VIBRATION ANALYSIS OF TAPERED TRUNCATED CANTILEVER BEAMS USING DEEP LEARNING

Naman Yadav

Asim Tewari

Department of Mechanical Engineering, IIT Bombay

IMEC 2019

OVERVIEW

- Motivation
- Deep Learning Model
- Data Generation
- Model Training
  - Training parameters and results
  - Percent deviations of predicted values
  - Graphical comparison with actual values
- Extrapolation Analysis
  - Prediction ability over unseen data range
- Conclusion
- Future Research

OVERVIEW

- Motivation

MOTIVATION

- ✓Some problems are really difficult or impossible to solve theoretically.
- ✓In some cases these can be solved by fitting curve to some data-points generated for special cases.
- ✓Data driven approach:
  1. Removes necessity of field specific knowledge.
  2. There is a large amount of data present.
     - Highly optimized algorithms are there to process this data.
  3. Tremendous computational power available.

OVERVIEW

- Motivation
- Deep Learning Model

DEEP LEARNING MODEL

1. Training data was generated using MATLAB.
2. Neural network features:
    1. Activation function: PReLU
    2. Loss function: Mean squared percent error
    3. Optimizer: Adam optimizer
3. Training of Neural network was done on Python using PyTorch.

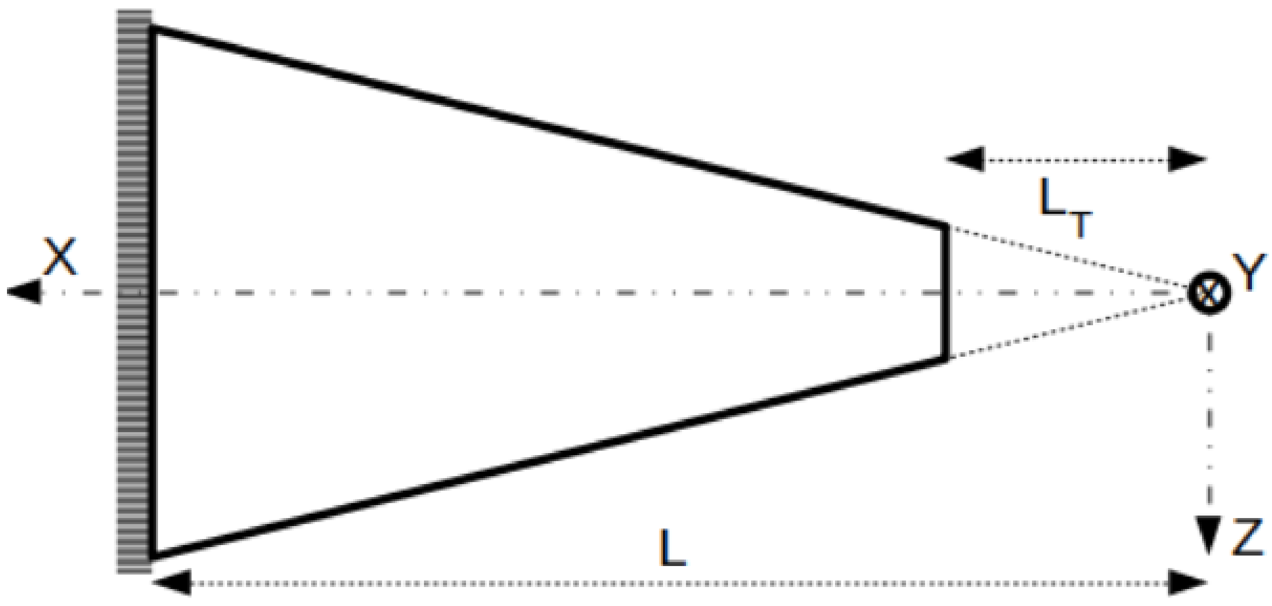
INPUT VARIABLES

- Variation in cross-sectional properties:

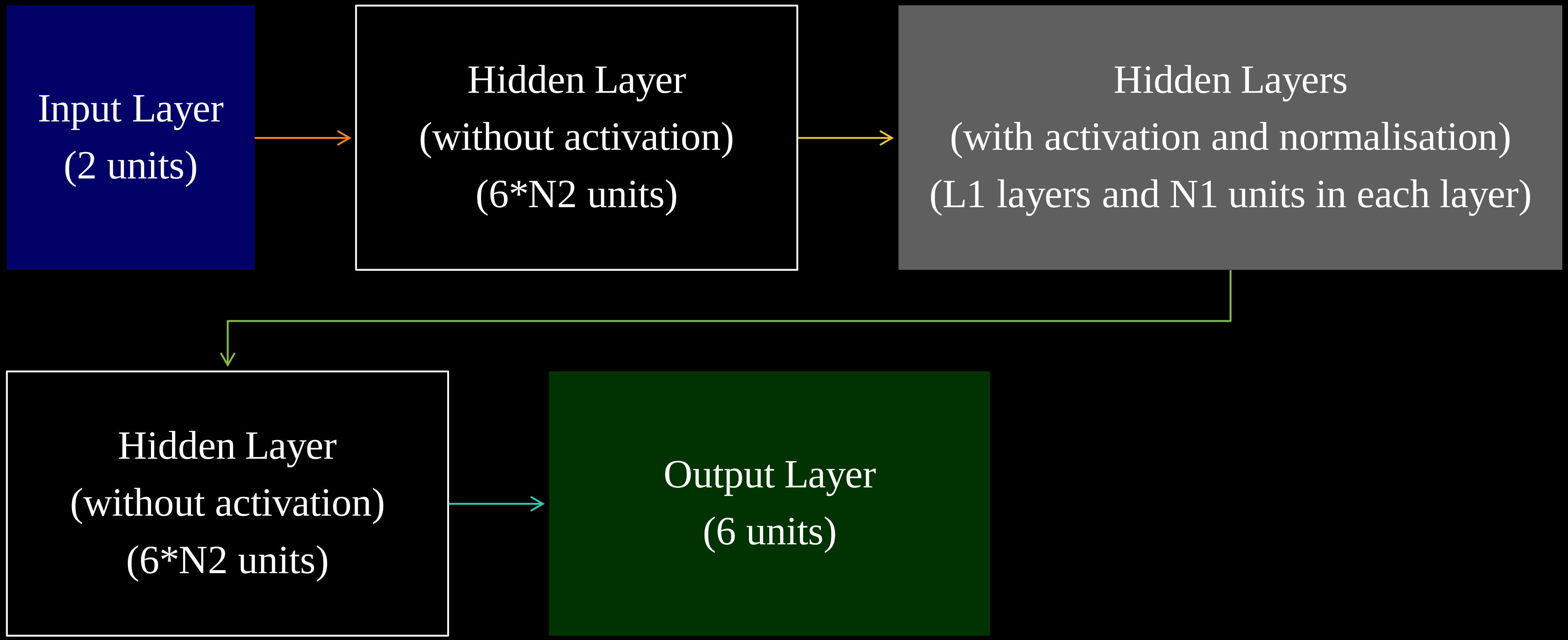
$$A = A_C \delta^{\eta}$$
$$I = I_C \delta^{\eta+2}$$

(where, $\delta = \frac{x}{L}$, $\delta \in [\delta_0, 1]$ and η is a non−negative real number)

- η and δ_0 are inputs to the neural network.

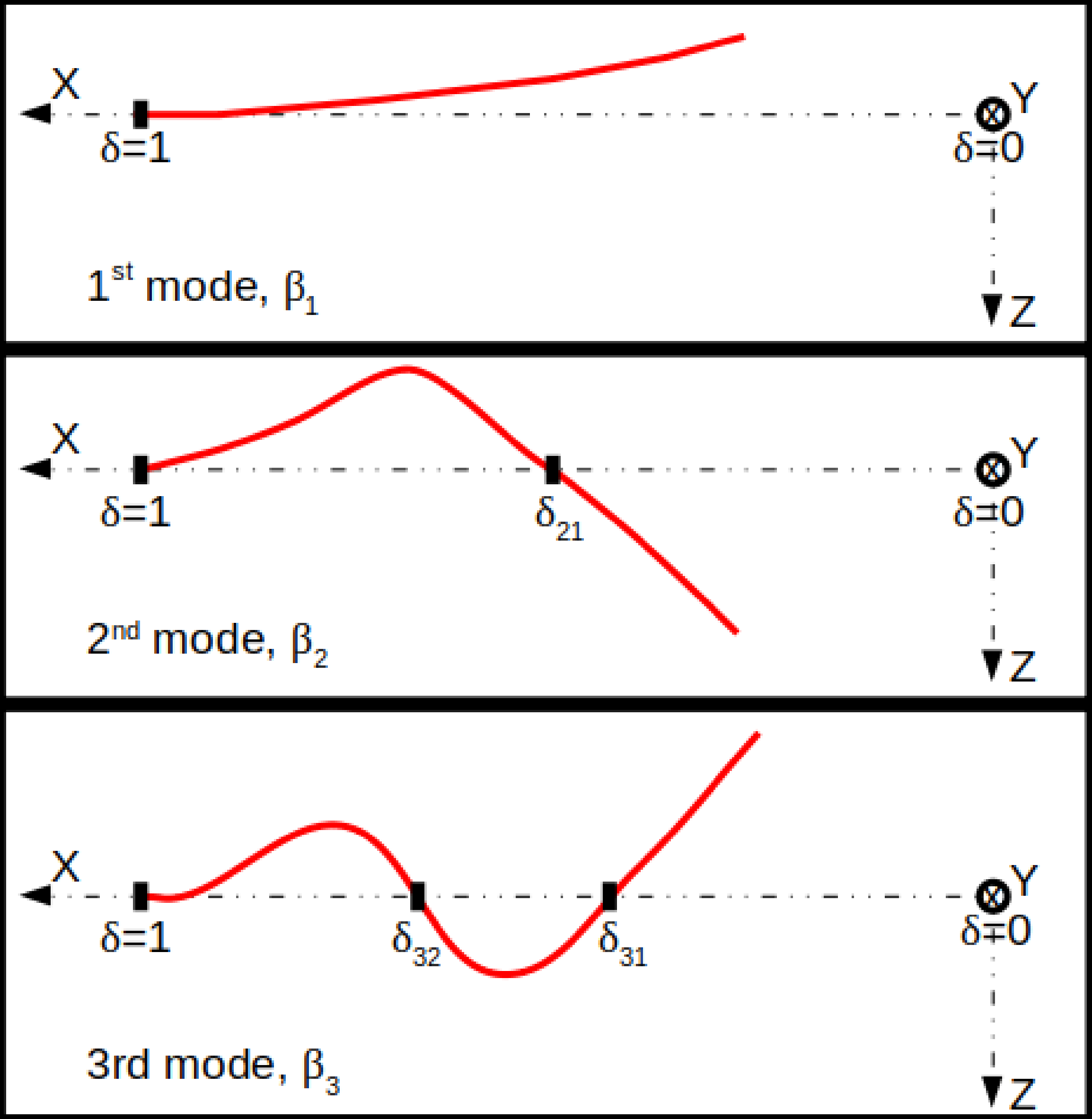


NEURAL NETWORK ARCHITECTURE



OUTPUT VARIABLES

- $\beta_1, \beta_2, \beta_3, \delta_{21}, \delta_{31}, \delta_{32}$ (6 variables)
  - $\beta_1, \beta_2, \beta_3$ are the natural frequency parameters
  - $\delta_{21}, \delta_{31}, \delta_{32}$ are the node locations



ACTUAL VARIATION OF OUTPUT VARIABLES

1. Euler-Bernoulli equation

$$\frac{\partial^2}{\partial x^2}\left(EI(x)\frac{\partial^2 y(x,t)}{\partial x^2}\right)+\rho A(x)\frac{\partial^2 y(x,t)}{\partial t^2}=0$$

2. Using variable separation

$$y(x,t)=W(x)e^{i\omega t}$$

3. General solution of $W(x)$

$$W(\delta)=(L\delta)^{-\eta/2}\left[c_1 J_\eta(z)+c_2 Y_\eta(z)+c_3 I_\eta(z)+c_4 K_\eta(z)\right]$$

where, $z=2\beta\delta^{1/2}$, $\beta=\omega^2 L^4\left(\frac{\rho A_C}{EI_C}\right)$, $\delta=\frac{x}{L}$

J_η and Y_η are η^{th} order Bessel Functions of first and second kind, respectively.
I_η and K_η are modified η^{th} order Bessel Function of first and second kind, respectively.

4. Boundary conditions

$$\frac{\partial}{\partial \delta}\left(EI(\delta)\frac{\partial^2 W(\delta)}{\partial \delta^2}\right)=\frac{\partial^2 W(\delta)}{\partial \delta^2}=0, at\delta=\delta_0$$

$$W(\delta)=\frac{\partial W(\delta)}{\partial \delta}=0, at\delta=\delta_1=1$$

ACTUAL VARIATION OF OUTPUT VARIABLES

6. After applying boundary conditions we will get 4 linear equations:

$$\Delta = \begin{vmatrix} J_{\eta+1}(z_0) & Y_{\eta+1}(z_0) & I_{\eta+1}(z_0) & -K_{\eta+1}(z_0) \\ J_{\eta+2}(z_0) & Y_{\eta+2}(z_0) & I_{\eta+2}(z_0) & K_{\eta+2}(z_0) \\ J_{\eta}(z_1) & Y_{\eta}(z_1) & I_{\eta}(z_1) & K_{\eta}(z_1) \\ J_{\eta+1}(z_1) & Y_{\eta+1}(z_1) & -I_{\eta+1}(z_1) & K_{\eta+1}(z_1) \end{vmatrix} = 0$$

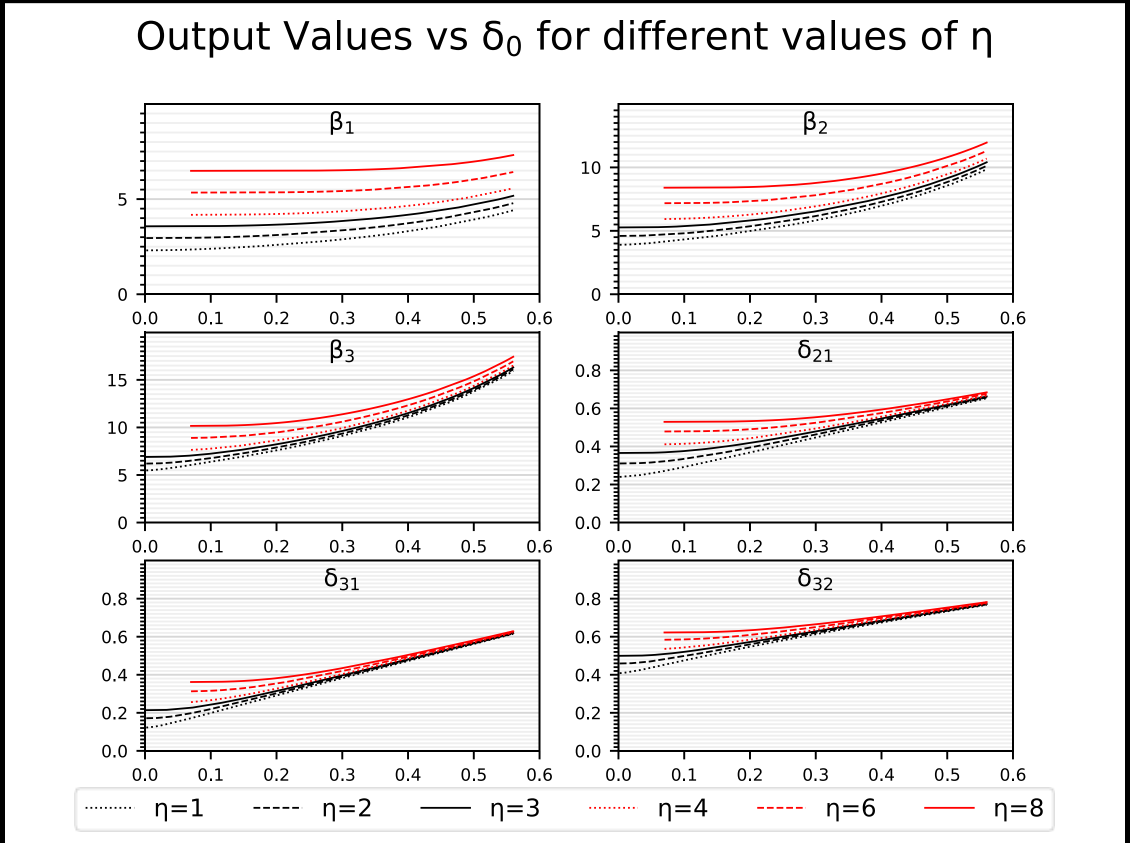
7. After solving $\Delta = 0$, we can get values of $\beta = \beta_r$, where, r is the mode number. From which, we can compute ω_r and z_r using following relations,

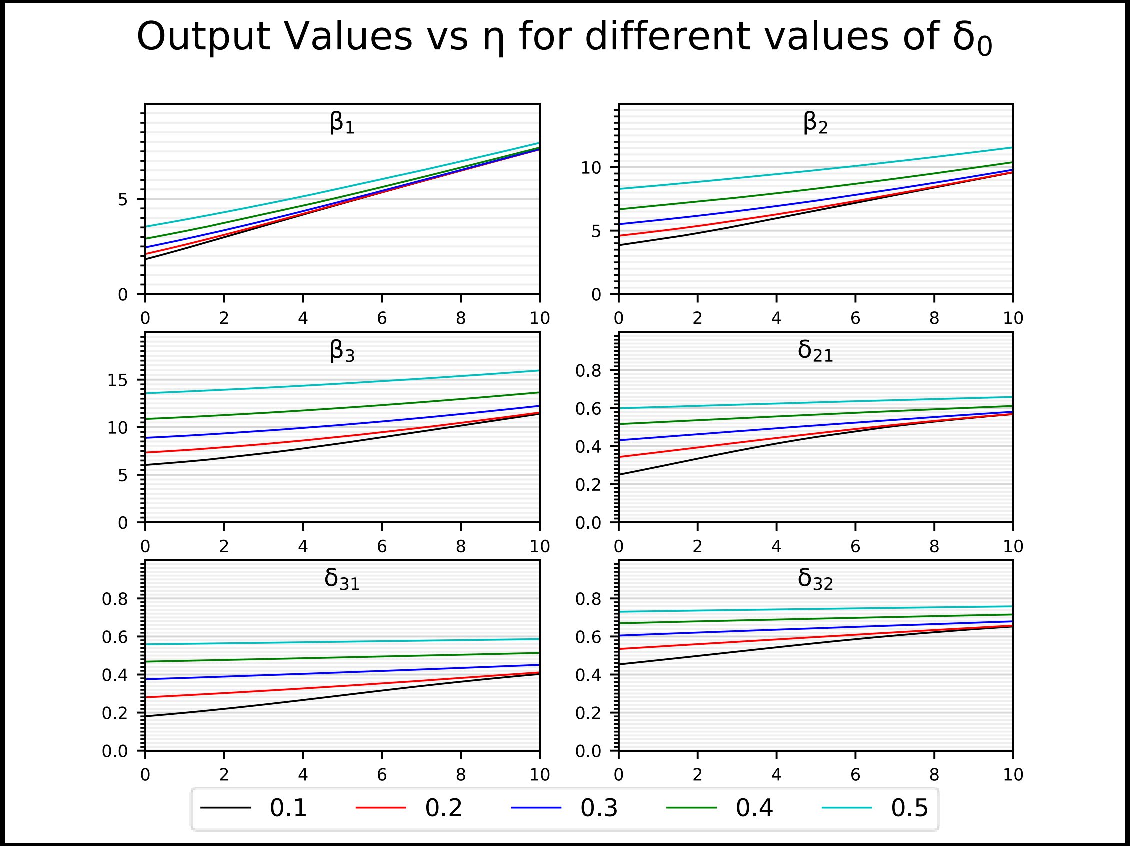
$$\omega_r = \left(\frac{\beta_r}{L}\right)^2 \sqrt{\frac{EI_C}{\rho A_C}},\ z_r = 2\beta_r \delta^{1/2}$$

8. Finally, mode shapes can be obtained as follows,

$$W(\delta) = (L\delta)^{-\eta/2}\left[c_1 J_\eta(z_r) + c_2 Y_\eta(z_r) + c_3 I_\eta(z_r) + c_4 K_\eta(z_r)\right]$$

ACTUAL VARIATION OF OUTPUT VARIABLES





OVERVIEW

- Motivation
- Deep Learning Model
- Data Generation

DATA GENERATION

1. **η** ϵ [0, 3], **δ_0** ϵ [2e-3, 0.56] => 301*15,000 datapoints.
2. **η** ϵ [3, 10], **δ_0** ϵ [7e-2, 0.56] => 701*15,000 datapoints.
3. For **η** ϵ [0, 10], data from the two cases was merged:
   - **η** ϵ [0, 3], **δ_0** ϵ [2e-3, 0.56] => 301*15,000 = 4,515,000.
   - **η** ϵ (3, 10], **δ_0** ϵ [7e-2, 0.56] => 700*15,000 = 10,500,000.

➢ In total 15,015,000 datapoints were generated.

➢ But only a fraction was used for training.

OVERVIEW

- Motivation
- Deep Learning Model
- Data Generation
- Model Training

MODEL TRAINING

- Training and validation data
  - Generated data was shuffled.
  - Few data points were sampled from shuffled data to train the neural network.
  - Some datapoints for validation were also selected.
- Test data
  - A separate data set was created by randomly sampling points.
- Results
  - Trained model was evaluated over the unseen test dataset to report the results.

OVERVIEW

- Motivation
- Deep Learning Model
- Data Generation
- Model Training
  - Training parameters and results

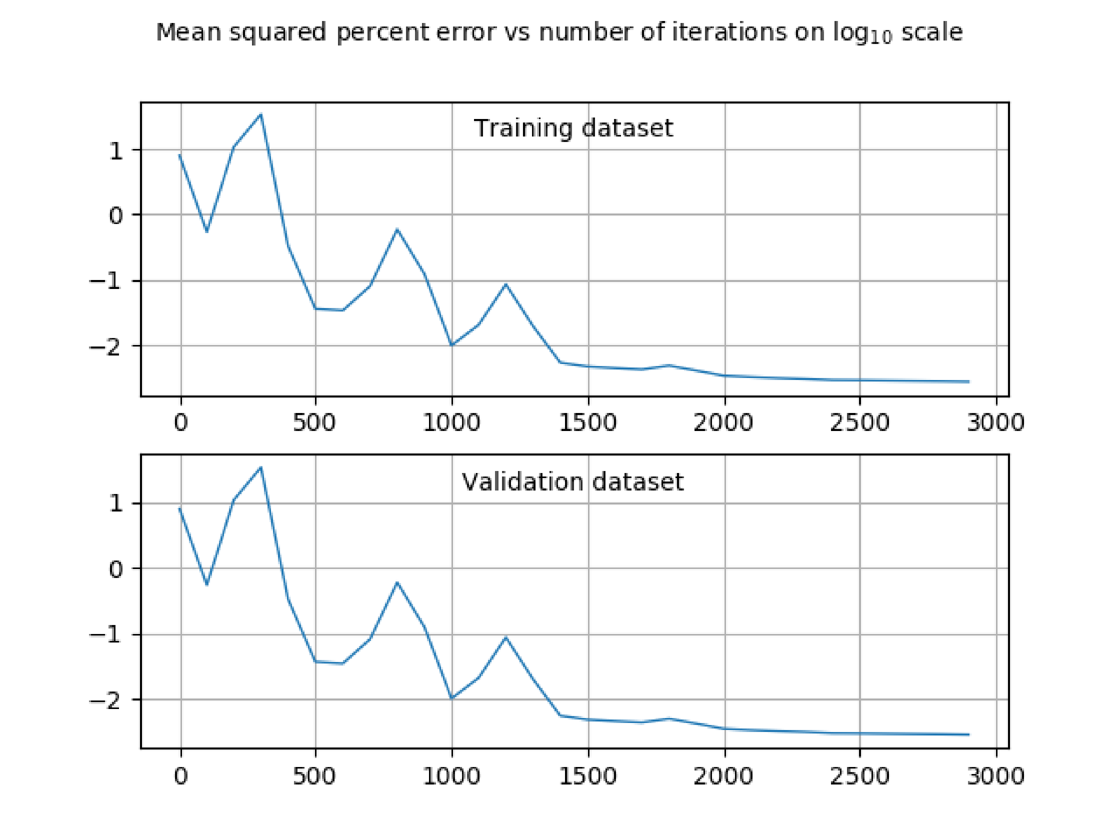
TRAINING PARAMETERS

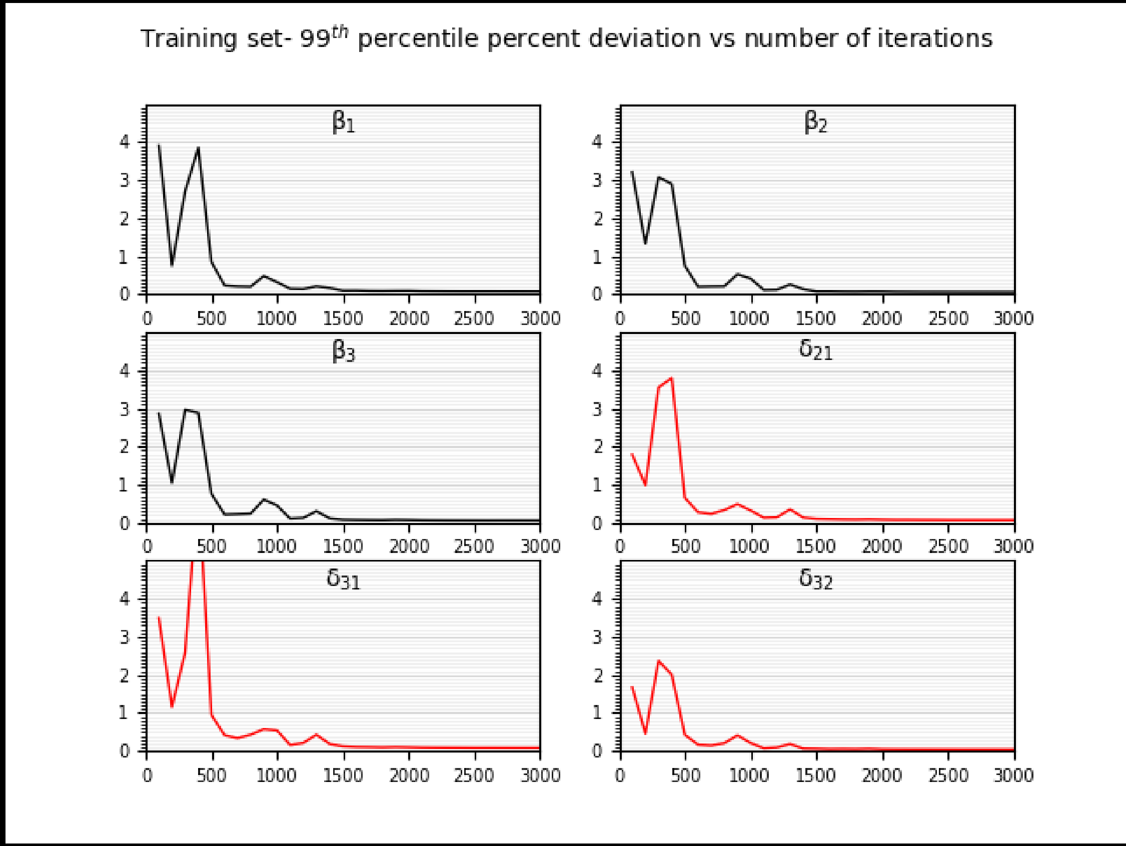
Table 1: Deep Learning model training parameters		
	SD (Small Dataset)	**BD (Big Dataset)**
# Training datapoints	$2*2^{11}$	$20*2^{11}$
# Validation datapoints	1,000	
# Iterations	30,000	
Initial Learning Rate	0.01	
Update period	5,000	
Update factor	5	

TRAINING RESULTS

	SD (Small Dataset)	BD (Big Dataset)
# Training datapoints	$2*2^{11}$	$20*2^{11}$
L1	3	7
N1	173	207
N2	3	3
	SN (Small Network)	BN (Big Network)

TRAINING RESULTS





OVERVIEW

- Motivation
- Deep Learning Model
- Data Generation
- Model Training
  - Training parameters and results
  - Percent deviations of predicted values

PERCENT DEVIATIONS

Table 2: For $\eta \in [0, 3]$, $\delta_0 \in [2e\text{-}3, 0.56]$

Output	Mean Percent dev.		Max. Percent dev.	
	SD-SN	BD-BN	SD-SN	BD-BN
β_1	0.022	0.011	0.115	0.106
β_2	0.020	0.008	0.110	0.071
β_3	0.021	0.009	0.111	0.052
δ_{21}	0.016	0.012	0.172	0.101
δ_{31}	0.024	0.013	0.317	0.255
δ_{32}	0.012	0.007	0.159	0.083

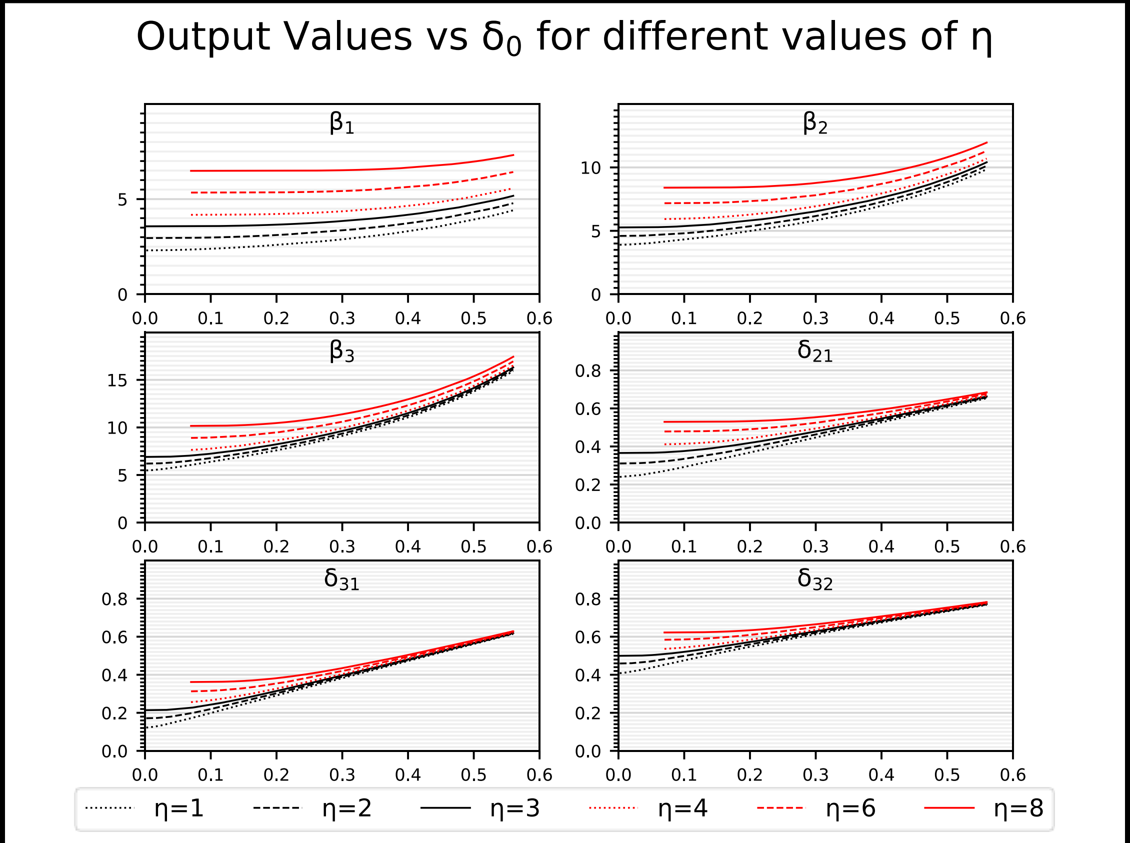
Table 3: For $\eta \in [3, 10]$, $\delta_0 \in [7e\text{-}2, 0.56]$

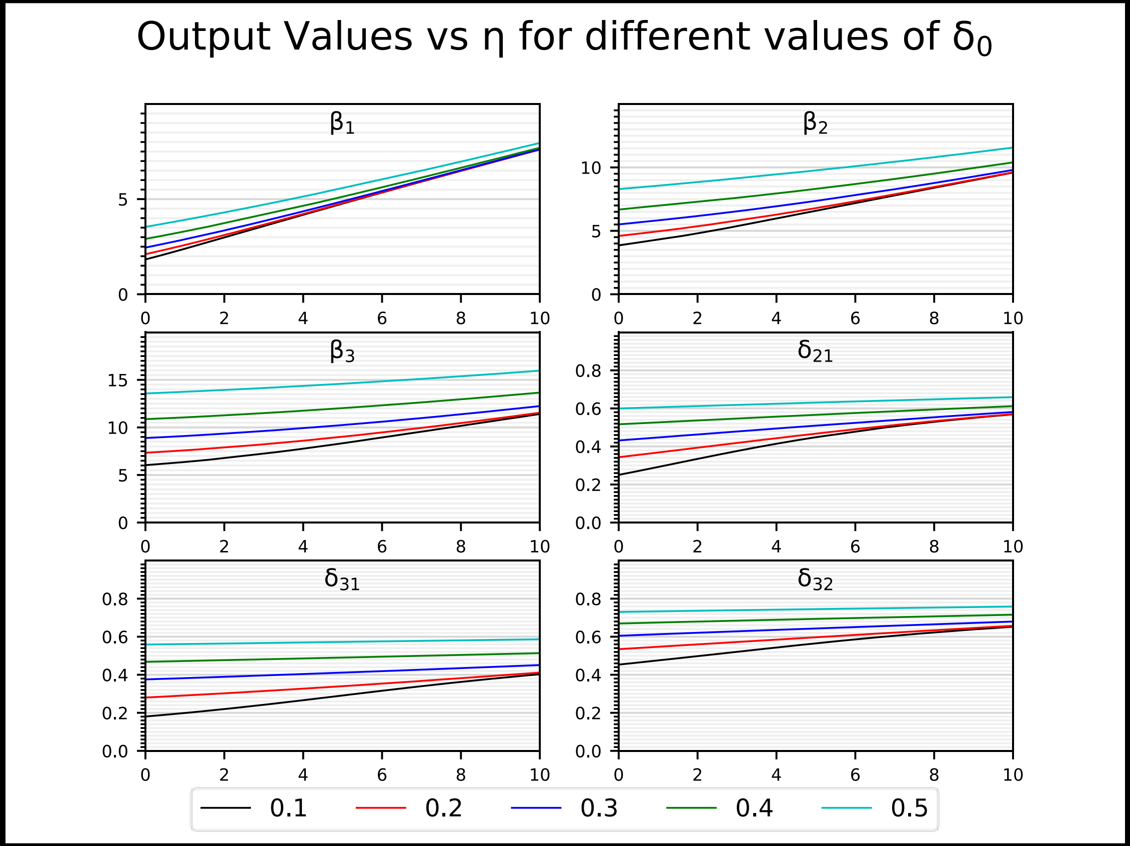
Output	Mean Percent dev.		Max. Percent dev.	
	SD-SN	BD-BN	SD-SN	BD-BN
β_1	0.019	0.011	0.130	0.079
β_2	0.018	0.010	0.130	0.046
β_3	0.019	0.010	0.123	0.054
δ_{21}	0.013	0.009	0.090	0.050
δ_{31}	0.019	0.009	0.169	0.051
δ_{32}	0.010	0.005	0.055	0.037

OVERVIEW

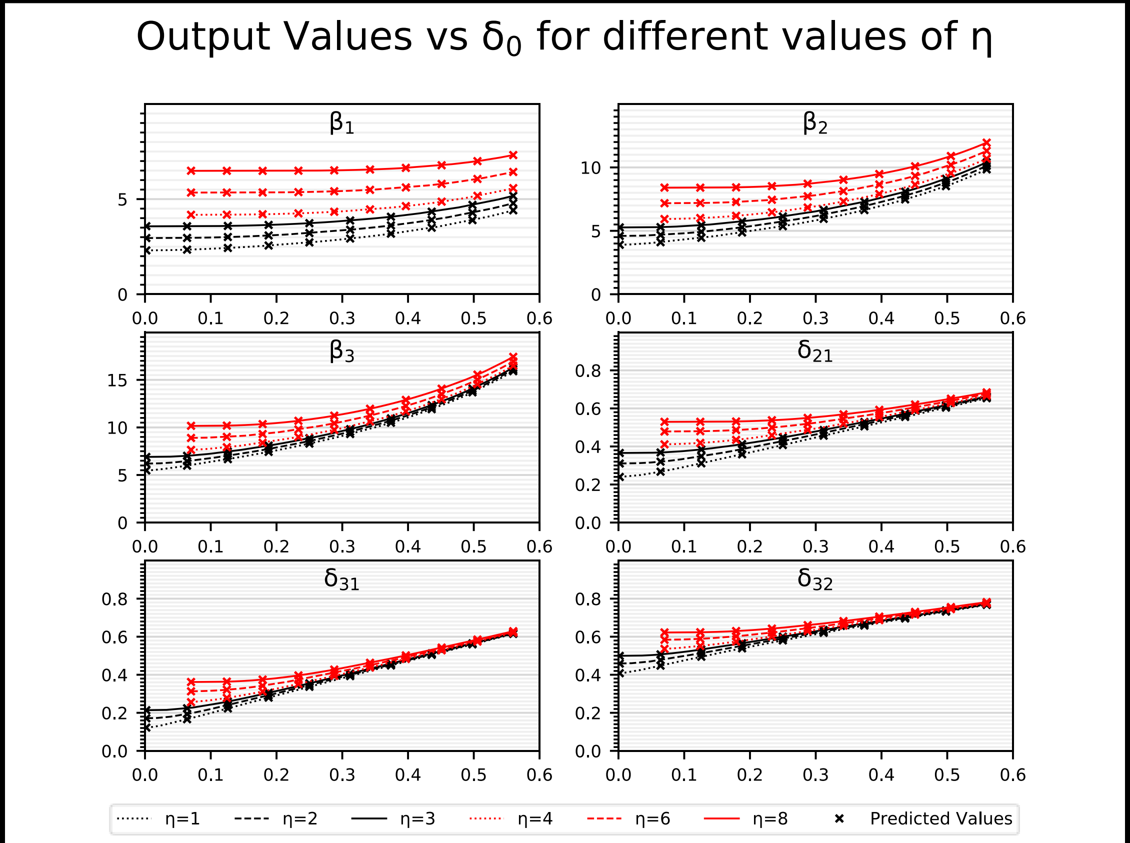
- Motivation
- Deep Learning Model
- Data Generation
- Model Training
  - Training parameters and results
  - Percent deviations of predicted values
  - Graphical comparison with actual values

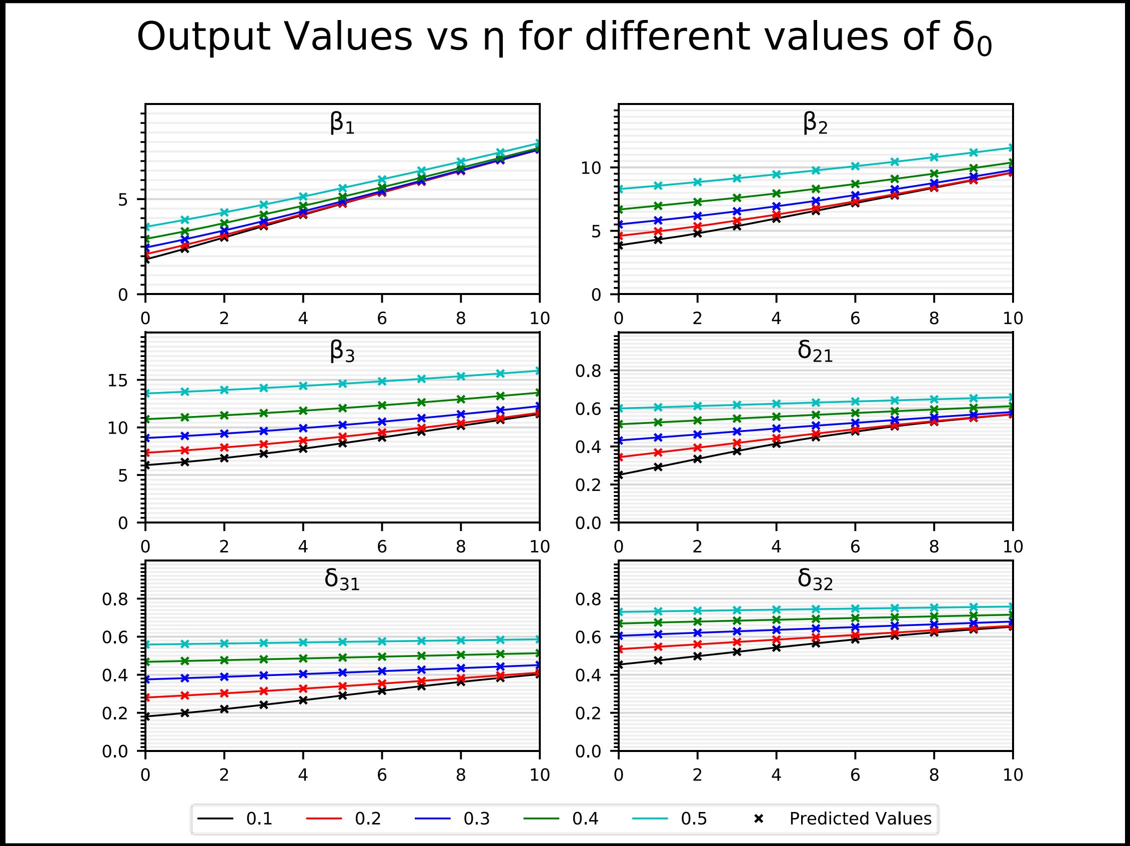
COMPARISON BETWEEN ACTUAL AND PREDICTED VALUES





COMPARISON BETWEEN ACTUAL AND PREDICTED VALUES





OVERVIEW

- Motivation
- Deep Learning Model
- Data Generation
- Model Training
  - Training parameters and results
  - Percent deviations of predicted values
  - Graphical comparison with actual values
- Extrapolation Analysis

EXTRAPOLATION ANALYSIS

- Data was then split back into two parts as generated:
  Part 1: $\boldsymbol{\eta}$ ϵ [0, 3], $\boldsymbol{\delta_0}$ ϵ [2e-3, 0.56]
  Part 2: $\boldsymbol{\eta}$ ϵ [3, 10], $\boldsymbol{\delta_0}$ ϵ [7e-2, 0.56]

- Models corresponding to SD and BD mentioned earlier were trained on one part and then evaluated on the other to see its extrapolation capabilities.

OVERVIEW

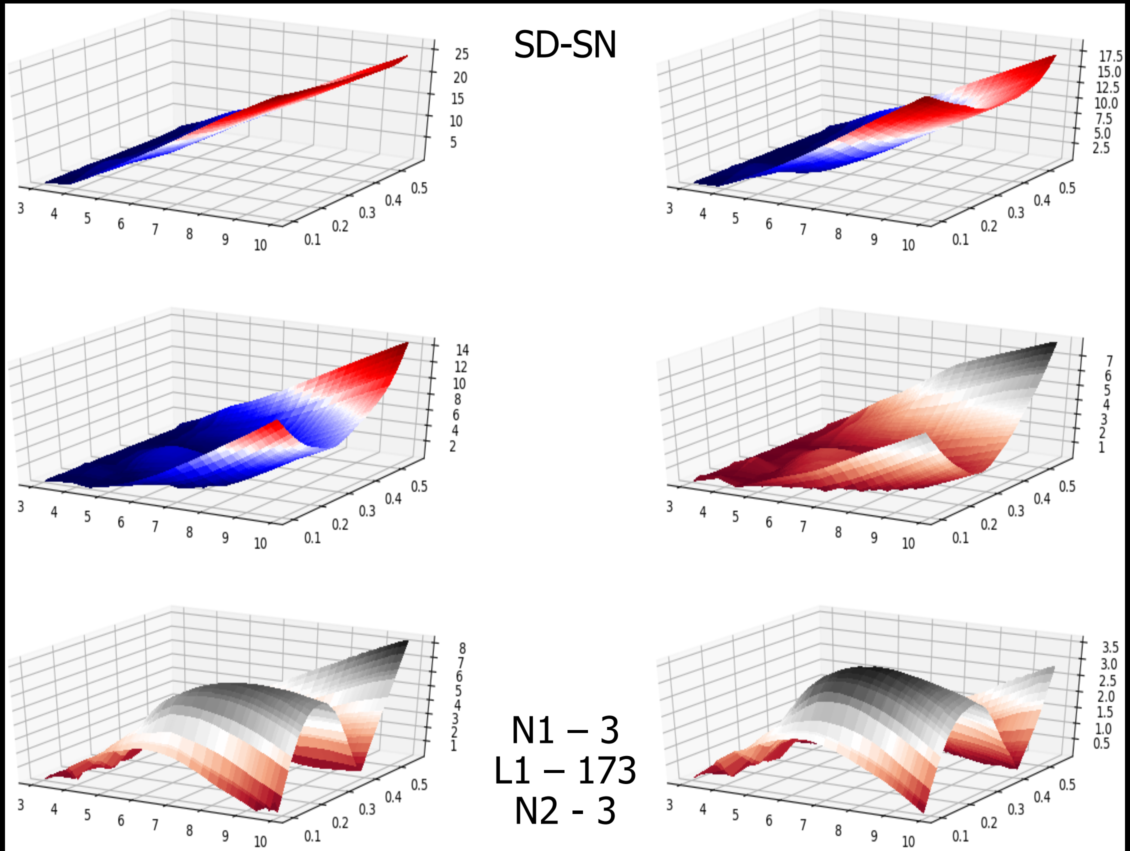
- Motivation
- Deep Learning Model
- Data Generation
- Model Training
  - Training parameters and results
  - Percent deviations of predicted values
  - Graphical comparison with actual values
- Extrapolation Analysis
  - Prediction ability over unseen data range

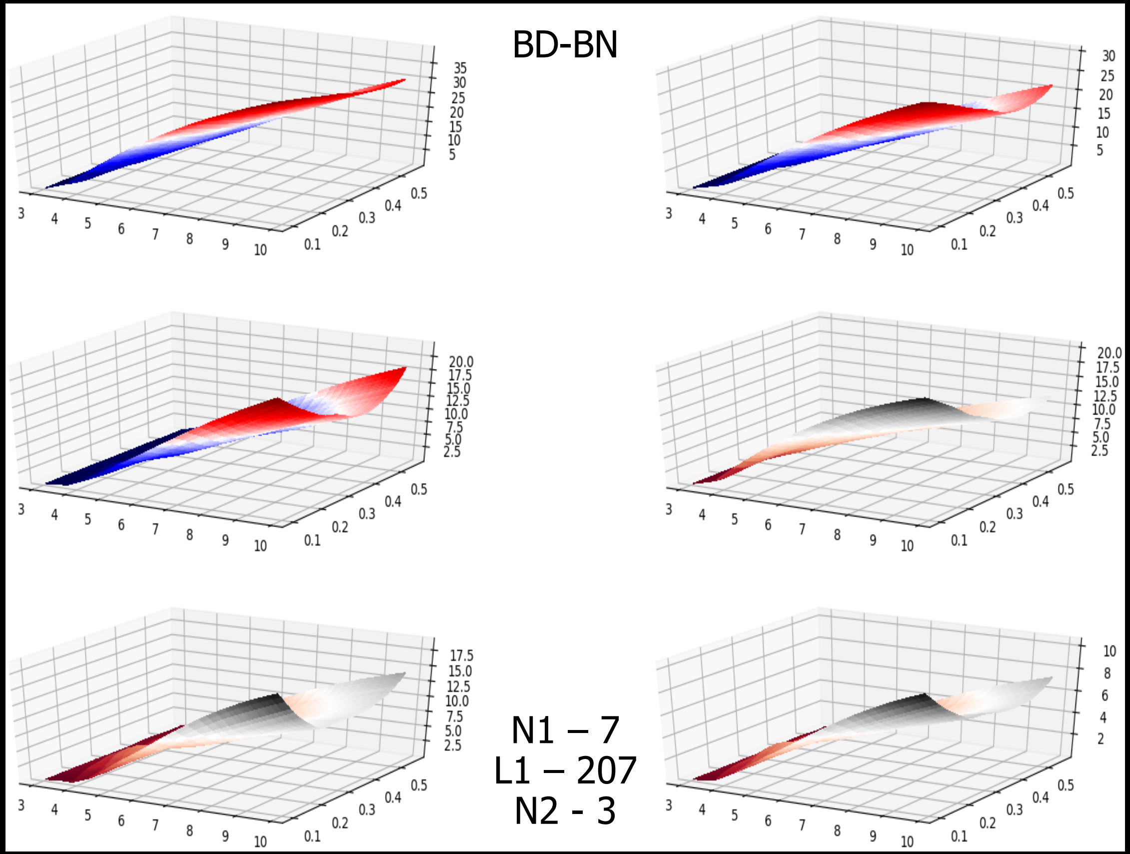
TRAINED ON PART 1 EVALUATED OVER PART 2

Table 4: Trained over η є [0, 3] and evaluated over η є [3, 10]

Output	Mean Percent dev.		Max. Percent dev.	
	SD-SN	BD-BN	SD-SN	BD-BN
β_1	12.922	19.327	26.671	40.068
β_2	7.082	12.582	19.416	31.156
β_3	3.737	8.609	13.828	23.150
δ_{21}	1.866	8.318	7.643	21.166
δ_{31}	2.919	7.453	7.816	20.621
δ_{32}	1.331	4.310	3.593	11.100

PERCENT DEVIATION VARIATION

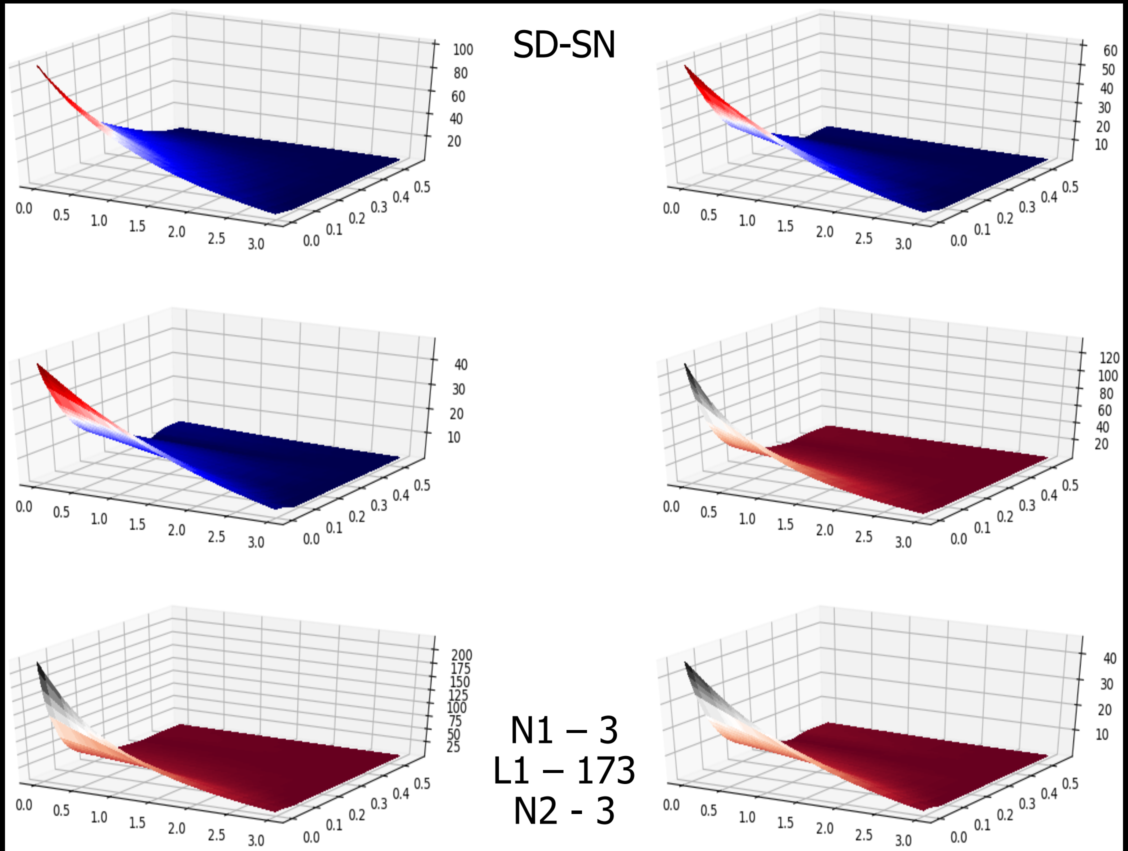


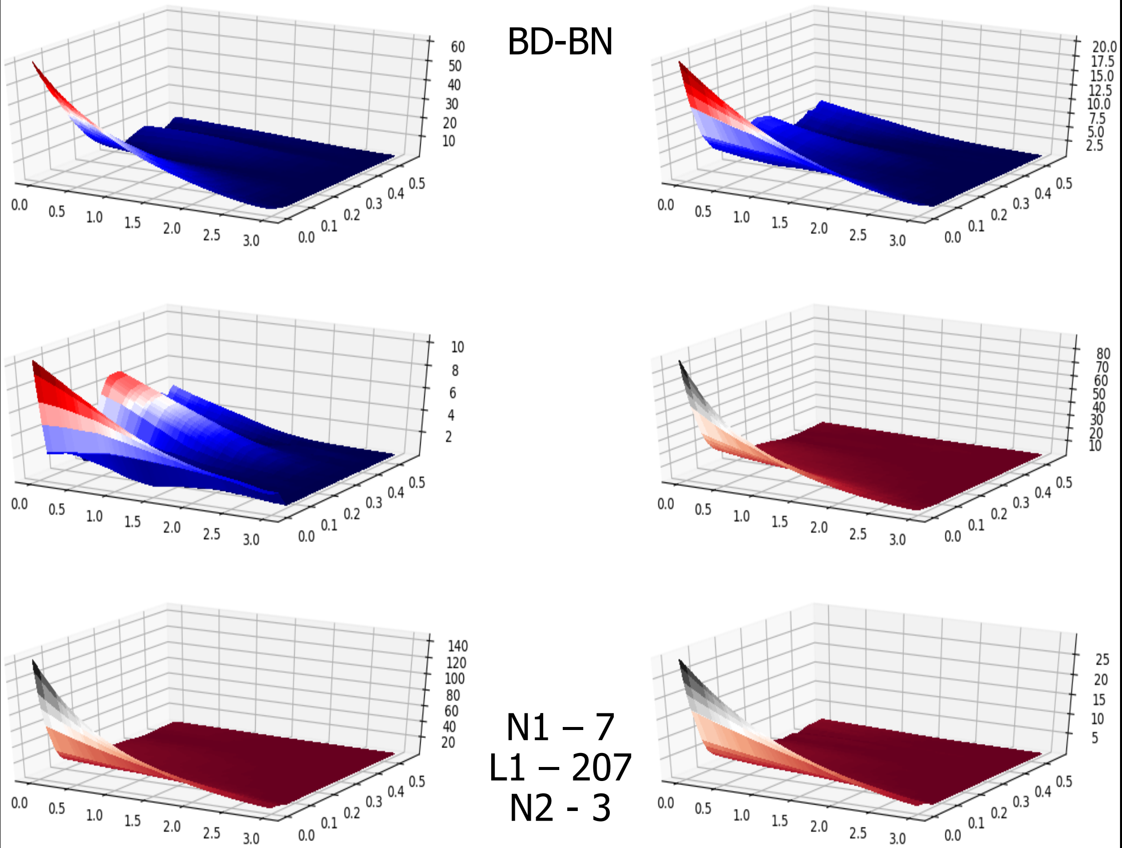


TRAINED ON PART 2 EVALUATED OVER PART 1

Table 5: Trained over η ϵ [3, 10] and evaluated over η ϵ [0, 3]				
Output	Mean Percent dev.		Max. Percent dev.	
	SD-SN	BD-BN	SD-SN	BD-BN
β_1	10.620	4.630	94.813	57.242
β_2	6.181	1.808	58.538	19.544
β_3	4.826	1.361	45.678	10.128
δ_{21}	7.060	3.664	121.945	80.435
δ_{31}	9.475	4.975	202.181	134.401
δ_{32}	3.174	1.408	43.375	27.963

PERCENT DEVIATION VARIATION





OVERVIEW

- Motivation
- Deep Learning Model
- Data Generation
- Model Training
  - Training parameters and results
  - Percent deviations of predicted values
  - Graphical comparison with actual values
- Extrapolation Analysis
  - Prediction ability over unseen data range
- Conclusion

CONCLUSION

	Small Neural Network	Large Neural Network
Simple trend in Training Data compared to evaluation data	Not Recommended	Recommended
Complex trend in Training Data compared to evaluation data	Recommended	Not Recommended

OVERVIEW

- Motivation
- Deep Learning Model
- Data Generation
- Model Training
  - Training parameters and results
  - Percent deviations of predicted values
  - Graphical comparison with actual values
- Extrapolation Analysis
  - Prediction ability over unseen data range
- Conclusion
- Future Research

FUTURE RESEARCH

- These models can also be trained on data for beams subjected to other boundary conditions.
  - Due to similar underlying equations, we can expect reasonably high accuracy in predictions.
- Other better models could be developed to improve the extrapolation accuracy of the neural network far away from trained range.
  - Challenging part in doing this is the complex nature of Bessel functions, which contain integrations of factorial, factorials in denominator, etc.

THANK YOU

Questions ?