Noname manuscript No.
(will be inserted by the editor)

Detecting Failure Modes in Image Reconstructions with Interval Neural Network Uncertainty

Luis Oala* · Cosmas Heiß* · Jan Macdonald* · Maximilian März* · Gitta Kutyniok · Wojciech Samek

Received: date / Accepted: date

Abstract

Purpose The quantitative detection of failure modes is important for making deep neural networks reliable and usable at scale. We consider three examples for common failure modes in image reconstruction and demonstrate the potential of uncertainty quantification as a fine-grained alarm system.

Methods We propose a deterministic, modular and lightweight approach called Interval Neural Network (INN) that produces fast and easy to interpret uncertainty scores for deep neural networks. Importantly, INNs can be constructed *post-hoc* for already trained prediction networks. We compare it against state-of-the-art baseline methods (MCDrop, ProbOut).

Results We demonstrate on controlled, synthetic inverse problems the capacity of INNs to capture uncertainty due to noise as well as directional error information. On a real-world inverse problem with human CT scans we can show that INNs produce uncertainty scores which improve the detection of all considered failure modes compared to the baseline methods.

Conclusion Interval Neural Networks offer a promising tool to expose weaknesses of deep image reconstruction models and ultimately make them more reliable. The fact that they can be applied *post-hoc* to equip already trained deep neural network models with uncertainty scores makes them particularly interesting for deployment.

* Equal contribution

L. Oala · W. Samek
AI Department, Fraunhofer HHI
E-mail: {luis.oala,wojciech.samek}@hhi.fraunhofer.de

C. Heiß · J. Macdonald · M. März
Institut für Mathematik, Technische Universität Berlin
E-mail: cosmas.heiss@gmail.com,{macdonald,maerz}@math.tu-berlin.de

G. Kutyniok
Mathematisches Institut, Ludwig-Maximilians-Universität München
E-mail: kutyniok@math.lmu.de

1 Introduction

The reconstruction of unknown signals from indirect measurements plays an important role in many applications, including medical imaging [14, 2]. Typically, such tasks are modelled as finite-dimensional linear inverse problems

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the 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. Important examples include magnetic resonance imaging and computed tomography, where $\mathbf{A}$ is a subsampled discrete Fourier or Radon transform respectively. Solving the inverse problem (1) requires computing an approximate reconstruction of $\boldsymbol{x}$ from the observed measurements $\boldsymbol{y}$.

Classical reconstruction methods, e.g., based on sparse regularization models, constitute the state-of-the-art for solving (1) in many cases and are backed by theoretical guarantees [8]. Recently, data-driven deep learning methods are increasingly gaining attention and are repeatedly able to outperform traditional solvers in terms of empirical reconstruction performance or speed, see for example [2].

Despite the advantages, the use of deep learning methods in sensitive applications such as clinical diagnosis is still a concern [23], due to questions regarding the reliability and robustness of the obtained reconstructions when compared to traditional approaches [13, 1]. What is more, erroneous artifacts in the reconstructed signals can be hard to detect as they tend to "blend in" well with the rest of the signal.

Various approaches for incorporating uncertainty quantification (UQ) into deep learning have been proposed to address these issues [22, 16, 10, 18]. However, as we demonstrate, existing UQ approaches come with limitations regarding their capacity to detect failure modes or their post-hoc applicability to trained deep learning models.

In this work, we consider a straight-forward approach to solving (1) by employing a neural network to post-process a standard model-based inversion as in [14]. This reconstruction is given by

$$\boldsymbol{x}_{\texttt{rec}} = \left(\boldsymbol{\Phi} \circ \boldsymbol{A}^\dagger\right)(\boldsymbol{y}),$$

where $\boldsymbol{\Phi}\colon \mathbb{R}^n \to \mathbb{R}^n$ is a neural network trained to minimize the loss $\|\boldsymbol{x} - \boldsymbol{\Phi}(\boldsymbol{A}^\dagger(\boldsymbol{y}))\|_2^2$ and $\boldsymbol{A}^\dagger\colon \mathbb{R}^m \to \mathbb{R}^n$ denotes the non-learned model-based inversion (e.g., the filtered back-projection in the case of Radon measurements). We will denote $\boldsymbol{z} = \boldsymbol{A}^\dagger(\boldsymbol{y})$ in the following. Given $\boldsymbol{y}$ or $\boldsymbol{z}$, a UQ method is supposed to extend the predicted reconstruction $\boldsymbol{\Phi}(\boldsymbol{z})$ by a component-wise uncertainty score $\boldsymbol{u}(\boldsymbol{z})$ that provides additional information regarding the reliability of the reconstruction. Therefore, $\boldsymbol{u}(\boldsymbol{z})$ should be correlated with the component-wise error $|\boldsymbol{x} - \boldsymbol{\Phi}(\boldsymbol{z})|$. We evaluate this for three different failure modes [7] that can arise during inference (see Sections 4.2.1 to 4.2.3 for more details):

(i) Errors caused solely by the ill-posedness of (1), which is mostly determined by the strength of measurement noise and the amount of undersampling,
(ii) Errors caused by adversarial perturbations to the network inputs,
(iii) Errors caused by atypical artifacts that have not been seen during the training.

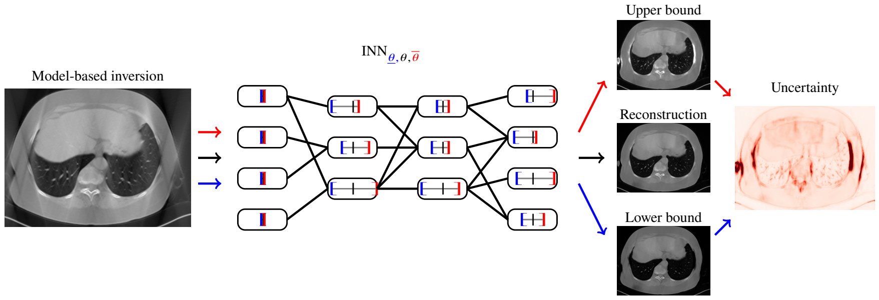

Fig. 1 A schematic overview of the proposed Interval Neural Networks for image reconstruction.

Our main contributions can be summarized as follows: We present a deterministic, modular and fast UQ-method for deep neural networks (DNNs), called Interval Neural Networks (INN). We evaluate INNs for the detection of the three different image reconstruction failure modes and demonstrate that they provide improved results compared to two existing UQ methods.

2 Related Work

Whereas a number of methods from classical statistical learning theory, such as Gaussian processes and approximations thereof [6,19], come with built-in uncertainty estimates, DNNs have been limited in this regard. A surge of efforts to treat neural networks from a variational perspective [3,16] started to change that. In addition, there exist strands of research in deep learning explicitly occupied with the detection of failure modes caused by adversarial and out of distribution (OoD) inputs. These include Maximum Mean Discrepancy, Kernel Density Estimation and other tools, see [5] or the Minimum Covariance Determinant method [26], Support Vector Data Description [28], among others. We refer to [27] for a comprehensive overview. The detection of adversarial and OoD inputs in these works is typically done in the classification setting. We emphasize that image-to-image regression is a fundamentally different task: While classification is inherently discontinuous, image reconstruction addresses a problem that allows for stable solution methods in many cases, e.g. by sparse regularization. Furthermore, we are not interested in a crude, outright rejection of data points in the *input space* but rather seek to obtain fine-grained information about erroneous artifacts in the *output space*. More closely related to our goal is Monte Carlo dropout (MCDrop) [10], and direct variance estimation (ProbOut) [12], where epistemic and aleatoric uncertainty quantification was considered for segmentation and depth-estimation tasks. Hence, we include their approaches as baseline comparison methods, see Section 3.4.

3 Methods

Popular existing UQ frameworks for DNNs place parametric densities, most commonly Gaussian densities, over the DNN parameters or predictions. Instead of using specific parametrized densities, our INN method relies on bounding distributions using intervals. This results in a flexible and modular method that can be applied post-hoc to a given DNN $\boldsymbol{\Phi}$ that has already been trained. A schematic illustration is provided in Fig. 1: the INN is formed by wrapping additional weight and bias intervals around the weights and biases of the underlying prediction DNN. This allows us to equip the DNN $\boldsymbol{\Phi}$ with uncertainty capabilities without the need to modify $\boldsymbol{\Phi}$ itself. After training the INN we obtain prediction intervals that are guaranteed to contain the original prediction of the underlying network and are easy to interpret. They provide exact upper and lower bounds for the range of possible values that the DNN prediction may take when slightly modifying the network parameters within the prescribed weight and bias intervals.

Previously, the capacity of neural networks with interval weights and biases was evaluated for fitting interval valued functions [11]. In contrast to [11] our targets $\boldsymbol{x}_i$ are neither interval-valued nor univariate, leading to a different loss function which allows us to equip trained neural networks with uncertainty capabilities *post-hoc*. For a direct comparison see Equation (3) in Section 3.2 and Equation (18) in [11]. Further, [30,17] explored neural networks implementing interval arithmetic for robust classifications. However, in their setting, the focus is purely on representing the *inputs* or *outputs* as intervals but not the *weights* and *biases*. In contrast, our proposed INNs determine interval bounds for all network parameters with the goal of providing uncertainty scores for the predictions of an underlying DNN.

3.1 Arithmetic of Interval Neural Networks

We will now give a description of those INN mechanisms that deviate from standard DNNs. The forward propagation of a single input z through a DNN is replaced by the forward propagation of a component-wise interval valued input $[\underline{z}, \overline{z}]$ through the INN. This can be expressed similarly to standard feed-forward neural networks but using interval arithmetic instead. For interval valued weight matrices $[\underline{\boldsymbol{W}}, \overline{\boldsymbol{W}}]$ and bias vectors $[\underline{\boldsymbol{b}}, \overline{\boldsymbol{b}}]$ the propagation through the ℓ-th network layer can be expressed as

$$[\underline{z}, \overline{z}]^{(\ell+1)} = \varrho\left(\left[\underline{\boldsymbol{W}}, \overline{\boldsymbol{W}}\right]^{(\ell)} [\underline{z}, \overline{z}]^{(\ell)} + \left[\underline{\boldsymbol{b}}, \overline{\boldsymbol{b}}\right]^{(\ell)}\right). \tag{2}$$

For non-negative $[\underline{z}, \overline{z}]^{(\ell)}$, for example when using a non-negative activation function ϱ such as the ReLU in the previous layer, we can explicitly rewrite (2) as

$$\begin{aligned} \overline{z}^{(\ell+1)} &= \varrho\left(\min\left\{\overline{\boldsymbol{W}}^{(\ell)}, \boldsymbol{0}\right\} \underline{z}^{(\ell)} + \max\left\{\overline{\boldsymbol{W}}^{(\ell)}, \boldsymbol{0}\right\} \overline{z}^{(\ell)} + \overline{\boldsymbol{b}}^{(\ell)}\right), \\ \underline{z}^{(\ell+1)} &= \varrho\left(\max\left\{\underline{\boldsymbol{W}}^{(\ell)}, \boldsymbol{0}\right\} \underline{z}^{(\ell)} + \min\left\{\underline{\boldsymbol{W}}^{(\ell)}, \boldsymbol{0}\right\} \overline{z}^{(\ell)} + \underline{\boldsymbol{b}}^{(\ell)}\right), \end{aligned}$$

where the maximum and minimum are computed component-wise. Similarly, for point intervals $\underline{z}^{(\ell)} = \overline{z}^{(\ell)} =: z^{(\ell)}$, for example as inputs to the first network layer, we can rewrite (2) as

$$\begin{aligned}\overline{z}^{(\ell+1)} &= \varrho\left(\overline{\boldsymbol{W}}^{(\ell)}\max\{z^{(\ell)},\mathbf{0}\} + \underline{\boldsymbol{W}}^{(\ell)}\min\{z^{(\ell)},\mathbf{0}\} + \overline{\boldsymbol{b}}^{(\ell)}\right),\\ \underline{z}^{(\ell+1)} &= \varrho\left(\underline{\boldsymbol{W}}^{(\ell)}\max\{z^{(\ell)},\mathbf{0}\} + \overline{\boldsymbol{W}}^{(\ell)}\min\{z^{(\ell)},\mathbf{0}\} + \underline{\boldsymbol{b}}^{(\ell)}\right),\end{aligned}$$

regardless of whether $z^{(\ell)}$ is non-negative or not. Optimizing the INN parameters requires obtaining the gradients of these operations. This can be achieved using automatic differentiation (backpropagation) in the same way as for standard neural networks.

3.2 Training Interval Neural Networks

Let $\boldsymbol{W}^{(\ell)}$ and $\boldsymbol{b}^{(\ell)}$ be the weights and biases of the underlying prediction network $\boldsymbol{\Phi}$ and let $\overline{\boldsymbol{\Phi}}\colon \mathbb{R}^n \to \mathbb{R}^n$ and $\underline{\boldsymbol{\Phi}}\colon \mathbb{R}^n \to \mathbb{R}^n$ denote the functions mapping a point interval input z to the upper and the lower interval bounds in the output layer of the INN respectively. Given data samples $\{z_i, \boldsymbol{x}_i\}_{i=1}^m$ the INN parameters $[\underline{\boldsymbol{W}}, \overline{\boldsymbol{W}}]^{(\ell)}$ and $[\underline{\boldsymbol{b}}, \overline{\boldsymbol{b}}]^{(\ell)}$ are trained by minimizing the empirical loss

$$\sum_{i=1}^{m} \left\| \max\{\boldsymbol{x}_i - \overline{\boldsymbol{\Phi}}(z_i), \mathbf{0}\}\right\|_2^2 + \left\| \max\{\underline{\boldsymbol{\Phi}}(z_i) - \boldsymbol{x}_i, \mathbf{0}\}\right\|_2^2 + \beta \cdot \left\|\overline{\boldsymbol{\Phi}}(z_i) - \underline{\boldsymbol{\Phi}}(z_i)\right\|_1, \quad (3)$$

subject to the constraints $\underline{\boldsymbol{W}}^{(\ell)} \leq \boldsymbol{W}^{(\ell)} \leq \overline{\boldsymbol{W}}^{(\ell)}$ and $\underline{\boldsymbol{b}}^{(\ell)} \leq \boldsymbol{b}^{(\ell)} \leq \overline{\boldsymbol{b}}^{(\ell)}$ for each layer. This way $\underline{\boldsymbol{\Phi}}(z) \leq \boldsymbol{\Phi}(z) \leq \overline{\boldsymbol{\Phi}}(z)$ is always guaranteed. The first two terms in (3) encourage that the predicted interval $[\underline{\boldsymbol{\Phi}}(z_i), \overline{\boldsymbol{\Phi}}(z_i)]$ should contain the target signal $\boldsymbol{x}_i$, while penalizing each component that lies outside with the squared distance to the nearest interval bound. The second term penalizes the interval size, so that the predicted intervals cannot grow arbitrarily large. While a quadratic penalty of the interval size is also possible and leads to similar theoretical bounds as in (4), we choose to minimize the ℓ_1-norm to make the intervals more outlier inclusive. In addition, the tightness parameter $\beta > 0$ can further tune the outlier-sensitivity of the intervals. This allows for a calibration of the INN uncertainty scores according to an application specific risk-budget. In practice, we found that choosing β similar to the mean absolute error of the underlying prediction network yields a good trade-off between coverage [9] and tightness.

3.3 Properties of Interval Neural Networks

The uncertainty estimate of an INN is given by the width of the prediction interval, i.e., $\boldsymbol{u}(z) = \overline{\boldsymbol{\Phi}}(z) - \underline{\boldsymbol{\Phi}}(z)$. In terms of computational overhead, INNs scale linearly in the cost of evaluating the underlying prediction DNN with a constant factor 2. In contrast, the popular MCDrop [10] scales linearly with a factor T which is proportional to the number of stochastic forward passes and at least $T = 10$ is recommended by the authors, see Section 3.4.

Further, INNs come with theoretical coverage guarantees that can be derived from the Markov inequality: Assuming that the loss (3) is optimized during training to yield an INN with vanishing expected gradient with respect to the data distribution, we obtain

$$\mathbb{P}_{(z,\boldsymbol{x})}\left[\underline{\boldsymbol{\Phi}}(z)_i - \lambda\beta < \boldsymbol{x}_i < \overline{\boldsymbol{\Phi}}(z)_i + \lambda\beta\right] \geq 1 - \frac{1}{\lambda}, \tag{4}$$

for any $\lambda > 0$. In other words, for input and target pair $(z, \boldsymbol{x})$ the probability of any component of the target lying inside the predicted interval enlarged by $\lambda\beta$ is at least $1 - \frac{1}{\lambda}$. As β is usually very small, this ensures a fast decay of the probability of the components of $\boldsymbol{x}$ lying outside the predicted interval bounds. Consequently, a component with a small uncertainty score was correctly reconstructed up to small error with a high probability. Of course, the training distribution needs to be well representative of the true data distribution to extrapolate this property to unseen data.

Finally, the optimization of the loss (3) yields additional information: If the prediction $\boldsymbol{\Phi}(z)$ lies closer to one boundary of the predicted interval, the true target $\boldsymbol{x}$ has a higher probability of lying on the other side of the interval. Consequently, INNs can provide directional uncertainty scores. A quantitative assessment of this capability is given in Fig. 3c+d. We note that it is also possible to explore asymmetric uncertainty estimates in the probabilistic setting, e.g., via exponential family distributions [29] or quantile regression [24]. In contrast to INNs, these methods cannot be applied post-hoc as they require substantial modifications to the underlying prediction network.

3.4 Baseline UQ Methods

In addition to our INN approach we consider two other related and popular UQ baseline methods for comparison. First, Monte Carlo dropout (MCDrop) [10] obtains uncertainty scores as the sample variance of multiple stochastic forward passes of the same input signal. In other words, if $\boldsymbol{\Phi}_1, \ldots, \boldsymbol{\Phi}_T$ are realizations of independent draws of random dropout masks for the same underlying network $\boldsymbol{\Phi}$, the component-wise uncertainty estimate is $\boldsymbol{u}_{\text{MCDrop}}(z) = (\frac{1}{T-1}(\sum_{t=1}^{T}\boldsymbol{\Phi}_t(z)^2 - \frac{1}{T}(\sum_{t=1}^{T}\boldsymbol{\Phi}_t(z))^2))^{1/2}$. Second, a direct variance estimation (ProbOut) was proposed in [22] and later expanded in [12]. Here, the number of output components of the prediction network is doubled and trained to approximate the mean and variance of a Gaussian distribution. The resulting network $\boldsymbol{\Phi}_{\text{ProbOut}} : \mathbb{R}^n \to \mathbb{R}^n \times \mathbb{R}^n, z \mapsto (\boldsymbol{\Phi}_{\text{mean}}(z), \boldsymbol{\Phi}_{\text{var}}(z))$ is trained by minimizing the empirical loss $\sum_i \|(\boldsymbol{y}_i - \boldsymbol{\Phi}_{\text{mean}}(z_i))/\sqrt{\boldsymbol{\Phi}_{\text{var}}(z_i)}\|_2^2 + \|\log \boldsymbol{\Phi}_{\text{var}}(z_i)\|_1$. The component-wise uncertainty score of ProbOut is $\boldsymbol{u}_{\text{ProbOut}}(z) = (\boldsymbol{\Phi}_{\text{var}}(z))^{1/2}$. Note that, in contrast to INN and MCDrop, the ProbOut approach requires the incorporation of UQ already during training. Thus, it cannot be employed as a post-hoc evaluation of an already trained, underlying network $\boldsymbol{\Phi}$. The role of the actual prediction network is taken by $\boldsymbol{\Phi}_{\text{mean}}$.

4 Experiments

We present experiments for two different inverse problems. First, a deconvolution task with 1D signals, and second a tomography task on real-world 2D image signals. Both setups are described in more detail below. The description of all hyperparameters for the experiments is kept brief and we refer to our publicly available code at https://github.com/luisoala/inn for full details.

4.1 Case Study A: Deconvolution of 1D Signals

We start with a synthetic, didactic experiment, inspired by a one-dimensional deconvolution task, to demonstrate the properties of INNs discussed in Section 3.3. For this purpose, we choose $n = m = 512$ and $\boldsymbol{A} = \boldsymbol{D}^\top \boldsymbol{S} \boldsymbol{D}$, where $\boldsymbol{D}$ is a discrete cosine transform (Type I DCT) and $\boldsymbol{S}$ is a diagonal matrix with entries $s_j = \left(\frac{n-j}{n-1}\right)^\nu \in [0, 1]$, that decay with a fixed exponent $\nu = 8$. We draw synthetically generated signals $\boldsymbol{x}$ from a distribution of piecewise constant functions with random jump positions and heights, see Fig. 2. The corresponding measurements $\boldsymbol{y}$ are computed according to (1). We generate a data set consisting of 2000 sample pairs $(\boldsymbol{y}_i, \boldsymbol{x}_i)$, 1600 of which were used for training, 200 for validation and 200 for testing. The underlying prediction network $\boldsymbol{\Phi}$ is a convolutional neural network (consisting of ten convolutional layers and three dropout layers in between) trained to directly map $\boldsymbol{y}$ to $\boldsymbol{x}$, i.e. we use $\boldsymbol{A}^\dagger = \mathrm{Id}$ and thus $z = \boldsymbol{A}^\dagger \boldsymbol{y} = \boldsymbol{y}$ in this experiment. We trained the underlying network $\boldsymbol{\Phi}$ for 100 epochs using Adam [15]. The interval parameters of the INN were subsequently trained for another 100 epochs with $\beta = 2 \cdot 10^{-3}$. For the MCDrop comparison we use $T = 64$ samples. The ProbOut model was trained in the same way as $\boldsymbol{\Phi}$ using 100 Adam epochs. Note that all subsequent evaluations, as well as the plots in Fig. 2 are computed using test samples.

In order to evaluate the UQ methods' abilities to capture uncertainty due to noisy data, we consider additive Gaussian noise $\boldsymbol{\eta} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \cdot \mathrm{Id})$ on the measurements over a range of noise levels σ (Fig. 3a) as well as $\boldsymbol{\eta}_1, \boldsymbol{\eta}_2 \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \cdot \mathrm{Id})$ on the measurements and targets, where (1) is adjusted to $\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x} + \boldsymbol{\eta}_1) + \boldsymbol{\eta}_2$ (Fig. 3b and right column of Fig. 2). In this case, INNs are able to capture the additional uncertainty of $\boldsymbol{\eta}_1$ using the bias parameters of the final network layer. In Fig. 3, it can be observed how in contrast to MCDrop, our method and ProbOut are able to capture independent noise in the data with ProbOut reacting to a lesser degree than the INN. Note also that in Fig. 3 some of the ProbOut evaluations are shifted to the right, indicating a reduced reconstruction performance compared to the other methods.

Finally, we determine the directional information of the INN uncertainty scores as discussed in Section 3.3. For this we define the component-wise *directionality ratio* by $\mathrm{DR}(z) = \max\{\overline{\boldsymbol{\Phi}}(z) - \boldsymbol{\Phi}(z), \boldsymbol{\Phi}(z) - \underline{\boldsymbol{\Phi}}(z)\}/\min\{\overline{\boldsymbol{\Phi}}(z) - \boldsymbol{\Phi}(z), \boldsymbol{\Phi}(z) - \underline{\boldsymbol{\Phi}}(z)\}$, i.e. as the ratio between the larger and smaller part of the interval $[\underline{\boldsymbol{\Phi}}(z), \overline{\boldsymbol{\Phi}}(z)]$ when divided by the prediction $\boldsymbol{\Phi}(z)$. The *directionality accuracy* (DA) is the relative frequency of target components corresponding to a given DR that are contained in the larger interval

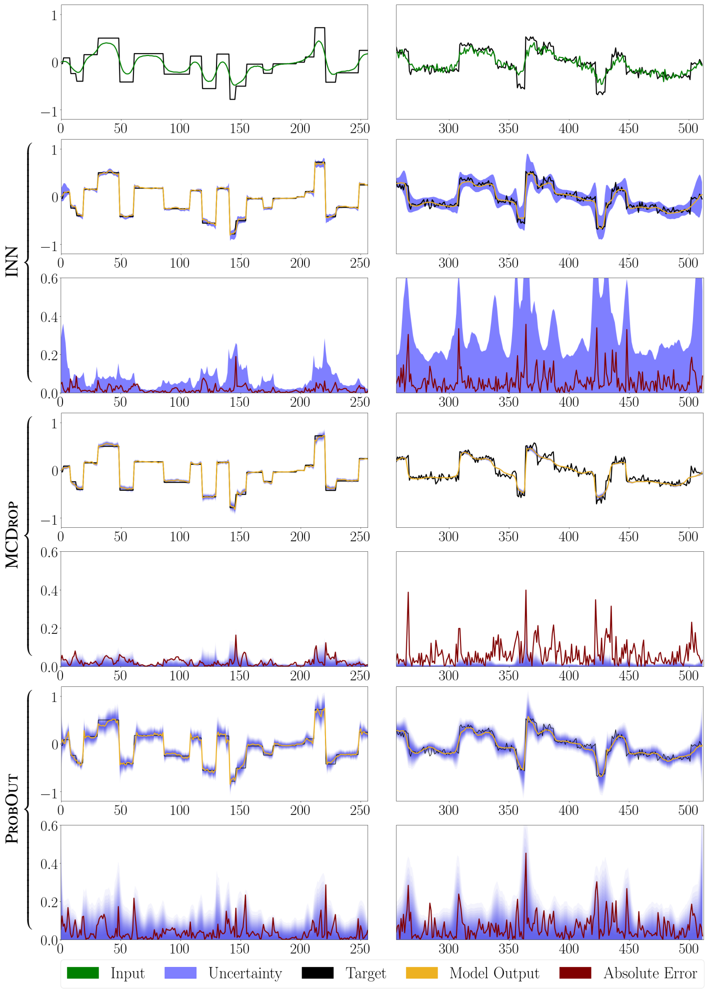

Fig. 2 Results for the deconvolution task for one exemplary signal without noise (left) and with additive Gaussian noise ($\sigma = 0.05$) on both the measurements $\boldsymbol{y}$ and signal $\boldsymbol{x}$ (right). The first row shows inputs $\boldsymbol{z} = \boldsymbol{y}$ and targets $\boldsymbol{x}$. Below the target $\boldsymbol{x}$, prediction $\boldsymbol{\Phi}(\boldsymbol{z})$, and uncertainty score $\boldsymbol{u}(\boldsymbol{z})$ as well as the uncertainty compared to the absolute error $|\boldsymbol{\Phi}(\boldsymbol{z}) - \boldsymbol{x}|$ are shown for the three UQ methods.

part. As displayed in Fig. 3c+d, INNs achieve a DA consistently above 0.5 (chance) indicating that the interval uncertainty scores contain directional information.

4.2 Case Study B: Limited Angle Computed Tomography

Next, we consider a 2D computed tomography (CT) task on real-world data in order to evaluate the detection capabilities of the UQ methods with respect to the three failure modes (i)–(iii). More precisely, we consider limited angle CT, which has applications in dental tomography, breast tomosynthesis or electron tomography. For this, $\boldsymbol{A}$ is a

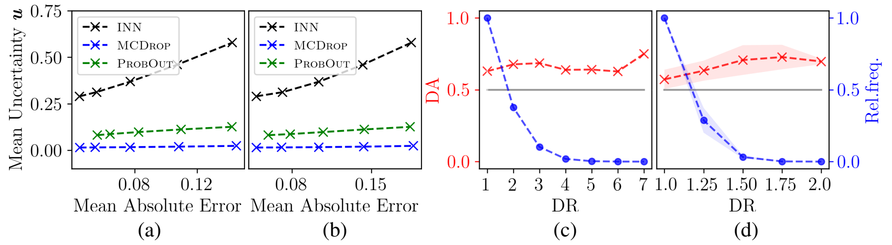

Fig. 3 (a) Mean uncertainty of the three UQ methods for varying levels σ of additive Gaussian on the measurements $\boldsymbol{y}$ for the deconvolution task. (b) Corresponding results for additive noise on both the measurements $\boldsymbol{y}$ and signals $\boldsymbol{x}$. (c) Illustration of the directional information contained in the INN output intervals for the deconvolution task. The additional right axis (in blue) displays the relative frequency of signal components for each directionality ratio. (d) Corresponding results for the CT task. The mean and standard deviation across three independent complete experimental runs are shown.

subsampled discrete Radon transform with subsampling corresponding to a moderate missing wedge of 30°. Limited angle measurements are simulated according to (1) and the non-learned inversion $\boldsymbol{A}^{\dagger}$ is based on the filtered backprojection algorithm (FBP) [21]. The underlying prediction network is a U-Net [25] variant. Our experiments are based on a data set consisting of 512×512 human CT scans from the AAPM Low Dose CT Grand Challenge data [20][1]. In total, it contains 2580 full-dose images with a slice thickness of 3mm from 10 patients. Eight of these ten patients were used for training (2036 samples), one for validation (214 samples) and one for testing (330 samples). We trained the underlying network $\boldsymbol{\Phi}$ for 400 epochs using Adam [15]. The interval parameters of the INN were subsequently trained for another 15 epochs with $\beta = 10^{-4}$. We limited the interval training to the last twelve layers. For the MCDrop comparison we use $T = 128$ samples. The ProbOut model was trained in the same way as $\boldsymbol{\Phi}$ using 400 Adam epochs.

4.2.1 Experiment B (i): General Prediction Error Detection

First, we evaluate how helpful UQ scores are for estimating the prediction error caused by the ill-posedness of the challenging CT task, see Fig. 4. The wedge of missing angles in the measurements results in reconstruction artifacts especially at vertical edges in the images. In order to best visualize these geometric effects of the very structured null-space of the limited angle CT forward operator, we do not add noise in this experiment. INNs are clearly able to reveal the reconstruction uncertainty along the "missing edges". For a more quantitative comparison of the UQ methods, we use the *performance weighted correlation coefficient* $\mathrm{PWCC}(z,\boldsymbol{x}) = \mathrm{corr}(|\boldsymbol{\Phi}(z)-\boldsymbol{x}|,\boldsymbol{u}(z))/\|\boldsymbol{\Phi}(z)-\boldsymbol{x}\|_2^2$ between the uncertainty score $\boldsymbol{u}$ and the absolute prediction error. Performance weighting (normalizing by the mean squared error of the prediction) is necessary to discourage rewards for poor prediction

[1] See: https://www.aapm.org/GrandChallenge/LowDoseCT/. We would like to thank Dr. Cynthia McCollough, the Mayo Clinic, and the American Association of Physicists in Medicine as well as the grants EB017095 and EB017185 from the National Institute of Biomedical Imaging and Bioengineering for providing the AAPM data.

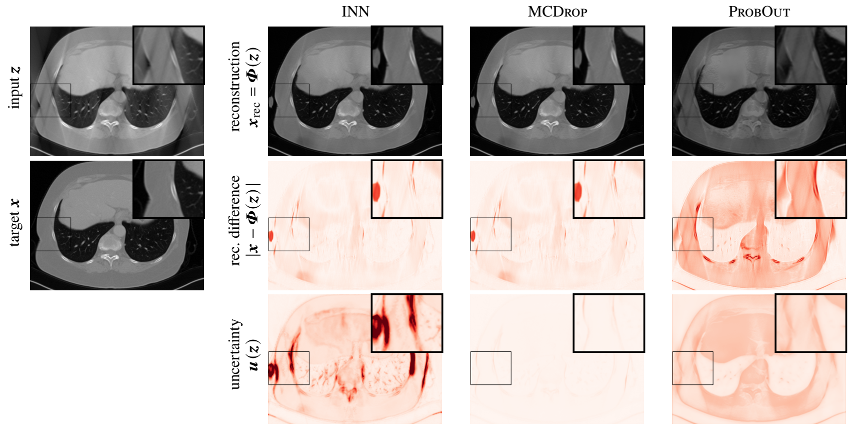
INN
MCDrop
ProbOut
input z
target x
reconstruction $x_{\text{rec}} = \Phi(z)$
rec. difference $|x - \Phi(z)|$
uncertainty $u(z)$

Fig. 4 Results of three UQ methods for the Error Detection experiment for one exemplary data sample of the limited angle CT task. The plotting windows are equally adjusted for better contrast.

models with high uncertainties everywhere. The average results over the test set for three independent complete experimental runs are summarized in Table 1. Both INNs and MCDrop are able to detect prediction errors, with INNs achieving slightly higher correlations. In Fig. 3d the directional accuracy of the INN is illustrated analogously to the corresponding experiment in Section 4.1. Again it is consistently above 0.5 (chance).

4.2.2 Experiment B (ii): Adversarial Artifact Detection

Second, we assess the capacity of UQ methods to capture artifacts in the output that were caused by adversarial perturbations. To that end, we create perturbed inputs for each input sample z in the test set by employing the box-constrained L-BFGS algorithm [4] to minimize $\|\boldsymbol{\Phi}(z_{\text{adv}}) - \boldsymbol{x}_{\text{adv. tar.}}\|_2^2$ subject to $z_{\text{adv}} \in [0,1]^n$. The adversarial targets $\boldsymbol{x}_{\text{adv. tar.}}$ are created by subtracting 1.5 times its mean value from $\boldsymbol{x}_{\text{rec}}$ within a random 50×50 square, leading to clearly visible artifacts in the corresponding reconstructions; see Fig. 5. It is arguable, whether the technical aspects of such an adversarial perturbation (i.e., attacking subsequently to a model-based inversion) is a realistic scenario in the context of inverse problems. However, for our purposes, such a simple setup (see also [13]) is sufficient. We refer to [1], where adversarial noise is mapped to the measurement domain. In order to assess the detection capacity for this failure mode, the different UQ schemes are then used to produce uncertainty heatmaps for the generated adversarial inputs. A quantitative evaluation is carried out by computing the mean Pearson correlation coefficient between the pixel-wise change in the uncertainty heatmaps $|\boldsymbol{u}(z) - \boldsymbol{u}(z_{\text{adv}})|$ and the change of reconstructions $|\boldsymbol{x}_{\text{rec}} - \boldsymbol{\Phi}(z_{\text{adv}})|$. The results are summarized in Table 1 and illustrated in Fig. 5. We observe that both INN and ProbOut are able to detect the image region of adversarial perturbations, with INN achieving the highest correlation. This shows that both methods are able to visually highlight the effect that visually almost imperceptible input perturbations can have on the reconstructions.

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

UQ Method	AdvDetect	ArtDetect	ErrDetect	
			PWCC	MSE
INN	**0.56 ± 0.05**	**0.52 ± 0.03**	**2211 ± 403**	$7.4 \pm 0.65 \times 10^{-4}$
MCDrop	0.28 ± 0.02	0.26 ± 0.01	2170 ± 513	$7.4 \pm 0.65 \times 10^{-4}$
ProbOut	0.48 ± 0.12	0.34 ± 0.04	190 ± 28	$6.7 \pm 2 \times 10^{-3}$

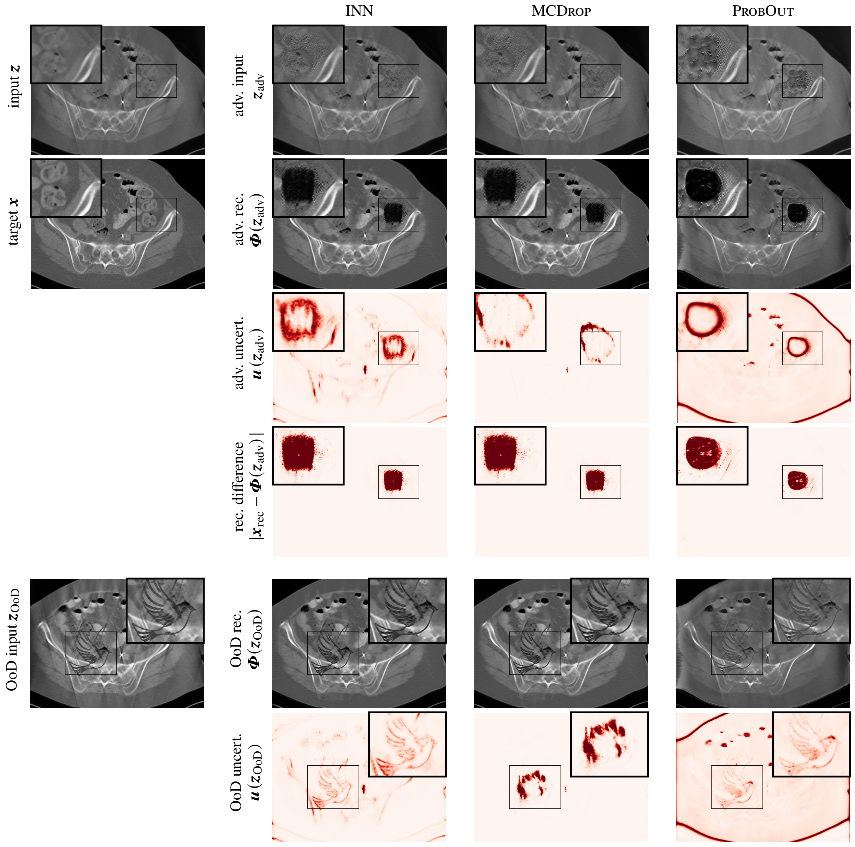
INN
MCDrop
ProbOut
input z
target x
adv. input z_{adv}
adv. rec. $\Phi(z_{\mathrm{adv}})$
adv. uncert. $u(z_{\mathrm{adv}})$
rec. difference $|x_{\mathrm{rec}} - \Phi(z_{\mathrm{adv}})|$
OoD input z_{OoD}
OoD rec. $\Phi(z_{\mathrm{OoD}})$
OoD uncert. $u(z_{\mathrm{OoD}})$

Fig. 5 Results of three UQ methods for the AdvDetect and ArtDetect experiments for one exemplary data sample of the limited angle CT task. The plotting windows are equally adjusted for better contrast.

4.2.3 Experiment B (iii): Atypical Artifact Detection

The third experiment is designed analogous to the setup described by [1], i.e., an atypical artifact, which was not present in the training data, is randomly placed in the input to produce z_{OoD}. More precisely, the silhouette of a peace dove is inserted in each image of the test set; see Fig. 5. The simulation of the measurements and model-based inversions is carried out as before. A quantitative evaluation is carried out by computing the mean Pearson correlation coefficient between the change in the uncertainty heatmaps $|u(z) - u(z_{\mathrm{OoD}})|$ and a binary mask marking the region of change in the inputs. This evaluation isolates the uncertainty caused by atypical artifacts and allows us to verify in a controlled manner how the uncertainty scores of each UQ method react to the artifacts. During deployment such controlled isolation is not possible. Instead, the joint uncertainty heatmaps $\boldsymbol{u}(z_{\mathrm{OoD}})$ will also capture other sources of uncertainty, thus providing a more comprehensive alarm system. The results are summarized in Table 1 and illustrated in Fig. 5. All three UQ methods are correlated with the input change, however INN again achieves the highest correlation. This shows that UQ in general, and INNs in particular, can serve as a warning system for inputs containing atypical features that might otherwise lead to unnoticed and possibly erroneous reconstruction artifacts.

5 Conclusion

We introduced INNs as a deterministic, post-hoc and fast approach for computing upper and lower bounds and subsequently uncertainty maps for pre-trained neural networks. We demonstrated that UQ in general and INNs in particular can be used to provide a fine-grained detection of failure modes of image reconstruction DNNs. INNs are able to capture uncertainty due to noise and can be used to obtain directional information. They perform well as an alarms system for errors due ill-posedness, adversarial noise and atypical artifacts and thus offer a promising tool to expose the weaknesses of deep image reconstruction models.

Declarations

Funding: J.M. acknowledges support by DFG-RTG 2260 BIOQIC. M.M. acknowledges support by DFG-SPP 1798 Grants KU 1446/21 and KU 1446/23. G.K. is grateful to MATH+ – BMRC Project EF1x1 for financial support. W.S. acknowledges support by BMBF/BIFOLD (ref. 01IS18025A and ref 01IS18037I). **Conflict of interest:** L.O. co-chairs the DAISAM working group at the ITU/WHO Focus Group AI4H. **Ethical approval:** This article does not contain any studies with human participants or animals performed by any of the authors. **Informed consent:** No new patient data was acquired as part of this work, public data was used from https://www.aapm.org/GrandChallenge/LowDoseCT/.

References

1. Antun, V., Renna, F., Poon, C., Adcock, B., Hansen, A.C.: On instabilities of deep learning in image reconstruction and the potential costs of AI. Proc. Natl. Acad. Sci. **117**(48), 30088–30095 (2020)
2. Arridge, S., Maass, P., Öktem, O., Schönlieb, C.B.: Solving inverse problems using data-driven models. Acta Numer. **28**, 1–174 (2019)
3. Barber, D., Bishop, C.: Ensemble learning in bayesian neural networks. In: Generalization in Neural Networks and Machine Learning, pp. 215–237. Springer Verlag (1998)
4. Byrd, R.H., Lu, P., Nocedal, J., Zhu, C.: A limited memory algorithm for bound constrained optimization. SIAM J. Sci. Comput. **16**(5), 1190–1208 (1995)
5. Carlini, N., Wagner, D.: Adversarial examples are not easily detected: Bypassing ten detection methods. In: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, p. 3–14 (2017)
6. Denker, J.S., Schwartz, D.B., Wittner, B.S., Solla, S.A., Howard, R.E., Jackel, L.D., Hopfield, J.J.: Large Automatic Learning, Rule Extraction, and Generalization. Complex Syst. **1** (1987)
7. Dietterich, T.G.: Robust artificial intelligence and robust human organizations. Front. Comput. Sci. **13**(1), 1–3 (2019)

8. Foucart, S., Rauhut, H.: A Mathematical Introduction to Compressive Sensing. Applied and Numerical Harmonic Analysis. Birkhäuser (2013)
9. Foygel Barber, R., Candès, E.J., Ramdas, A., Tibshirani, R.J.: The limits of distribution-free conditional predictive inference. Information and Inference: A Journal of the IMA **10**(2), 455–482 (2020). DOI 10.1093/imaiai/iaaa017. URL https://doi.org/10.1093/imaiai/iaaa017
10. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: M.F. Balcan, K.Q. Weinberger (eds.) Proceedings of The 33rd International Conference on Machine Learning, *Proceedings of Machine Learning Research*, vol. 48, pp. 1050–1059. PMLR, New York, New York, USA (2016)
11. Garczarczyk, Z.: Interval neural networks. In: 2000 IEEE International Symposium on Circuits and Systems. Emerging Technologies for the 21st Century. Proceedings (IEEE Cat No.00CH36353), vol. 3, pp. 567–570. Presses Polytech. Univ. Romandes, Geneva, Switzerland (2000)
12. Gast, J., Roth, S.: Lightweight Probabilistic Deep Networks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition pp. 3369–3378 (2018)
13. Huang, Y., Würfl, T., Breininger, K., Liu, L., Lauritsch, G., Maier, A.: Some investigations on robustness of deep learning in limited angle tomography. In: A.F. Frangi, J.A. Schnabel, C. Davatzikos, C. Alberola-López, G. Fichtinger (eds.) Medical Image Computing and Computer Assisted Intervention – MICCAI 2018, pp. 145–153. Springer International Publishing, Cham (2018)
14. Jin, K.H., McCann, M.T., Froustey, E., Unser, M.: Deep Convolutional Neural Network for Inverse Problems in Imaging. IEEE Trans. Image Process. **26**, 4509–4522 (2017)
15. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Y. Bengio, Y. LeCun (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015)
16. Kingma, D.P., Salimans, T., Welling, M.: Variational dropout and the local reparameterization trick. In: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2, NIPS'15, p. 2575–2583. MIT Press, Cambridge, MA, USA (2015)
17. Kowalski, P.A., Kulczycki, P.: Interval probabilistic neural network. Neural. Comput. Appl. **28**(4), 817–834 (2017)
18. Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. In: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (eds.) Advances in Neural Information Processing Systems, vol. 30, pp. 6402–6413. Curran Associates, Inc. (2017)
19. MacKay, D.J.C.: Bayesian methods for adaptive models. Phd, California Institute of Technology (1992)
20. McCollough, C.: Tu-fg-207a-04: Overview of the low dose ct grand challenge. Med. Phys. **43**(6 Part 35), 3759–3760 (2016)
21. Natterer, F.: The Mathematics of Computerized Tomography. SIAM (2001)
22. Nix, D.A., Weigend, A.S.: Estimating the mean and variance of the target probability distribution. In: Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94), vol. 1, pp. 55–60 vol.1 (1994). DOI 10.1109/ICNN.1994.374138
23. Oala, L., Fehr, J., Gilli, L., Balachandran, P., Leite, A.W., Calderon-Ramirez, S., Li, D.X., Nobis, G., Alvarado, E.A.M.n., Jaramillo-Gutierrez, G., Matek, C., Shroff, A., Kherif, F., Sanguinetti, B., Wiegand, T.: Ml4h auditing: From paper to practice. In: Proceedings of the Machine Learning for Health NeurIPS Workshop, *Proceedings of Machine Learning Research*, vol. 136, pp. 280–317. PMLR (2020). URL http://proceedings.mlr.press/v136/oala20a.html
24. Rodrigues, F., Pereira, F.C.: Beyond expectation: deep joint mean and quantile regression for spatiotemporal problems. IEEE Trans. Neural Netw. Learn. Syst. (2020)
25. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: N. Navab, J. Hornegger, W.M. Wells, A.F. Frangi (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Lecture Notes in Computer Science, pp. 234–241. Springer International Publishing (2015)
26. Rousseeuw, P.J.: Least median of squares regression. J. Am. Stat. Assoc. **79**(388), 871–880 (1984)
27. Ruff, L., Kauffmann, J.R., Vandermeulen, R.A., Montavon, G., Samek, W., Kloft, M., Dietterich, T.G., Müller, K.R.: A unifying review of deep and shallow anomaly detection. Proceedings of the IEEE pp. 1–40 (2021). DOI 10.1109/JPROC.2021.3052449
28. Tax, D.M.J., Duin, R.P.W.: Support vector data description. Mach. Learn. **54**(1), 45–66 (2004)
29. Wang, H., Xingjian, S., Yeung, D.Y.: Natural-parameter networks: A class of probabilistic neural networks. In: Advances in Neural Information Processing Systems, pp. 118–126 (2016)
30. Yang, D., Wu, W.: A smoothing interval neural network. Discrete Dyn. Nat. Soc. **2012** (2012)