Deep Interval Neural Network in Computational Mechanics

By: David Betancourt and Rafi L. Muhanna

Georgia Tech

REC

Georgia Tech Computational Science and Engineering
College of Computing

Georgia Tech School of Civil and Environmental Engineering
College of Engineering

Presentation Outline

- Part I: Introduction
  - Motivations and broader impact
  - Aleatory and epistemic uncertainty
  - Interval Deep Neural Network algorithms
- Part II: Forward Problem under Uncertainty
  - Predicting input for finite element models
- Part III: Conclusion

Part I

Motivations and Broader Impact

Goals:

(1) quantify interval uncertainty in data analysis model (i.e., deep learning model).

(2) propagate interval uncertainty in data analysis and IFEM model (i.e., deep learning model).

Georgia Tech

Uncertainty in critical infrastructure systems

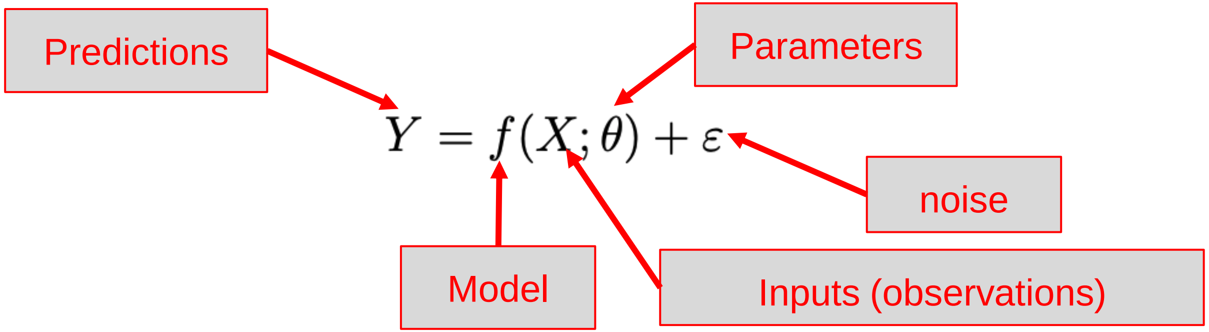
- Insufficient uncertainty modeling can lead to catastrophic events
  - COVID-19
  - Economic policy
  - Social policy
  - Wars

- Need better frameworks to quickly make decisions under uncertainty.
- AI is being deployed to more critical systems
  - **Therefore, need to make AI safer.**

Deep Neural Networks → Function Approximation

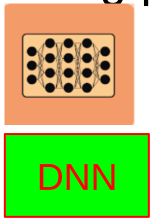
- A function approximation model, also known as curve estimation or regression model, is generally defined as

$$Y = f(X;\theta) + \varepsilon$$



- Deep neural networks act as the function approximation model, which is oftentimes needed to solve engineering problems.

$$f(X;\theta) =$$



Uncertainty Modeling in Engineering and DL (cont)

Frequentist and Bayesian approach:

- Need to make probabilistic assumptions.
- Under Frequentist, need a lot of data. No uncertainty about parameters.
- Under Bayesian, need to update beliefs.

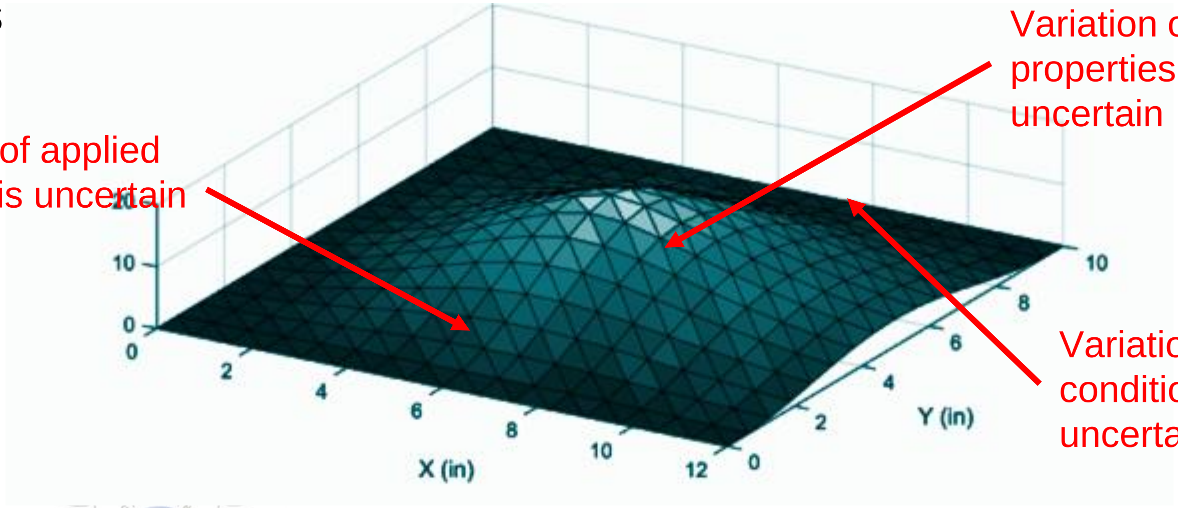
Interval uncertainty approach:

- Do not need to make distribution assumptions.
- Only need to numeric bounds of the input.
- Uses Interval Analysis to propagate uncertainty to guarantee the bounds.

Georgia Tech

Forward problem under uncertainty: *Quantifying spatiotemporal uncertainty in systems inputs for response prediction*

- Many times in civil engineering we lack enough knowledge about the system inputs



Deep Interval Neural Network

Deep Interval Neural Network (DINN)

- DINN is a predictive model *F*. In regression setting:

  $$F : [\underline{X}, \overline{X}] \rightarrow [\underline{Y}, \overline{Y}] \in \mathbb{IR}$$

- Classification setting:

  $$F : [\underline{X}, \overline{X}] \rightarrow y \in \{1, \ldots, C\}$$

- To find the optimal parameter of the DINN (i.e. *training*):
  - Need a training dataset with *n* examples.
  - Each example has *d* dimensionality

- Learning of function **Y** by minimizing a loss (error) function $\mathcal{L}(\hat{F}(\mathbf{X}_i), \mathbf{Y}_i; \mathbf{W})$
  - Mean Square Error for regression, cross-entropy loss (monotonic and gradient is Lipschitz!)

Georgia Tech

DINN: training set (regression setting)

$$\mathcal{T} = \{(\mathbf{X}_1, \mathbf{Y}_1), (\mathbf{X}_2, \mathbf{Y}_2), \ldots, (\mathbf{X}_n, \mathbf{Y}_n)\}$$

Features | Targets

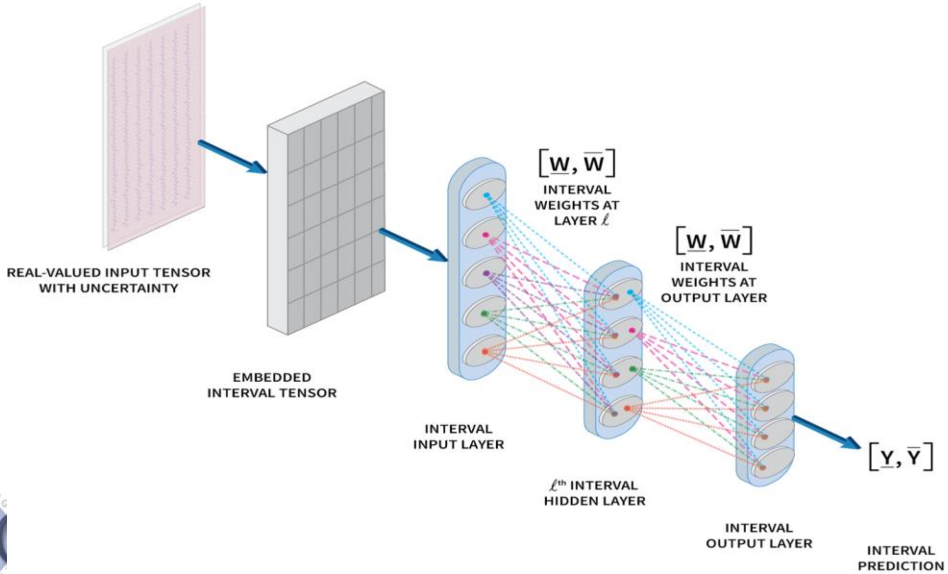
	$\mathbf{X}_{i,1}$	$\mathbf{X}_{i,2}$	...	$\mathbf{X}_{i,d}$	$\mathbf{Y}_i$
example 1	[100.2, 110.3]	[250.1, 268.8]	...	[83.4, 85.7]	[31,33.1]
example 2	[93.7, 95.1]	[260.1, 271.1]	...	[72.1, 73.7]	[25.1, 26.7]
...	...	...	...	...	...
example n	[85.3, 88.1]	[240.4, 255.7]		[78.8, 79.6]	[35.1, 36.1]

$\mathbf{X} \in \mathbb{R}^{n \times d}$ $\mathbf{Y} \in \mathbb{R}^{n}$

Georgia Tech | REC

DINN Training

- Steps for training the network:
    a. Compute the loss at each training iteration via forward propagation.
    b. Compute the gradient of the loss w.r.t. the model interval parameters **W** via Backpropagation.
    c. Minimize the loss function via mini-batch stochastic gradient descent or variants.



Interval Adam

- Adaptive learning rate

Algorithm 3: Adam Algorithm for Interval Input

Input: Step size α

Input: Decay rates $\beta_1, \beta_2 \in [0, 1)$

Input: Initial parameters $\mathbf{W} \in \mathbb{IR}$

Initialize $\mathbf{m}_0 \in \mathbb{IR} = 0$

Initialize $v_0 \in \mathbb{R} = 0$

Initialize $k = 0$

while *stopping criteria not met* **do**

$k \leftarrow k + 1$

$\mathbf{G}_k \leftarrow \nabla_{\mathbf{W}} \mathcal{L}_k$ compute interval gradients at step k

$\mathbf{m}_k \leftarrow \beta_1 \mathbf{m}_{k-1} + (1 - \beta_1)\mathbf{G}_k$ biased interval first moment estimate (interval)

$v_k \leftarrow \beta_2 v_{k-1} + (1 - \beta_2) \cdot \mathtt{mid}(\mathbf{G}_k)^2$ biased real second moment estimate (real)

$\widehat{\mathbf{m}}_k \leftarrow \mathbf{m}_k / (1 - \beta_1^k)$

$\widehat{v}_k \leftarrow v_k / (1 - \beta_2^k)$

$\mathbf{W}_k \leftarrow \mathbf{W}_{k-1} - \alpha \cdot \widehat{\mathbf{m}}_k / (\sqrt{\widehat{v}_k} + \epsilon)$

return $\mathbf{W}_k$

Mini-batch SGD

- Optimization is done via stochastic gradient descent
  - Update rule:

$$\mathbf{W}_{k+1}^{(l)} = \mathbf{W}_k^{(l)} - \alpha_k \nabla_{\mathbf{W}_k^{(l)}} \mathcal{L}(\hat{F}(\mathbf{X}_i), \mathbf{Y}_i)$$

- Also:
  - SGD + momentum: accounts for past gradients
  - Interval Adam: Adaptive

Part II

Forward Problem Under Uncertainty

Supervised Interval Field

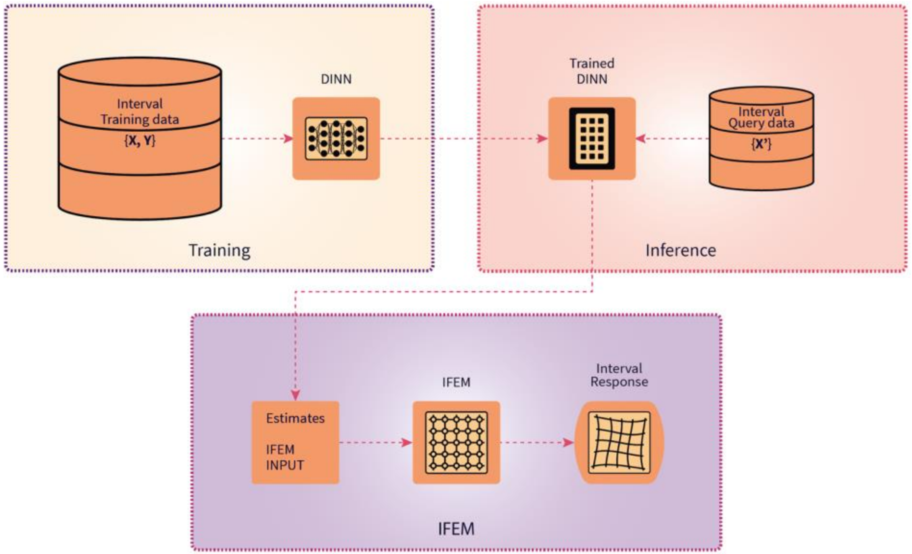
- IFEM:
  - Need a way to quantify spatiotemporal-varying uncertainty in the model input.
- Solution:
  - Use *real-valued* deep neural network to infer the uncertain properties in the field from indirect measurements:
    - ***Called: "Supervised Interval Field"***
  - Works for any domain dimension (1D, 2D, 3D).
  - Independent of IFEM mesh discretization.

Quantify Input Variation and Interval Uncertainty

- Quantifying the variation of a property **Y** using observations **X**, considering interval uncertainty in **X** and **Y**.

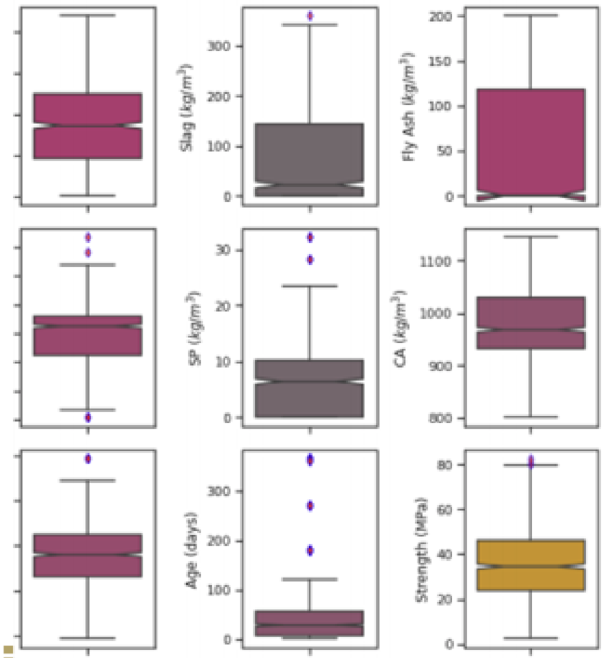
SIF-IFEM a unified framework

- SIF propagates to the IFEM the uncertain property with respect to the model’s mesh.
- SIF is independent of mesh, therefore, must discretize and map into IFEM mesh.
- Average of the prediction taken over the size of FE.



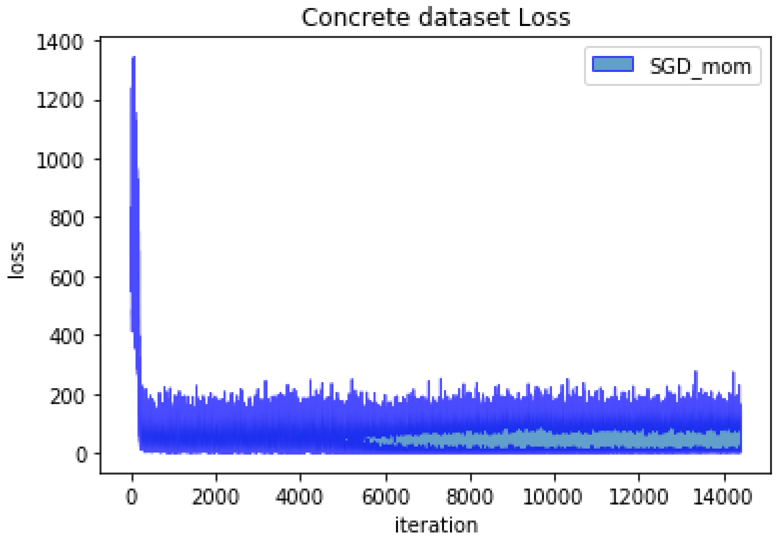
Experiments: UCI Concrete Dataset

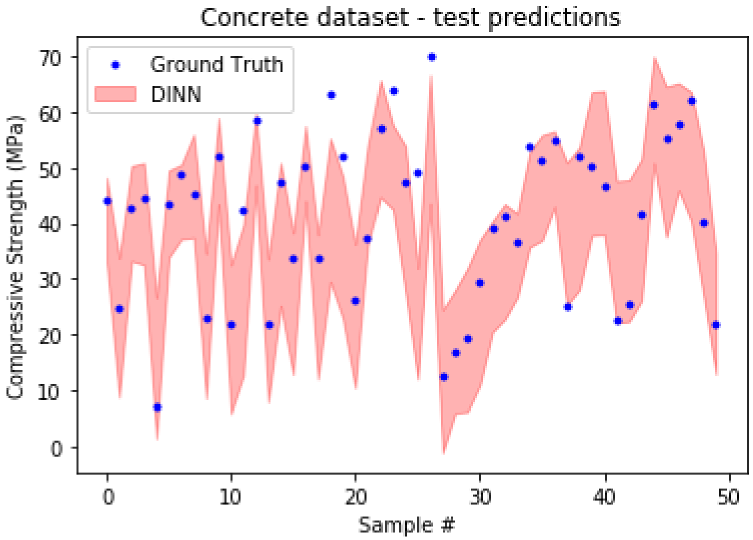
- Boxplots for dataset



- DINN details:
    - Hidden Dimensions: H1 = 500, H2 = 500, H3 = 500.
    - Mini-batch size = 16
    - Learning Rate = 0.005
    - L2 regularization.
    - Features: 8 chemical/physical measurements.
    - Targets: Concrete Strength.
    - Interval Uncertainty for targets: 15% for superplasticizer, 5% for all other features.

Experiments: UCI Concrete Dataset





Part III

Conclusion

Conclusion

- Deep Interval Neural Network was developed in the regression setting for engineering problems.

Acknowledgements: This research is supported by the National Science Foundation grant No. 1634483.

Thank you!

Questions?