Citation for published version:
Mukherjee, S, Hauptmann, A, Öktem, O, Pereyra, M & Schönlieb, C-B 2022 'Learned reconstruction methods with convergence guarantees'.

Publication date:
2022

Link to publication

University of Bath

Alternative formats

If you require this document in an alternative format, please contact:
openaccess@bath.ac.uk

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 11. Nov. 2022

arXiv:2206.05431v3 [cs.CV] 14 Sep 2022

Learned reconstruction methods with convergence guarantees

Subhadip Mukherjee[*1], Andreas Hauptmann[*2,3], Ozan Öktem[4], Marcelo Pereyra[5], and Carola-Bibiane Schönlieb[1]

[1]University of Cambridge, UK; [2]University of Oulu, Finland; [3]University College London, UK; [3]KTH–Royal Institute of Technology, Sweden; [4]Maxwell Institute for Mathematical Sciences & Heriot-Watt University, UK; [*]Equal contribution

E-mails:andreas.hauptmann@oulu.fi, sm2467@cam.ac.uk, ozan@kth.se, m.pereyra@hw.ac.uk, cbs31@cam.ac.uk

Abstract

In recent years, deep learning has achieved remarkable empirical success for image reconstruction. This has catalyzed an ongoing quest for precise characterization of correctness and reliability of data-driven methods in critical use-cases, for instance in medical imaging. Notwithstanding the excellent performance and efficacy of deep learning-based methods, concerns have been raised regarding their stability, or lack thereof, with serious practical implications. Significant advances have been made in recent years to unravel the inner workings of data-driven image recovery methods, challenging their widely perceived black-box nature. In this article, we will specify relevant notions of convergence for data-driven image reconstruction, which will form the basis of a survey of learned methods with mathematically rigorous reconstruction guarantees. An example that is highlighted is the role of input-convex neural networks (ICNNs), offering the possibility to combine the power of deep learning with classical convex regularization theory for devising methods that are provably convergent.

This survey article is aimed at both methodological researchers seeking to advance the frontiers of our understanding of data-driven image reconstruction methods as well as practitioners, by providing an accessible description of useful convergence concepts and by placing some of the existing empirical practices on a solid mathematical foundation.

Index Terms

Inverse problems, data-driven regularization, convexity, convergence guarantees, Bayesian methods.

I. Introduction

Image reconstruction problems are virtually ubiquitous in scientific and engineering applications; ranging from astronomy to clinical diagnosis, from tomographic imaging in 3D electron microscopy to X-ray crystallography. In such problems, an image of interest needs to be recovered from its incomplete and noisy observation. The data generation process is governed by an underlying physical process, precise knowledge of which is a prerequisite to formulate an accurate image reconstruction problem.

It is of paramount importance, especially for critical applications such as medical imaging, to obtain image reconstructions with reliable content as they contain critical structural information about the object of interest. Moreover, in many applications of computational imaging, image reconstruction is used as a tool for scientific discovery without any ground-truth being available. Therefore, one needs to be able to rely on the correctness of the reconstruction result. Correctness and reliability of reconstruction algorithms have been traditionally provided in terms of convergence guarantees, and more specifically, in the framework of model-based variational regularization [1], [2]. Mathematically, image reconstruction is an example of an inverse problem, where research is primarily concerned with the development and analysis of mathematical theory and algorithms in a fairly general setting. Such guarantees also offer a principled framework for practitioners to control the reconstruction process.

The emergence of deep learning and the availability of high-quality training data and computing capabilities have considerably transformed the research landscape of image reconstruction [3], [4]. Image quality has improved to a considerable extent as compared to the classical model-based techniques, both qualitatively and quantitatively, driven by task-specific image data sets and sophisticated machine learning algorithms. Some concerns have been voiced nevertheless, pointing out the lack of adequate comprehension of what happens within the learned reconstruction methods, putting forward the argument that a more thorough understanding is needed for reliable utilization of these techniques. Consequently, in parallel with the ongoing enterprise of designing more efficient and better-performing data-driven reconstruction approaches, researchers have begun to investigate theoretical properties of these methods for image reconstruction.

A specific question is whether one can provide reconstruction guarantees and convergence results for deep learning-based methods. This question is important to theoreticians and practitioners alike, since successful answers would place empirically well-performing methods, often based on heuristics, on a rigorous theoretical foundation. Nevertheless, the notion of *theoretical guarantees* is imprecise and can have different meanings depending on the specific context.

This survey attempts to define different notions of convergence within the realm of image reconstruction, elaborate on their practical implications, and present an overview of recent deep learning-based image reconstruction methods that fit into the different notions of convergence. We argue that many of the recently proposed deep learning-based approaches come with more theoretical backing than it is often communicated and are, in fact, not as much a black-box as they are generally referred to be. In particular, identifying the connections between model- and data-driven methods will help unite the two seemingly disparate paradigms and broaden our knowledge of this exciting new line of research.

The article is organized as follows. In Sec. II, we provide the mathematical preliminaries for inverse problems and explain different training strategies for data-driven image reconstruction. This section facilitates precise characterization of different convergence notions, which appears in Sec. III, in a rigorous yet accessible manner. The ideas of convergence considered in this article are primarily derived from the classical regularization, convex analysis, and statistics literature. Sections IV and V provide a review of recent notable data-driven methods that come with convergence guarantees introduced in the preceding section, in the classical functional-analytic and Bayesian settings, respectively. Our survey is not exhaustive by any means, in that it excludes approaches that are based on heuristics and are not provably convergent (except for a few pioneering methods that inspired new lines of research). It is also worth emphasizing that the methods chosen for review are not necessarily the best-performing methods empirically, but they are more interpretable in the classical sense and hence more transparent as compared to competing techniques with possibly superior numerical performance. Finally, we present a summary and make some concluding remarks in Sec. VI.

II. Mathematical foundations

In order to characterize what convergence and reconstruction guarantees mean for an image reconstruction problem, we need to mathematically formulate the reconstruction task.

In the traditional deterministic *functional analytical* setting, this is viewed as solving an operator equation. More precisely, one seeks to find $x^* \in \mathbb{X}$, which is unknown but deterministic, from measured data $y \in \mathbb{Y}$ under the measurement model $y = \mathcal{A}x^* + e$, where $e \in \mathbb{Y}$ denotes observation error. Here, $\mathcal{A}\colon \mathbb{X} \to \mathbb{Y}$ (*forward operator*) models how an image gives rise to data in the absence of observation error and is typically derived from a careful modeling of the involved physics. In what follows, we focus on the linear setting where $\mathcal{A}$ is a linear operator that can be represented by a matrix for discretized images. As we discuss convergence results for the linear setting, we will comment on extensions to the nonlinear case wherever appropriate.

The spaces $\mathbb{X}$ and $\mathbb{Y}$ can be fairly general and potentially infinite-dimensional function spaces. However for the purpose of our exposition, it suffices to consider them as finite-dimensional vector spaces endowed with an inner-product and norm (in particular, subsets of Euclidean spaces). Given the forward model, an estimate of the desired image x^* is obtained by formulating a *reconstruction method*, represented by a mapping $\mathcal{R}\colon \mathbb{Y} \to \mathbb{X}$. In the functional analytic setting, this corresponds to a regularized inverse of $\mathcal{A}$.

Bayesian inversion extends the above by modeling the true (unknown) image $x^* \in \mathbb{X}$ as a realization of an $\mathbb{X}$-valued random variable $\mathbb{x}$ [5]. The distribution of $\mathbb{x}$, known in the literature as the prior distribution, constitutes a statistical model for images in $\mathbb{X}$ that encodes desired and expected properties about the solution. For example, in the context of limited-angle CT chest imaging, $\mathbb{x}$ would represent our expectations about what a CT chest image looks like at a population level in noise-free high-resolution conditions, with x^* understood as a realization of $\mathbb{x}$ stemming from performing a CT scan for an individual in that population. The measured data $y \in \mathbb{Y}$, acquired by the scanner in limited-angle conditions and corrupted by noise, is regarded as a realization of a $\mathbb{Y}$-valued random variable $\mathbb{y}$

conditioned on $\mathbb{x} = x^*$, with the two random variables being related by the forward model $\mathbb{y} = \mathcal{A}\,\mathbb{x} + \mathbb{e}$, where $\mathbb{e}$ is a random variable for the measurement noise. Given y, Bayesian image reconstruction becomes a statistical inference task underpinned by the conditional distribution of $(\mathbb{x}|\mathbb{y} = y)$, commonly known as the *posterior* distribution of $\mathbb{x}$. Bayesian estimators stem from seeking to optimally summarize this posterior distribution as a single point in $\mathbb{X}$, with different optimality criteria leading to different Bayesian estimators [6].

A key challenge in both the functional analytical and the Bayesian frameworks is to handle the *ill-posedness* of inversion, which arises from the fact that the forward operator $\mathcal{A}$ in practical inverse problems is either under-determined or poorly-conditioned with an unstable inverse. This leads to non-uniqueness and instability of reconstruction, meaning that many candidate images explain the measured data even in the absence of noise, or a small amount of noise in the data results in large changes in the recovered image. One can circumvent ill-posedness by introducing a *regularization* to stabilize the reconstruction, for instance by enforcing certain regularity conditions, such as smoothness. In Bayesian inversion, regularization is often achieved by selecting a prior distribution on images, which assigns low likelihood to images with unwanted features. In both settings, regularization involves a handcrafted a model that encodes prior knowledge as well as expected properties of the solution.

A. Model-based reconstruction

The design of regularized reconstruction methods is traditionally based on the functional analytic view. Early approaches sought to provide an analytic pseudo-inverse to the forward operator (*direct regularization*). An example is the filtered back-projection (FBP) method for tomographic image reconstruction, which regularizes by recovering the band-limited part of the image. A drawback of such an approach is that it is problem-specific, and does not generalize to a different forward operator.

Thus, it is desirable to formulate a general class of reconstruction methods that allow us to replace the forward operator in a plug-and-play manner. This leads to *variational models*, where the reconstruction task is formulated as a minimization problem of some penalty function $\mathcal{J} \colon \mathbb{X} \times \mathbb{Y} \to \mathbb{R}$ that ensures data-consistency and incorporates a regularizer. More precisely, such a penalty function typically takes the form $\mathcal{J}_\theta(x, y) := \mathcal{L}_\mathbb{Y}\big(\mathcal{A}\,x, y\big) + \mathcal{S}_\theta(x)$, where $\mathcal{L}_\mathbb{Y} \colon \mathbb{Y} \times \mathbb{Y} \to \mathbb{R}$ quantifies consistency in data space $\mathbb{Y}$ and the regularizer $\mathcal{S}_\theta \colon \mathbb{X} \to \mathbb{R}$ penalizes undesirable solutions. The data consistency term is often taken to be the least-squares loss $\mathcal{L}_\mathbb{Y}\big(\mathcal{A}\,x, y\big) := \|\,\mathcal{A}\,x - y\|_2^2$, which, if minimized without regularization, leads to overfitting the measurement noise. This underlines the need of including a regularization term $\mathcal{S}_\theta$ with possible hyper-parameters $\theta \in \mathbb{R}^d$, which allows to encode prior knowledge about desirable solutions, such as sparsity assumptions [2], [7], [8], [9]. The reconstruction method $\mathcal{R}_\theta \colon \mathbb{Y} \to \mathbb{X}$ with $\theta \in \mathbb{R}^d$ is now defined as the solution operator for the minimization problem

$$\mathcal{R}_\theta(y) \in \operatorname*{arg\,min}_{x \in \mathbb{X}} \mathcal{J}_\theta(x, y) \quad \text{where} \quad \mathcal{J}_\theta(x, y) := \mathcal{L}_\mathbb{Y}\big(\mathcal{A}\,x, y\big) + \mathcal{S}_\theta(x). \tag{1}$$

The hyper-parameter θ needs to be chosen beforehand depending on the noise level in the data. A simple yet widely popular special case is to construct $\mathcal{S}_\theta(x)$ as $\mathcal{S}_\theta(x) = \lambda\, \mathscr{S}_\vartheta(x)$, where $\theta = (\lambda, \vartheta)$ with $\lambda > 0$. Typically, only λ is adjusted depending on the noise level δ, while the parameter ϑ of the functional $\mathscr{S}_\vartheta$ is kept fixed (either hand-crafted or pre-trained on some training data set). Solutions to (1) are then often computed through an *iterative scheme* by (proximal) gradient-based methods, which are the basis for the unrolling techniques discussed in Sec. IV.

Reconstructions arising from minimizing a variational potential such as (1) can alternatively be interpreted as Bayes estimators. If the data-discrepancy $\mathcal{L}_{\mathbb{Y}}\big(\mathcal{A}\,x, y\big)$ is proportional to the negative log-likelihood for $(\mathbb{y}|\mathbb{x} = x)$, then minimizing it corresponds to maximum-likelihood estimation. In addition, if the regularizer $\mathcal{S}_\theta(x)$ in (1) is proportional to the negative log-prior density of $\mathbb{x}$, then (1) can be interpreted as the maximum a-posteriori probability (MAP) estimation [6].

Finally, it is important to note that the majority of physical phenomena are of nonlinear nature, but often a linearity assumption can be made under ideal measurement conditions. Nevertheless, for more advanced techniques or when accurate quantitative estimates are needed, the full nonlinear model is necessary. In this case, convergence guarantees often hold only locally [10].

A relevant question in practice is now what happens if the model in the physics-driven reconstruction is inaccurately approximated or linearized. Unfortunately, in general a recovery of the true unknown can only be guaranteed if the model mismatch is taken into account [11]. But, as these approximation errors are often of nonlinear nature, modern data-driven approaches offer a new and promising avenue to take these into account [12].

B. Data driven reconstruction

Although model-based inversion such as variational regularization with highly complex and sophisticated analytical regularizers represents a promising approach to solve ill-posed inverse problems, they pose two key challenges: (i) *handcrafting* a sufficiently expressive regularizer (or prior) and (ii) ensuring computational feasibility of hyper-parameter selection and evaluation. These challenges are more pronounced for Bayesian inversion. Algorithms to approximate the posterior mean and to perform uncertainty quantification are typically based on Markov chain Monte Carlo (MCMC) methods, which require carefully constructed priors and tend to be computationally infeasible for time-critical applications.

The development of data-driven reconstruction is inspired by the need to address the above challenges of achieving computational feasibility and selection of a domain-adapted regularizer/prior. Instead of handcrafting a reconstruction method, data-driven methods use training data to learn an *optimal* reconstruction method based on statistical learning. We will focus here on (*learned reconstruction*) methods $\mathcal{R}_\theta \colon \mathbb{Y} \to \mathbb{X}$ that are typically parameterized by some suitably chosen deep neural network

(DNN) and thus learning refers to selecting optimal parameters $\widehat{\theta}$ from the training data. This, however, depends on the statistical properties of the training data as discussed in the following and, as we will see later, has an effect on the type of convergence we obtain.

a) Supervised learning: In this case, one has access to pairs of ground-truth images and the corresponding measurements following the measurement model $y = \mathcal{A}x + e$. That is, training data are given as i.i.d. samples $(x_1, y_1) \ldots, (x_n, y_n) \in \mathbb{X} \times \mathbb{Y}$ of $(\mathbbm{x}, \mathbbm{y})$. An optimal set of parameters $\widehat{\theta}$ for the reconstruction method is found by empirical risk minimization given a suitable loss function $\mathcal{L}_{\mathbb{X}} \colon \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ in the image domain:

$$\widehat{\theta} \in \arg\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}_{\mathbb{X}}\big(\mathcal{R}_\theta(y_i), x_i\big). \tag{2}$$

Usual choices for the loss function include the squared ℓ^2-norm, where $\mathcal{L}_{\mathbb{X}}(x, x') := \|x - x'\|_2^2$. So far as the training data is concerned, we assume here that we have access to appropriate training data distributions. Nevertheless, the choice of the training data plays an important role in the performance of the trained models and can lead to overly optimistic results, a phenomenon referred to as *implicit data crime* by [13].

The formulation in (2) does not explicitly include the forward operator, or more generally, the data likelihood. Nevertheless, this is implicitly incorporated by the choice of training data that satisfy $y_i \approx \mathcal{A}x_i$. One can now select a parametrization $\mathcal{R}_\theta \colon \mathbb{Y} \to \mathbb{X}$ that accounts for the fact that a trained estimator should represent a regularized inversion method. A popular example of such a domain-adapted parametrization is to combine a DNN $\mathcal{C}_\theta \colon \mathbb{X} \to \mathbb{X}$ with a handcrafted pseudo inverse $\mathcal{A}^\dagger \colon \mathbb{Y} \to \mathbb{X}$, which incorporates a direct regularization, as discussed in Sec. II-A. The learned reconstruction operator is then represented by the composition $\mathcal{R}_\theta := \mathcal{C}_\theta \circ \mathcal{A}^\dagger$ where $\mathcal{C}_\theta \colon \mathbb{X} \to \mathbb{X}$ acts as a learned post-processing operator that removes noise and under-sampling artifacts. Popular architectures in imaging are based on convolutional neural networks (CNNs), more specifically convolutional autoencoders with an encoding/decoding branch, such as the popular U-Net [14] and related architectures inspired by harmonic analysis [15]. The supervised setting applies to the case where pairs of high- and low-quality reconstructions are available, such as low-dose and high-dose computed tomography (CT) scans. In this case, the high-dose reconstruction can be identified with x and the low-dose measurements provides the data y to obtain an initial reconstruction by applying the pseudo-inverse operator $\mathcal{A}^\dagger$, which can be considered a pre-computation step before the training procedure.

Another highly popular parameterization for $\mathcal{R}_\theta$ is based on so-called *unrolling*. The idea is to start with some iterative scheme, like one designed to minimize $x \mapsto \mathcal{L}_{\mathbb{Y}}\big(\mathcal{A}x, y\big)$ or the objective in (1). Next, the iterative scheme is truncated to a fixed number of iterations and unrolled by replacing selected handcrafted updates with (possibly shallow) neural networks (NNs). Notably, for model-based learning, one does not usually replace $\mathcal{A}$ and its adjoint $\mathcal{A}^\top$, but other components such as the proximal operator

[3, Sec. 4.9.1]. Hence, the DNN for $\mathcal{R}_\theta$ is formed by combining (shallow) NNs with physics-driven operators. Popular examples of unrolled networks include learned primal-dual [16], and variational networks [17]. At this point, we would like to stress that *unrolling is merely a way to select an architecture for the DNN parametrizing* $\mathcal{R}_\theta$. In particular, training an unrolled reconstruction network following (2) does not, in general, lead to an $\mathcal{R}_\theta$ that minimizes a variational objective as in (1), even though the architecture of $\mathcal{R}_\theta$ is constructed by unrolling an optimization solver. One can instead interpret $\mathcal{R}_\theta$ as a Bayes estimator that summarizes the posterior distribution of $\mathbbm{x}$ conditioned on $\mathbbm{y} = y$ using a point estimate. The statistical characterization of this estimate depends on the choice of the loss function $\mathcal{L}_\mathbb{X}$. For instance, when $\mathcal{L}_\mathbb{X}$ is the squared ℓ_2 distance, $\mathcal{R}_\theta$ approximates the classical minimum mean-squared error (MMSE) estimate (i.e., the conditional posterior mean $\mathrm{E}\left[\mathbbm{x}|\mathbbm{y} = y\right]$) in the limit of infinite training data.

b) Unsupervised learning: This type of learning comes in two flavors, depending on whether high-quality images or measurement data are available. In the former case, training data $x_1, \ldots, x_n \in \mathbb{X}$ are i.i.d. samples of $\mathbbm{x}$. One can then consider reconstruction methods $\mathcal{R}_\theta$ of the form in (1) with a regularizer/prior $\mathcal{S}_\theta \colon \mathbb{X} \to \mathbb{R}$ that is learned from training data through generative modeling, like generative adversarial networks (GANs) or variational auto-encoders (VAEs). GAN-based approaches implicitly regularize the reconstruction by restricting it to the range of a pre-trained generator [18], whereas the other alternative is to learn an explicit regularizer parametrized by a DNN [19], [20].

The other variant is when training data $y_1, \ldots, y_n \in \mathbb{Y}$ are i.i.d. samples of $\mathbbm{y}$. One option is to learn a data-driven solver for optimization problems of the type in (1) where the objective is parameterized by $y \in \mathbb{Y}$ (see [21] for further details). In this setting, it is quite natural to parametrize $\mathcal{R}_\theta$ by a DNN whose architecture is given by unrolling an optimization solver for (1). Unlike the supervised setting where the reconstruction operator does not necessarily minimize a variational potential, the trained DNN in this case is designed specifically to solve an underlying variational problem of the form (1).

c) Weakly supervised learning: This refers to the case where samples of measurement data and ground-truth are available, but they are not paired. Training data then consists of un-paired $x_1, \ldots, x_n \in \mathbb{X}$ and $y_1, \ldots, y_n \in \mathbb{Y}$ in the sense that $x_i \in \mathbb{X}$ and $y_i \in \mathbb{Y}$ are i.i.d. samples of the $\mathbbm{x}$- and $\mathbbm{y}$–marginal distributions of $(\mathbbm{x}, \mathbbm{y})$, respectively. Such data can be used to set-up a learning problem which quantifies consistency against data and image error similar to (2), but using loss functions on both $\mathbb{X}$ and $\mathbb{Y}$. This is complemented with a term that quantifies similarity of probability distributions induced by training data, thus resulting in an elaborate learning problem, see [3, eq. (5.3)]. The learning problem is recast into a more suitable form by using deep generative models.

Notable methods that fall within the weakly supervised training paradigm are adversarial regularizer (AR) [20], [22] and its convex variant given by adversarial convex regularizer (ACR) [23]. Both of these methods seek to learn a regularizer in a variational model as a critic that can tell apart noisy reconstructions (obtained through some simple baseline approach, e.g., by applying the pseudo-inverse of $\mathcal{A}$ on the measurements) from the ground-truth image samples x_i. The regularizer is restricted to be Lipschitz-continuous (and convex in case of ACR), which plays an important role in the stability analysis of the resulting variational model. Lipschitz-continuity of the regularizer is enforced via a soft-penalty (see [20], [23] and references therein), whereas convexity is enforced by the choice of DNN architecture.

III. Types of convergence

Now that we have established the concept of reconstruction methods, regularization, and learned reconstruction, the question that remains is: what are the theoretically desired properties of a learned reconstruction? For instance, if we think about iterative reconstruction, the question of convergence is relevant. Does the iterative scheme converge to a fixed point? Does this fixed point correspond to a minimizer of a variational loss, such as in (1)? Moreover, is it, in fact, a regularization method? These are not only purely academic questions, but provide interpretability and guarantees of correctness of the obtained solution. This section outlines important notions of convergence relevant for the image reconstruction task at hand in an accessible way, while not compromising on the mathematical rigor.

a) Formal stability: The weakest and arguably the most fundamental mathematical guarantee comes in the form of *stability*, which has its origin in Hadamard's definition of well-posedness [2]. *Stability* refers to a smooth variation of the reconstruction with respect to changes in the observed data (see box). The notion of stability can also be applied to Bayesian inversion, in which case it refers to the stability of posterior probabilities, statistical moments, and/or Bayesian estimators. When we refer to provable stability in the following, we mean that it is possible to estimate and control the maximal error in the reconstruction with respect to deviations in the measurement (for instance, in terms of the Lipschitz constant). It should be also noted, that the notion of stability (or lack thereof) itself can be quite meaningless, if no further conditions are provided. For instance, a reconstruction method that always produces the same image from varying data is in fact stable, but useless in practice.

Stability versus accuracy

Consider a trained reconstruction operator $\mathcal{R}_\theta$ with fixed network parameters (learned from training data). The reconstruction produced by $\mathcal{R}_\theta$ is said to be *stable* if $\mathcal{R}_\theta : \mathbb{Y} \to \mathbb{X}$ is a continuous function of the observed data. Formally, stability demands that

$$\|\mathcal{R}_\theta(y+w) - \mathcal{R}_\theta(y)\|_{\mathbb{X}} \to 0 \text{ as } \|w\|_{\mathbb{Y}} \to 0.$$

One possibility for a stability analysis is to consider the Lipschitz constant L of the mapping

$\mathcal{R}_\theta$, which is given by the smallest $L > 0$, such that

$$\| \mathcal{R}_\theta(y_1) - \mathcal{R}_\theta(y_2)\| \leq L\|y_1 - y_2\|, \text{ for all } y_1, y_2 \in \mathbb{Y}. \tag{3}$$

It is important to note that since DNNs are compositions of affine functions and smoothly varying nonlinear activations, a mapping $\mathcal{R}_\theta$ modeled using DNNs is continuous and a constant L satisfying (3) exists. However, although $\mathcal{R}_\theta$ is formally stable, the constant L might be large, leading to large deviations in the reconstruction for small changes in the measurement.

Additionally, a consequence of (3) is that the reconstruction of a slightly perturbed image must satisfy

$$\| \mathcal{R}_\theta(\mathcal{A}(x + \eta)) - \mathcal{R}_\theta(\mathcal{A}\, x)\| \leq L\| \mathcal{A}\, \eta\|, \text{ for any perturbation } \eta.$$

Since the forward operator $\mathcal{A}$ is ill-posed and with possibly non-trivial null space, $\| \mathcal{A}\, \eta\|$ could be arbitrarily small for some perturbation η. As a result, the reconstruction remains insensitive to such changes, thereby compromising in accuracy if L is small, and consequently an accurate $\mathcal{R}_\theta$ for small perturbations must have a large Lipschitz constant L.

Adversarial robustness

Adversarial robustness of a trained reconstruction operator $\mathcal{R}_\theta$ is measured by the largest deviation caused in the reconstruction by a small amount of noise in the data. For a given $y_0 = \mathcal{A}\, x_0 \in \mathbb{Y}$, where x_0 is the underlying image, and a given noise level ϵ_0, this is defined formally as [24]:

$$\delta_{\text{adv}} = \sup_{w : \|w\| \leq \epsilon_0} \|\mathcal{R}_\theta(y_0 + w) - \mathcal{R}_\theta(y_0)\|_2 \,. \tag{4}$$

If δ_{adv} is small for small ϵ_0, the reconstruction method $\mathcal{R}_\theta$ is said to be adversarially robust.

Adversarial robustness of image recovery methods, learning-based or classical, is crucial for their safe deployment in decision-critical applications such as medical imaging. Some skepticism about adversarial stability (or lack thereof) of deep learning-based approaches has been raised in [25]. Nevertheless, subsequent work on adversarial robustness of learned methods in [24] put this concern into perspective and performed a systematic comparison of data-driven methods with the classical (and provably stable) total variation (TV)-regularized solution. In a compressed sensing experiment with random Gaussian measurements, the learned methods were found to be as robust as TV to adversarial noise, and superior to TV for statistical noise. Further, on the fastMRI knee dataset[a], the learned methods were shown to be even more resilient to large adversarial perturbations as compared to TV. Very recently, [26] considered an ℓ_∞-norm-based measure of adversarial stability, as opposed to the ℓ_2-norm, owing to its relevance in capturing *localized reconstruction artifacts*. The authors showed that neural network-based methods are more robust to ℓ_∞-based adversarial perturbations in comparison with TV.

The studies of adversarial robustness as discussed above pertain to supervised methods trained end-to-end, and only concern robustness to noise in the measurement. A principled empirical and theoretical analyses are needed for a quantitative understanding of the robustness of unsupervised methods (such as AR, ACR, network Tikhonov (NETT), plug-and-play (PnP) methods, etc.) to noise and possibly to other types of distortions that go beyond noise (e.g., forward-model and/or prior mismatch).

[a] fastMRI data set available at: https://fastmri.med.nyu.edu/

b) Fixed-point convergence: To solve a reconstruction problem iteratively starting from an initial guess, one applies an operator updating $\mathcal{T}\colon \mathbb{X} \to \mathbb{X}$ on the previous iterate(s): $x_{k+1} := \mathcal{T}(x_k)$. The operator $\mathcal{T}$ typically involves the forward operator, its adjoint, and the observed data. It is important to determine whether the iterates converge. One such notion is *fixed-point convergence*, which means that $\lim_{k\to\infty} x_k = x_\infty$, where $x_\infty := \mathcal{T}(x_\infty)$ is a fixed-point of $\mathcal{T}$. If fixed-point convergence holds, the iterates stabilize after sufficiently many steps, which is clearly desirable for an iterative scheme. However, it does not necessarily tell anything about what kind of solutions the iteration converges to.

c) Convergence to the minimum of a variational loss (objective convergence): For a stronger notion of convergence of an iterative algorithm, one can consider minimizing a variational loss, similar to (1). That is, we consider the loss function of the form $\mathcal{J}(x) := \mathcal{L}_{\mathbb{Y}}\big(\mathcal{A}x, y\big) + \mathcal{S}(x)$ for data $y \in \mathbb{Y}$. In an optimization algorithm, an initial guess is refined iteratively by exploiting, for instance, information about the gradient $\nabla\mathcal{J}$ at each step to compute a minimizer. Such an iterative scheme $x_{k+1} := \psi_{\vartheta_k}\big(x_k, \nabla\mathcal{J}(x_k)\big)$ with an updating rule $\psi_{\vartheta_k}\colon \mathbb{X}\times\mathbb{X} \to \mathbb{X}$ and iteration-dependent parameters $\vartheta := \{\vartheta_1, \vartheta_2, \ldots\}$ is said to converge to a minimizer if $x_k \to \arg\min_{x\in\mathbb{X}} \mathcal{J}(x)$ as $k \to \infty$. That is, we can now characterize the point of convergence as the minimizer of an objective function. We will refer to this notion of convergence as *objective convergence* in the remainder of the paper.

d) Convergent regularization: The strongest form of convergence we discuss here considers whether a regularized solution for an ill-posed inverse problem with (linear) forward operator $\mathcal{A}\colon \mathbb{X} \to \mathbb{Y}$ tends to the solution corresponding to noise-free data $y^0 \in \mathbb{Y}$ as the noise level vanishes. This could be viewed as a second level of convergence, as convergence of the iterative scheme is needed.

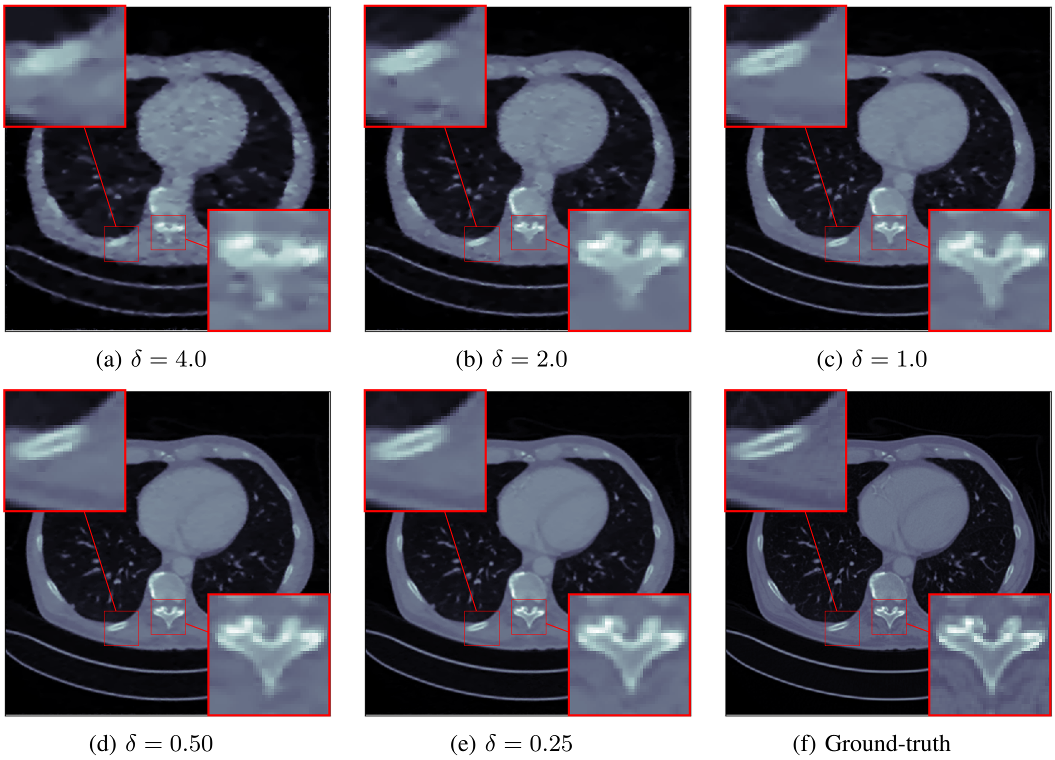
Formally, a regularization method can be understood as a parameterized family $\{\mathcal{R}_\theta\}_{\theta\in\mathbb{R}^d}$ of reconstruction methods. Here, the parameter θ depends on the noise level $\delta > 0$, where $\|y^\delta - y^0\| \le \delta$ with $y^0 := \mathcal{A}x^*$ denoting noise-free data. A regularization method is convergent if there exists a parameter choice rule $\delta \mapsto \theta(\delta, y^\delta)$ such that reconstructions converge to a pseudo-inverse solution as noise vanishes, i.e., $\mathcal{R}_{\theta(\delta,y^\delta)}(y^\delta) \to \mathcal{A}^\dagger(y^0)$ as $\delta \to 0$.

In the context of variational models (1), one can re-formulate the above as follows: Let $x_{\theta,\delta} \in \mathbb{X}$ denote a minimizer to the objective in (1) for given θ and data $y^\delta \in \mathbb{Y}$ with noise level $\|e\| = \|y - y^0\| < \delta$.

Next, assume also that there is a parameter choice rule $\delta \mapsto \theta(\delta, y^\delta)$, such that $\theta(\delta, y^\delta) \to \theta_0$ as $\delta \to 0$. The variational model defined by (1) is said to *converge to an $\mathcal{S}$-minimizing solution* if $x_{\theta(\delta,y^\delta),\delta} \to x^\dagger$ as $\delta \to 0$. Here, $x^\dagger \in \mathbb{X}$ denotes a minimizer of the regularization functional $\mathcal{S}_{\theta_0}$ among all solutions that are consistent with the clean measured data y^0. The $\mathcal{S}$-minimizing solution is formally defined as

$$x^\dagger \in \arg\min_{x \in \mathbb{X}} \ \mathcal{S}_{\theta_0}(x) \quad \text{subject to } y^0 = \mathcal{A}\,x, \text{ where } \theta_0 := \lim_{\delta \to 0} \theta(\delta, y^\delta). \tag{5}$$

Fig. 1 shows such convergence in the context of ACR [23], a learned convex regularizer. Here, the regularizer is constructed as $\mathcal{S}_\theta(x) = \lambda\, \mathscr{S}_\vartheta(x)$, where the functional $\mathscr{S}_\vartheta$ is modeled using an ICNN and the parameters ϑ are learned from training data. For reconstruction, the variational optimization problem in (1) is solved with $\theta = (\lambda, \vartheta^*)$, where ϑ^* denotes the parameter of a trained model and $\lambda : \delta \mapsto \lambda(\delta)$, the regularization penalty, should be chosen appropriately based on δ to guarantee convergence to the $\mathscr{S}_{\vartheta^*}$-minimizing solution [23].



(a) $\delta = 4.0$

(b) $\delta = 2.0$

(c) $\delta = 1.0$

(d) $\delta = 0.50$

(e) $\delta = 0.25$

(f) Ground-truth

Fig. 1: ACR [23] as a convergent regularization strategy: The reconstructed image converges to the ground-truth as $\delta \to 0$, subject to an appropriate parameter choice rule for the scalar regularization parameter: $\delta \mapsto \lambda(\delta) > 0$. Here, the regularizer parameter ϑ is learned from data and kept fixed during reconstruction. The highlighted regions show the key differences between the reconstructed images for different values of the pair $(\delta, \lambda(\delta))$.

IV. Provable stability and convergence

Most learned reconstruction methods $\mathcal{R}_\theta \colon \mathbb{Y} \to \mathbb{X}$ are formally stable since they are continuous mappings. This is the case with one-step methods $\mathcal{R}_\theta := \mathcal{C}_\theta \circ \mathcal{A}^\dagger$ whenever the pseudo-inverse $\mathcal{A}^\dagger \colon \mathbb{Y} \to \mathbb{X}$ and the learned post-processor $\mathcal{C}_\theta \colon \mathbb{X} \to \mathbb{X}$ are continuous. Likewise, common unrolling architectures for $\mathcal{R}_\theta$, like variational networks [17] and learned primal-dual (LPD) [16], are continuous. The above claims are supported by Fig. 2, which shows performance of FBPconvNet (one-step method) and LPD (unrolling architecture) for sparse-view CT reconstruction. In contrast, a reconstruction operator given by a variational scheme with a learned regularizer, as in AR, is not necessarily continuous. Further, even if the reconstruction operator is formally continuous, its Lipschitz constant can be large, resulting in loose stability bounds.

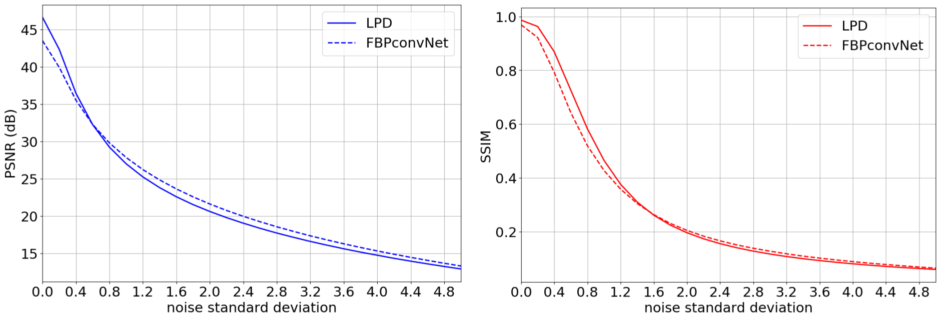


Fig. 2: Reconstruction quality (PSNR (dB) on the left and SSIM on the right) as a function of noise level in data for LPD (unrolling) [16] and FBPconvNet (one-step) [14] methods. Both methods are trained against simulated noise-free 2D sparse-view CT data generated from 2D images that are cross sections of anthropomorphic phantoms in the AAPM low-dose CT challenge [27].

Regarding convergence, most results can be placed within the variational framework with either an explicit or an implicit regularizer, see Fig. 3 for an overview of the discussed methods. Explicit regularization schemes model the regularizer directly through a neural network, whereas the implicit schemes regularize the solution via a denoiser (also known as plug-and-play methods). The explicit regularization schemes [20], [19], [23] typically come with a stability or convergent regularization guarantees, whereas the plug-and-play methods have been shown to possess either fixed-point or objective convergence subject to different constraints on the denoiser [28], [29].

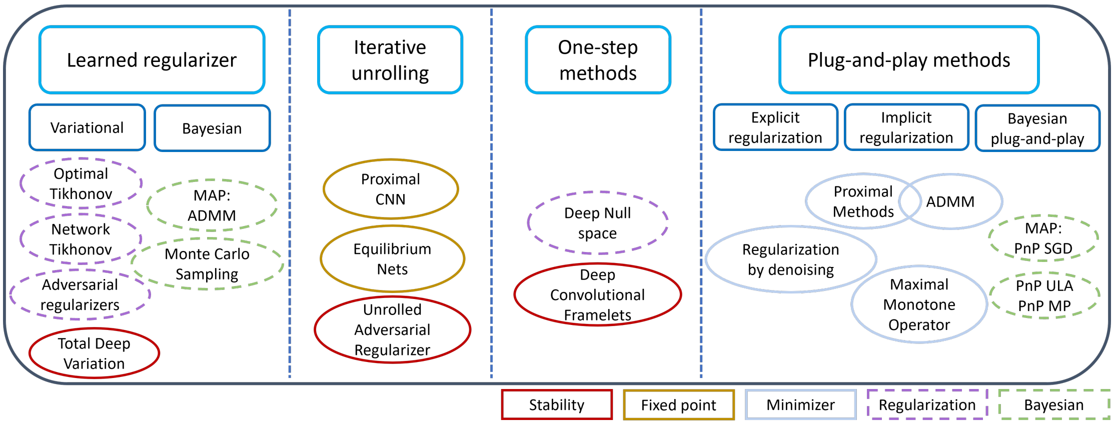


Fig. 3: Categorization of data-driven reconstruction approaches, which are color-coded based on the strongest type of convergence guarantee they satisfy.

A. *Learned regularization methods*

These methods are based on learning a DNN representing a regularizer $\mathcal{S}_\theta\colon \mathbb{X} \to \mathbb{R}$ in a variational model of the form (1) (learned regularizer) or reconstruction by considering images generated by a DNN (regularization-by-architecture).

Inspired by the optimal transport theory, an adversarial framework for learning the regularizer was proposed in [20]. This method, also referred to as the adversarial regularizer (AR) method, comes with stability guarantees subject to the regularizer being 1-Lipschitz and coercive. Its convex counterpart (abbreviated as ACR) models the regularizer using an ICNN [30]. The analysis of ACR [23] follows from the classical convex regularization theory and one can formally establish well-posedness of the reconstruction problem (i.e., existence and uniqueness of the solution, and its continuous dependence on the observed data) using (strong) convexity of the regularizer. Moreover, ACR can be shown to be a convergent regularization technique using function-analytic tools in classical regularization theory for inverse problems. Another approach to learning a regularizer is the NETT method [19], which considers a learned regularizer given by a CNN that is trained using an encoder-decoder set-up. The resulting variational model is shown to be well-posed and convergent subject to mild conditions (see [19, Condition 2.2]) on the neural network that parameterizes the regularizer. Other variants and extensions of NETT (such as augmented NETT [31] and the synthesis counterpart of NETT [32]) are also provably convergent regularization methods.

A slightly different approach is [33], which can be seen as a regularization-by-architecture scheme akin to deep image prior [34]. Regularization properties for this method are shown rigorously by constructing the generator as a multi-level sparse coding network. The approach proposed in [35] is similar in spirit with [33], in the sense that regularization is achieved by restricting the image to lie in the range of an untrained generator network. The paper provides recovery guarantees that are similar to the compressed sensing guarantees in flavor (using the so-called set-restricted eigenvalue condition). This condition is essentially the same as the *restricted isometry condition* [9, Definition 2], but defined for all images in the range of the generator. In contrast with [34], [35], the method developed in [36] seeks a reconstruction in the range of a pre-trained generator. A proximal gradient-descent (PGD) algorithm is used for recovery by replacing the projection operator onto the range of the generator with a learned network. The recovery algorithm is provably convergent.

Adversarial regularizers: Why convexity matters

Consider a denoising problem on the real line[a] (i.e., $\mathbb{X} = \mathbb{R}$), where the distribution of ground-truth is given by $p^\star(x) = \frac{1}{2}\big(\delta_{-1}(x) + \delta_1(x)\big)$, two Dirac pulses at $+1$ and -1. Let the noisy data have distribution $p_{\text{noisy}} = U\big([-\frac{1}{2}, \frac{1}{2}]\big)$, the uniform distribution over $[-\frac{1}{2}, \frac{1}{2}]$. Recall the adversarial learning framework [20], wherein the regularizer is trained to discern the distribution of the ground-truth from that of some baseline reconstruction by minimizing the functional

$$\mathcal{L}_{\text{AR}}(\mathscr{S}) = \mathrm{E}_{\mathbb{x}\sim p^{\star}}\mathscr{S}(\mathbb{x}) - \mathrm{E}_{\mathbb{x}\sim p_{\text{noisy}}}\mathscr{S}(\mathbb{x}),$$

over $\mathscr{S} \in \operatorname{Lip}(\mathbb{X})$, where $\operatorname{Lip}(\mathbb{X})$ denotes the class of 1-Lipschitz functionals on $\mathbb{X}$. Thanks to the Kantorovich-Rubinstein duality, the optimal regularizer $\mathscr{S}^{*}$ satisfies $\mathcal{L}_{\text{AR}}(\mathscr{S}^{*}) = -\mathbb{W}_1\left(p_{\text{noisy}}, p^{\star}\right)$, the negative Wasserstein-1 distance between $p^{\star}$ and p_{noisy}. As apparent in this case, the optimal transport map from p_{noisy} to $p^{\star}$ is given by

$$T(x) = \begin{cases} 1 & \text{for } x \geq 0 \\ -1 & \text{for } x < 0. \end{cases} \tag{6}$$

Consequently, the Wasserstein-1 distance evaluates to:

$$\mathbb{W}_1\left(p_{\text{noisy}}, p^{\star}\right) = \int_{-\frac{1}{2}}^{0} |x-(-1)|\,\mathrm{d}x + \int_{0}^{\frac{1}{2}} |x-1|\,\mathrm{d}x = \frac{3}{4}.$$

The function $\mathscr{S}^{*}(x) = -|x|$ is 1-Lipschitz and achieves $\mathcal{L}_{\text{AR}}(\mathscr{S}^{*}) = -\frac{3}{4}$, and is therefore the optimal regularizer. The resulting variational problem for denoising reads

$$\min_{x} \frac{1}{2}(x-y)^2 - \lambda\,|x|, \quad \text{where} \quad -\frac{1}{2} \leq y \leq \frac{1}{2} \text{ and } \lambda > 0. \tag{7}$$

One can solve (7) in closed form as

$$\hat{x}(y) = \begin{cases} y+\lambda & \text{for } y \geq 0 \\ y-\lambda & \text{for } y < 0. \end{cases} \tag{8}$$

Clearly, the reconstruction given by (8) changes drastically as the data y changes sign and is therefore discontinuous at $y = 0$. Note that it does not violate the stability guarantee of AR, which only ensures convergence up to sub-sequences. For instance, consider a sequence $y_k = (-1)^k \frac{1}{k}$ for $k = 1, 2, \ldots$, so $y_k \to 0$ as $k \to \infty$. The corresponding sequence of reconstructions is given by $\left\{(-1)^k \frac{1}{k} + (-1)^k \lambda\right\}_{k\geq 1}$, which does not converge, but has a sub-sequence converging to λ, which is a solution of (7) for $y = 0$.

Imposing (strong) convexity on the regularizer [23] helps achieve stronger forms of convergence and precludes such discontinuities in the reconstruction. More precisely, for two measurement vectors y_1 and y_2 that are δ apart (in norm on $\mathbb{Y}$), the corresponding reconstructions can vary (with respect to norm on $\mathbb{X}$) by at most $\frac{\beta\delta}{\lambda\rho}$, where β is the spectral norm of $\mathcal{A}$ and $\mathscr{S}_{\vartheta}$ is ρ-strongly convex [23, Prop. 2]. Notably, ACR is also a convergent regularization scheme, meaning that the reconstruction converges to the $\mathscr{S}_{\vartheta}$-minimizing solution of $\mathcal{A}x = y^0$, where y^0 denotes clean data, as the noise level $\delta \to 0$, provided that the regularization penalty $\delta \mapsto \lambda(\delta)$

satisfies $\lim\limits_{\delta\to 0}\lambda(\delta) = \lim\limits_{\delta\to 0}\frac{\delta}{\lambda(\delta)} = 0$ [23, Prop. 3]. The importance of convexity prior for stability is demonstrated through the example of limited-view CT reconstruction from [23][b] in Fig. 4.

Notably, such convergence results for strongly convex regularizers are not limited to only linear forward operators. If the forward operator $\mathcal{A}$ is nonlinear, it needs to satisfy some additional technical conditions for the variational reconstruction to converge to the $\mathcal{S}$-minimizing solution as $\delta \to 0$. One such condition is that the level sets $L_t := \{x \in \mathbb{X} : \mathcal{J}_\theta(x, y) \leq t\} \subset \mathbb{X}$ of the variational objective are required to be sequentially pre-compact for any $t > 0$, meaning that, every sequence in L_t must have a sub-sequence converging to some element in $\mathbb{X}$ (which does not necessarily have to be in L_t). For a complete and precise statement of such strong convergence results, we refer interested readers to Proposition 3.32 in [1].

[a] Thanks to Sebastian Lunz for providing this example

[b] Thanks in particular to Zakhar Shumaylov for the limited-view CT experiments.

B. Iterative unrolling with fixed-point convergence

The main philosophy behind algorithm unrolling is to construct a DNN architecture by unfolding a fixed number of iterations of an optimization algorithm. Subsequently, different components of the algorithm are replaced by learnable units (typically modeled using shallow neural networks) and the overall network is trained end-to-end to produce a reconstruction from its corresponding measurement. The origin of unrolling can be traced back to the seminal work by Gregor and LeCun [37] for solving sparse coding via unfolding the iterative soft-thresholding algorithm. The output of the k^{th} layer of a generic unrolled architecture can be expressed as $x_{k+1} = \psi_{\theta_k}(x_k, y)$, where ψ_{θ_k} is a non-linear mapping with parameters θ_k.

As we remarked in Sec. II.B.a, the reconstruction of an end-to-end trained unrolled network can be interpreted as a Bayes estimator. Recall that the conditional mean estimator, given by $\mathcal{R}^*(y) = \mathrm{E}\left[\mathbb{x}|\mathbb{y} = y\right]$ is a solution of

$$\mathcal{R}^* = \underset{\mathcal{R}:\mathbb{Y}\to\mathbb{X}}{\arg\min}\, \mathrm{E}_{\mathbb{x},\mathbb{y}} \left\|\mathcal{R}(\mathbb{y}) - \mathbb{x}\right\|_2^2, \tag{9}$$

where the minimization is carried out over all measurable mappings from $\mathbb{Y}$ to $\mathbb{X}$ (see Proposition 2 in [38]). Therefore, an unrolled network with a sufficiently powerful parameterization to approximate any measurable map from the data space to the image space essentially seeks to approximate $\mathcal{R}^*$ given enough training data. One can obtain an approximation to a different Bayes estimator by using a different loss function for training.

However, without any further assumptions on ψ_{θ_k}, it is generally not possible to characterize the estimate of an unrolled network as the stationary point of a variational potential or a fixed-point of a non-linear map. Further, one trains an unrolled architecture for a few iterations (typically ≤ 20) due to memory constraints, and the reconstruction deteriorates if more iterations are performed at test time than the number of iterations that were used in training.

The deep equilibrium (DEQ) model proposed in [39] provided a promising avenue to reduce the memory requirement of training unrolled networks. The key idea was to represent the output of a feed-forward model as a fixed-point of a nonlinear transformation, through which one can back-propagate using implicit differentiation. This approach can effectively learn an infinite depth network with a constant memory footprint. The DEQ models were leveraged in [40] for imaging inverse problems. Such models come with two notable characteristics: (i) weight-sharing, i.e., by using the same set of parameters at each layer ($\theta_k = \theta$ for all k), and (ii) explicitly constraining the output $x^*(y,\theta)$ of the unrolled network to be a fixed-point of ψ_θ, while further ensuring that the (nonlinear) operator ψ_θ is contractive. The training problem reads

$$\min_\theta \frac{1}{n}\sum_{i=1}^{n} \|x_i^*(y_i,\theta) - x_i\|_2^2 \text{ subject to } x_i^*(y_i,\theta) = \psi_\theta\left(x_i^*(y_i,\theta), y_i\right). \tag{10}$$

It is shown in [40] that the trained map ψ_θ can be applied iteratively beyond the number of iterations trained without any degradation in reconstruction quality. The DEQ architectures come with fixed-point convergence incorporated into them by construction.

Back-propagating through an infinite-depth network in DEQ models essentially requires computing inverse Jacobian-vector product, which is approximated using fixed-point iterations [40, Sec. 4.2] or via quasi-Newton (QN) methods, namely Broyden methods [39, Sec. 3.1.3]. Such an iterative approximation of inverse Jacobian-vector products could be computationally demanding, which slows down the training of DEQ models. Some recent works have sought to accelerate the training of DEQ networks, for instance via Jacobian-free back-propagation [41], or by using the SHINE approach in [42], which considered QN methods to compute the forward pass of DEQ models and used the QN matrices (that are available as a bi-product of the forward pass) to approximate the inverse Jacobian.

Generally, sharing the weights of different layers in an unrolling scheme leads to a less expressive model and the image quality might suffer as a consequence. However, we would like to emphasize that weight-sharing is not essential for having a provable unrolled scheme. For instance, the stochastic unrolled network proposed in [43] is a memory-efficient unrolling scheme that comes with recovery guarantees, but with no weight-sharing across the layers. Learned optimization solvers (see Sec. IV-E and [21] for more details) also rely on unrolling, but without any weight-sharing, and can be shown to converge to the minimizer of a variational potential.

Notably, unrolling an iterative scheme can be constructed to have objective convergence. The key idea behind such constructions is to parametrize the neural network in a manner such that it corresponds to the proximal operator of an underlying proper, convex, and lower semi-continuous function. One example is Parseval proximal neural network [44] that use tight frames to parametrize the affine layers.

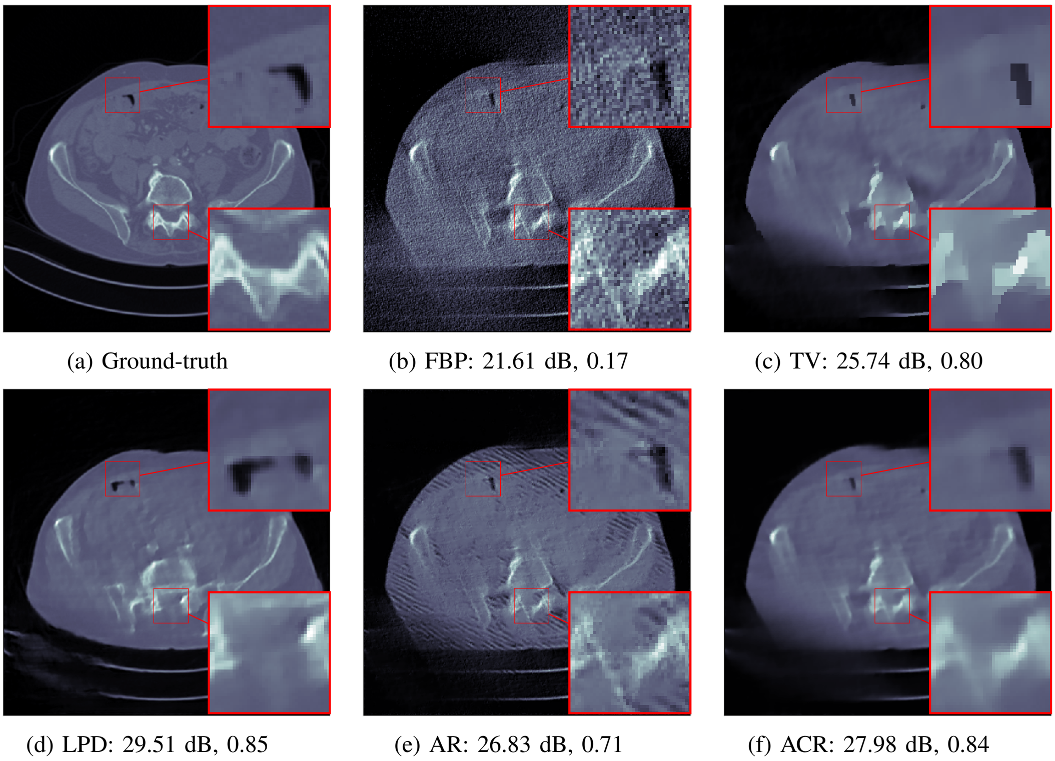
C. One-step methods

The one-step methods consist in learning a deep neural network-based post-processing of a model-based reconstruction based on pairs of input and target images [14]. More specifically, the reconstruction operator is parameterized as $\mathcal{R}_\theta := \mathcal{C}_\theta \circ \mathcal{B}$, where $\mathcal{B}\colon \mathbb{Y} \to \mathbb{X}$ denotes a classical reconstruction method (with no or few tune-able parameters, e.g., FBP or TV in X-ray CT) and $\mathcal{C}_\theta\colon \mathbb{X} \to \mathbb{X}$ represents a deep convolutional network with parameters θ. The reconstructed image produced by such a post-processing method fails to satisfy the data-consistency criterion. That is, a small value of $\left\|\mathcal{A}\,x^\dagger - y^\delta\right\|$ does not necessarily imply a small value for the data-fidelity term $\left\|\mathcal{A}\,\mathcal{C}_\theta(x^\dagger) - y^\delta\right\|$ corresponding to the output of $\mathcal{C}_\theta$, where $x^\dagger$ is the reconstruction obtained using $\mathcal{B}$. Consequently, such post-processing schemes do not lead to convergent regularization strategies. This issue was addressed in [45] by parametrizing the operator $\mathcal{C}_\theta$ as $\mathcal{C}_\theta = \mathrm{Id} + \left(\mathrm{Id} - \mathcal{A}^\dagger\,\mathcal{A}\right)\mathcal{Q}_\theta$, where $\mathcal{Q}_\theta$ is a Lipschitz-continuous DNN. Since $\left(\mathrm{Id} - \mathcal{A}^\dagger\,\mathcal{A}\right)$ is the projection operator onto the null-space of $\mathcal{A}$, the operator $\mathcal{C}_\theta$ (referred to as null-space network) always satisfies $\mathcal{A}\,\mathcal{C}_\theta(x^\dagger) = \mathcal{A}\,x^\dagger$, ensuring that the output of $\mathcal{C}_\theta$ explains the observed data. Null-space networks are shown to provide convergent regularization schemes [45].

Deep convolutional framelets [15] aim to gain a better understanding and interpretability of deep learning by establishing a link with wavelet theory. An image is here represented by convolving local and non-local bases. The convolutional framelets generalize the theory of low-rank Hankel matrix approaches for inverse problems and [15] extends the idea to obtain a DNN using multilayer convolutional framelets that enable perfect representation of an image. The analysis in [15] shows that the popular deep network components such as residual block, redundant filter channels, and concatenated rectified linear units (ReLU) do indeed offer perfect representation, while the pooling and unpooling layers should be augmented with high-pass branches to meet the perfect representation condition.

D. Plug-and-play denoising

Reconstruction in the variational setting typically requires an iterative algorithm to minimize the underlying variational loss. Popular iterative techniques such as PGD and alternating-directions method of multipliers (ADMM) entail applying the proximal operator corresponding to the (possibly non-smooth) regularizer to update the estimate. The seminal work by Venkatakrishnan et al. [46] pioneered the idea of replacing the proximal operator with an off-the-shelf denoiser. To give a specific example, such a PnP-denoising used inside the PGD algorithm leads to the iterative scheme $x_{k+1} = D_\sigma\left(x_k - \eta \nabla_x \mathcal{L}_{\mathbb{Y}}\big(\mathcal{A}\,x, y\big)\big|_{x=x_k}\right)$ for reconstruction, starting from an initial estimate x_0. Here, $D_\sigma\colon \mathbb{X} \to \mathbb{X}$ is typically a denoiser to remove Gaussian noise of standard deviation σ. Traditionally, the choice of the denoiser has been model-inspired (e.g., BM3D, dictionary-based denoisers like K-SVD, TV, etc.), but more recently, PnP algorithms have used off-the-shelf deep neural network-based denoisers (e.g., DnCNN). These methods have shown excellent empirical performance for practical


(a) Ground-truth

(b) FBP: 21.61 dB, 0.17

(c) TV: 25.74 dB, 0.80

(d) LPD: 29.51 dB, 0.85

(e) AR: 26.83 dB, 0.71

(f) ACR: 27.98 dB, 0.84

Fig. 4: A comparison of different model-driven and learned methods for limited-angle CT (with the PSNR (dB) and SSIM scores indicated below). While the FBP reconstruction is noisy, TV fails to preserve important details in the reconstructed image. The LPD method, which was trained on pairs of ground-truth images and limited-angle projection data in this case, does not faithfully reconstruct the image, especially the highlighted regions. One possible reason behind this is that LPD, or any other end-to-end supervised method for that matter, does not necessarily produce a data-consistent reconstruction (see Sec. IV-C for further explanation). The AR leads to artifacts in the reconstructed image, as it favors oscillations in the direction of blurring artifacts in the FBP images. Imposing convexity on the regularizer helps prevent such oscillations and resulting instability, as seen in the images reconstructed by ACR. Notably, unlike LPD and AR, ACR is a convergent regularization scheme and the reconstructions exemplify the importance of this type of theoretical guarantee.

inverse problems, which inspired a recent line of research to analyze their convergence theoretically. Generally, PnP methods, subject to appropriate conditions on the denoiser, have been shown to possess fixed-point and/or objective convergence.

One of the first results on the global objective convergence of PnP-ADMM was shown in [47]. Their theorem requires that the denoiser is continuously differentiable and has a doubly-stochastic gradient matrix, which are equivalent to the denoiser being a proximal operator for some convex function. Fixed-point convergence of PnP-ADMM with a continuation scheme and bounded denoisers was established in [28]. It was shown in [48] that the iterations of PnP-PGD and PnP-ADMM are contractive (and, hence converges to a fixed-point) if the denoiser satisfies a ‘Lipschitz-like’ condition (see Assumption A in [48]). Both objective and fixed-point convergence of PnP-forward-backward splitting (FBS) and PnP-ADMM were proved in [49] for linear denoisers $D(x) = \boldsymbol{W}x$, where $\boldsymbol{W}$ is diagonalizable with eigenvalues in $[0, 1]$. The denoiser scaling approach, which provides a systematic way to control the regularization of PnP denoisers, is shown to be fixed-point convergent using the consensus equilibrium (CE) framework [50].

Typically, the convergence results for PnP methods either assume the denoiser to satisfy a hard Lipschitz bound, or require the data-fidelity term to be strongly-convex in x. While the former is restrictive for deep denoisers, as it affects the denoising performance adversely, the latter assumption excludes inverse problems where the forward operator has a non-trivial null-space (e.g., sparse-view CT, or compressed sensing). Recently, convergence guarantees for PnP methods were derived in [29] with gradient-step (GS) denoisers, alleviating the need for such restrictive assumptions. GS denoisers are constructed as $D_\sigma = \text{Id} - \nabla g_\sigma$, where $g_\sigma(x) = \frac{1}{2}\left\|x - \mathcal{P}_\sigma(x)\right\|_2^2$, with $\mathcal{P}_\sigma$ being a deep network without any structural constraints. This parametrization was shown to have enough expressive power to achieve state-of-the-art denoising performance in [29]. Notably, GS denoisers have a scalar potential given by $h_\sigma(x) = \frac{1}{2}\left\|x\right\|_2^2 - g_\sigma(x)$. When the potential h_σ is convex, GS denoisers are proximal operators corresponding to a potentially non-convex function. Therefore, this approach can target to minimize a variational objective with a potentially non-convex regularizer.

A closely related PnP approach was adopted in [51] with a similar aim of providing an asymptotic characterization of iterative PnP solutions. The main idea was to model maximally monotone operators (MMO) using a NN and interpret the reconstruction as the solution of a monotone inclusion problem (which generalizes convex optimization problems). The parametrization of MMOs was done by modeling the resolvent via a non-expansive NN.

Regularization-by-denoising (RED) is another prominent PnP approach that utilizes an off-the-shelf denoiser $D(x)$ to construct an explicit regularizer $\mathcal{S}(x) = \lambda \cdot x^\top \left(x - D(x)\right)$. If the denoiser is such that the gradient condition $\nabla\mathcal{S}(x) = \lambda \cdot (x - D(x))$ holds, the RED algorithms recover the stationary point of $x \mapsto \frac{1}{2}\|\mathcal{A}x - y\|_2^2 + \mathcal{S}(x)$. It was shown in [52] that the gradient condition does not hold for denoisers with a non-symmetric Jacobian, which is the case for most practical (classical or data-driven). A new analysis framework based on score-matching was developed in [52] to explain the empirical success of the RED algorithms. Specifically, if the denoiser is such that $\frac{D(x)-x}{\epsilon}$ approximates the score function (i.e., the gradient of the log prior), the stationarity condition for the MAP estimation problem (with the prior replaced by a smooth surrogate) leads to the following fixed-point equation (see Sec. IV.C in [52]): $\frac{1}{2}\mathcal{A}^\top\left(\mathcal{A}x^* - y\right) + \lambda\left(x^* - D(x^*)\right) = 0,$ where $\lambda = \frac{\sigma^2}{\epsilon}$, with σ^2 being the variance of measurement noise. The equation above is identical to the fixed-point equation that the RED algorithm seeks to recover. Here, the denoiser D is not required to have a symmetric Jacobian. With this new interpretation, several variants of the RED algorithm were proposed in [52] with fixed-point convergence.

Objective convergence of PnP with GS denoisers [29]

The convergence of PnP denoisers used with half-quadratic splitting (HQS) was established in [29]. The denoiser is constructed as a GS denoiser as explained in Sec. IV-D, i.e., $D_\sigma = \text{Id} - \nabla g_\sigma$, where g_σ is proper, lower semi-continuous, and differentiable with an L-Lipschitz gradient. The PnP algorithm proposed in [29] takes the form $x_{k+1} = \text{prox}_{\tau\, f}\left(x_k - \tau\, \lambda\, \nabla g_\sigma(x_k)\right)$, where $f\colon \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ measures the data-fidelity and is assumed to be convex and lower semi-continuous. Under these assumptions on f and g_σ, the following guarantees hold for $\tau < \frac{1}{\lambda\, L}$:

1) The sequence $F(x_k)$, where $F = f + \lambda\, g_\sigma$, is non-increasing and convergent.
2) $\|x_{k+1} - x_k\|_2 \to 0$, which indicates that iterations are stable, in the sense that they do not diverge if one iterates indefinitely.
3) All limit points of $\{x_k\}$ are stationary points of $F(x)$.

Fixed-point convergence of PnP with Douglas–Rachford splitting (PnP-DRS) [48]

Consider the PnP-DRS algorithm, given by

$$x_{k+\frac{1}{2}} = \text{prox}_{\tau\, f}\left(z_k\right), x_{k+1} = D_\sigma\left(2x_{k+\frac{1}{2}} - z_k\right), \text{ and } z_{k+1} = z_k + x_{k+1} - x_{k+\frac{1}{2}}. \tag{11}$$

Here, f denotes the data-fidelity term and is assumed to be μ-strongly convex. One can equivalently express (11) as the fixed-point iteration $z_{k+1} = \mathcal{T}(z_k)$, where

$$\mathcal{T} = \frac{1}{2}\text{Id} + \frac{1}{2}\left(2D_\sigma - \text{Id}\right)\left(2\text{prox}_{\tau\, f} - \text{Id}\right). \tag{12}$$

Suppose, the denoiser satisfies

$$\left\|\left(D_\sigma - \text{Id}\right)(u) - \left(D_\sigma - \text{Id}\right)(v)\right\|_2 \leq \epsilon\, \|u - v\|_2\,, \tag{13}$$

for all $u, v \in \mathbb{X}$ and some $\epsilon > 0$. It was shown in [48] that if the strong convexity parameter μ is such that $\dfrac{\epsilon}{(1+\epsilon-2\epsilon^2)\,\mu} < \tau$ holds, the operator $\mathcal{T}$ is contractive and the PnP-DRS algorithm is fixed-point convergent. That is, $(x_k, z_k) \to (x_\infty, z_\infty)$, where (x_∞, z_∞) satisfy

$$x_\infty = \text{prox}_{\tau\, f}\left(z_\infty\right) \text{ and } x_\infty = D_\sigma\left(2x_\infty - z_\infty\right). \tag{14}$$

As remarked in [48], the convergence of PnP-DRS follows from monotone operator theory if $(2D_\sigma - \text{Id})$ is non-expansive, but (13) imposes a less restrictive condition on the denoiser.

E. Learned optimization solvers

Reconstruction in imaging inverse problems is framed as an optimization problem as in (1), which could be computationally demanding to solve, especially when the image lives in a high-dimensional vector space. Some recent works [21] have developed data-driven solvers with convergence guarantees for minimizing convex variational objectives. They seek to learn a solver for a family of optimization problems of the form $\min\limits_{x\in\mathbb{X}} F_y(x)$, parametrized by y. In the context of inverse problems, the functional $F_y(x)$ is the variational objective defined in (1). The key idea is to build a parametric solver of the form $\mathcal{T}_{N,\theta}\colon \mathbb{Y} \to \mathbb{X}$ by unrolling a fixed number of iterations (denoted as N) of a gradient-based algorithm. Subsequently, the parameters θ of the solver are learned in an unsupervised manner by minimizing $\frac{1}{n}\sum_{i=1}^{n} F_{y_i}\left(\mathcal{T}_{N,\theta}(y_i)\right)$ over θ, where $(y_i)_{i=1}^{n}$ are n i.i.d. samples drawn from the marginal distribution of the data y. The iterative solver is constructed by adding a neural network-based deviation term to the gradient-based update, and convergence is shown in the case where the deviation term lies in an appropriately defined set. In practice, such learned solvers converge significantly faster than a conventional first-order solver with suitably chosen step-size parameters. See Sections 3 and 4 in [21] for more technical details about the construction and convergence proof of learned optimization solvers. Similar provably convergent data-driven optimization solvers based on mirror-descent with an ICNN-based Bregman distance were recently proposed in [53].

V. Provable learned Bayesian methods

A. *Bayesian inference in imaging inverse problems*

As explained in Sec. II, the Bayesian framework represents the unknown image x^* as a realization of a random variable $\mathbb{x}$ taking values in $\mathbb{X}$ according to a prior distribution with density $p(x)$. The measured data is modeled by a $\mathbb{Y}$-valued random variable $\mathbb{y}$ that is related to $\mathbb{x}$ through the measurement equation $\mathbb{y} = \mathcal{A}\,\mathbb{x} + \mathbb{e}$. The observed data $y \in \mathbb{Y}$ is then understood as a realization of $(\mathbb{y}|\mathbb{x} = x^*)$. The statistical representation of this forward model is given by the conditional density $p(y|x)$, which is known in the literature as the data likelihood function. Given the prior and the likelihood, one can use Bayes' theorem to derive the posterior distribution for $(\mathbb{x}|\mathbb{y} = y)$ with density given by [54]

$$p(x|y) = \frac{p(y|x)p(x)}{\int_{\mathbb{X}} p(y|\tilde{x})p(\tilde{x})\mathrm{d}\tilde{x}}. \tag{15}$$

The posterior distribution describes how the probability mass is distributed over the solution space $\mathbb{X}$ and plays a central role in Bayesian inversion. The posterior distribution not only underpins Bayesian estimators such as the minimum mean-squared error (MMSE) estimator, given by the posterior mean $\mathrm{E}[\mathbb{x}|\mathbb{y} = y] = \displaystyle\int_{\mathbb{X}} x\, p(x|y)\,\mathrm{d}x$, but also has an important role in uncertainty quantification and model selection techniques [54].

From a computational perspective, Bayesian inference often requires computing posterior probabilities and expectations, which is challenging because of the high-dimensional integrals involved. Bayesian computational algorithms based on stochastic sampling address this difficulty by using Monte Carlo integration, wherein one constructs a sequence of random variables $\{\tilde{\mathbb{x}}_i\}_{i=1}^{m}$ such that averages computed along the sequence coincide with the desired probabilities and expectations as $m \to \infty$. For instance, the posterior mean is approximated by the Monte Carlo estimator $\mathrm{E}[\mathbb{x}|\mathbb{y} = y] \approx \dfrac{1}{m}\displaystyle\sum_{i=1}^{m} \tilde{\mathbb{x}}_i$, with the approximation error vanishing as $m \to \infty$. The same strategy is used to compute posterior probabilities and other quantities of interest, e.g., higher-order statistical moments that can be used for uncertainty quantification [55].

Constructing a sequence $\{\tilde{\mathbb{x}}_i\}_{i=1}^m$ leading to provably fast converging Monte Carlo estimators is challenging when the dimension of $\mathbb{X}$ is large. The main approach in imaging is to construct such a sequence iteratively, as a Markov chain, by using a stochastic update rule that contracts the iterates towards $(\mathbb{x}|\mathbb{y} = y)$ and has it as its unique fixed point. In philosophy, this is similar to iterative optimization schemes that contract iterates towards a fixed-point of interest, except for the important distinction that one works with random variables here. This approach is known as Markov chain Monte Carlo (MCMC) [55] and plays an instrumental role in Bayesian inferencing.

Convergence results for MCMC algorithms characterize how quickly the iterates contract towards $(\mathbb{x}|\mathbb{y} = y)$ w.r.t. a given probability measure distance, with different distances describing different forms of convergence. MCMC algorithms that contract geometrically fast as m increases are particularly interesting for Bayesian computation. A detailed convergence analysis seeks to explicitly bound the distance to $(\mathbb{x}|\mathbb{y} = y)$ as a function of the number of iterations m and the initialization, and also characterizes the dependence of the geometric contraction rate on various key aspects of the problem (e.g., dimension, conditioning, and tail behavior). In principle, such results allow bounding the number of iterations m that are required to reach a desired numerical precision, but the bounds are too loose to be useful in practice. Instead, m is often set by monitoring the quantities of interest and stopping the algorithm once these quantities are sufficiently stable. It is also recommended to discard the first 10% of the Markov chain to reduce the initialization bias. Improving MCMC convergence theory to provide sharper bounds that are useful for setting m is an active research area.

Moreover, while conventional Bayesian computation approaches are asymptotically unbiased, i.e., they converge exactly to $(\mathbb{x}|\mathbb{y} = y)$, modern large-scale strategies often accept some bias, by allowing convergence to a controlled neighborhood of $(\mathbb{x}|\mathbb{y} = y)$ to achieve significantly faster convergence rates. Similarly as optimization methods, the convergence properties of MCMC methods depend crucially on the regularity properties of the posterior $p(x|y)$ (e.g., smoothness, convexity, and tail behavior). Checking that $p(x|y)$ satisfies relevant regularity properties is challenging for Bayesian modeling strategies that use learned priors encoded by neural networks, which we will discuss later in this section.

Distances between probability measures for studying convergence of Markov chains

To define probability measures, one needs an underlying Borel-measurable space $(\mathbb{X}, \mathcal{B}(\mathbb{X}))$, where $\mathcal{B}(\mathbb{X})$ is the Borel σ-algebra of $\mathbb{X}$. It is the smallest σ-algebra generated by the open subsets of $\mathbb{X}$ and contains all those subsets of $\mathbb{X}$ to which one can assign probabilities consistently using a probability measure. Any probability measure on $(\mathbb{X}, \mathcal{B}(\mathbb{X}))$ is formally defined as a function that maps $\mathcal{B}(\mathbb{X})$ to the interval $[0, 1]$.

Consider two such probability measures π_1 and π_2 defined on $(\mathbb{X}, \mathcal{B}(\mathbb{X}))$. A notion of distance between π_1 and π_2 is generally of the form

$$\mathcal{D}(\pi_1, \pi_2) = \sup_{f \in \mathscr{F}} \left| \int f \mathrm{d}\pi_1 - \int \mathcal{Q} f \mathrm{d}\pi_2 \right|, \tag{16}$$

where $\mathscr{F}$ is some class of functions defined on $\mathbb{X}$ and $\mathcal{Q} : \mathscr{F} \mapsto \mathscr{F}$.

When $\mathscr{F} = \{\mathbb{I}_{\mathscr{A}}; \mathscr{A} \in \mathcal{B}(\mathbb{X})\}$ is the class of all indicator functions of the Borel subsets of $\mathbb{X}$ and $\mathcal{Q}f = f$, (16) leads to the well-known total variation (TV) distance given by

$$\mathrm{TV}(\pi_1, \pi_2) = \sup_{\mathscr{A} \in \mathcal{B}(\mathbb{X})} |\pi_1(A) - \pi_2(A)| \, . \tag{17}$$

In other words, the total-variation distance measures the largest deviation in the probabilities assigned by the two measures to any event $\mathscr{A} \in \mathcal{B}(\mathbb{X})$. For Markov chain Monte Carlo methods, convergence in TV metric implies that probabilities computed from the Markov chain converge to the posterior probabilities of interest.

In the case where $\mathscr{F}$ contains all 1-Lipschitz functions on $\mathbb{X}$ and $\mathcal{Q}f = f$, (16) recovers the Wasserstein-1 distance between π_1 and π_2. Similarly, (16) leads to the Wasserstein-2 distance between π_1 and π_2 when $\mathscr{F}$ contains all bounded continuous functions on $\mathbb{X}$ and

$$(\mathcal{Q}f)(x) = \sup_{u \in \mathbb{X}} f(u) - \|u - x\|_2^2,$$

for any $x \in \mathbb{X}$. Wasserstein distances play a complementary role to the TV distance in the analysis of MCMC methods. Unlike TV, convergence in a Wasserstein-p distance guarantees the convergence of expectations of bounded continuous functions, as well as the convergence of the first p statistical moments, but not convergence of probabilities.

Convergence issues aside, theoretical analysis of a Bayesian imaging procedure might also seek to prove that the posterior distribution associated with $(\mathrm{x}|\mathrm{y} = y)$ is well-posed, and that the posterior quantities of interest exist and inherit this stability. A Bayesian inverse problem is said to be *well-posed* if the posterior distribution of $(\mathrm{x}|\mathrm{y} = y)$ is well-defined and unique, and varies continuously w.r.t. y under a given distance metric $\mathcal{D}$ (e.g. the total-variation (TV) or the Wasserstein-p distances). Formally, we consider two data realizations $y_1 \in \mathbb{Y}$ and $y_2 \in \mathbb{Y}$ and denote by π_1 and π_2 the two probability measures related to the posterior distributions of $(\mathrm{x}|\mathrm{y} = y_1)$ and $(\mathrm{x}|\mathrm{y} = y_2)$. Stability in the Bayesian setting requires that $\mathcal{D}(\pi_1, \pi_2)$ is a continuous function w.r.t. $(y_1, y_2) \in \mathbb{Y} \times \mathbb{Y}$. Different choices for $\mathcal{D}$ can capture different forms of stability: well-posedness in the TV distance guarantees that the posterior probabilities are stable w.r.t. perturbations in y, whereas well-posedness in a Wasserstein-p distance describes the stability of expectations of bounded continuous functions (but not probabilities), as well as the stability of the first p posterior moments [56].

B. Bayesian inference with learned priors

The recent advances in data-driven modeling have inspired Bayesian strategies similarly as for variational regularization. In this case, instead of hand-crafting a plausible prior distribution, we now assume to have access to i.i.d. samples $\{x_i\}_{i=1}^n$ from the *true* marginal distribution of $\mathbb{x}$, say for instance, by taking MR scans of randomly sampled people on the street. We henceforth denote by $p^\star(x)$ the density associated with this marginal distribution, and by $p^\star(x|y)$ the corresponding posterior density resulting from Bayes' theorem. We view $p^\star(x)$ and $p^\star(x|y)$ as the *true* prior and posterior, respectively, inasmuch they represent how nature assigns probability mass for $\mathbb{x}$. Unfortunately, it is not possible to perform inference directly with $p^\star(x|y)$ because the prior is only known through representative samples. Moreover, the lack of an analytical expression for $p^\star(x)$ means that we cannot guarantee that $p^\star(x|y)$ is well-posed and that the quantities of interest exist and are stable w.r.t. perturbations of y, or that the regularity conditions required for efficient gradient-based MCMC computation are satisfied. Learned Bayesian imaging methods use the available samples to construct an approximation of $p^\star(x|y)$. In the following we focus on theoretically rigorous strategies for this task that are by construction well-posed and amenable to provably convergent computation[1]. Specifically, we discuss two strategies: a) PnP Bayesian methods that encode the prior distribution in the form of an end-to-end denoising neural network that is used within a Bayesian computation algorithm, and b) algorithms that rely on a generative model trained to reproduce the prior distribution from its samples, such as generative adversarial networks (GANs) or variational auto-encoders (VAEs).

a) Bayesian methods with plug-and-play priors: The aim is now to learn a prior by exploiting its relation to a learned denoiser, similarly as in Sec. IV-D. For that purpose, we denote by $D_\sigma^\star : \mathbb{X} \mapsto \mathbb{X}$ the optimal minimum mean-squared-error (MMSE) denoiser that estimates $\mathbb{x}$ from its noisy observation $\mathbb{u} = \mathbb{x} + \sigma\mathbb{z}$, where $\mathbb{z}$ is a standard Gaussian random variable. From Bayesian decision theory, $D_\sigma^\star$ is given by $D_\sigma^\star(u) = \mathrm{E}[\mathbb{x}|\mathbb{u} = u]$ for any $u \in \mathbb{X}$, where the expectation is computed under the assumption that $\mathbb{x}$ has marginal density $p^\star$. In practice, $D_\sigma^\star$ is of course unknown, because $p^\star$ is unknown. However, it can be approximated using a deep neural network D_σ trained on the available samples from $p^\star$.

Plug-and-play (PnP) Bayesian methods now use this denoiser D_σ trained to mimic $D_\sigma^\star$ in order to perform approximate inference w.r.t. the oracle $p^\star(x|y)$. More precisely, this can be achieved by mimicking gradient-based Bayesian computation algorithms that target a regularized approximation of $p^\star(x|y)$, which, by construction, verifies the regularity properties required for fast convergence [59], [60]. In particular, PnP Bayesian methods stem from the observation that $D_\sigma^\star$ is related to $p^\star$ by Tweedie's identity:

$$\sigma^2 \nabla \log p_\sigma^\star(x) = D_\sigma^\star(x) - x,$$

[1] Here we focus on non-asymptotic convergence results that can be applied to a broad class of models. There are other results, e.g., [57], [58], that provide some guarantees for approximate message passing algorithms, but they require stronger assumptions on the forward model and they only hold asymptotically, in a limit where the dimension of $\mathbb{X}$ and $\mathbb{Y}$ diverge in a specific way.

for all $x \in \mathbb{X}$ and $\sigma > 0$, where $p^\star_\sigma$ is a regularized approximation of $p^\star$ obtained via the convolution of $p^\star$ with a Gaussian smoothing kernel of bandwidth σ. Unlike $p^\star$ which may be degenerate or non-smooth, $p^\star_\sigma$ is by construction proper and smooth, with its gradient $x \mapsto \nabla \log p^\star_\sigma(x)$ being globally Lipschitz continuous under mild conditions [59]. Also, $p^\star_\sigma(x)$ can be made arbitrarily close to $p^\star(x)$ by reducing σ to control the approximation error involved.

Equipped with this regularized prior, PnP Bayesian methods use Bayes' theorem to derive the regularized posterior density $p^\star_\sigma(x|y) \propto p(y|x)p^\star_\sigma(x)$. Under gentle assumptions on the likelihood $p(y|x)$, $p^\star_\sigma(x|y)$ inherits the favorable regularity properties of $p^\star_\sigma(x)$ and provides an approximation to $p^\star(x|y)$ that is amenable to efficient computation by gradient-based algorithms such as the unadjusted Langevin algorithm (ULA) and stochastic gradient-descent (SGD) [59]. By controlling σ, the approximation $p^\star_\sigma(x|y)$ can be made as close to $p^\star(x|y)$ as required in order to control the estimation bias. This, however, comes at the expense of additional computation due to slower convergence of gradient-based algorithms. In addition, [59] provides verifiable conditions that guarantee that $p^\star_\sigma(x|y)$ is well-posed w.r.t. the TV distance, and key quantities such as the posterior moments exist.

With regards to the maximum a-posteriori probability (MAP) estimation for $p^\star_\sigma(x|y)$, defined as

$$\hat{x}_{\text{map}} \in \arg\max_{x \in \mathbb{X}} \, p^\star_\sigma(x|y),$$

it is worth mentioning two main results from [60] before discussing practical computational issues. First, MAP solutions of $p^\star_\sigma(x|y)$ lie in a neighborhood of MAP solutions of $p^\star(x|y)$ and they vary in a stable manner w.r.t. σ, with the two sets of solutions coinciding as $\sigma \to 0$. Second, MAP solutions of $p^\star_\sigma(x|y)$ are locally Lipschitz continuous w.r.t. to perturbations in $y \in \mathbb{Y}$, which is a weak form of well-posedness (see [60] for details).

From a Bayesian computation viewpoint, the main theoretical insight underpinning PnP Bayesian methods such as the PnP-ULA and PnP-SGD studied in [59], [60] is that one can use $\nabla \log p^\star_\sigma$ to formulate idealized ULA and SGD algorithms for inference w.r.t. $p^\star_\sigma$, and subsequently substitute $\nabla \log p^\star_\sigma(x) = (D^\star_\sigma(x) - x)/\sigma^2$ within these algorithms by the learned approximation $(D_\sigma(x) - x)/\sigma^2$ without significantly affecting their convergence properties, even if $D_\sigma(x)$ is not a gradient or a maximally monotone operator. Approximating the oracle denoiser $D^\star_\sigma$ by a learned denoiser D_σ introduces some bias; i.e., the algorithms produce a solution in the neighborhood of the oracle solution that would be produced by the idealized algorithms. The magnitude of this bias w.r.t. the oracle depends primarily on how close D_σ is to $D^\star_\sigma$.

The plug-and-play unadjusted Langevin algorithm (PnP-ULA) and stochastic gradient-descent (PnP-SGD)

PnP-ULA to sample from the regularized posterior $p^\star_\sigma(x|y)$ is defined by the following recursion

[59], where $k \in \mathbb{N}$:

$$\tilde{\mathbbm{x}}_{k+1} = \tilde{\mathbbm{x}}_k + \delta \nabla \log p(y|\tilde{\mathbbm{x}}_k) + \frac{\delta}{\sigma^2}\left[D_\sigma(\tilde{\mathbbm{x}}_k) - \tilde{\mathbbm{x}}_k\right] + \frac{\delta}{\lambda}\left[\Pi_{\mathrm{C}}(\tilde{\mathbbm{x}}_k) - \tilde{\mathbbm{x}}_k\right] + \sqrt{2\delta}\, \mathbbm{z}_{k+1}. \tag{18}$$

Here, δ is a step-size, λ is a tail regularization parameter, $\mathrm{C} \subset \mathbb{X}$ denotes a compact convex set that contains most of the prior probability mass of $\mathbbm{x}$, Π_{C} is the projection operator onto C, and $\{\mathbbm{z}_k\}_{k\in\mathbb{N}}$ are i.i.d. standard Gaussian random variables.

For maximum a-posteriori probability (MAP) estimation for $p^\star(x|y)$, the PnP-SGD algorithm is defined by the following recursion [60], for $k \in \mathbb{N}$:

$$\tilde{\mathbbm{x}}_{k+1} = \tilde{\mathbbm{x}}_k + \delta_k \nabla \log p(y|\tilde{\mathbbm{x}}_k) + \frac{\delta_k}{\sigma^2}\left[D_\sigma(\tilde{\mathbbm{x}}_k) - \tilde{\mathbbm{x}}_k\right] + \delta_k \mathbbm{z}_{k+1}\,, \tag{19}$$

where $\{\delta_k\}_{k\in\mathbb{N}}$ is a family of decreasing positive step-sizes and $\{\mathbbm{z}_k\}_{k\in\mathbb{N}}$ are again i.i.d. standard Gaussian random variables.

Convergence guarantees for these two algorithms require that the denoiser satisfies the same condition as in (13), for all $u, v \in \mathbb{X}$, and for some $\epsilon > 0$. In addition, to characterize the error introduced by approximating $D^\star_\sigma$ through D_σ, one needs that for any $R > 0$ there exists $M_R \geq 0$ such that $\|D_\sigma(u) - D^\star_\sigma(u)\|_2 \leq M_R$ for all $u \in \mathbb{X}$ with $\|u\|_2 < R$. Further, the likelihood $p(y|x)$ is assumed to be finite and differentiable w.r.t. x, with $\nabla \log p(y|x)$ being L_y-Lipschitz. Then, for any $\delta < \frac{1}{3}\left(\frac{\epsilon}{\sigma} + L_y + \frac{1}{\lambda}\right)$ and $\lambda \in \left(0, \frac{1}{2}\left(\frac{\epsilon}{\sigma} + 2L_y\right)\right)$, the Markov chain generated by (18) converges geometrically fast to a neighborhood of $p^\star_\sigma(x|y)$, in TV and Wasserstein-1 distances. The asymptotic bias and the convergence rate depend on $\delta, \lambda, \mathrm{C}, M_R$ and the dimension of $\mathbb{X}$ (see [59, Section 3.2] for details).

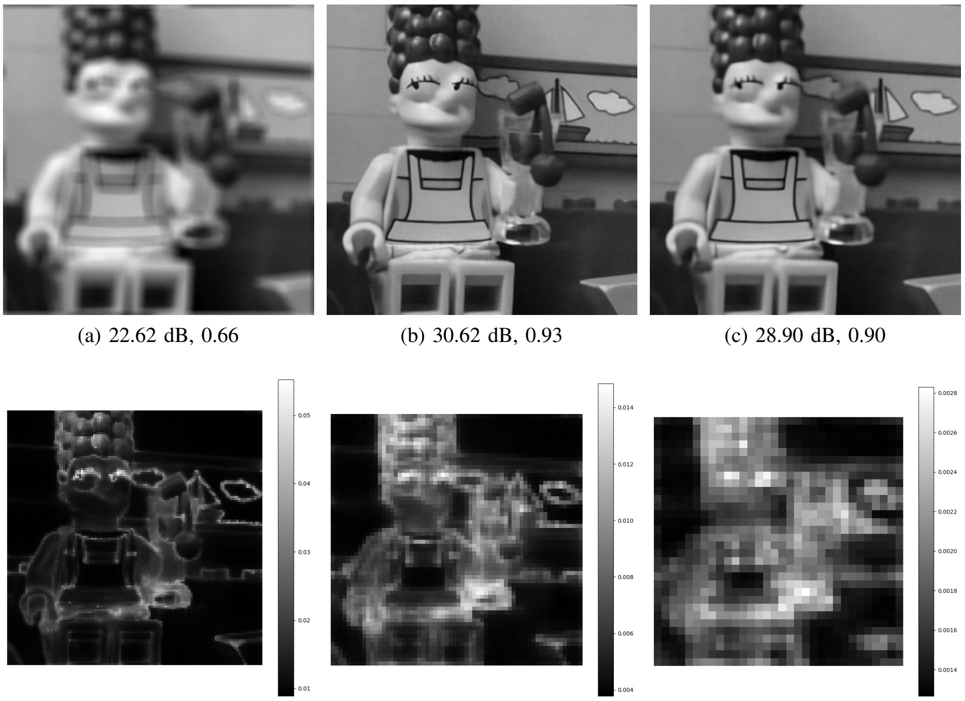
For the PnP-SGD algorithm in (19), assume that the step-sizes satisfy

$$\lim_{k\to\infty} \delta_k = 0\,, \quad \sum_{k=0}^{\infty} \delta_k = +\infty\,, \quad \sum_{k=0}^{\infty} \delta_k^2 < +\infty\,.$$

Then, it is shown in [60] that all *stable* sequences generated by (19) converge to a neighborhood of the stationary points of $\nabla \log p^\star_\sigma(x|y)$, and the size of the neighborhood is controlled by M_R (see [60, Proposition 3] for technical details). Whether all the sequences generated by (19) are stable is an open question.

For illustration, Fig. 5 presents the results of an image deblurring experiment where PnP-ULA and PnP-SGD are used to compute the MMSE and MAP estimators, respectively, using the denoiser of [48] with $\sigma = 5/255$. The main strength of Monte Carlo algorithms such as PnP-ULA is their capacity for Bayesian analyses beyond point estimation, such as uncertainty quantification, as depicted in the bottom row of Fig. 5.

b) Bayesian methods with generative priors: A prominent alternative to plug-and-play denoising is to construct a prior distribution for $\mathbbm{x}$ by leveraging recent developments in deep generative modeling,



(a) 22.62 dB, 0.66 (b) 30.62 dB, 0.93 (c) 28.90 dB, 0.90

(d) scale 2×2 (e) scale 4×4 (f) scale 8×8

Fig. 5: Image deblurring using the denoiser [48] in a PnP fashion. Top row shows data (left), a blurred 256×256 image with additive Gaussian noise with standard deviation $1/255$, along with MMSE solution from PnP-ULA (middle) and MAP solution from PnP-SGD (right). The corresponding PSNR (dB) and SSIM values are indicated below the images. The bottom row shows uncertainty plots at the scales of 2×2 (left), 4×4 (middle), and 8×8 pixels (right), computed by PnP-ULA. See [59] for details.

such as VAEs, GANs, and normalizing flows. The Bayesian approach in [61] adopts the manifold hypothesis to construct a prior distribution that is supported on a sub-manifold of $\mathbb{X}$, the dimension of which is much smaller than the ambient dimension of $\mathbb{X}$. Operating on this manifold of dramatically reduced dimensionality simultaneously allows to effectively regularize the posterior and perform computations efficiently. The reduction in dimensionality is achieved by introducing a latent Gaussian variable $\mathbbm{z} \sim \mathcal{N}(0, \mathbb{I}_d)$ on $\mathbb{R}^d$ and a deep generative network $\mathcal{G}_\theta \colon \mathbb{R}^d \mapsto \mathbb{X}$ designed such that the random variable $\mathcal{G}_\theta(\mathbbm{z})$ is close in distribution to $\mathbbm{x}$. In [61], $\mathcal{G}_\theta$ is derived from a VAE architecture, but the strategy is agnostic to the choice of the specific generative model, and a GAN or a normalizing flow-based model could be used instead. Given this construction, the latent posterior is derived as $p(z|y) \propto p(y|z)p(z)$, where $p(z)$ is a standard Gaussian density on $\mathbb{R}^d$ and the likelihood is $p(y|z) = p_{\mathbbm{y}|\mathbbm{x}}(y|\,\mathcal{G}_\theta(z))$. The posterior distribution of $(\mathbbm{x}|\mathbbm{y} = y)$ is then given by using $\mathcal{G}_\theta$ to map $(\mathbbm{z}|\mathbbm{y} = y)$ onto $\mathbb{X}$. The resulting Bayesian inversion is shown to be well-posed under mild conditions on the likelihood, and important quantities such as the posterior mean exist.

Well-posedness of linear Bayesian inverse problems with generative priors

Consider the generative model $\mathbbm{x} = \mathcal{G}_\theta(\mathbbm{z})$ where $\mathbbm{z} \sim \mathcal{N}(0, \mathbb{I}_d)$ is a latent random variable and $\mathcal{G}_\theta \colon \mathbb{R}^d \mapsto \mathbb{X} \subseteq \mathbb{R}^p$ is a Lipschitz-continuous neural network. Suppose that the observed data $y \in \mathbb{R}^n$ is a realization of a random variable $\mathbb{y} = \mathcal{A}\mathcal{G}_\theta(\mathbb{z}) + \mathbb{e}$, where $\mathcal{A} \in \mathbb{R}^{n \times p}$ and $\mathbb{e}$ is Gaussian noise. Then, it is shown in [61] that the posterior distributions associated with $(\mathbb{z}|\mathbb{y} = y)$ and $(\mathbb{x}|\mathbb{y} = y)$ are well-posed w.r.t. the TV and Wasserstein-2 distances. Moreover, all posterior moments exist, and in particular the minimum mean-squared error (MMSE) Bayesian estimators for $(\mathbb{z}|\mathbb{y} = y)$ and $(\mathbb{x}|\mathbb{y} = y)$ are well-posed.

With regards to computation, [61] takes advantage of the fact that $\mathbb{z}$ is relatively low-dimensional and has a Gaussian prior to generate samples for $(\mathbb{z}|\mathbb{y} = y)$ by using the following simple MCMC procedure specialized for this class of models: for any $k \in \mathbb{N}$, draw $\mathbb{z}^\dagger \sim \mathcal{N}(\sqrt{1-\delta^2}\mathbb{z}_k, \mathbb{I}_d)$ and set $\mathbb{z}_{k+1} = \mathbb{z}^\dagger$ with probability $\min(1, p(y|\mathbb{z}^\dagger)/p(y|\mathbb{z}_k))$; otherwise set $\mathbb{z}_{k+1} = \mathbb{z}_k$. This MCMC algorithm, known as the preconditioned Crank-Nicolson algorithm, is provably convergent under mild assumptions on $p(y|z)$ that are verified in particular for linear Gaussian observation models of the form $\mathbb{y} = \mathcal{A}\mathcal{G}_\theta(\mathbb{z}) + \mathbb{e}$. Given Monte Carlo samples for the latent variable $(\mathbb{z}|\mathbb{y} = y)$, samples for $(\mathbb{x}|\mathbb{y} = y)$ are obtained by applying the generator $\mathcal{G}_\theta$.

A Bayesian model with a prior encoded by a VAE was considered in [62], with an objective of MAP estimation. Unlike the approach in [61], where one constraints $\mathbb{x}$ to take values in the range of $\mathcal{G}_\theta$ in order to reduce dimensionality, [62] considers an augmented model $p(x, z)$ on $\mathbb{X} \times \mathbb{R}^d$ that concentrates mass in the neighborhood of $x = \mathcal{G}_\theta(z)$ while carefully allowing for deviations from this sub-manifold to better fit the training images. This leads to an augmented posterior distribution $p(x, z|y) \propto p(y|x)p(x, z)$ that is a more accurate model than the marginal posterior model considered in [61]. Inference with $p(x, z|y)$ is significantly more computationally challenging, a difficulty that was addressed by focusing exclusively on MAP estimation. Crucially, the authors established that the potential $(x, z) \mapsto -\log p(x, z|y)$ is weakly bi-convex under realistic conditions on the VAE, and subsequently proposed three provably convergent alternating optimization schemes to compute a critical point of this potential efficiently.

VI. Conclusions and Outlook

In scientific disciplines where imaging drives new discoveries or in real-world applications where imaging is used for making critical decisions, it is essential to have mathematical correctness guarantees for the algorithms used for image recovery. While the classical variational approaches come with such certificates, they fall short in terms of empirical performance as compared to the modern data-driven imaging algorithms. We formalized different notions of correctness as they apply to image reconstruction methods and surveyed some of the notable deep learning-based approaches, both deterministic and stochastic, that fit within these notions. We discussed some of the essential components, e.g., network architecture design, training strategies, etc. that typically aid deriving such theoretical certificates. While we sought to dispel the widely held belief about the black-box nature of deep learning algorithms for image reconstruction, we also highlighted the gaps in theoretical understanding about well-performing methods rooted in robust heuristics. The methods we reviewed in this article broadly derive their origin from the variational regularization framework and convex analysis, two of the major theoretical pillars that the classical methods rest on. In fact, convexity arose as a recurring theme for proving convergence results in both deterministic and stochastic settings, which underscores the importance of ICNNs for combining classical theory with data-driven learning. In summary, we argued that the classical mathematical machinery can go a long way when it comes to devising and analyzing data-driven methods, leading to better reliability and transparency of deep learning for imaging.

References

[1] O. Scherzer *et al.*, "Variational methods in imaging," 2009.

[2] M. Benning and M. Burger, "Modern regularization methods for inverse problems," *Acta Numerica*, vol. 27, pp. 1–111, 2018.

[3] S. Arridge, P. Maass, O. Öktem, and C.-B. Schönlieb, "Solving inverse problems using data-driven models," *Acta Numerica*, vol. 28, pp. 1–174, 2019.

[4] A. Lucas, M. Iliadis, R. Molina, and A. K. Katsaggelos, "Using deep neural networks for inverse problems in imaging: beyond analytical methods," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 20–36, 2018.

[5] J. Kaipio and E. Somersalo, *Statistical and computational inverse problems*. Springer Science & Business Media, 2006, vol. 160.

[6] M. Pereyra, "Revisiting maximum-a-posteriori estimation in log-concave models," *SIAM Journal on Imaging Sciences*, vol. 12, pp. 650–670, 01 2019.

[7] R. Tibshirani, "Regression shrinkage and selection via the lasso: a retrospective," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 73, no. 3, pp. 273–282, 2011.

[8] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[9] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, pp. 21–30, 2008.

[10] B. Kaltenbacher, A. Neubauer, and O. Scherzer, "Iterative regularization methods for nonlinear ill-posed problems," in *Iterative Regularization Methods for Nonlinear Ill-Posed Problems*. de Gruyter, 2008.

[11] S. Arridge *et al.*, "Approximation errors and model reduction with an application in optical diffusion tomography," *Inverse Problems*, vol. 22, no. 1, p. 175, 2006.

[12] S. Lunz *et al.*, "On learned operator correction in inverse problems," *SIAM Journal on Imaging Sciences*, vol. 14, no. 1, pp. 92–127, 2021.

[13] E. Shimron, J. I. Tamir, K. Wang, and M. Lustig, "Implicit data crimes: Machine learning bias arising from misuse of public data," *Proceedings of the National Academy of Sciences*, vol. 119, no. 13, p. e2117203119, 2022.

[14] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4509–4522, 2017.

[15] J. C. Ye, Y. Han, and E. Cha, "Deep convolutional framelets: A general deep learning framework for inverse problems," *SIAM Journal on Imaging Sciences*, vol. 11, no. 2, pp. 991–1048, 2018.

[16] J. Adler and O. Öktem, "Learned primal-dual reconstruction," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1322–1332, 2018.

[17] K. Hammernik *et al.*, "Learning a variational network for reconstruction of accelerated MRI data," *Magnetic Resonance in Medicine*, vol. 79, no. 6, pp. 3055–3071, 2018.

[18] A. Bora, A. Jalal, E. Price, and A. G. Dimakis, "Compressed sensing using generative models," in *Proceedings of Machine Learning Research*, vol. 70, 2017, proceedings of the 34th International Conference on Machine Learning, Sydney, Australia.

[19] H. Li, J. Schwab, S. Antholzer, and M. Haltmeier, “NETT: solving inverse problems with deep neural networks,” *Inverse Problems*, vol. 36, no. 6, 2020.

[20] S. Lunz, O. Öktem, and C.-B. Schönlieb, “Adversarial regularizers in inverse problems,” in *32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada*, 2018, pp. 8507–8516.

[21] S. Banert, J. Rudzusika, O. Öktem, and J. Adler, “Accelerated forward-backward optimization using deep learning,” *arXiv:2105.05210v1*, 2021.

[22] S. Mukherjee, M. Carioni, O. Öktem, and C.-B. Schönlieb, “End-to-end reconstruction meets data-driven regularization for inverse problems,” in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 21 413–21 425.

[23] S. Mukherjee *et al.*, “Learned convex regularizers for inverse problems,” *arXiv:2008.02839v2*, 2020.

[24] M. Genzel, J. Macdonald, and M. Marz, “Solving inverse problems with deep neural networks - robustness included,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2022.

[25] V. Antun *et al.*, “On instabilities of deep learning in image reconstruction and the potential costs of AI,” *Proceedings of the National Academy of Sciences*, vol. 117, no. 48, pp. 30 088–30 095, 2020.

[26] R. Alaifari, G. S. Alberti, and T. Gauksson, “Localized adversarial artifacts for compressed sensing mri,” *arXiv:2206.05289v1*, 2022.

[27] C. McCollough, “Tfg-207a-04: Overview of the low dose CT grand challenge,” *Medical Physics*, vol. 43, no. 6, pp. 3759–3760, 2016.

[28] S. H. Chan, X. Wang, and O. A. Elgendy, “Plug-and-play ADMM for image restoration: Fixed-point convergence and applications,” *IEEE Transactions on Computational Imaging*, vol. 3, no. 1, pp. 84–98, 2017.

[29] S. Hurault, A. Leclaire, and N. Papadakis, “Gradient step denoiser for convergent plug-and-play,” *CoRR*, vol. abs/2110.03220, 2021. [Online]. Available: https://arxiv.org/abs/2110.03220

[30] B. Amos, L. Xu, and J. Z. Kolter, “Input convex neural networks,” in *International Conference on Machine Learning*, 2017, pp. 146–155.

[31] D. Obmann, L. Nguyen, J. Schwab, and M. Haltmeier, “Augmented NETT regularization of inverse problems,” *Journal of Physics Communications*, vol. 5, no. 10, p. 105002, Oct. 2021.

[32] D. Obmann, J. Schwab, and M. Haltmeier, “Deep synthesis network for regularizing inverse problems,” *Inverse Problems*, vol. 37, no. 1, p. 015005, Dec. 2020.

[33] A. Habring and M. Holler, “A generative variational model for inverse problems in imaging,” *SIAM J. Mathematics of Data Science*, vol. 4, no. 1, pp. 306–335, 2022.

[34] D. Ulyanov, A. Vedaldi, and V. Lempitsky, “Deep image prior,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9446–9454.

[35] G. Jagatap and C. Hegde, “Algorithmic guarantees for inverse imaging with untrained network priors,” in *33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada*, 2019.

[36] A. Raj, Y. Li, and Y. Bresler, “GAN-based projector for faster recovery in compressed sensing with convergence guarantees,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[37] K. Gregor and Y. LeCun, “Learning fast approximations of sparse coding,” in *Intl. Conf. on Machine Learning*, 2010.

[38] J. Adler and O. Öktem, “Deep bayesian inversion,” *arXiv:1811.05910v1*, 2018.

[39] S. Bai, J. Z. Kolter, and V. Koltun, “Deep equilibrium models,” in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[40] D. Gilton, G. Ongie, and R. Willett, “Deep equilibrium architectures for inverse problems in imaging,” *IEEE Transactions on Computational Imaging*, vol. 7, pp. 1123–1133, 2021.

[41] S. W. Fung *et al.*, “Jfb: Jacobian-free backpropagation for implicit networks,” *arXiv:2103.12803v4*, 2021.

[42] Z. Ramzi *et al.*, “SHINE: SHaring the INverse estimate from the forward pass for bi-level optimization and implicit models,” in *International Conference on Learning Representations*, 2022.

[43] J. Tang, S. Mukherjee, and C.-B. Schönlieb, “Stochastic primal-dual deep unrolling,” *arXiv:2110.10093v4*.

[44] M. Hasannasab *et al.*, “Parseval proximal neural networks,” *Journal of Fourier Analysis and Applications*, vol. 26, no. 4, Jul. 2020.

[45] J. Schwab, S. Antholzer, and M. Haltmeier, “Deep null space learning for inverse problems: convergence analysis and rates,” *Inverse Problems*, vol. 35, no. 2, p. 025008, 2019.

[46] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, “Plug-and-play priors for model based reconstruction,” in *2013 IEEE Global Conference on Signal and Information Processing*, 2013, pp. 945–948.

[47] S. Sreehari *et al.*, “Plug-and-play priors for bright field electron tomography and sparse interpolation,” *IEEE Transactions on Computational Imaging*, vol. 2, no. 4, pp. 408–423, 2016.

[48] E. Ryu *et al.*, “Plug-and-play methods provably converge with properly trained denoisers,” in *Proceedings of the 36th International Conference on Machine Learning*, vol. 97. PMLR, 09–15 Jun 2019, pp. 5546–5557.

[49] P. Nair, R. G. Gavaskar, and K. N. Chaudhury, “Fixed-point and objective convergence of plug-and-play algorithms,” *IEEE Transactions on Computational Imaging*, vol. 7, pp. 337–348, 2021.

[50] X. Xu *et al.*, “Boosting the performance of plug-and-play priors via denoiser scaling,” in *54th Asilomar Conference on Signals, Systems, and Computers*, 2020, pp. 1305–1312.

[51] J.-C. Pesquet, A. Repetti, M. Terris, and Y. Wiaux, “Learning maximally monotone operators for image recovery,” *SIAM Journal on Imaging Sciences*, vol. 14, no. 3, pp. 1206–1237, 2021.

[52] E. T. Reehorst and P. Schniter, “Regularization by denoising: clarifications and new interpretations,” *IEEE Transactions on Computational Imaging*, vol. 5, no. 1, pp. 52–67, 2019.

[53] H. Y. Tan, S. Mukherjee, J. Tang, and C.-B. Schönlieb, “Data-driven mirror descent with input-convex neural networks,” *arXiv:2206.06733v1*, 2022.

[54] C. P. Robert, *The Bayesian choice: a decision-theoretic motivation*. Springer-Verlag, 1994.

[55] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. Springer-Verlag New York Inc., 2004.

[56] J. Latz, “On the well-posedness of Bayesian inverse problems,” *SIAM/ASA Journal on Uncertainty Quantification*, vol. 8, no. 1, pp. 451–482, 2020.

[57] A. K. Fletcher *et al.*, “Plug in estimation in high dimensional linear inverse problems a rigorous analysis,” *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2019, no. 12, p. 124021, 2019.

[58] P. Pandit *et al.*, “Inference with deep generative priors in high dimensions,” *IEEE Journal on Selected Areas in Information Theory*, vol. 1, pp. 336–347, 2020.

[59] R. Laumont *et al.*, “Bayesian imaging using plug & play priors: When langevin meets tweedie,” *SIAM Journal on Imaging Sciences*, vol. 15, no. 2, pp. 701–737, 2022.

[60] ——, “On Maximum-a-Posteriori estimation with Plug and Play priors and stochastic gradient descent,” *HAL preprint hal-0334873*, 2021.

[61] M. Holden, M. Pereyra, and K. C. Zygalakis, “Bayesian imaging with data-driven priors encoded by neural networks: Theory, methods, and algorithms,” *SIAM J. Imaging Sciences*, 2021, to appear.

[62] M. González, A. Almansa, and P. Tan, “Solving inverse problems by joint posterior maximization with autoencoding prior,” *arXiv:2103.01648v3*, 2021.

Subhadip Mukherjee is currently a Lecturer (Assistant Professor) of machine learning and AI at the Department of Computer Science, University of Bath, UK. Prior to this, he held two postdoctoral positions, at the Department of Applied Mathematics and Theoretical Physics, University of Cambridge, UK (2020-2022); and at the Department of Mathematics, KTH, Sweden (2018-2020). He completed his Ph.D. in 2018 from the Department of Electrical Engineering, Indian Institute of Science, Bangalore, specializing in sparsity-regularized inverse problems. His research interests include areas at the interface of machine learning, inverse problems, convex optimization, and statistics. Specifically, he is interested in developing novel machine learning algorithms for computational imaging problems.

Andreas Hauptmann is Academy Research Fellow and Associate Professor of Computational Mathematics at the Research Unit of Mathematical Sciences, University of Oulu, Finland. He has worked prior to this as Research Associate at the Department of Computer Science, University College London, UK. He received his Ph.D. in 2017 in Applied Mathematics from the University of Helsinki, Finland. His research interest is in combining model-based inversion techniques with data driven methods for tomographic reconstructions.

Ozan Öktem is a Professor in Computational Science at the Department of Information Technology, Uppsala University and Associate Professor in Numerical Analysis at the Department of Mathematics, KTH - Royal Institute of Technology, Stockholm, Sweden. He received his Ph.D. in 1999 in Mathematics from Stockholm University, Sweden. He worked for 13 years in industry before returning to academia in 2009. His recent focus is on combining model based approaches with DNNs for uncertainty quantification and task adapted reconstruction in large scale inverse problems, with concrete challenges in imaging applications from various scientific fields.

Marcelo Pereyra is an Associate Professor at the Maxwell Institute for Mathematical Sciences and the School of Mathematical and Computer Sciences, Heriot-Watt University. He obtained a Ph.D. degree in Signal Processing from the University of Toulouse in 2012 and was a Research Fellow in Statistics at the University of Bristol (2012 - 2016), funded by a Marie Curie Intra-European Fellowship for Career Development, a Brunel Postdoctoral Research Fellowship in Statistics, and a Postdoctoral Research Fellowship from French Ministry of Defence. In 2019 he was an Invited Professor at the Institut Henri Poincaré in Paris during the "Mathematics of Imaging" trimester. His research focuses on new Bayesian statistical theory, methodology and algorithms to solve challenging inverse problems related to computational imaging.

Carola-Bibiane Schönlieb is Professor of Applied Mathematics at the University of Cambridge, where she is head of the Cambridge Image Analysis group and co-Director of the EPSRC Cambridge Mathematics of Information in Healthcare Hub. Carola graduated from the Institute for Mathematics, University of Salzburg (Austria) in 2004. From 2004 to 2005 she held a teaching position in Salzburg. She received her PhD degree from the University of Cambridge (UK) in 2009. After one year of postdoctoral activity at the University of Göttingen (Germany), she became a Lecturer at Cambridge in 2010, promoted to Reader in 2015 and promoted to Professor in 2018. Since 2011 she is a fellow of Jesus College Cambridge and since 2016 a fellow of the Alan Turing Institute, London. Her current research interests focus on variational methods, partial differential equations and machine learning for image analysis, image processing and inverse imaging problems.