Detecting Failure Modes in Image Reconstructions with Interval Neural Network Uncertainty

Luis Oala[1*], Cosmas Heiß[2*], Jan Macdonald[2*], Maximilian März[2*], Gitta Kutyniok[2] and Wojciech Samek[1]

[1]Fraunhofer HHI, [2]Technical University of Berlin
[*]Equal contribution

ICML 2020
Workshop on Uncertainty & Robustness in Deep Learning
July 17, 2020

Setup

Problem Setting

- Data set $\{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^m$ consisting of inputs $\boldsymbol{x}_i \in \mathcal{X}$ and targets $\boldsymbol{y}_i \in \mathcal{Y}$
- Inverse problem: $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{y} + \boldsymbol{\eta}$ where $\boldsymbol{y} \in \mathbb{R}^n$ is the unknown signal of interest, $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ denotes the forward operator representing a physical measurement process, and $\boldsymbol{\eta} \in \mathbb{R}^m$ is modelling noise in the measurements
- Prediction function $\boldsymbol{\Phi}\colon \mathcal{X} \to \mathcal{Y}$

Goal

A high-resolution alarm system in output-space that is *post hoc*, *efficient*, *easy to interpret* and *effective*.

Method: Interval Neural Network Uncertainty I

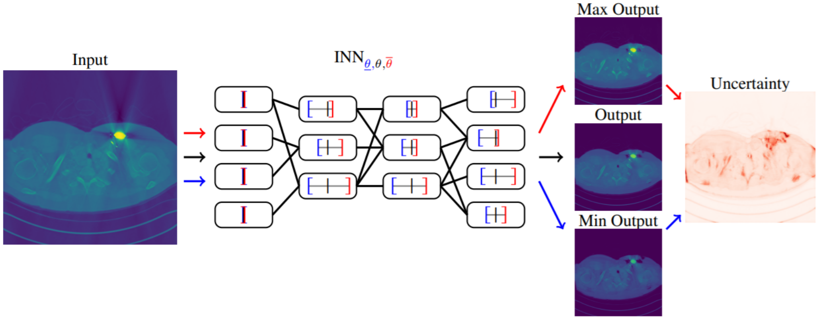

Figure 1: Schematic INN overview

Method: Interval Neural Network Uncertainty II

For positive values of $[\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}]^{(l)}$, we can express the interval propagation as

$$\overline{\boldsymbol{x}}^{(l+1)} = \varrho\left(\min\left\{\overline{\boldsymbol{W}}^{(l)}, 0\right\}\underline{\boldsymbol{x}}^{(l)} + \max\left\{\overline{\boldsymbol{W}}^{(l)}, 0\right\}\overline{\boldsymbol{x}}^{(l)} + \overline{\boldsymbol{b}}^{(l)}\right)$$

$$\underline{\boldsymbol{x}}^{(l+1)} = \varrho\left(\max\left\{\underline{\boldsymbol{W}}^{(l)}, 0\right\}\underline{\boldsymbol{x}}^{(l)} + \min\left\{\underline{\boldsymbol{W}}^{(l)}, 0\right\}\overline{\boldsymbol{x}}^{(l)} + \underline{\boldsymbol{b}}^{(l)}\right)$$

These formulas can then be used in existing deep learning frameworks to optimize the bounds of the interval parameters via backpropagation and the following cost function:

$$\mathcal{L}(\underline{\boldsymbol{\Phi}}, \overline{\boldsymbol{\Phi}}) = \sum_{i=1}^{m} \max\{\boldsymbol{y}_i - \overline{\boldsymbol{\Phi}}(\boldsymbol{x}_i), 0\}^2 + \max\{\underline{\boldsymbol{\Phi}}(\boldsymbol{x}_i) - \boldsymbol{y}_i, 0\}^2$$

$$+\beta \cdot \left(\overline{\boldsymbol{\Phi}}(\boldsymbol{x}_i) - \underline{\boldsymbol{\Phi}}(\boldsymbol{x}_i)\right)$$

Method: Interval Neural Network Uncertainty III

INN Perks

- **Modular**: Plug in a finished prediction function and get uncertainty features on top without retraining
- **Quick**: INNs scale linearly in the number of prediction DNN operations K with a constant factor of 2, in contrast to a factor of $T \geq 10$ for [1]
- **Interpretable**: Interval values and analytic coverage bounds[1] $\mathbb{P}(\underline{\boldsymbol{\Phi}}(\boldsymbol{x}^*) - \lambda\beta < \boldsymbol{y}^* < \overline{\boldsymbol{\Phi}}(\boldsymbol{x}^*) + \lambda\beta \mid \boldsymbol{x}^*) \geq 1 - \frac{1}{\lambda}$
- **Effective**: ?

[1] On the training distribution, see Section 3 of the paper

Experiments I

Failure Modes

- Adversarial Artifact Detection (AdvDetect)
- Atypical Artifact Detection (ArtDetect)
- Error Correlation (EC)

UQ Methods

- Interval Neural Network (INN):
  $\boldsymbol{u}_{\mathrm{INN}}(\widetilde{\boldsymbol{x}}) = \overline{\boldsymbol{\Phi}}(\widetilde{\boldsymbol{x}}) - \underline{\boldsymbol{\Phi}}(\widetilde{\boldsymbol{x}})$
- Monte Carlo dropout (MCDrop)[1, 3]:
  $\boldsymbol{u}_{\mathrm{MCDrop}}(\widetilde{\boldsymbol{x}}) = \frac{1}{T-1}\left(\sum_{t=1}^{T} \boldsymbol{\Phi}_t(\widetilde{\boldsymbol{x}})^2 - \frac{1}{T}\left(\sum_{t=1}^{T} \boldsymbol{\Phi}_t(\widetilde{\boldsymbol{x}})\right)^2\right)$
- Mean and Variance Estimation (ProbOut)[4, 2]:
  $\boldsymbol{u}_{\mathrm{ProbOut}}(\widetilde{\boldsymbol{x}}) = \boldsymbol{\Phi}_{\mathrm{var}}(\widetilde{\boldsymbol{x}})$

Experiments II

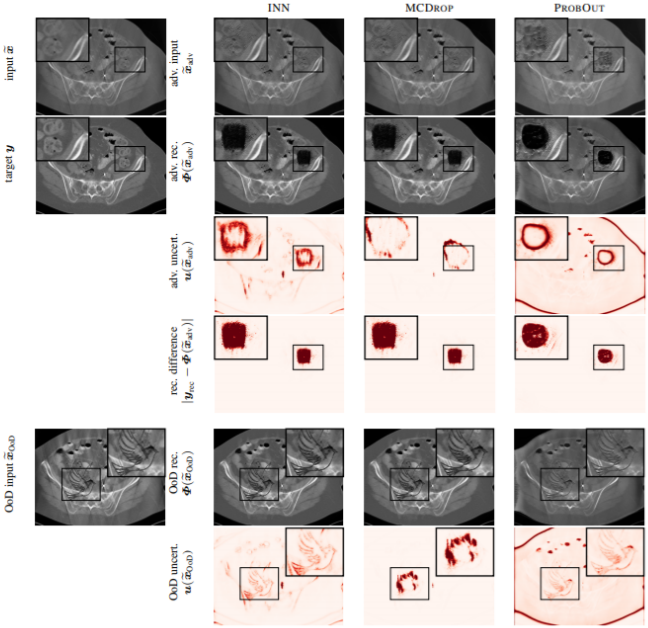

Figure 2: Results of three UQ methods for the AdvDetect and ArtDetect experiments. Plotting windows slightly adjusted for better contrast.

Experiments III

Table 1: Mean test results (± standard deviation) averaged over three experimental runs. Pearson correlation coefficients for the Adversarial Artifact Detection and Atypical Artifact Detection experiments and PWCC with MSE for the EC experiment.

	AdvDetect		ArtDetect		EC	
UQ Method	CT	Denoise	CT	Denoise	PWCC	MSE
INN	$\mathbf{0.56 \pm 0.05}$	0.77 ± 0.008	$\mathbf{0.52 \pm 0.03}$	$\mathbf{0.69 \pm 0.006}$	$\mathbf{2211 \pm 403}$	$7.4 \pm 0.65 \times 10^{-4}$
MCDrop	0.28 ± 0.02	0.20 ± 0.001	0.26 ± 0.01	0.44 ± 0.02	2170 ± 513	$7.4 \pm 0.65 \times 10^{-4}$
ProbOut	0.48 ± 0.12	$\mathbf{0.81 \pm 0.002}$	0.34 ± 0.04	0.44 ± 0.01	190 ± 28	$6.7 \pm 2 \times 10^{-3}$

Musings

- + The advertisements above
- - Dealing with INN activation functions other than ReLU
- - How can we incorporate batch normalization in the INN?
- ? Beyond inverse problems: classification
- ? Deeper probabilistic interpretation of INNs beyond ELBO and the approximate posterior [2]
- ? Application of INNs in DNN compression

[2] See Appendix D

References I

Yarin Gal and Zoubin Ghahramani. "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning". In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, 20–22 Jun 2016, pp. 1050–1059.

Jochen Gast and Stefan Roth. "Lightweight Probabilistic Deep Networks". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), pp. 3369–3378.

References II

Alex Kendall and Yarin Gal. “What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?” In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Long Beach, California, USA: Curran Associates Inc., 2017, pp. 5580–5590. isbn: 9781510860964.

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)*. Vol. 1. June 1994, 55–60 vol.1. doi: 10.1109/ICNN.1994.374138.