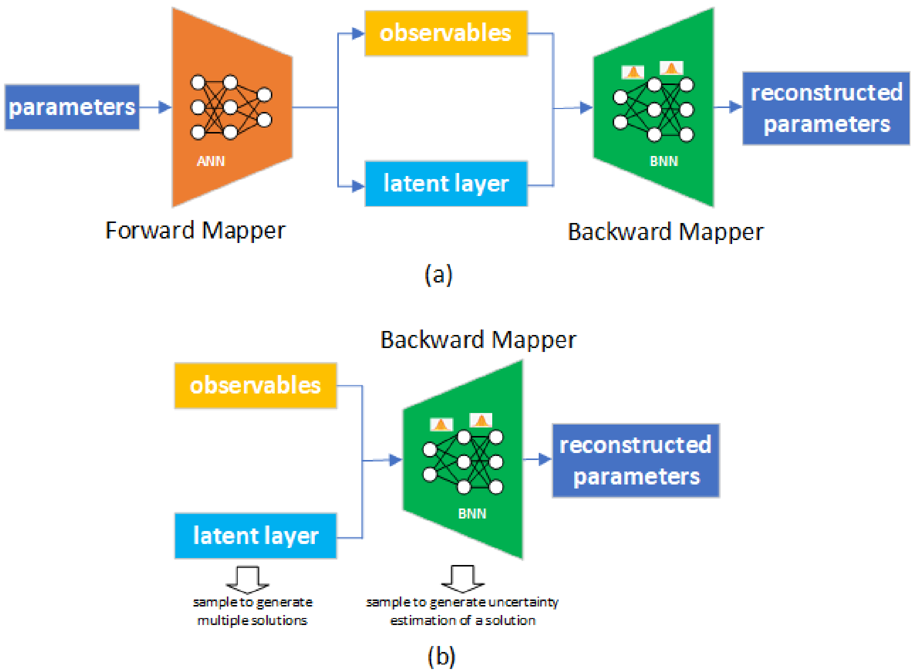
	Multiple Solutions	Latent Space Distribution
$f(x) = x^2$	 Parameter x distribution for $f(x) = 1.0$  Parameter x distribution for $f(x) = 0.36$	Sign Information 
$f(x) = \sin(x)$ $x \in [-2\pi, 2\pi]$	 $f(x) = 1.0$  $f(x) = -0.5$  $f(x) = 0.0$	Period Information 
$f(x_0, x_1)$ $= {x_0}^2 + {x_1}^2$	$f(x_0, x_1) = 1.0$ 	Radius and Polar Angle Information  

BNN-VAIM

VAIM

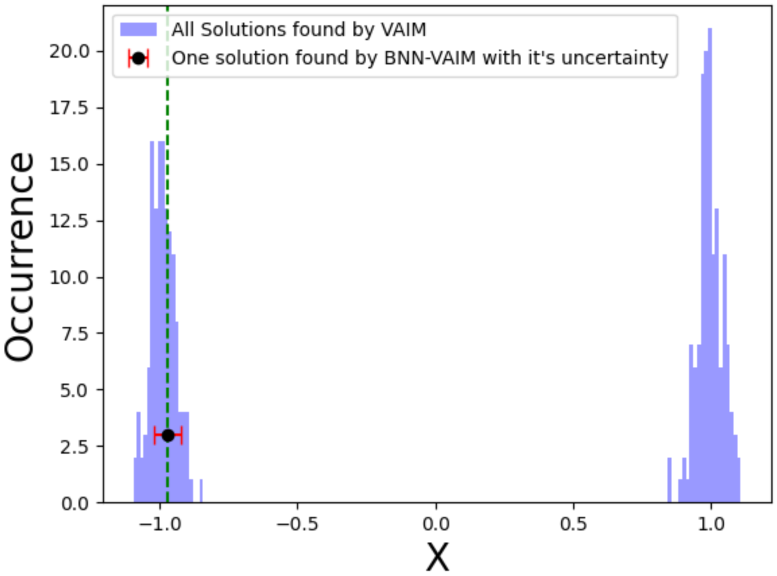


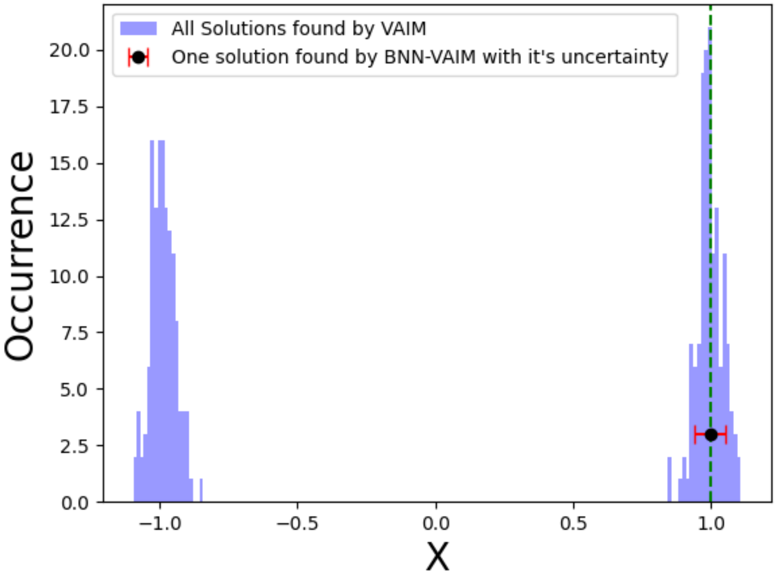
BNN-VAIM



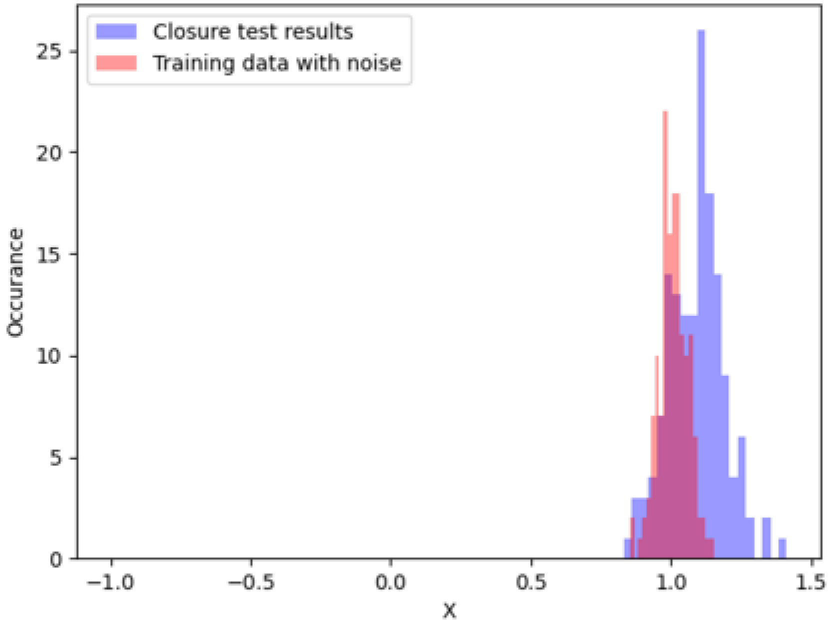
BNN-VAIM on the inverse problem of $f(x) = x^2$

- Input y = 1



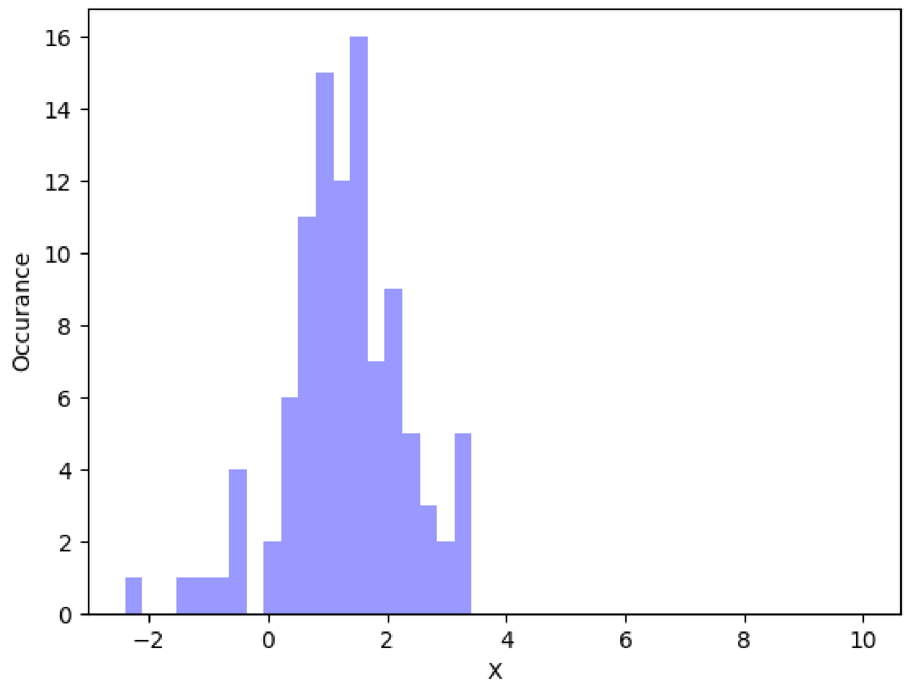


Closure Test



BNN-VAIM Results

- Input y = -10



Conclusion

- **Central Limit Behavior**
  - The model uncertainty complicates $O(N^{-1/2})$
- **BNN**
  - Can be used to estimate aleatory and epistemic uncertainty
  - Seems to be able to represent the posterior distribution even the mapping is not invertible
    - Still need to work out the math
- **Issues of BNN**
  - Much harder to train
  - Much harder to fine-tune
  - More computational intensive
  - More unstable

Acknowledgements

Support from DOE SciDAC and EXCLAIM projects.

AIDAPT Collaboration.