

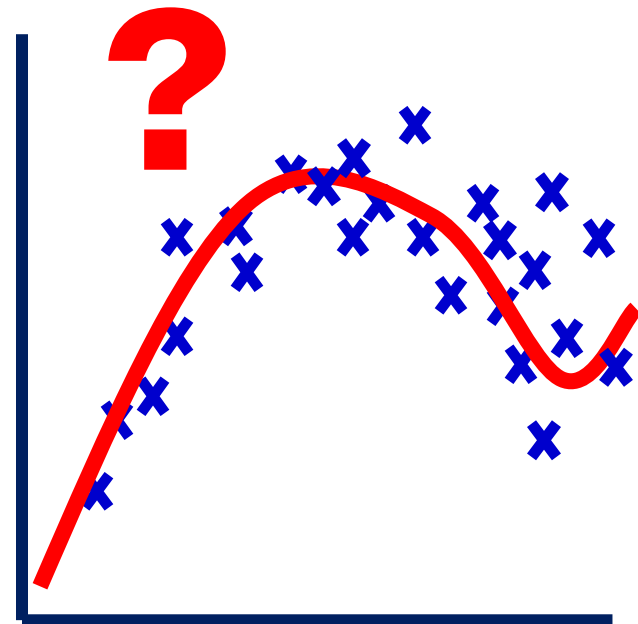
Probabilistic Neural Networks with Embedded Linear Equality Constraints

Matthew Marsh¹,
Benoît Chachuat¹,
Antonio del Rio Chanona¹
1. Sargent Centre for Process Systems Engineering, Department of
Chemical Engineering, Imperial College London

Background

Neural Networks are powerful modelling tools – *But...*

- Require lots of data
- Lack uncertainty quantification
- May violate known physical constraints



Methodology

Probabilistic Neural Network

Model parametrises output distribution as a Multivariate Gaussian:

$$p(\mathbf{y} | \mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_P, \boldsymbol{\Sigma}_P); \quad \boldsymbol{\mu}_P, \boldsymbol{\Sigma}_P = \mathbf{f}_\theta(\mathbf{x})$$

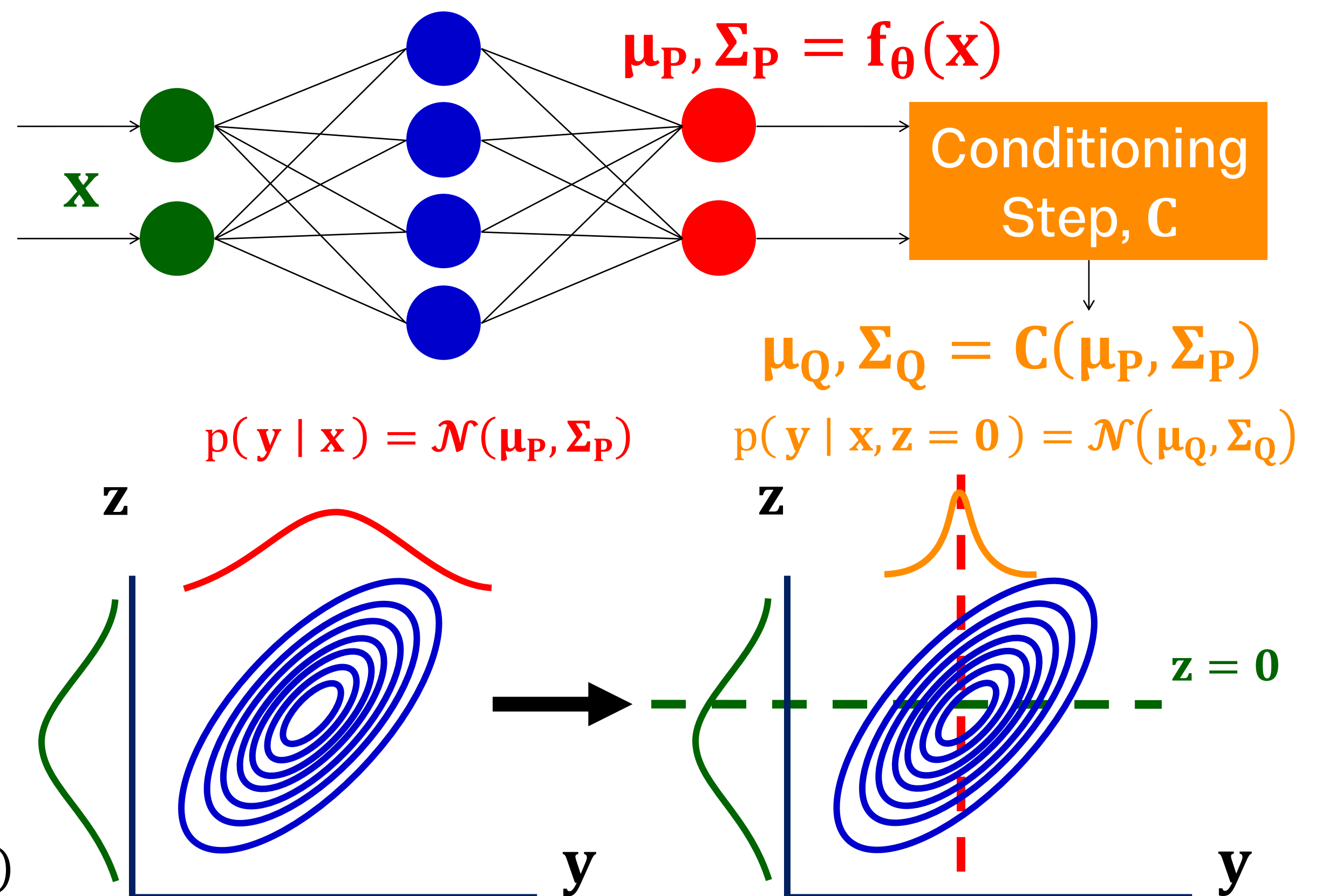
Conditioning on Linear-Gaussian Systems

Defining the residual of an affine constraint:

$$\mathbf{z} = \mathbf{Ax} + \mathbf{By} - \mathbf{b} + \boldsymbol{\epsilon}; \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_\epsilon \mathbf{I})$$

As $\mathbf{z}$ itself is Gaussian, condition the original output distribution on constraint satisfaction:

$$p(\mathbf{y} | \mathbf{x}, \mathbf{z} = \mathbf{0}) = \mathcal{N}(\boldsymbol{\mu}_Q, \boldsymbol{\Sigma}_Q)$$



The conditioning projects the model output onto the constraint manifold in closed form – always satisfied by design:

$$\begin{aligned} \boldsymbol{\mu}_Q &= \boldsymbol{\mu}_P + \boldsymbol{\Sigma}_P \mathbf{B}^\top (\mathbf{B} \boldsymbol{\Sigma}_P \mathbf{B}^\top + \boldsymbol{\sigma}_\epsilon \mathbf{I})^{-1} (\mathbf{b} - \mathbf{Ax} - \mathbf{B} \boldsymbol{\mu}_P) \\ \boldsymbol{\Sigma}_Q &= \boldsymbol{\Sigma}_P - \boldsymbol{\Sigma}_P \mathbf{B}^\top (\mathbf{B} \boldsymbol{\Sigma}_P \mathbf{B}^\top + \boldsymbol{\sigma}_\epsilon \mathbf{I})^{-1} \mathbf{B} \boldsymbol{\Sigma}_P \end{aligned}$$

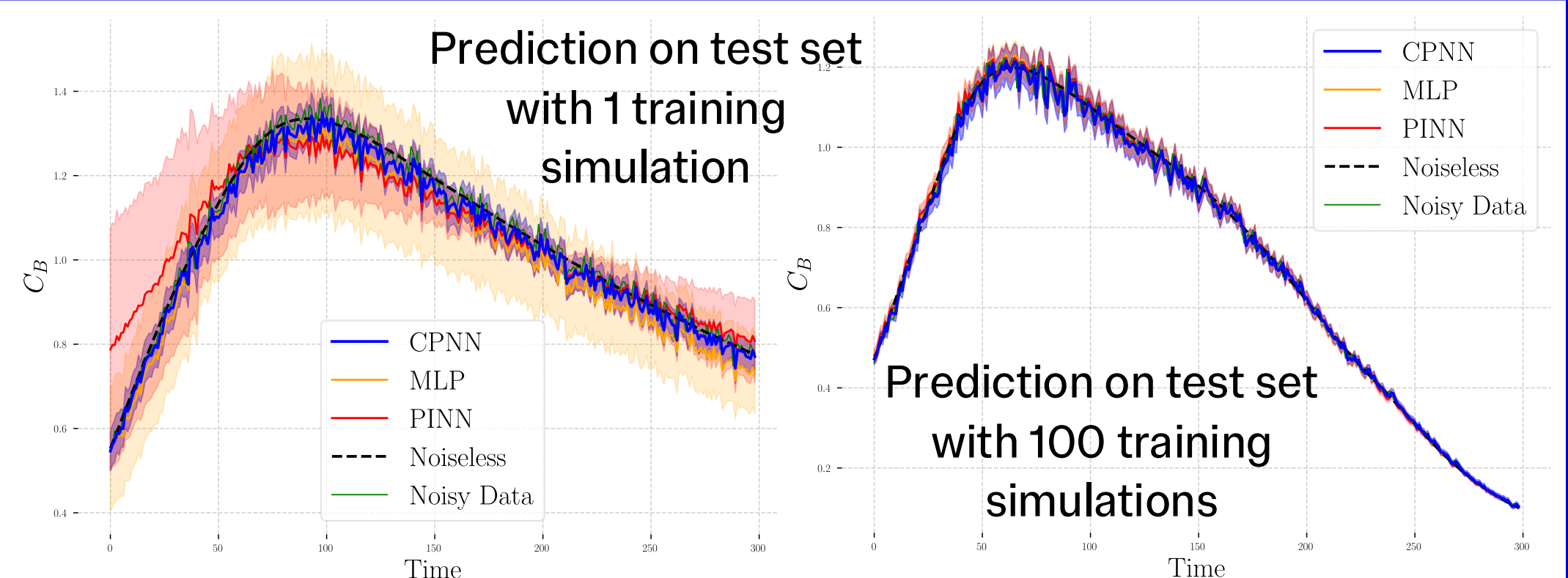
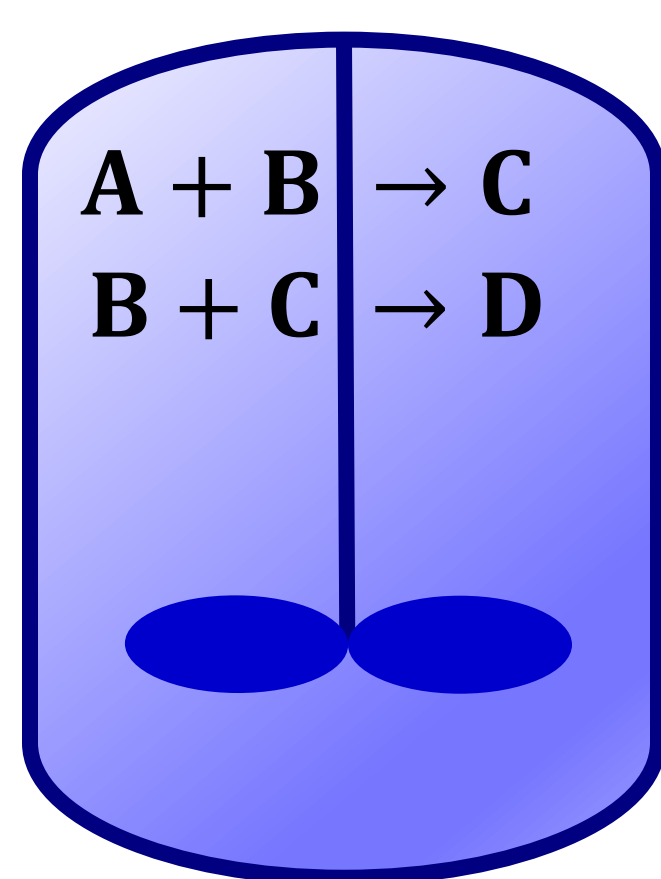
Then train on the Conditional Gaussian Negative-Log-Likelihood

$$\mathcal{L}(\theta) = \frac{1}{2} \left[(\mathbf{y} - \boldsymbol{\mu}_Q)^\top \boldsymbol{\Sigma}_Q^{-1} (\mathbf{y} - \boldsymbol{\mu}_Q) + \log |\boldsymbol{\Sigma}_Q| \right]$$

Results

Case Study

- Modelling of a dynamic batch reactor, with embedded mass conservation
- Investigated performance on noisy data, with varying amounts of data



Key Findings

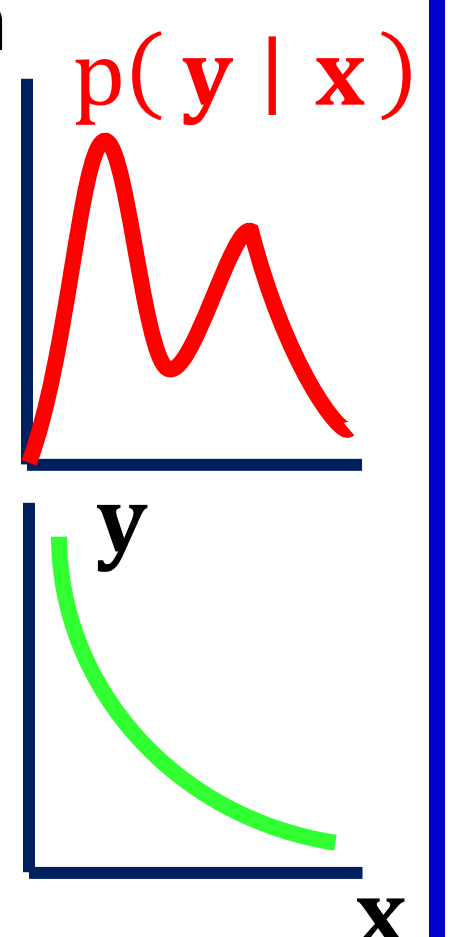
- Data was main differentiator between performance, with impressive one-shot performance compared to baselines

← Increasing CPNN Advantage Increasing Data →

- Also had up to **2.5x** speed increase when training CPNN compared to baselines – altering the loss landscape
- However - *all models* overestimated uncertainty prediction

Outlook

- Improve uncertainty calibration
- Implement with a non-Gaussian uncertainty profiles
- Enforce non-linear constraints
- Large scale system modelling and warm start training
- Alternative frameworks, Neural ODEs - constraints on the gradients



[1] Hao Chen, Gonzalo E. Constante Flores, and Can Li. Physics-informed neural networks with hard linear equality constraints. Computers & Chemical Engineering, 189:108764, October 2024

[2] D.A. Nix and A.S. Weigend. Estimating the mean and variance of the target probability distribution. In Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94), pages 55–60 vol.1, June 1994.

[3] Derek Hansen, Danielle C. Maddix, Shima Alizadeh, Gaurav Gupta, and Michael W. Mahoney. Learning Physical Models that Can Respect Conservation Laws. In Proceedings of the 40th International Conference on Machine Learning, pages 12469–12510. PMLR, July 2023. ISSN: 2640-3498