Acta Numerica (2019), pp. 1–174
doi:10.1017/S0962492919000059 Printed in the United Kingdom

Solving inverse problems using data-driven models

Simon Arridge
Department of Computer Science,
University College London,
Gower Street, London WC1E 6BT, UK
E-mail: S.Arridge@cs.ucl.ac.uk

Peter Maass
Department of Mathematics,
University of Bremen, Postfach 330 440,
28344 Bremen, Germany
E-mail: pmaass@math.uni-bremen.de

Ozan Öktem
Department of Mathematics,
KTH – Royal Institute of Technology,
SE-100 44 Stockholm, Sweden
E-mail: ozan@kth.se

Carola-Bibiane Schönlieb
Department of Applied Mathematics and Theoretical Physics,
Cambridge University, Wilberforce Road,
Cambridge, CB3 0WA, UK
E-mail: C.B.Schoenlieb@damtp.cam.ac.uk

Recent research in inverse problems seeks to develop a mathematically coherent foundation for combining data-driven models, and in particular those based on deep learning, with domain-specific knowledge contained in physical–analytical models. The focus is on solving ill-posed inverse problems that are at the core of many challenging applications in the natural sciences, medicine and life sciences, as well as in engineering and industrial applications. This survey paper aims to give an account of some of the main contributions in data-driven inverse problems.

CONTENTS

1. Introduction

In several areas of science and industry there is a need to reliably recover a hidden multi-dimensional model parameter from noisy indirect observations. A typical example is when imaging/sensing technologies are used in medicine, engineering, astronomy and geophysics. These so-called inverse problems are often ill-posed, meaning that small errors in data may lead to large errors in the model parameter, or there are several possible model parameter values that are consistent with observations. Addressing ill-posedness is critical in applications where decision making is based on the recovered model parameter, for example in image-guided medical diagnostics. Furthermore, many highly relevant inverse problems are large-scale: they involve large amounts of data and the model parameter is high-dimensional.

Traditionally, an inverse problem is formalized as solving an equation of the form

$$g = \mathcal{A}(f) + e.$$

Here $g \in Y$ is the measured data, assumed to be given, and $f \in X$ is the model parameter we aim to reconstruct. In many applications, both g and f are elements in appropriate function spaces Y and X, respectively. The mapping $\mathcal{A}\colon X \to Y$ is the forward operator, which describes how the model parameter gives rise to data in the absence of noise and measurement errors, and $e \in Y$ is the observational noise that constitutes random corruptions in the data g. The above view constitutes a knowledge-driven approach, where the forward operator and the probability distribution of the observational noise are derived from first principles.

Classical research on inverse problems has focused on establishing conditions which guarantee that solutions to such ill-posed problems exist and on

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

methods for approximating solutions in a stable way in the presence of noise (Engl, Hanke and Neubauer 2000, Benning and Burger 2018, Louis 1989, Kirsch 2011). Despite being very successful, such a knowledge-driven approach is also associated with some shortcomings. First, the forward model is always an approximate description of reality, and extending it might be challenging due to a limited understanding of the underlying physical or technical setting. It may also be limited due to computational complexity. Accurate analytical models, such as those based on systems of nonlinear partial differential equations (PDEs), may reach a numerical complexity beyond any feasible real-time potential in the foreseeable future. Second, most applications will have inputs which do not cover the full model parameter space, but stem from an unknown subset or obey an unknown stochastic distribution. The latter shortcoming in particular has led to the advance of methods that incorporate information about the structure of the parameters to be determined in terms of sparsity assumptions (Daubechies, Defrise and De Mol 2004, Jin and Maass 2012*b*) or stochastic models (Kaipio and Somersalo 2007, Mueller and Siltanen 2012). While representing a significant advancement in the field of inverse problems, these models are, however, limited by their inability to capture very bespoke structures in data that vary in different applications.

At the same time, data-driven approaches as they appear in machine learning offer several methods for amending such analytical models and for tackling these shortcomings. In particular, deep learning (LeCun, Bengio and Hinton 2015), which has had a transformative impact on a wide range of tasks related to artificial intelligence, ranging from computer vision and speech recognition to playing games (Igami 2017), is starting to show its impact on inverse problems. A key feature in these methods is the use of *generic* models that are adapted to specific problems through learning against example data (training data). Furthermore, a common trait in the success stories for deep learning is the abundance of training data and the explicit agnosticism from *a priori* knowledge of how such data are generated. However, in many scientific applications, the solution method needs to be robust and there is insufficient training data to support an entirely data-driven approach. This seriously limits the use of entirely data-driven approaches for solving problems in the natural and engineering sciences, in particular for inverse problems.

A recent line of development in computational sciences combines the seemingly incompatible data- and knowledge-driven modelling paradigms. In the context of inverse problems, ideally one uses explicit knowledge-driven models when there are such available, and learns models from example data using data-driven methods only when this is necessary. Recently several algorithms have been proposed for this combination of model- and data-driven approaches for solving ill-posed inverse problems. These results are still

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

primarily experimental and lack a thorough theoretical foundation; nevertheless, some mathematical concepts for treating data-driven approaches for inverse problems are emerging.

This survey attempts to provide an overview of methods for integrating data-driven concepts into the field of inverse problems. Particular emphasis is placed on techniques based on deep neural networks, and our aim is to pave the way for future research towards providing a solid mathematical theory. Some aspects of this development are covered in recent reviews of inverse problems and deep learning, for instance those of McCann, Jin and Unser (2017), Lucas, Iliadis, Molina and Katsaggelos (2018) and McCann and Unser (2019).

1.1. Overview

This survey investigates algorithms for combining model- and data-driven approaches for solving inverse problems. To do so, we start by reviewing some of the main ideas of knowledge-driven approaches to inverse problems, namely functional analytic inversion (Section 2) and Bayesian inversion (Section 3), respectively. These knowledge-driven inversion techniques are derived from first principles of knowledge we have about the data, the model parameter and their relationship to each other.

Knowledge- and data-driven approaches can now be combined in several different ways depending on the type of reconstruction one seeks to compute and the type of training data. Sections 4 and 5 represent the core of the survey and discuss a range of inverse problem approaches that introduce data-driven aspects in inverse problem solutions. Here, Section 4 is the data-driven sister section to functional analytic approaches in Section 2. These approaches are primarily designed to combine data-driven methods with functional analytic inversion. This is done either to make functional analytic approaches more data-driven by appropriate parametrization of these approaches and adapting these parametrizations to data, or to accelerate an otherwise costly functional analytic reconstruction method.

Many reconstruction methods, however, are not naturally formulated within the functional analytic view of inversion. An example is the posterior mean reconstruction, whose formulation requires adopting the Bayesian view of inversion. Section 5 is the data-driven companion to Bayesian inversion in Section 3, and surveys methods that combine data- and knowledge-driven methods in Bayesian inversion. The simplest is to apply data-driven post-processing of a reconstruction obtained via a knowledge-driven method. A more sophisticated approach is to use a learned iterative scheme that integrates a knowledge-driven model for how data are generated into a data-driven method for reconstruction. The latter is done by unrolling a knowledge-driven iterative scheme, and both approaches, which compute

statistical estimators, can be combined with forward operators that are partially learned via a data-driven method.

The above approaches come with different trade-offs concerning demands on training data, statistical accuracy and robustness, functional complexity, stability and interpretability. They also impact the choice of machine learning methods and algorithms for training. Certain recent – and somewhat anecdotal – topics of data-driven inverse problems are discussed in Section 6, and exemplar practical inverse problems and their data-driven solutions are presented in Section 7.

Within data-driven approaches, deep neural networks will be a focus of this survey. For an introduction to deep neural networks the reader might find it helpful to consult some introductory literature on the topic. We recommend Courville, Goodfellow and Bengio (2017) and Higham and Higham (2018) for a general introduction to deep learning; see also Vidal, Bruna, Giryes and Soatto (2017) for a survey of work that aims to provide a mathematical justification for several properties of deep networks. Finally, the reader may also consult Ye, Han and Cha (2018), who give a nice survey of various types of deep neural network architectures.

Detailed structure of the paper. In Section 2 we discuss functional analytic inversion methods, and in particular the mathematical notion of ill-posedness (Section 2.3) and regularization (Section 2.4) as a means to counteract the latter. A special focus is on variational regularization methods (Sections 2.5–2.7), as those reappear in bilevel learning in Section 4.3 in the context of data-driven methods for inverse problems.

Statistical – and in particular Bayesian – approaches to inverse problems are described in Section 3. In contrast to functional analytic approaches (Section 2.4), in Bayesian inversion (Section 3.1) the model parameter is a random variable that follows a prior distribution. A key difference between Bayesian and functional analytic inversion is that in Bayesian inversion an approximation to the whole distribution of the model parameter conditioned on the measured data (posterior distribution) is computed, rather than a single model parameter as in functional analytic inversion. This means that reconstructed model parameters can be derived via different estimates of its posterior distribution (a concept that we will encounter again in Section 5, and in particular Section 5.1.2, where data-driven reconstructions are phrased as results of different Bayes estimators), but also that uncertainty of reconstructed model parameters can be quantified (Section 3.2.5). When evaluating different reconstructions of the model parameter – which is again important when defining learning, *i.e.* optimization criteria for inverse problem solutions – aspects of statistical decision theory can be used (Section 3.3). Also, the parallel concept of regularization, introduced in Section 3 for the functional analytic approach, is outlined in Section 3.2 for

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

statistical approaches. The difficult problem of selecting a prior distribution for the model parameter is discussed in Section 3.4.

In Section 4 we present some central examples of machine learning combined with functional analytic inversion. These encompass classical parameter choice rules for inverse problems (Section 4.1) and bilevel learning (Section 4.3) for parameter learning in variational regularization methods. Moreover, dictionary learning is discussed in Section 4.4 as a companion to sparse reconstruction methods in Section 2.7, but with a data-driven dictionary. Also, the concept of a black-box denoiser, and its application to inverse problems by decoupling the regularization from the inversion of the data, is presented in Section 4.6. Two recent approaches that use deep neural network parametrizations for data-driven regularization in variational inversion models are investigated in Section 4.7. In Section 4.9 we discuss a range of learned optimization methods that use data-driven approximations as a means to speed up numerical computation. Finally, in Section 4.10 we introduce a new idea of using the recently introduced concept of deep inverse priors for solving inverse problems.

In Section 5 learning data-driven inversion models are phrased in the context of statistical regularization. Section 5.1.2 connects back to the difficulty in Bayesian inversion of choosing an appropriate prior (Section 3.4), and outlines how model learning can be used to compute various Bayes estimators. Here, in particular, fully learned inversion methods (Section 5.1.3), where the whole inversion model is data-driven, are put in context with learned iterative schemes (Section 5.1.4), in which data-driven components are interwoven with inverse model assumptions. In this context also we discuss post-processing methods in Section 5.1.5, where learned regularization together with simple knowledge-driven inversion methods are used sequentially. Section 5.2 addresses the computational bottleneck of Bayesian inversion methods by using learning, and shows how one can use learning to efficiently sample from the posterior.

Section 6 covers special topics of learning in inverse problems, and in Section 6.1 includes task-based reconstruction approaches that use ideas from learned iterative reconstruction (Section 5.1.4) and deep neural networks for segmentation and classification to solve joint reconstruction-segmentation problems, learning physics-based models via neural networks (Section 6.2.1), and learning corrections to forward operators by optimization methods that perform joint reconstruction-operator correction (Section 6.2).

Finally, Section 7 illustrates some of the data-driven inversion methods discussed in the paper by applying them to practical inverse problems. These include an introductory example on inversion of ill-conditioned linear systems to highlight the intricacy of using deep learning for inverse problems as a black-box approach (Section 7.1), bilevel optimization from Section 4.3 for parameter learning in TV-type regularized problems and

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

variational models with mixed-noise data fidelity terms (Section 7.2), the application of learned iterative reconstruction from Section 5.1.4 to computed tomography (CT) and photoacoustic tomography (PAT) (Section 7.3), adversarial regularizers from Section 4.7 for CT reconstruction as an example of variational regularization with a trained neural network as a regularizer (Section 7.4), and the application of deep inverse priors from Section 4.10 to magnetic particle imaging (MPI) (Section 7.5).

In Section 8 we finish our discussion with a few concluding remarks and comments on future research directions.

2. Functional analytic regularization

Functional analysis has had a strong impact on the development of inverse problems. One of the first publications that can be attributed to the field of inverse problems is that of Radon (1917). This paper derived an explicit inversion formula for the so-called Radon transform, which was later identified as a key component in the mathematical model for X-ray CT. The derivation of the inversion formula, and its analysis concerning missing stability, makes use of operator formulations that are remarkably close to the functional analysis formulations that would be developed three decades later.

2.1. The inverse problem

There is no formal mathematical definition of an inverse problem, but from an applied viewpoint such problems are concerned with determining causes from desired or observed effects. It is common to formalize this as solving an operator equation.

Definition 2.1. An *inverse problem* is the task of recovering the model parameter $f_{\text{true}} \in X$ from measured data $g \in Y$, where

$$g = \mathcal{A}(f_{\text{true}}) + e. \tag{2.1}$$

Here, X (model parameter space) and Y (data space) are vector spaces with appropriate topologies and whose elements represent possible model parameters and data, respectively. Moreover, $\mathcal{A}\colon X \to Y$ (forward operator) is a known continuous operator that maps a model parameter to data in absence of observation noise and $e \in Y$ is a sample of a Y-valued random variable modelling the observation noise.

In most imaging applications, such as CT image reconstruction, elements in X are images represented by functions defined on a fixed domain $\Omega \subset \mathbb{R}^d$ and elements in Y represent imaging data by functions defined on a fixed manifold $\mathbb{M}$ that is given by the acquisition geometry associated with the measurements.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

2.2. Introduction to some example problems

In the following, we briefly introduce some of the key inverse problems we consider later in this survey. All are from imaging, and we make a key distinction between (i) *image restoration* and (ii) *image reconstruction*. In the former, the data are a corrupted (*e.g.* noisy or blurry) realization of the model parameter (image) so the reconstruction and data spaces coincide, whereas in the latter the reconstruction space is the space of images but the data space has a definition that is problem-dependent. As we will see when discussing data-driven approaches to inverse problems in Sections 4 and 5, this differentiation is particularly crucial as the difference between image and data space poses additional challenges to the design of machine learning methods. Next, we describe very briefly some of the most common operators that we will refer to below. Here the inverse problems in Sections 2.2.1–2.2.3 are image restoration problems, while those in Sections 2.2.4 and 2.2.5 are examples of image reconstruction problems.

2.2.1. Image denoising

The observed data are the ideal solution corrupted by additive noise, so the forward operator in (2.1) is the identity transform $\mathcal{A} = \mathrm{id}$, and we get

$$g = f_{\text{true}} + e, \tag{2.2}$$

In the simplest case the distribution of the observational noise is known. Furthermore, this distribution may in more advanced problems be correlated, spatially varying and of mixed type.

In Section 7.2 we will discuss bilevel learning of total variation (TV)-type variational models for denoising of data corrupted with mixed noise distributions.

2.2.2. Image deblurring

The observed data are given by convolution with a known filter function K together with additive noise, so (2.1) becomes

$$g = f_{\text{true}} * K + e. \tag{2.3}$$

Any inverse problem of the type (2.1) with a linear forward operator that is translation-invariant will be of this form.

In the absence of noise, the inverse problem (*image deconvolution*) is exactly solvable by division in the Fourier domain, *i.e.* $f_{\text{true}} = \mathcal{F}^{-1}[\mathcal{F}[g]/\mathcal{F}[K]]$, provided that $\mathcal{F}[K]$ has infinite support in the Fourier domain. In the presence of noise, the estimated solution is corrupted by noise whose frequency spectrum is the reciprocal of the spectrum of the filter K. The distribution of the observational also has the same considerations as in (2.2). Finally, extensions include the case of a spatially varying kernel and the case where K is unknown (*blind deconvolution*).

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

2.2.3. Image in-painting

Here, the observed data represents a noisy observation of the true model parameter $f_{\text{true}}\colon \Omega \to \mathbb{R}$ restricted to a fixed measurable set $\Omega_0 \subset \mathbb{R}^n$:

$$g = f_{\text{true}}\big|_{\Omega_0} + e. \tag{2.4}$$

In the above, $f_{\text{true}}\big|_{\Omega_0}$ is the restriction of f_{true} to Ω_0. Solutions take different forms depending on the size of connected components in Ω_0. Extensions include the case where Ω_0 is unknown or only partially known.

2.2.4. Computed tomography (CT)

The simplest physical model for CT assumes mono-energetic X-rays and disregards scattering phenomena. The model parameter is then a real-valued function $f\colon \Omega \to \mathbb{R}$ defined on a fixed domain $\Omega \subset \mathbb{R}^d$ ($d = 2$ for two-dimensional CT and $d = 3$ for three-dimensional CT) that has unit mass per volume. The forward operator is the one given by the Beer–Lambert law:

$$\mathcal{A}(f)(\omega, x) = \mathrm{e}^{-\mu \int_{-\infty}^{\infty} f(x+s\omega)\,\mathrm{d}s}. \tag{2.5}$$

Here, the unit vector $\omega \in S^{d-1}$ and $x \in \omega^{\perp}$ represent the line $\ell : s \mapsto x + s\omega$ along which the X-rays travel, and we also assume f decays fast enough for the integral to exist. In medical imaging, μ is usually set to a value that approximately corresponds to water at the X-ray energies used. The above represents pre-logarithm (or pre-log) data, and by taking the logarithm (or log) of data, one can recast the inverse problem in CT imaging to one where the forward model is the linear ray transform:

$$\mathcal{A}(f)(\omega, x) = \int_{-\infty}^{\infty} f(x + s\omega)\,\mathrm{d}s. \tag{2.6}$$

For low-dose imaging, pre-log data are Poisson-distributed with mean $\mathcal{A}(f_{\text{true}})$, where $\mathcal{A}$ is given as in (2.5), that is, $g \in Y$ is a sample of $\mathrm{g} \sim \text{Poisson}(\mathcal{A}(f_{\text{true}}))$. Thus, to get rid of the non-linear exponential in (2.5), it is common to take the log of data. With such post-log data the forward operator is linear and given as in (2.6). A complication with such post-log data is that the noise model becomes non-trivial, since one takes the log of a Poisson-distributed random variable (Fu *et al.* 2017). A common approximate noise model for post-log data is (2.1), with observational noise e which is a sample of a Gaussian or Laplace-distributed random variable.

In the case of *complete* data, that is, where a full angular set of data is measured, an exact inverse is obtained by the (Fourier-transformed) data backprojected on the same lines as used for the measurements and scaled by the absolute value of the spatial frequency, followed by the inverse Fourier transform. Thus, as in deblurring, the noise is amplified, but only linearly in spatial frequency, making the problem mildly ill-posed. Extensions

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

include the *emission tomography* problem (single photon emission computed tomography (SPECT) and positron emission tomography (PET)) where the line integrals are exponentially attenuated by a function μ that may be unknown. A major challenge in tomography is to consider *incomplete* data, in particular the case where only a subset of lines is measured. This problem is much more ill-posed.

See Sections 7.3, 7.4 and 7.6 for instances of CT reconstruction that use deep neural networks in the solution of the inverse problem.

2.2.5. Magnetic resonance imaging (MRI)

The observed data are often considered to be samples of the Fourier transform of the ideal signal, so the MRI image reconstruction problem is an inverse problem of the type (2.1), where the forward operator is given as a discrete sampling operator concatenated with the Fourier transform. A correct description of the problem takes account of the complex-valued nature of the data, which implies that when e is normally distributed then the noise model of $|\mathcal{F}^{-1}[g]|$ is Rician. As in CT, the case of under-sampled data is of high practical importance. In MRI, the subsampling operator has to consist of connected trajectories in Fourier space but is not restricted to straight lines.

Extensions include the case of *parallel MRI* where the forward operator is combined with (several) spatial sensitivity functions. More exact forward operators take account of other non-linear physical effects and can reconstruct several functions in the solution space.

2.3. Notion of ill-posedness

A difficulty in solving (2.1) is that the solution is sensitive to variations in data, which is referred to as ill-posedness. More precisely, the notion of ill-posedness is usually attributed to Hadamard, who postulated that a well-posed problem must have three defining properties, namely that (i) it has a solution (existence) that is (ii) unique and that (iii) depends continuously on the data g (stability). Problems that do not fulfil these criteria are ill-posed and, according to Hadamard, should be modelled differently (Hadamard 1902, Hadamard 1923).

For example, instability arises when the forward operator $\mathcal{A}\colon X \to Y$ in (2.1) has an unbounded or discontinuous inverse. Hence, every non-degenerate compact operator between infinite-dimensional Hilbert spaces whose range is infinite naturally leads to ill-posed inverse problems. Slightly more generally, one can prove that continuous operators with non-closed range yield unbounded inverses and hence lead to ill-posed inverse problems. This class includes non-degenerate compact operators as well as convolution operators on unbounded domains.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Another way of describing ill-posedness is in terms of the set of $f \in X$ such that $\|\mathcal{A}(f) - g\| \leq \|e\|$ for given noise e in (2.1). This is an unbounded set for continuous operators with non-closed range. Finally, yet another way to understand the ill-posedness of a compact linear operator $\mathcal{A}$ is by means of its singular value decomposition. The decay of the spectrum $\{\sigma_k\}_{k\in\mathbb{N}}$ is strongly related to the ill-posedness: faster decay implies a more ill-posed problem. This allows us to determine the severity of the ill-posedness. More precisely, (2.1) is weakly ill-posed if σ_k decays with polynomial rate as $k \to \infty$ and strongly ill-posed if the decay is exponential: see Engl *et al.* (2000), Derevtsov, Efimov, Louis and Schuster (2011) and Louis (1989) for further details. As a final note, such classification is not possible when the forward operator is non-linear. In such cases, either linearized forward operators are analysed or a non-linear spectral analysis is considered for determining the degree of ill-posedness: see *e.g.* Hofmann (1994). Moreover, extensions to non-compact linear operators are considered by Hofmann *et al.* (2010), for example.

2.4. Regularization

Unfortunately, Hadamard's dogma stigmatized the study of ill-posed problems and thereby severely hampered the development of the field. Mathematicians' interest in studying ill-posed problems was revitalized by the pioneering works of Calderón and Zygmund (1952, 1956), Calderón (1958) and John (1955*a*, 1955*b*, 1959, 1960), who showed that instability is an intrinsic property in some of the most interesting and challenging problems in mathematical physics and applied analysis. To some extent, these papers constitute the origin of the modern theory of inverse problems and regularization.

The aim of functional analytic regularization theory is to develop stable schemes for estimating f_{true} from data g in (2.1) based on knowledge of $\mathcal{A}$, and to prove analytical results for the properties of the estimated solution. More precisely, a regularization of the inverse problem in (2.1) is formally a scheme that provides a well-defined parametrized mapping $\mathcal{R}_\theta \colon Y \to X$ (*existence*) that is continuous in Y for fixed θ (*stability*) and *convergent*. The latter means there is a way to select θ so that $\mathcal{R}_\theta(g) \to f_{\text{true}}$ as $g \to \mathcal{A}(f_{\text{true}})$.

Besides existence, stability and convergence, a complete mathematical analysis of a regularization method also includes proving *convergence rates* and *stability estimates*. Convergence rates provide an estimate of the difference between a regularized solution $\mathcal{R}_\theta(g)$ and the solution of (2.1) with $e = 0$ (provided it exists), whereas stability estimates provide a bound on the difference between $\mathcal{R}_\theta(g)$ and $\mathcal{R}_\theta(\mathcal{A}(f_{\text{true}}))$ depending on the error $\|e\|$. These theorems rely on 'source conditions': for example, convergence rate results are obtained under the assumption that the true solution f_{true} is in

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

the range of $[\partial \mathcal{A}(f_{\text{true}})]^*: Y \to X$. One difficulty is to formulate source conditions that are verifiable. This typically relates to a regularity assumption for f_{true} that ensures a certain convergence rate: see *e.g.* Engl, Kunisch and Neubauer (1989), Hofmann, Kaltenbacher, Pöschl and Scherzer (2007), Schuster, Kaltenbacher, Hofmann and Kazimierski (2012), Grasmair, Haltmeier and Scherzer (2008) and Hohage and Weidling (2016) for an example of this line of development.

From an algorithmic viewpoint, functional analytic regularization methods are subdivided into essentially four categories.

Approximate analytic inversion. These methods are based on stabilizing a closed-form expression for $\mathcal{A}^{-1}$. This is typically achieved by considering reconstruction operators that give a mollified solution, so the resulting approaches are highly problem-specific.

Analytic inversion has been hugely successful: for example, filtered backprojection (FBP) (Natterer 2001, Natterer and Wübbeling 2001) for inverting the ray transform is still the standard method for image reconstruction in CT used in clinical practice. Furthermore, the idea of recovering a mollified version can be stated in a less problem-specific manner, which leads to the method of approximate inverse (Louis 1996, Schuster 2007, Louis and Maass 1990).

Iterative methods with early stopping. Here one typically considers iteration methods based on gradient descent for the data misfit or discrepancy term $f \mapsto \| \mathcal{A}(f) - g\|^2$. The ill-posedness of the inverse problem leads to semiconvergent behaviour, meaning that the reconstruction error decreases until a certain data fit is achieved and then starts diverging. Hence a suitable stopping needs to be designed, which acts as a regularization.

Well-known examples are the iterative schemes of Kaczmarz and Landweber (Engl *et al.* 2000, Natterer and Wübbeling 2001, Kirsch 2011). There is also large body of literature addressing iteration schemes in Krylov spaces for inverse problems (conjugate gradient (CG) type methods) as well as accelerated and discretized versions thereof (*e.g.* CGLS, LSQR, GMRES): see Hanke-Bourgeois (1995), Hanke and Hansen (1993), Frommer and Maass (1999), Calvetti, Lewis and Reichel (2002) and Byrne (2008) for further reference. A different approach that has a statistical interpretation uses a fixed-point iteration for the maximum *a posteriori* (MAP) estimator leading to the maximum likelihood expectationmaximization (ML-EM) algorithm (Dempster *et al.* 1977).

Discretization as regularization. Projection or Galerkin methods, which search for an approximate solution of an inverse problems in a predefined subspace, are also a powerful tool for solving inverse problems. The level of discretization controls the approximation of the forward operator but it also stabilizes the inversion process: see Engl *et al.* (2000), Plato and

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Vainikko (1990) and Natterer (1977). Such concepts have been discussed in the framework of parameter identification for partial differential equations (quasi-reversibility): see Lattès and Lions (1969) for an early reference and Hämarik, Kaltenbacher, Kangro and Resmerita (2016) and Kaltenbacher, Kirchner and Vexler (2011) for some recent developments.

Variational methods. The idea here is to minimize a measure of data misfit that is penalized using a regularizer (Kaltenbacher, Neubauer and Scherzer 2008, Scherzer *et al.* 2009):

$$\mathcal{R}_\theta(g) := \arg\min_{f \in X} \{\mathcal{L}(\mathcal{A}(f), g) + \mathcal{S}_\theta(f)\}, \tag{2.7}$$

where we make use of the notation in Definitions 2.2, 2.4 and 2.5. This is a generic, yet highly adaptable, framework for reconstruction with a natural plug-and-play structure where the forward operator $\mathcal{A}$, the data discrepancy $\mathcal{L}$ and the regularizer $\mathcal{S}_\theta$ are chosen to fit the specific aspects of the inverse problem. Well-known examples are classical Tikhonov regularization and TV regularization.

Sections 2.5 and 2.6 provide a closer look at the development of variational methods since these play an important role in Section 4, where data-driven methods are used in functional analytic regularization. To simplify these descriptions it is convenient to establish some key notions.

Definition 2.2. A *regularization functional* $\mathcal{S} : X \to \mathbb{R}_+$ quantifies how well a model parameter possesses desirable features: a larger value usually means less desirable properties.

In variational approaches to inverse problems, the value of $\mathcal{S}$ is considered as a *penalty term*, and in Bayesian approaches it is seen as the negative log of a *prior probability distribution*. Henceforth we will use $\mathcal{S}_\theta$ to denote a regularization term that depends on a parameter set $\theta \in \Theta$; in particular we will use θ as parameters that will be *learned*.

Remark 2.3. In some cases θ is a single scalar. We will use the notation $\lambda\mathcal{S}(f) \equiv \mathcal{S}_\theta(f)$ wherever such usage is unambiguous, and with the implication that $\theta = \lambda \in \mathbb{R}_+$. Furthermore, we will sometimes express the set θ explicitly, *e.g.* $\mathcal{S}_{\alpha,\beta}$, where the usage is unambiguous.

Definition 2.4. A *data discrepancy functional* $\mathcal{L}\colon Y \times Y \to \mathbb{R}$ is a scalar quantification of the similarity between two elements of data space Y.

The data discrepancy functional is considered to be a data fitting term in variational approaches to inverse problems. Although often taken to be a metric on data space, choosing it as an affine transformation of the negative log-likelihood of data allows for a statistical interpretation, since

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

minimizing $f \mapsto \mathcal{L}(\mathcal{A}(f), g)$ amounts to finding a maximum likelihood solution. For Gaussian observational noise $\mathrm{e} \sim \mathrm{N}(0, \Gamma)$, the corresponding data discrepancy is then given by the Mahalanobis distance:

$$\mathcal{L}(g, v) := \|g - v\|^2_{\Gamma^{-1}} \quad \text{for } g, v \in Y. \tag{2.8}$$

If data are Poisson-distributed, *i.e.* $\mathrm{g} \sim \mathrm{Poisson}(\mathcal{A}(f_{\mathrm{true}}))$, then an appropriate data discrepancy functional is the Kullback–Leibler (KL) divergence. When elements in data space Y are real-valued functions on $\mathbb{M}$, then the KL divergence becomes

$$\mathcal{L}(g, v) := \int_{\mathbb{M}} [v(y) \log g(y) - \log v(y) - g(y) + v(y)] \, \mathrm{d}y \quad \text{for } g, v \in Y. \tag{2.9}$$

Similarly, Laplace-distributed observational noise corresponds to a data discrepancy that is given by the 1-norm. One can also express the data log-likelihood for Poisson-distributed data with an additive observational noise term that is Gaussian, but the resulting expressions are quite complex (Benvenuto *et al.* 2008).

Definition 2.5. A *reconstruction operator* $\mathcal{R} \colon Y \to X$ is a mapping which gives a point estimate $\hat{f}$ as the solution to (2.1).

Henceforth we will use $\mathcal{R}_\theta$ to denote a reconstruction operator that depends on a parameter set $\theta \in \Theta$; in particular we will use θ as parameters that will be *learned*.

Remark 2.6. In variational approaches we will typically use the notation $\mathcal{R}_\lambda$ to denote an operator parametrized by a single scalar $\lambda > 0$ which corresponds (explicitly or implicitly) to optimizing a functional that includes a regularization penalty $\lambda \mathcal{S}$ as in Definition 2.2. More generally θ will also be used to define the parameters of an algorithm or neural network used to generate a solution f_θ that may or may not explicitly specify a regularization functional. Again we will assume the context provides an unambiguous justification for the choice between $\mathcal{R}_\theta$ and $\mathcal{R}_\lambda$. Furthermore, we will sometimes express the set θ explicitly, *e.g.* $\mathcal{R}_{\mathcal{W},\psi}$, where the usage is unambiguous.

2.5. Classical Tikhonov regularization

Tikhonov (or Tikhonov–Phillips) regularization is arguably the most prominent technique for inverse problems. It was introduced by Tikhonov (1943, 1963), Phillips (1962) and Tikhonov and Arsenin (1977) for solving ill-posed inverse problems of the form (2.1), and can be stated in the form

$$\mathcal{R}_\lambda(g) := \operatorname*{arg\,min}_{f \in X} \left\{ \frac{1}{2} \| \mathcal{A}(f) - g \|^2 + \lambda \mathcal{S}(f) \right\}. \tag{2.10}$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Bearing in mind the notation in Remarks 2.3 and 2.6, note that (2.10) has the form of (2.7) with $\mathcal{L}$ given by the squared L^2-distance. Here X and Y are Hilbert spaces (typically both are L^2 spaces). Moreover, the choice $\mathcal{S}(f) := \frac{1}{2}\|f\|^2$ is the most common one for the penalty term in (2.10). In fact, if $\mathcal{A}$ is linear, with $\mathcal{A}^*$ denoting its adjoint, then standard arguments for minimizing quadratic functionals yield

$$\mathcal{R}_\lambda = (\mathcal{A}^* \circ \mathcal{A} + \lambda \,\mathrm{id})^{-1} \circ \mathcal{A}^* . \tag{2.11}$$

Until the late 1980s the analytical investigations of regularization schemes of the type in (2.10) were restricted either to linear operators or to rather specific approaches for selected non-linear problems. Many inverse problems, such as parameter identification in linear differential operators, lead to non-linear parameter-to-state maps, so the corresponding forward operator becomes non-linear (Arridge and Schotland 2009, Greenleaf, Kurylev, Lassas and Uhlmann 2007, Bal, Chung and Schotland 2016, Jin and Maass 2012*b*, Jin and Maass 2012*a*).

Analysis of (2.10) for non-linear forward operators is difficult, for example, singular value decompositions are not available. A major theoretical breakthrough came with the publications of Seidman and Vogel (1989) and Engl *et al.* (1989), which extended the theoretical investigation of Tikhonov regularization to the non-linear setting by introducing radically new concepts. Among others, it extended the notion of minimum norm solution used in theorems dealing with convergence rates to f_0-minimum norm solutions. This means one assumes the knowledge of some meaningful parameter $f_0 \in X$ and the aim of the regularization method is to approximate a solution that in the limit minimizes $\|f - f_0\|$ amongst all solutions of $\mathcal{A}(f) = g$. Hence, f_0 acts as a kind of prior. Among the main results of Engl *et al.* (1989) is a theorem that estimates the convergence rate assuming sufficient regularity of the forward operator $\mathcal{A}$ and a source condition that relates the penalty term to the functional $\mathcal{A}$ at f_{true}. This theorem, which is stated below, opened the path to many successful applications, particularly for parameter identification problems related to partial differential equations, and such assumptions occur in different variations in all theorems related to the variational approach.

Theorem 2.7. Consider the inverse problem in (2.1) where $\mathcal{A}\colon X \to Y$ is continuous, weakly sequentially closed and with a convex domain. Next, assume there exists a f_0-minimum norm solution f_{true} for some fixed $f_0 \in X$ and let data $g \in Y$ in (2.1) satisfy $\|\mathcal{A}(f_{\text{true}}) - g\| \leq \delta$. Also, let $f_\lambda^\delta \in X$ denote a minimizer of (2.10) with $\mathcal{S}(f) := \frac{1}{2}\|f\|^2$. Finally, assume the following.

(1) $\mathcal{A}$ has a continuous Fréchet derivative.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

(2) There exists a $\gamma > 0$, such that

$$\|[\partial\mathcal{A}(f_{\text{true}})] - [\partial\mathcal{A}(f)]\|_{L(X,Y)} \leq \gamma\|f_{\text{true}} - f\|_X$$

for all $f \in \text{dom}(\mathcal{A}) \cap B_r(f_{\text{true}})$ with $r > 2\|f_{\text{true}} - f_0\|$. Here, $L(X, Y)$ is the vector space of Y-valued linear maps on X and $B_r(f_{\text{true}}) \subset X$ denotes a ball of radius r around f_{true}.

(3) There exists $v \in Y$ with $\gamma\|v\| < 1$ such that $f_{\text{true}} - f_0 = [\partial\mathcal{A}(f_{\text{true}})]^*(v)$.

Then, choosing $\lambda \propto \delta$ as $\delta \to 0$ yields

$$\|f_\lambda^\delta - f_{\text{true}}\| = O(\sqrt{\delta}) \quad \text{and} \quad \|\mathcal{A}(f_\lambda^\delta) - g\| = O(\delta).$$

2.6. Extension of classical Tikhonov regularization

Tikhonov regularization is a particular case of the more general variational regularization schemes (2.7). In fact, penalty terms for Tikhonov-type functionals, as well as suitable source conditions, have been generalized considerably in a series of papers. A key issue in solving an inverse problem is to use a forward operator $\mathcal{A}$ that is sufficiently accurate. It is also important to choose an appropriate data discrepancy $\mathcal{L}$, regularizer $\mathcal{S}_\theta$, and to have a parameter choice rule for setting θ. Depending on these choices, different reconstruction results are obtained.

The data discrepancy. Here the choice is ideally guided by statistical considerations for the observation noise (Bertero, Lantéri and Zanni 2008). Ideally one selects $\mathcal{L}$ as an appropriate affine transform of the negative log-likelihood of data, in which case minimizing $f \mapsto \mathcal{L}(\mathcal{A}(f), g)$ becomes the same as computing an maximum likelihood estimator. Hence, Poisson-distributed data that typically appear in photography (Costantini and Susstrunk 2004) and emission tomography applications (Vardi, Shepp and Kaufman 1985) lead to a data discrepancy given by the Kullback–Leibler divergence (Sawatzky, Brune, Müller and Burger 2009, Hohage and Werner 2016), while additive normally distributed data, as for Gaussian noise, result in a least-squares fit model.

The regularizer. As stated in Definition 2.2, $\mathcal{S}$ acts as a penalizer and is chosen to enforce stability by encoding *a priori* information about f_{true}. How to set the (regularization) parameter θ reflects noise level in data: see Section 4.1.

Classical Tikhonov regularization (2.10) uses Hilbert-space norms (or semi-norms) to regularize the inverse problems. In more recent years, Banach-space regularizers have become more popular in the context of sparsity-promoting and discontinuity-preserving regularization, which are revisited in Section 2.7. TV regularization was introduced by Rudin, Osher

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

and Fatemi (1992) for image denoising due to its edge-preserving properties, favouring images f that have a sparse gradient. Here, the TV regularizer is given as

$$\mathcal{S}(f) := \mathrm{TV}(f) := |Df|(\Omega) = \int_\Omega \mathrm{d}|Df|, \tag{2.12}$$

where $\Omega \subset \mathbb{R}^d$ is a fixed open and bounded set. The above functional (TV regularizer) uses the total variation measure of the distributional derivative of f defined on Ω (Ambrosio, Fusco and Pallara 2000). A drawback of using such a regularization procedure is apparent as soon as the true model parameter not only consists of constant regions and jumps but also possesses more complicated, higher-order structures, *e.g.* piecewise linear parts. In this case, TV introduces jumps that are not present in the true solution, which is referred to as staircasing (Ring 2000). Examples of generalizations of TV for addressing this drawback typically incorporate higher-order derivatives, *e.g.* total generalized variation (TGV) (Bredies, Kunisch and Pock 2011) and the infimal-convolution total variation (ICTV) model (Chambolle and Lions 1997). These read as

$$\begin{aligned}\mathcal{S}_{\alpha,\beta}(f) &:= \mathrm{ICTV}_{\alpha,\beta}(f)\\ &= \min_{\substack{v\in W^{1,1}(\Omega)\\ \nabla v\in BV(\Omega)}} \{\alpha\|Df-\nabla v\|_{\mathcal{M}(\Omega;\mathbb{R}^2)} + \beta\|D\nabla v\|_{\mathcal{M}(\Omega;\mathbb{R}^{2\times 2})}\},\end{aligned} \tag{2.13}$$

and the second-order TGV (Bredies and Valkonen 2011, Bredies, Kunisch and Valkonen 2013) reads as

$$\begin{aligned}\mathcal{S}_{\alpha,\beta}(f) &:= \mathrm{TGV}^2_{\alpha,\beta}(f)\\ &= \min_{w\in BD(\Omega)} \{\alpha\|Df-w\|_{\mathcal{M}(\Omega;\mathbb{R}^2)} + \beta\|Ew\|_{\mathcal{M}(\Omega;\mathrm{Sym}^2(\mathbb{R}^2))}\}.\end{aligned} \tag{2.14}$$

Here

$$BD(\Omega) := \{w \in L^1(\Omega;\mathbb{R}^d) \mid \|Ew\|_{\mathcal{M}(\Omega;\mathbb{R}^{d\times d})} < \infty\}$$

is the space of vector fields of bounded deformation on Ω with E denoting the *symmetrized gradient* and $\mathrm{Sym}^2(\mathbb{R}^2)$ the space of symmetric tensors of order 2 with arguments in $\mathbb{R}^2$. The parameters α, β are fixed positive parameters. The main difference between (2.13) and (2.14) is that we do not generally have $w = \nabla v$ for any function v. That results in some qualitative differences of ICTV and TGV regularization: see *e.g.* Benning, Brune, Burger and Müller (2013). One may also consider Banach-space norms other than TV, such as Besov norms (Lassas, Saksman and Siltanen 2009), which behave more nicely with respect to discretization (see also Section 3.4). Different TV-type regularizers and their adaption to data by bilevel learning of parameters (*e.g.* α and β in ICTV and TGV) will be discussed in more detail in Section 4.3.1 and numerical results will be given in Section 7.2.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Finally, in applied harmonic analysis, ℓ_p-norms of wavelets have been proposed as regularizers (Daubechies *et al.* 1991, Mallat 2009, Unser and Blu 2000, Kutyniok and Labate 2012, Eldar and Kutyniok 2012, Foucart and Rauhut 2013). Other examples are non-local regularization (Gilboa and Osher 2008, Buades, Coll and Morel 2005), anisotropic regularizers (Weickert 1998) and, in the context of free discontinuity problems, the representation of images as a composition of smooth parts separated by edges (Blake and Zisserman 1987, Mumford and Shah 1989, Carriero, Leaci and Tomarelli 1996).

2.7. Sparsity-promoting regularization

Sparsity is an important concept in conceiving inversion models as well as learning parts of them. In what follows we review some of the main approaches to computing sparse solutions, and postpone learning sparse representations to Section 4.4.

2.7.1. Notions of sparsity

Let X be a separable Hilbert space, that is, we will assume that it has a countable orthonormal basis. A popular approach to sparse reconstruction uses the notion of a *dictionary* $\mathbb{D} := \{\phi_i\} \subset X$, whose elements are called atoms. Here, $\mathbb{D}$ is either given, *i.e.* knowledge-driven, or data-driven and derived from a set of realizations $f_i \in X$: see Bruckstein, Donoho and Elad (2009), Daubechies *et al.* (2004), Rubinstein, Bruckstein and Elad (2010), Lanusse, Starck, Woiselle and Fadili (2014) and Chen and Needell (2016).

A special class of dictionaries is that of *frames*. A dictionary $\mathbb{D} := \{\phi_i\}$ is a frame if there exists $C_1, C_2 > 0$ such that

$$C_1\|f\|^2 \leq \sum_i |\langle f, \phi_i\rangle|^2 \leq C_2\|f\|^2 \quad \text{for any } f \in X. \tag{2.15}$$

A frame is called *tight* if $C_1 = C_2 = 1$; it is called *over-complete* or *redundant* if $\mathbb{D}$ does not form a basis for X. Redundant dictionaries, *e.g.* translation-invariant wavelets, often work better than non-redundant dictionaries: see Peyré, Bougleux and Cohen (2011) and Elad (2010).

To construct sparse representations, *i.e.* sparsity-promoting regularizers from such a parametrization, we need the notions of an analysis and a synthesis operator. Given a dictionary $\mathbb{D}$, the analysis operator $\mathcal{E}_{\mathbb{D}}\colon X \to \Xi$ maps an element in X to a sequence in Ξ, which is typically in ℓ^2, such that

$$\boldsymbol{\xi} = \mathcal{E}_{\mathbb{D}}(f) \in \Xi \text{ has components } \xi_i = \langle f, \phi_i\rangle.$$

The corresponding synthesis operator $\mathcal{E}^*_{\mathbb{D}}\colon \Xi \to X$ is the adjoint of the analysis operator, that is,

$$\mathcal{E}^*_{\mathbb{D}}(\boldsymbol{\xi}) := \sum_i \xi_i \phi_i. \tag{2.16}$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

We further define the frame operator as $\mathcal{E}_{\mathbb{D}}^* \circ \mathcal{E}_{\mathbb{D}} \colon X \to X$, that is,

$$\mathcal{E}_{\mathbb{D}}^* \circ \mathcal{E}_{\mathbb{D}}(f) := \sum_i \langle f, \phi_i \rangle \phi_i. \tag{2.17}$$

Now, we can define an $f \in X$ to be s-sparse with respect to $\mathbb{D}$ if

$$\| \mathcal{E}_{\mathbb{D}}(f) \|_0 = \#\{i | \langle f, \phi_i \rangle \neq 0\} \leq s. \tag{2.18}$$

In most applications, the model parameter is not sparse in the strict sense, which leads to the weaker notion of compressibility. A model parameter $f \in X$ is compressible with respect to $\mathbb{D}$ if the following power decay law holds:

$$|\widetilde{\xi}_k| \leq C k^{-1/q} \text{ as } k \to \infty \text{ for some } C > 0 \text{ and } 0 < q < 1. \tag{2.19}$$

Here, $\widetilde{\boldsymbol{\xi}}$ is a non-increasing rearrangement of the sequence $\boldsymbol{\xi} = \{\langle f, \phi_i \rangle\} = \mathcal{E}_{\mathbb{D}}(f)$. Note that sparse signals are compressible, and in particular, if q is small in (2.19), then compressibility is equivalent to sparsity from any practical viewpoint. We further define $\boldsymbol{\xi}_s$ to consist of the s largest (in magnitude) coefficients of the sequence $\widetilde{\boldsymbol{\xi}}$.

2.7.2. Sparse recovery

Here we consider solving (2.1) when our prior model assumes f_{true} to be compressible with respect to a given dictionary $\mathbb{D} := \{\phi_i\}$. Then sparse recovery of f can be done by either the synthesis approach (*i.e.* sparse coding) or the analysis approach. In the synthesis approach the reconstruction operator $\mathcal{R}_\theta$ is given as

$$\mathcal{R}_\theta(g) := \mathcal{E}_{\mathbb{D}}^*(\hat{\xi}), \tag{2.20}$$

where

$$\hat{\xi} = \arg\min_{\boldsymbol{\xi} \in \Xi} \{\mathcal{L}(\mathcal{A}(\mathcal{E}_{\mathbb{D}}^*(\boldsymbol{\xi})), g) + \lambda \|\boldsymbol{\xi}\|_0\},$$

with $\theta = \{\mathbb{D}, \lambda\}$, *i.e.* θ is the scalar $\lambda > 0$ and the entire dictionary that defines the synthesis operator $\mathcal{E}_{\mathbb{D}}^*$. In the corresponding analysis approach, we get

$$\mathcal{R}_\theta(g) := \arg\min_{f \in X} \{\mathcal{L}(\mathcal{A}(f), g) + \lambda \| \mathcal{E}_{\mathbb{D}}(f) \|_0\}. \tag{2.21}$$

If $\mathbb{D}$ is an orthonormal basis then synthesis and analysis formulations are equivalent.

There are different strategies for numerically computing sparse representations that solve (2.20) and (2.21).

Greedy approaches. These build up an approximation in a greedy fashion, computing one non-zero entry of $\hat{\xi}$ at a time by making locally optimal choices at each step. One example of a greedy approach is iterative

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

(hard) thresholding (Blumensath and Davies 2008, Blumensath 2013, Foucart 2016), where for an initial guess $\boldsymbol{\xi}^{(i0}$ and $i = 0, 1, \ldots$ one iterates

$$\boldsymbol{\xi}^{(i+1)} = T_s(\boldsymbol{\xi}^{(i)} - \mathcal{A}_{\mathbb{D}}^*(\mathcal{A}_{\mathbb{D}}(\boldsymbol{\xi}^{(i)}) - g)), \tag{2.22}$$

where $\mathcal{A}_{\mathbb{D}} := \mathcal{A} \circ \mathcal{E}_{\mathbb{D}}^*$. Here, $T_s(\boldsymbol{\xi})$ sets all but the largest (in magnitude) s elements of $\boldsymbol{\xi}$ to zero. This is therefore a proximal-gradient method with the proximal of the function being 0 at 0 and 1 everywhere else (see Section 8.2.7). Other examples are matching pursuit (MP) (Mallat and Zhang 1993), orthogonal matching pursuit (OMP) (Tropp and Gilbert 2007) and variants thereof such as StOMP (Donoho, Tsaig, Drori and Starck 2012), ROMP (Needell and Vershynin 2009) and CoSamp (Needell and Tropp 2009).

Convex relaxation. One of the most common approaches to solving sparse recovery is to replace the ℓ_0-(semi)norm with the ℓ_1-norm in (2.20) and (2.21). This leads to basis pursuit (Candès, Romberg and Tao 2006), also called Lasso in the statistics literature (Tibshirani 1996). The optimization literature for solving the resulting ℓ_1-type problems is vast. A few examples are interior-point methods (Candès *et al.* 2006, Kim *et al.* 2007), projected gradient methods (Figueiredo, Nowak and Wright 2007), iterative soft thresholding (see Section 8.2.7) (Daubechies *et al.* 2004, Fornasier and Rauhut 2008) and fast proximal gradient methods (FISTA and variants) (Bubeck 2015), to name just a few.

Combinatorial algorithms. A third class of approaches for sparse coding is that of combinatorial algorithms. They are particularly suitable when acquiring highly structured samples of the signal so that rapid reconstruction via group testing is efficient. This class of approaches includes Fourier sampling, chaining pursuit and HHS pursuit (Berinde *et al.* 2008).

The above approaches for solving (2.20) and (2.21) all have their advantages and disadvantages. First of all, greedy methods will generally not give the same solution as convex relaxation. However, if the restricted isometry property (RIP) from Section 2.7.3 holds, then both approaches have the same solution. Convex relaxation has the advantage that it succeeds with a very small number of possibly noisy measurements. However, their numerical solution tends to be computationally burdensome. Combinatorial algorithms, on the other hand, can be extremely fast (sublinear in the length of the target signal) but they require a very specific structure of the forward operator $\mathcal{A}$ and a large number of samples. The performance of greedy methods falls in between those of convex relaxation and combinatorial algorithms in their run-time and sampling efficiency.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

2.7.3. Error estimates for sparse recovery

When the true model parameter is compressible, then it is possible to estimate the error committed by performing sparse recovery. Such estimates have been derived in the finite-dimensional setting when the matrix representing the linear forward operator satisfies the RIP:

$$(1-\epsilon_s)\|f\|_2^2 \leq \|\mathcal{A}(f)\|_2^2 \leq (1+\epsilon_s)\|f\|_2^2 \quad \text{for all } s\text{-sparse } f \in X, \tag{2.23}$$

for sufficiently small $\epsilon_s > 0$. Then we have the following error estimate for sparse recovery.

Theorem 2.8 (Candès, Romberg and Tao 2006). Let $\mathcal{A}\colon \mathbb{R}^n \to \mathbb{R}^m$ be a linear mapping whose matrix satisfies the RIP. If $g = \mathcal{A}(f_{\text{true}}) + e$ with $\|e\| \leq \delta$ and

$$\hat{f}_\delta := \arg\min_{f \in X} \|f\|_1 \quad \text{subject to } \|\mathcal{A}(f) - g\|_2 \leq \delta, \tag{2.24}$$

then

$$\|\hat{f}_\delta - f_{\text{true}}\|_2 \leq C\left[\delta + \frac{\|f_{\text{true}} - f_{\text{true}}^{(s)}\|_2}{\sqrt{s}}\right]. \tag{2.25}$$

In the above, $f_{\text{true}}^{(s)} \in \mathbb{R}^n$ is a vector consisting of the s largest (in magnitude) coefficients of f_{true} and zeros otherwise.

Examples of matrices satisfying RIP are sub-Gaussian matrices and partial bounded orthogonal matrices (Chen and Needell 2016). Theorem 2.8 states that the reconstruction error is at most proportional to the norm of the noise in the data plus the *tail* $f_{\text{true}} - f_{\text{true}}^{(s)}$ of the signal. Cohen, Dahmen and DeVore (2009) show that this error bound is optimal (up to the precise value of C). Moreover, if f_{true} is s-sparse and $\delta = 0$ (noise-free data), then f_{true} can be reconstructed exactly. Furthermore, if f_{true} is compressible with (2.19), then

$$\|\hat{f}_\delta - f_{\text{true}}\|_2 \leq C(\delta + C' s^{1/2-1/q}). \tag{2.26}$$

Finally, error estimates of the above type have been extended to the infinite-dimensional setting in Adcock and Hansen (2016).

The choice of dictionary is clearly a central topic in sparsity-promoting regularization and, as outlined in Section 4.4, the dictionary can be learned beforehand or jointly alongside the signal recovery.

2.7.4. Error estimates for convex relaxation

Applying the convex relaxation to the analysis approach of (2.21) yields the regularized Tikhonov functional

$$\mathcal{R}_\theta(g) := \arg\min_{f \in X} \{\mathcal{L}(\mathcal{A}(f), g) + \lambda \|\mathcal{E}_{\mathbb{D}}(f)\|_p\} \tag{2.27}$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

with $1 \leq p < 2$. In the context of ill-posed inverse problems this functional was introduced and analysed in the ground-breaking paper by Daubechies *et al.* (2004). We emphasize that this analysis holds in infinite-dimensional function spaces, and does not depend on the finite-dimensional concepts used in compressive sampling or finite-dimensional sparse recovery concepts.

Since then, this model has been studied intensively, and in particular the case $p = 1$. Similar to Theorem 2.7, error estimates and regularizing properties, such as existence of minimizers, well-posedness, stability, convergence rates and error estimates, have been obtained for linear and non-linear operators: see *e.g.* Scherzer *et al.* (2009). Minimizers of this functional with $p = 1$ are indeed sparse even if the true solution is not. Using the notion of Bregman distances, such sparsity-promoting approaches have been extended to rather general choices of data discrepancy and regularizers. For a more complete introduction, see Chan and Shen (2006), Scherzer *et al.* (2009) and Bredies *et al.* (2011) and the recent survey by Benning and Burger (2018, Section 2).

3. Statistical regularization

Statistical regularization, and Bayesian inversion in particular, is a complete statistical inferential methodology for inverse problems. It offers a rich set of tools for incorporating data into the recovery of the model parameter, so it is a natural framework to consider when data-driven approaches from machine learning are to be used for solving ill-posed inverse problems.

A key element is to treat *both* the data and model parameter as realizations of certain random variables and phrase the inverse problem as a statistical inference question. In contrast, the functional analytic viewpoint (Section 2) allows for data to be interpreted as samples generated by a random variable, but there are no statistical assumptions on the model parameters.

Remark 3.1. In functional analytic regularization, a statistical model for data is mostly used to justify the choice of data discrepancy in a variational method and for selecting an appropriate regularization parameter. Within functional analytic regularization, one can more carefully account for statistical properties of data which can be useful for uncertainty quantification (Bissantz, Hohage, Munk and Ruymgaart 2007).

Bayesian statistics offers a natural setting for such a quest since it is natural to interpret measured data in an inverse problem as a sample of a random variable conditioned on data whose distribution is the data likelihood. The data likelihood can often be derived using knowledge-driven modelling. Solving an inverse problem can then be stated as finding the distribution of the model parameter conditioned on data (posterior distribution). The

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

posterior describes all possible solutions given measured data, so in particular it provides an estimate of the statistical uncertainty of the solution that can be used for uncertainty quantification. *Many of the challenges in Bayesian inversion are associated with realizing these advantages without having access to the full posterior.* In particular, designing a 'good' prior and to have a computationally feasible means for exploring the posterior is essential for implementing and using Bayesian inversion. This, along with investigating its regularizing properties, drives much of the research in Bayesian inversion.

Motivated by the above, the focus of this brief survey is on Bayesian inversion. As one would expect, the theory was first developed in the finite-dimensional setting, *i.e.* when both model parameter and space and data spaces are finite-dimensional. Important early developments were made in the geophysics community (Tarantola and Valette 1982, Tarantola 2005), which had a great impact in the field; see also Calvetti and Somersalo (2017, Section 2) for a brief historical survey. Nice surveys of the finite-dimensional theory are given in Kaipio and Somersalo (2005) and Calvetti and Somersalo (2017), where many further references can be found.

Our focus is primarily on later developments that deal with Bayesian inversion in the infinite-dimensional (non-parametric) setting. Early work in this direction can be found in Mandelbaum (1984) and Lehtinen, Päivärinta and Somersalo (1989). Our brief survey in Sections 3.1, 3.2.1 and 3.2.2 is based on the excellent survey papers by Evans and Stark (2002), Stuart (2010) and Dashti and Stuart (2017). The sections concerning convergence (Section 3.2.3), convergence rates (Section 3.2.3) and characterization of the Bayesian posterior (Section 3.2.5) are based on the excellent short survey by Nickl (2017*b*).

3.1. Basic notions for Bayesian inversion

The vector spaces X and Y play the same role as in functional analytic regularization (Section 2), that is, elements in X represent model parameters and elements in Y represent data. For technical reasons, we also assume that both X and Y are separable Banach spaces. Both these spaces are also equipped with a Borel σ-algebra, and we let $\mathscr{P}_Y$ and $\mathscr{P}_Y$ denote the class of probability measures on X and Y, respectively. We also assume there exists a $(X \times Y)$-valued random variable $(\mathbb{f}, \mathbb{g}) \sim \mu$ that is distributed according to some joint law μ. Here, $\mathbb{f}$ generates elements in X (model parameters) and $\mathbb{g}$ generates elements in Y (data).

Remark 3.2. In this setting, integrals over X and/or Y, which are needed for defining expectation, are interpreted as a Bochner integral that extends the Lebesgue integral to functions that take values in a Banach space. See

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Dashti and Stuart (2017, Section A.2) for a brief survey of Banach and Hilbert space-valued random variables.

3.1.1. The data model

A key assumption is that the conditional distribution of $(\mathbb{g} \mid \mathbb{f} = f) \sim \Pi^f_{\text{data}}$ exists (data likelihood) under the joint law μ for any $f \in X$. This allows us to define the *data model*, which is the statistical analogue of the forward operator, as the following $\mathscr{P}_Y$-valued mapping defined on X:

$$f \mapsto \Pi^f_{\text{data}} \quad \text{for any } f \in X. \tag{3.1}$$

Remark 3.3. The existence of regular conditional distributions can be ensured under very general assumptions. In particular, let $\mathbb{f}$ be an X-valued random variable and $\mathbb{g}$ a Y-valued random variable, where X is a Polish space, *i.e.* a complete and separable metric space, and Y is a general measurable space. Then there exists a regular conditional distribution of the conditional random variable $(\mathbb{f} \mid \mathbb{g})$ (Kallenberg 2002, Theorem 6.3). In particular, when both X and X are Polish spaces, then both $(\mathbb{f} \mid \mathbb{g})$ and $(\mathbb{g} \mid \mathbb{f})$ exist.

The most common data model is the statistical analogue of (2.1) where the model parameter is allowed to be a random variable (Kallenberg 2002, Lemma 1.28 and Corollary 3.12):

$$\mathbb{g} = \mathcal{A}(\mathbb{f}) + \mathbb{e}. \tag{3.2}$$

Here, $\mathcal{A}\colon X \to Y$ is the same forward operator as in (2.1), which models how data are generated in the absence of noise. Likewise, $\mathbb{e} \sim \Pi_{\text{noise}}$, with $\Pi_{\text{noise}} \in \mathscr{P}_Y$ known, is the random variable that generates the observation noise. If $\mathbb{e}$ is independent from $\mathbb{f}$, then (3.2) amounts to the data model

$$\Pi^f_{\text{data}} = \delta_{\mathcal{A}(f)} \circledast \Pi_{\text{noise}} = \Pi_{\text{noise}}(\,\cdot\, - \mathcal{A}(f)) \quad \text{for any } f \in X, \tag{3.3}$$

where $\circledast$ denotes convolution between measures.

Remark 3.4. Another common data model is when Π^f_{data} is a Poisson random measure on Y with mean equal to $\mathcal{A}(f)$ (Hohage and Werner 2016, Streit 2010, Vardi *et al.* 1985, Besag and Green 1993). This is suitable for modelling statistical properties of low-dose imaging data, such as data that is measured in line of response PET (Kadrmas 2004, Calvetti and Somersalo 2008, Section 3.2 of Natterer and Wübbeling 2001) and variants of fluorescence microscopy (Hell, Schönle and Van den Bos 2007, Diaspro *et al.* 2007).

3.1.2. The inverse problem

Following Evans and Stark (2002), a *(statistical) inverse problem* requires us to perform the task of recovering the posterior given a single sample

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

(measured data) from the data model with unknown true model parameter. A more precise statement reads as follows.

Definition 3.5. A *statistical inverse problem* is the task of recovering the conditional distribution $\Pi^g_{\text{post}} \in \mathscr{P}_Y$ of $(\mathbb{f} \mid \mathbb{g} = g)$ under μ from measured data $g \in Y$, where

$$g \text{ is a single sample of } (\mathbb{g} \mid \mathbb{f} = f_{\text{true}}) \sim \Pi^{f_{\text{true}}}_{\text{data}}. \tag{3.4}$$

Here $f_{\text{true}} \in X$ is unknown while $f \mapsto \Pi^f_{\text{data}}$, which describes how data are generated, is known.

The conceptual difference that comes from adopting such a statistical view brings with it several potential advantages. The posterior, assuming it exists, describes all possible solutions, so recovering it represents a more complete solution to the inverse problem than recovering an approximation of f_{true}, which is the goal in functional analytic regularization. This is particularly the case when one seeks to quantify the uncertainty in the recovered model parameter in terms of statistical properties of the data. However, recovering the entire posterior is often not feasible, such as in inverse problems that arise in imaging. As an alternative, one can settle for exploring the posterior by computing suitable estimators (Section 3.3). Some may serve as approximations of f_{true} whereas others are designed for quantifying the uncertainty.

3.1.3. Bayes' theorem

A key part of solving the above inverse problem is to utilize a relation between the unknown posterior that one seeks to recover and the known data likelihood. Such a relation is given by Bayes' theorem.

A very general formulation of Bayes' theorem is given by Schervish (1995, Theorem 1.31). For simplicity we consider the formulation that holds for the special case where the data likelihood is given as in (3.2), with $\mathcal{A}\colon X \to Y$ a measurable map. Assume furthermore that $\mathbb{f} \sim \Pi_{\text{prior}}$ and $\mathbb{e} \sim \Pi_{\text{noise}}$, with $\mathbb{e}$ independent of $\mathbb{f}$. Then the data model is given as in (3.3), that is, at f it yields the translate of Π_{noise} by $\mathcal{A}(f)$.

Assume next that $\Pi^f_{\text{data}} \ll \Pi_{\text{noise}}$ (*i.e.* Π^f_{data} is absolutely continuous with respect to Π_{noise}) Π_{noise}-almost surely for all $f \in X$, so there exists some measurable map $\Phi\colon X \times Y \to \mathbb{R}$ (potential) such that

$$\frac{\mathrm{d}\Pi^f_{\text{data}}}{\mathrm{d}\Pi_{\text{noise}}}(g) = \exp(-\mathcal{L}(f, g)) \quad \text{for all } f \in X. \tag{3.5}$$

with

$$\mathbb{E}_{\mathbb{g} \sim \Pi_{\text{noise}}}[\exp(-\mathcal{L}(f, \mathbb{g}))] = 1.$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

The mapping $f \mapsto -\mathcal{L}(f, g)$ is called the (*data*) *log-likelihood* for the data $g \in Y$.

Remark 3.6. The equality for the Radon–Nikodym derivative in (3.5) means that

$$\mathbb{E}_{\mathrm{g}\sim\Pi^f_{\mathrm{data}}}[F(\mathrm{g})] = \mathbb{E}_{\mathrm{g}\sim\Pi_{\mathrm{noise}}}[\exp(-\mathcal{L}(f,\mathrm{g}))F(\mathrm{g})]$$

holds for any measurable $F\colon Y \to \mathbb{R}$.

Finally, assume $\mathcal{L}$ is μ_0-measurable, where $\mu_0 := \Pi_{\mathrm{prior}} \otimes \Pi_{\mathrm{noise}}$ and $\mu \ll \mu_0$, which means in particular that the joint law $(\mathbb{f}, \mathrm{g}) \sim \mu$ can be written as

$$\frac{\mathrm{d}\mu}{\mathrm{d}\mu_0}(f, g) = \exp(-\mathcal{L}(f, g)) \quad \text{for } (f, g) \in X \times Y.$$

Bearing in mind the above assumptions, we can now state Bayes' theorem (Dashti and Stuart 2017, Theorem 14), which expresses the posterior in terms of the data likelihood and the prior.

Theorem 3.7 (Bayes' theorem). The normalization constant $Z\colon Y \to \mathbb{R}$ is defined by

$$Z(g) := \mathbb{E}_{\mathbb{f}\sim\Pi_{\mathrm{prior}}}[\exp(-\mathcal{L}(\mathbb{f}, g))] \tag{3.6}$$

and we assume $Z(g) > 0$ holds Π_{noise}-almost surely for $g \in Y$. Then the posterior Π^g_{post}, which is the conditional distribution of $(\mathbb{f} \mid \mathrm{g} = g)$, exists under μ and it is absolutely continuous with respect to the prior, *i.e.* $\Pi^g_{\mathrm{post}} \ll \Pi_{\mathrm{prior}}$. Furthermore,

$$\frac{\mathrm{d}\Pi^g_{\mathrm{post}}}{\mathrm{d}\Pi_{\mathrm{prior}}}(f) = \frac{1}{Z(g)} \exp(-\mathcal{L}(f, g)) \tag{3.7}$$

holds μ_0-almost surely for $(f, g) \in X \times Y$.

Bayes' theorem is the basis for Bayesian inversion, where one seeks to solve the statistical inverse problem assuming access to *both* a prior and a data likelihood, whereas $f_{\mathrm{true}} \in X$ remains unknown. The data likelihood is given by the data model, which is in turn derived from knowledge about how data are generated. The choice of prior, however, is more subtle: it needs to act as a regularizer, and ideally it also encodes subjective prior beliefs about the unknown model parameter f_{true} by giving high probability to model parameters similar to f_{true} and low probability to other 'unnatural' model parameters. A brief survey of hand-crafted priors is provided in Section 3.4.

3.2. Regularization theory for Bayesian inversion

In functional analytic regularization, existence, stability and convergence are necessary if a reconstruction method is to be a regularization (see

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Section 2.4). Moreover, a mathematical analysis also seeks to provide convergence rates and stability estimates. There is an ongoing effort to develop a similar theory for Bayesian inversion.

Methods for functional analytic regularization of ill-posed inverse problems typically regularize by a variational procedure or a spectral cut-off. In Bayesian inversion, regularization is mainly through the choice of an appropriate prior distribution. Since many different priors can serve as regularizers, a large portion of the theory seeks to characterize properties of Bayesian inversion methods that are independent of the prior. Much of the analysis in Sections 3.2.3–3.2.5 is therefore performed in the large-sample or small-noise limit, and under the assumption that data g is generated from $(\mathrm{g} \mid \mathbb{f} = f)$ where $f = f_{\text{true}}$ (the true model parameter), instead of having f as a random sample of $\mathbb{f} \sim \Pi_{\text{prior}}$ (prior).

Remark 3.8. The parametric setting refers to the case when the dimensions of X and Y are finite. In this context, a large-sample limit refers to an asymptotic analysis performed when the dimension of X is kept fixed and independent of the dimension of Y (sample size), which is allowed to grow. In the non-parametric setting, either both Y and X are infinite-dimensional from the outset, or one lets the dimension of X increase as the dimension of Y (sample size) increases.

3.2.1. Existence

Existence for Bayesian inversion follows when Bayes' theorem holds. Below we state the precise existence theorem (Dashti and Stuart 2017, Theorem 16) for the setting in Section 3.1.3, which covers the case when model parameter and data spaces are infinite-dimensional separable Banach spaces.

Theorem 3.9 (existence for Bayes inversion). Assume that $\mathcal{L}\colon X \times Y \to \mathbb{R}$ in (3.5) is continuously differentiable on some $X' \subset X$ that contains the support of the prior Π_{prior} and $\Pi_{\text{prior}}(X' \bigcap B) > 0$ for some bounded set $B \subset X$. Also, assume there exists mappings $M_1, M_2\colon \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}_+$ that are component-wise monotone, non-decreasing and where the following holds:

$$\begin{aligned} -\mathcal{L}(f,g) &\le M_1(r, \|f\|), \\ |\mathcal{L}(f,g) - \mathcal{L}(f,v)| &\le M_2(r, \|f\|)\|g - v\| \end{aligned} \tag{3.8}$$

for $f \in X$ and $g, v \in B_r(0) \subset Y$. Then, Z in (3.6) is finite, *i.e.* $0 < Z(g) < \infty$ for any $g \in Y$, and the posterior given by (3.7) yields a well-defined $\mathscr{P}_X$-valued mapping on Y: $g \mapsto \Pi^g_{\text{post}}$.

Under certain circumstances it is possible to work with improper priors on X, for example by computing posterior distributions that approximate the posteriors one would have obtained using proper conjugate priors whose extreme values coincide with the improper prior.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

3.2.2. Stability

One can show that small changes in the data lead to small changes in the posterior distribution (in Hellinger metric) on $\mathscr{P}_X$. The precise formulation given by Dashti and Stuart (2017, Theorem 16) reads as follows.

Theorem 3.10 (stability for Bayes inversion). Let the assumptions in Theorem 3.9 and assume in addition that

$$f \mapsto \exp(M_1(r, \|f\|))(1 + M_2(r, \|f\|))^2 \quad \text{is } \Pi_{\text{prior}}\text{-integrable on } X \tag{3.9}$$

for some fixed $r > 0$. Then

$$d_{\text{H}}(\Pi^g_{\text{post}}, \Pi^v_{\text{post}}) \leq C(r)\|g + v\|$$

for some $C(r) > 0$ and $g, v \in B_r(0) \subset Y$. In the above, $d_{\text{H}} \colon \mathscr{P}_X \times \mathscr{P}_X \to \mathbb{R}$ is the Hellinger metric (see Dashti and Stuart 2017, Definition 4).

Theorem 3.10 holds in particular when the negative log-likelihood of data is locally Hölder-continuous, which is the case for many standard probability distribution functions, for example when the negative log-likelihood is continuously differentiable.

By Theorem 3.10 the posterior is Lipschitz in the data with respect to the Hellinger metric, so rephrasing the inverse problem as the task of recovering the posterior instead of a model parameter acts as a regularization of an inverse problem that is ill-posed in the functional analytic sense. Note, however, that the above does not automatically imply that a particular estimator is continuous with respect to data. However, the Hellinger distance possesses the convenient property that continuity with respect to this metric implies the continuity of moments. Hence, a corollary to Theorem 3.10 is that *posterior moments, such as the mean and covariance, are continuous (Sprungk 2017, Corollary 3.23), that is, these estimators are regularizing.*

As a final note, one can also show that small changes in the data log-likelihood $\mathcal{L}$ in (3.5) lead to small changes in the posterior distribution, again in the Hellinger metric (Dashti and Stuart 2017, Theorem 18). This enables one to translate errors arising from inaccurate forward operator into errors in the Bayesian solution of the inverse problem, a topic that is also considered in Section 6.2.

3.2.3. Convergence

Posterior consistency is the Bayesian analogue of the notion of convergence in functional analytic regularization. More precisely, the requirement is that the posterior Π^g_{post}, where g is a sample of $(\mathrm{g} \mid \mathbb{f} = f_{\text{true}})$, concentrates in any small neighbourhood of the true model parameter $f_{\text{true}} \in X$ as information

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

in data g increases indefinitely.[1] Intuitively, this means our knowledge about the model parameter becomes more accurate and precise as the amount of data increases indefinitely.

In the finite-dimensional setting, consistency of the posterior holds if and only if $f_{\text{true}} \in X$ is contained in the support of the prior Π_{prior} (Freedman 1963, Schwartz 1965) (provided the posterior is smooth with respect to the model parameter).

The situation is vastly more complex in the infinite-dimensional (non-parametric) setting. It is known that the posterior is consistent at every model parameter except possibly on a set of measure zero (Doob 1948, Breiman, Le Cam and Schwartz 1965), that is, Bayesian inversion is almost always consistent in the measure-theoretic sense. However, the situation changes if 'smallness' is measured in a topological sense, as shown by a classical counter-example in Freedman (1963) involving the simplest non-parametric problem where consistency fails. This is not a pathological counter-example: it is a generic property in the sense that most priors are 'bad' in a topological sense (Freedman 1965). In fact, consistency may fail for non-parametric models for very natural priors satisfying the support condition, which means even an infinite amount of data may not be sufficient to 'correct' for errors introduced by a prior. Hence, unlike the finite-dimensional setting, many priors do not 'wash out' as the information in the data increases indefinitely, so the prior may have a large influence on the corresponding posterior even in the asymptotic setting. Examples of posterior consistency results for Bayesian inversion are given in Ghosal, Ghosh and Ramamoorthi (1999), Ghosal, Ghosh and van der Vaart (2000), Neubauer and Pikkarainen (2008), Bochkina (2013), Agapiou, Larsson and Stuart (2013), Stuart and Teckentrup (2018), Kekkonen, Lassas and Siltanen (2016) and Kleijn and Zhao (2018).

To summarize, there are many reasonable priors for which posterior consistency holds at every point of the model parameter space X. A general class of such 'good' priors is that of tail-free priors (Freedman 1963) and neutral-to-the-right priors (Doksum 1974); see also Le Cam (1986, Section 17.7). The fact that there are too many 'bad' priors may therefore not be such a serious concern since there are also enough good priors approximating any given subjective belief. It is therefore important to have general results providing sufficient conditions for the consistency given a pair of model parameter and prior as developed, and Schwartz (1965) is an example in this direction.

[1] Increasing the 'information in data indefinitely' means $\varepsilon \to 0$ in (3.2), and if the data space Y is finite-dimensional, one also lets its dimension (sample size) increase. See Ghosal and van der Vaart (2017, Definition 6.1) for the precise definition.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

3.2.4. Convergence rates

Posterior consistency is a weak property shared by many different choices of priors. More insight into the performance of Bayesian inversion under the choice of different priors requires characterizing other properties of the posterior, such as quantifying how quickly it converges to the true solution as the observational noise goes to 0. This leads to results about *contraction rates*, which is the Bayesian analogue of convergence rate theorems in functional analytic regularization.

Formally, consider the setting in (3.2) where the observational noise tends to zero as some scalar $\delta \to 0$, that is, for some fixed $f_{\text{true}} \in X$ we have

$$\mathrm{g} = \mathcal{A}(f_{\text{true}}) + \mathrm{e}_\delta \quad \text{where } \|\mathrm{e}_\delta\| \to 0 \text{ as } \delta \to 0. \tag{3.10}$$

A contraction rate theorem seeks to find the base rate for $\epsilon(\delta) \to 0$ such that

$$\Pi^{g^\delta}_{\text{post}}(\{f \in X : \ell_X(f, f_{\text{true}}) \geq \epsilon(\delta)\}) \to 0$$

$\Pi^{g_0}_{\text{post}}$-almost surely as $\epsilon \to 0$. Here, $g^\delta \in Y$ is a single sample of g in (3.10), $g_0 = \mathcal{A}(f_{\text{true}})$ is the corresponding ideal data, $\{\Pi^\delta_{\text{prior}}\}_\delta \subset \mathscr{P}_X$ is a sequence of prior distributions with $f \sim \Pi^\delta_{\text{prior}}$, and $\ell_X \colon X \times X \to \mathbb{R}$ is a (measurable) distance function on X. Research in this area has mainly focused on (a) obtaining contraction rates for Bayesian inversion where the prior is from some large class, or (b) improving the rates by changing the parameters of the prior depending on the level of noise (and even the data). Most of the work is done for additive observational white Gaussian noise, that is, the setting in (3.10) with $\mathrm{e}_\delta = \delta\mathbb{W}$, where $\mathbb{W}$ is a centred Gaussian white noise process that can be defined by its action on a separable Hilbert space Y. See Dashti and Stuart (2017, Sections A.3 and A.4) for a survey related to Gaussian measures and Wiener processes on separable Banach spaces.

As one might expect, initial results were for linear forward operators. The first results restricted attention to conjugate priors where the posterior has an explicit expression (it is in the same family as the prior) (Liese and Miescke 2008, Section 1.2). However, Bayesian non-parametric statistics often carries over to the inverse setting via the singular value decomposition of the linear forward operator. This allowed one to prove contraction rate results for Bayesian inversion with non-conjugate priors (Knapik, van der Vaart and van Zanten 2011, Knapik, van der Vaart and van Zanten 2013, Ray 2013, Agapiou *et al.* 2013, Agapiou, Stuart and Zhang 2014). In particular, if the Gaussian prior is non-diagonal for the singular value decomposition (SVD), then the posterior is still Gaussian and its contraction rate will be driven by the convergence rate of its posterior mean (since the variance does not depend on the data). Furthermore, in the linear setting, the posterior mean is the MAP, so the convergence rate of the MAP will be a posterior contraction rate.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Unfortunately, not many of the methods developed for proving contraction rates in the linear setting carry over to proving contraction rates for general, non-linear, inverse problems. A specific difficulty in the Bayesian setting is that the noise term often does not take values in the natural range of the forward operator. For example, consider the data model in (3.3) with observational white Gaussian noise, *i.e.* (3.10) with $\mathbb{e}_\delta = \delta\mathbb{W}$, where $\mathbb{W}$ is a centred Gaussian white noise process that can be defined by its action on a separable Hilbert space Y. If we have a non-linear forward operator $\mathcal{A}$ given as the solution operator to an elliptic PDE, then the noise process $\mathbb{W}$ does not define a proper random element in $Y = L^2(\mathbb{M})$ for $\mathbb{M} \subset \mathbb{R}^d$: instead it defines a random variable only in a negative Sobolev space $W^{-\beta}$ with $\beta > d/2$. Nevertheless, there are also some results in the non-linear setting: for example, Kekkonen *et al.* (2016) derive contraction rate results for Bayesian inversion of inverse problems under Gaussian conjugate priors and where the forward operator is a linear hypoelliptic pseudodifferential operator. Recent developments that make use of techniques from concentration of measure theory have resulted in contraction rate theorems outside the conjugate setting. For example, Nickl and Söhl (2017) and Nickl (2017*a*) derive contraction rates for Bayesian inversion for parameter estimation involving certain classes of elliptic PDEs that are minimax-optimal in prediction loss: see *e.g.* Nickl (2017*a*, Theorem 28) for an example of a general contraction theorem. The case of a (possibly) non-linear forward operator and a Gaussian prior is considered in Nickl, van de Geer and Wang (2018), which studies properties of the MAP estimator. Finally, Gugushvili, van der Vaart and Yan (2018) derive contraction rates for 'general' priors expressed by scales of smoothness classes. The precise conditions are checked only for an elliptic example: it is not clear whether it works in other examples, such as the ray transform.

Remark 3.11. The proof techniques used for obtaining contraction rates in the non-linear setting depend on stability estimates for the forward problem that allow us to control $\|f - h\|$ in terms of $\|\mathcal{A}(f) - \mathcal{A}(h)\|$ in some suitable norm, and a dual form of the usual regularity estimates for solutions of PDEs that encodes the (functional analytic) ill-posedness of the problem. With estimates, one can use methods from non-parametric statistics to prove contraction rates for Bayesian inversion, with priors that do not require identifying a singular value-type basis underlying the forward operator.

3.2.5. Characterization of the posterior for uncertainty quantification

Posterior consistency and contraction rates are relevant results, but assessing the performance of Bayesian inversion methods in uncertainty quantification requires a more precise characterization of the posterior. The aim

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

is to characterize the fluctuations of $(\mathbb{f} \mid \mathbb{g} = g)$ near f_{true} when scaled by some inverse contraction rate.

One approach is to derive Bernstein–von Mises type theorems, which characterize the posterior distribution in terms of a canonical Gaussian distribution in the small-noise or large-sample limit. To better illustrate the role of such theorems, we consider the special case with Gaussian prior on the Hilbert space X and observational noise $\mathbb{e}_\delta = \delta\mathbb{W}$ in (3.10) (white noise model) where $\mathbb{W}$ denotes a Gaussian white noise process in the Hilbert space Y, that is,

$$\mathbb{g}^\delta = \mathcal{A}(f_{\text{true}}) + \delta\mathbb{W}.$$

Many of the results on contraction rates (Section 3.2.4) for Bayesian inversion in this setting are obtained for the distance function induced by the L^2-norm, so it is natural to initially consider the statistical fluctuations in L^2 of the random variable

$$\mathbb{z}_\delta := \frac{1}{\epsilon_\delta}(\mathbb{E}[\mathbb{f} \mid \mathbb{g} = g^\delta] - f_{\text{true}}).$$

Now, it turns out that there is no Bernstein–von Mises type asymptotics for $\mathbb{z}_\delta$ as the noise level δ tends to zero, that is, there is no Gaussian process $(\mathbb{G}(\phi))_{\phi \in C^\infty}$ such that

$$\left(\frac{1}{\delta}\langle \mathbb{z}_\delta - \mathbb{E}[\mathbb{z}_\delta], \phi\rangle_{L^2}\right)_{\phi \in C^\infty} \to (\mathbb{G}(\phi))_{\phi \in C^\infty} \quad \text{weakly as } \delta \to 0. \tag{3.11}$$

To sidestep this difficulty, Castillo and Nickl (2013, 2014) seek to determine maximal families Ψ that replace C^∞ in (3.11) and where such an asymptotic characterization holds. This leads to non-parametric Bernstein–von Mises theorems, and while Castillo and Nickl (2013, 2014) considered 'direct' problems in non-parametric regression and probability density estimation, recent papers have obtained non-parametric Bernstein–von Mises theorems for certain classes of inverse problems. For example, Monard, Nickl and Paternain (2019) consider the case of inverting the (generalized) ray transform, whereas Nickl (2017*a*) considers PDE parameter estimation problems. The case with general linear forward problem is treated by Giordano and Kekkonen (2018), who build upon the techniques of Monard *et al.* (2019).

To give a flavour of the type of results obtained, we consider Theorem 2.5 of Monard *et al.* (2019), which is relevant to tomographic inverse problems involving inversion of the ray transform for recovering a function (images) defined on $\Omega \subset \mathbb{R}^d$, *i.e.* $X \subset L^2(\Omega)$. This theorem states that

$$\frac{1}{\delta}\langle(\mathbb{f} \mid \mathbb{g} = g^\delta) - \mathbb{E}[\mathbb{f}|\mathbb{g} = g^\delta], \phi\rangle_{L^2} \to \mathrm{N}(0, \| \mathcal{A} \circ (\mathcal{A}^* \circ \mathcal{A})^{-1}(\phi)\|_Y) \tag{3.12}$$

as $\delta \to 0$ for any $\phi \in C^\infty(\Omega)$.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

The convergence is in Π_{prior}-probability and the Y-norm on the right-hand side is a natural L^2-norm on the range of the ray transform. The limiting covariance is also shown to be minimal, that is, it attains the semi-parametric Cramér–Rao lower bound (or 'inverse Fisher information') for estimating $\langle f, \phi \rangle_{L^2}$ near f_{true}. A key step in the proof is to show smoothness of the 'Fisher information' operator $(\mathcal{A}^* \circ \mathcal{A})^{-1} \colon X \to X$, which is done using techniques from microlocal analysis (Monard *et al.* 2019, Theorem 2.2). The existence and mapping properties of this inverse Fisher information operator in (3.12) also plays a crucial role in proving that Bernstein–von Mises theorem for other inverse problems. For those with a non-linear forward operator, the information operator to be inverted is found after linearization as shown in Nickl (2017*a*) for parameter estimation in an elliptic PDE.

Relevance to applications. For large-scale problems, such as those arising in imaging, it is computationally challenging to even explore the posterior beyond computing point estimators (Section 3.5). This holds in particular for the task of computing Bayesian credible sets, which are relevant for uncertainty quantification. Hence, the theoretical results in Section 3.2.5 can be of interest in practical applications, since these provide good analytic approximations of the posterior that are much simpler to compute.

In particular, if a Bernstein–von Mises theorem holds, then the Bayesian and maximum likelihood estimators have the same asymptotic properties and the influence of the prior diminishes asymptotically as information in the data increases. Then, Bayesian credible sets are asymptotically equivalent to frequentist confidence regions,[2] so Bayesian inversion with 95% posterior credibility will have approximately 0.95 chance of returning the correct decision in repeated trials.

Many of the results, however, assume a Gaussian prior and data likelihood: for non-Gaussian problems in an infinite-dimensional setting the structure of the posterior can be very chaotic and difficult to characterize. Furthermore, the issue with any such asymptotic characterization of the posterior is that it is based on increasing information in data indefinitely. In reality data are fixed, and it is quite possible that the posterior is (very) non-normal and possibly multi-modal even though it behaves asymptotically like a Gaussian (or more generally the sampling distribution of a maximum likelihood estimator). Hence, a regularizing prior that provides 'best' contraction rates may not necessarily be the best when it comes to image quality for data with a given noise level. Here one would need to consider a prior that acts as more than just a regularizer. Finally, the assumptions

[2] A Bayesian credible set is a subset of the model parameter space X that contains a predefined fraction, say 95%, of the posterior mass. A frequentist confidence region is a subset of X that includes the unknown true model parameter with a predefined frequency as the experiment is repeated indefinitely.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

of the Bernstein–von Mises theorem are fragile and easy to violate for a dataset or analysis, and it is difficult to know, without outside information, when this will occur. As nicely outlined in Nickl (2013, Section 2.25), the parametric (finite-dimensional) setting already requires many assumptions, such as a consistent maximum likelihood estimator, a true model parameter in the support of the prior, and a log-likelihood that is sufficiently regular. For example, data in an inverse problem are observational, and therefore it is unlikely that an estimator of f_{true}, such as maximum likelihood or the posterior mean, is consistent, in which case a Bernstein–von Mises theorem does not apply.

3.3. Reconstruction method as a point estimator

In most large-scale inverse problems it is computationally very challenging, if not impossible, to recover the entire posterior. However, a reconstruction method that seeks to compute an estimator for the posterior formally defines a mapping $\mathcal{R}\colon Y \to X$ (reconstruction operator). As such, it can be viewed as non-randomized decision rule (point estimator) in a statistical estimation problem: see Liese and Miescke (2008, Definition 3.1) for the formal definition. Computing a suitable point estimator is therefore an alternative to seeking to recover the posterior.

In the infinite-dimensional setting, the benefits of using one estimator rather than another (*e.g.* the conditional mean estimate rather than the MAP estimate) are not well understood. Statistical decision theory provides criteria for selecting and comparing decision rules, which can be used when selecting the estimator (reconstruction method). This requires phrasing the inverse problem as a statistical decision problem. More precisely, the tuple $((Y, \mathfrak{S}_Y), \{\Pi^f_{\text{data}}\}_{f\in X})$ defines a statistical model which is parametrized by the Banach space X. The inverse problem is now a statistical decision problem (Liese and Miescke 2008, Definition 3.4) where the statistical model is parametrized by X, the decision space is $D := X$, and a given *loss function*:

$$\ell_X\colon X \times X \to \mathbb{R}. \tag{3.13}$$

The loss function is used to define the risk, which is given as the Π^f_{data}-expectation of the loss function. The risk seeks to quantify the downside that comes with using a particular reconstruction operator $\mathcal{R}\colon Y \to X$.

3.3.1. Some point estimators

Here we briefly define the most common point estimators that are commonly considered in solving many inverse problems. Overall, estimators that include an integration over X are computationally demanding. These include the conditional mean and many estimators associated with uncertainty quantification.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Maximum likelihood estimator. This estimator maximizes the data likelihood. Under the data model in (3.5), the (maximum likelihood) reconstruction operator $\mathcal{R}\colon Y \to X$ becomes

$$\mathcal{R}(g) \in \operatorname*{arg\,max}_{f \in X} \{\exp(-\mathcal{L}(f, g))\} = \operatorname*{arg\,min}_{f \in X} \mathcal{L}(f, g).$$

The advantage of the maximum likelihood estimator is that it does not involve any integration over X, so it is computationally feasible to use on large-scale problems. Furthermore, it only requires access to the data likelihood (no need to specify a prior), which is known. On the other hand, it does not act as a regularizer, so it is not suitable for ill-posed problems.

Maximum a posteriori *(MAP) estimator.* This estimator maximizes the posterior probability, that is, it is the 'most likely' model parameter given measured data g. In the finite-dimensional setting, the prior and posterior distribution can typically be described by densities with respect to the Lebesgue measure, and the MAP estimator is defined as the reconstruction operator $\mathcal{R}\colon Y \to X$ given by

$$\mathcal{R}(g) := \operatorname*{arg\,max}_{f \in X} \pi_{\text{post}}(f \mid g) = \operatorname*{arg\,min}_{f \in X} \{\mathcal{L}(f, g) - \log \pi_{\text{prior}}(f)\}.$$

The second equality above assumes a data model as in (3.5).

For many of the infinite-dimensional spaces there exists no analogue of the Lebesgue measure, which makes it difficult to define a MAP estimator through densities. One way to work around this technical problem is to replace the Lebesgue measure with a Gaussian measure on X. Hence, we assume the posterior and prior have densities with respect to some fixed centred (mean-zero) Gaussian measure μ_0 and E denotes its Cameron–Martin space.[3] Now, following Dashti, Law, Stuart and Voss (2013), we consider the centre of a small ball in X with maximal probability, and then study the limit of this centre as the radius of the ball shrinks to zero. Stated precisely, given fixed data $g \in Y$, assume there is a functional $\mathcal{J}\colon E \to \mathbb{R}$ that satisfies

$$\lim_{r \to 0} \frac{\Pi^g_{\text{post}}(B_r(f_1))}{\Pi^g_{\text{post}}(B_r(f_2))} = \exp(\mathcal{J}(f_1) - \mathcal{J}(f_2)).$$

Here, $B_r(f) \subset X$ is the open ball of radius $r > 0$ centred at $f \in X$ and $\mathcal{J}$ is the Onsager–Machlup functional (Ikeda and Watanabe 1989, p. 533).

[3] The Cameron–Martin space E associated with $\Pi \in \mathscr{P}_X$ consists of elements $f \in X$ such that $\delta_f \circledast \Pi \ll \Pi$, that is, the translated measure $B \mapsto \Pi(B - f)$ is absolutely continuous with respect to Π. The Cameron–Martin space is fundamental when dealing with the differential structure in X, mainly in connection with integration by parts formulas, and it inherits a natural Hilbert space structure from the space X^*.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

For any fixed $f_1 \in X$, a model parameter $f_2 \in X$ for which the above limit is maximal is a natural candidate for the MAP estimator and is clearly given by minimizers of the Onsager–Machlup functional. The advantage of the MAP estimator is that it does not involve any integration over X. Furthermore, the prior often acts as a regularizer, so MAP can be useful for solving an ill-posed inverse problem. A disadvantage is that a MAP may not always exist (Section 3.3.2): one needs to provide an explicit prior, and many priors result in a non-smooth optimization problem. Due to the latter, MAP estimation is computationally more challenging than maximum likelihood estimation.

Conditional (posterior) mean. The reconstruction operator $\mathcal{R}\colon Y \to X$ is here defined as

$$\mathcal{R}(g) := \mathbb{E}[\mathbb{f} \mid \mathbb{g} = g] = \int_X f \, \mathrm{d}\Pi^g_{\mathrm{post}}(f).$$

This estimator involves integration over X, making it challenging to use on even small- to mid-scale problems. It also assumes access to the posterior, which in turn requires a prior. It does, however, act as a regularizer (Sprungk 2017, Corollary 3.23) and therefore it is suitable for ill-posed problems.

Bayes estimator. One starts by specifying a function $\ell_X\colon X \times X \to \mathbb{R}$ that quantifies proximity in X (it does not have to be a metric). Then, a Bayes estimator minimizes the expected loss with respect to the prior Π_{prior}, that is, $\mathcal{R}\colon Y \to X$ is defined as

$$\mathcal{R}(g) := \widehat{\mathcal{R}}(g), \quad \text{where } \widehat{\mathcal{R}} \in \operatorname*{arg\,min}_{\mathcal{R}\colon Y \to X} \mathbb{E}_{(\mathbb{f},\mathbb{g})\sim\mu}[\ell_X(\mathcal{R}(\mathbb{g}), \mathbb{f})]. \tag{3.14}$$

This estimator is challenging to compute even for small- to mid-scale problems, due to the integration over $X \times Y$ and the minimization over all non-randomized decision rules. It also requires access to the joint distribution μ, which by the law of total probability is expressible as $\mu = \Pi_{\mathrm{prior}} \otimes \Pi^f_{\mathrm{data}}$ with known data likelihood Π^f_{data}. An important property of the Bayes estimator is that it is a regularizer, so it is suitable for ill-posed problems. It is also equivalent to the conditional mean when the loss is taken as the square of the L^2-norm (Helin and Burger 2015) and the conditional median when the loss is the L^1-norm. Furthermore, Theorem 1 of Burger and Lucka (2014) shows that a MAP estimator with a Gibbs-type prior where the energy functional is Lipschitz-continuous and convex equals a Bayes estimator where the loss is the Bregman distance of the aforementioned energy functional. Finally, in the finite-dimensional setting, one can show (Banerjee, Guo and Wang 2005) that the Bayes estimator is the same as the conditional mean

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

if and only if ℓ_X is the Bregman distance of a strictly convex non-negative differentiable functional.[4]

Other estimators. Another important family of estimators in statistical decision theory is that of minimax estimators that minimize the maximum loss. Estimators for uncertainty quantification typically involve higher-order moments, such as the variance, or interval estimators that are given as a set of points in the model parameter space X.

3.3.2. Relation to functional analytic regularization

It is quite common to interpret a variational method as in (2.7) as a MAP estimators. This is especially the case when one chooses the data discrepancy $\mathcal{L}$ so that minimizing $f \mapsto \mathcal{L}(\mathcal{A}(f), g)$ corresponds to computing a maximum likelihood estimator.

Such an interpretation is almost always possible in the finite-dimensional setting since, as mentioned in Definition 2.2, the regularization functional $\mathcal{S}_\theta$ can be interpreted as a Gibbs-type prior $\rho_{\text{prior}}(f) \propto \exp(-\mathcal{S}_\theta(f))$ (Kaipio and Somersalo 2005) where ρ_{prior} is the density for the prior. The situation is more complicated in the infinite-dimensional setting since a MAP estimator does not always exist, and if it exists then there is no general scheme for connecting the topological description of a MAP estimate to a variational problem. The main reason is that the posterior no longer has a natural density representation, which significantly complicates the definition and study of the underlying conditional probabilities.

Assume the prior measure is specified by a Gaussian random field, and the likelihood satisfies conditions in Theorem 3.9 that are necessary for the existence of a well-posed posterior measure. Then the MAP estimator is well-defined as the minimizer of an Onsager–Machlup functional defined on the Cameron–Martin space of the prior. If one has Gaussian noise, then this becomes a least-squares functional penalized by a regularization functional that is the Cameron–Martin norm of the Gaussian prior (Dashti *et al.* 2013); see also Dashti and Stuart (2017, Section 4.3). To handle the case with non-Gaussian priors, Helin and Burger (2015) introduce the notion of weak MAP estimate (wMAP) and show that a wMAP estimate can be connected to a variational formulation also for non-Gaussian priors. Furthermore, any MAP estimate in the sense of Dashti *et al.* (2013) is a wMAP estimate, so this is a generalization in the strict sense. The wMAP approach, however, fails when the prior does not admit continuous densities. In order to handle this, Clason, Helin, Kretschmann and Piiroinen (2018) introduce the notion of a generalized mode of a probability measure (Clason *et al.* 2018, Definition 2.3) and define a (generalized) MAP estimator as a generalized

[4] The Bregman distance for $x \mapsto \langle x, x\rangle$ gives the L^2-loss.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

mode of the posterior measure. Generalized MAP reduces to the classical MAP in certain situations that include Gaussian priors but the former also exist for a more general class of priors, such as uniform priors. The main result in Clason *et al.* (2018) is that one can characterize a generalized MAP estimator as a minimizer of an Onsager–Machlup functional even in cases when the prior does not admit a continuous density.

As a final remark, while a MAP estimate with a Gaussian noise model does lead to an optimization problem with a quadratic data-fidelity term, Gribonval and Nikolova (2018) show via explicit examples that the converse is not true. They characterize those data models of the type in (3.2) where Bayes estimators can be expressed as the solution of a penalized least-squares optimization problem. One example is denoising in the presence of additive Gaussian noise and an arbitrary prior; another is a data model with (a variant of) Poisson noise and any prior probability on the unknowns. In these cases, the variational approach is rich enough to build all possible Bayes estimators via a well-chosen penalty.

3.4. Explicit priors

The choice of prior is important in Bayesian inversion, and much of the work has focused on characterizing families of priors that ensure posterior consistency holds and for which there are good convergence rates (Section 3.2.3).

From the general Bayesian statistics literature, much effort has gone into characterizing robust priors. These have limited influence (in the asymptotic regime) on the posterior (Bayesian robustness), and such non-informative priors are useful for Bayesian inversion when there is not enough information to choose a prior for the unknown model parameter, or when the information available is not easily translated into a probabilistic statement: see Berger (1985, Section 4.7.9) and Calvetti and Somersalo (2017). For example, hierarchical priors tend to be robust (Berger 1985, Section 4.7.9). Another class is that of conjugate priors, which are desirable from a computational perspective (Section 3.5.2) since they have tails that are typically of the same form as the likelihood function. Conjugate priors also remain influential when the likelihood function is concentrated in the (prior) tail, so natural conjugate priors are therefore not necessarily robust (Berger 1985, Section 4.7).

Priors on digitized images. A wide range of priors for Bayesian inversion have been suggested when the unknown model parameter represents a digitized image, the latter given by a real-valued function in some suitable class defined on a domain $\Omega \subset \mathbb{R}^n$.

One such class is that of *smoothness* priors, which reflect the belief that values of the model parameter at a point are close to the average of its values in a neighbourhood of that point. Another is *structural* priors, which allow

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

for more abrupt (discontinuous) changes in the values of the unknown model parameter at specific locations. Yet another is *sparsity-promoting* priors (see Section 2.7), which encode the *a priori* belief that the unknown model parameter is compressible with respect to some underlying dictionary, that is, it can be transformed into a linear combination of dictionary elements where most coefficients vanish (or are small). Finally, there are hierarchical priors which are formed by combining other priors hierarchically into an overall prior. This is typically done in a two-step process where one first specifies some underlying priors, often taken as natural conjugate priors, and then mixes them in a second stage over hyper-parameters. Recently, Calvetti, Somersalo and Strang (2019) have reformulated the question of sparse recovery as an inverse problem in the Bayesian framework, and expressed the sparsity criteria by means of a hierarchical prior mode. More information and further examples of priors in the finite-dimensional setting are given by Kaipio and Somersalo (2005), Calvetti and Somersalo (2008, 2017) and Calvetti *et al.* (2019).

Priors on function spaces. Defining priors when X is an infinite-dimensional function space is somewhat involved. A common approach is to consider a convergent series expansion and then let the coefficients be generated by a random variable.

More precisely, consider the case when X is a Banach space of real-valued functions on some fixed domain $\Omega \subset \mathbb{R}^d$. Following Dashti and Stuart (2017, Section 2.1), let $\{\phi_i\}_i \subset X$ be a countable sequence whose elements are normalized, *i.e.* $\|\phi_i\|_X = 1$. Now consider model parameters $f \in X$ of the form

$$f = f_0 + \sum_i \alpha_i \phi_i,$$

where $f_0 \in X$ is not necessarily normalized to 1. A probability distribution on the coefficients α_i renders a real-valued random function on Ω: simply define the deterministic sequence $\{\gamma_i\}_i \subset \mathbb{R}$ and the i.i.d. random sequence $\{\xi_i\}_i \subset \mathbb{R}$ and set $\alpha_i = \gamma_i \xi_i$. This generates a probability measure Π_{prior} on X by taking the pushforward of the measure on the i.i.d. random sequence $\{\xi_i\}_i \subset \mathbb{R}$ under the map which takes the sequence into the random function.

Using the above technique, one can construct a uniform prior that can be shown to generate random functions that are all contained in a subset $X \subset L^\infty(\Omega)$, which can be characterized (Dashti and Stuart 2017, Theorem 2). It is likewise possible to define Gaussian priors where the random function exists as an L^2 limit in Sobolev spaces of sufficient smoothness (Dashti and Stuart 2017, Theorem 8). For imaging applications, much effort has been devoted to constructing edge-preserving priors. An obvious candidate is the TV-prior, but it is not 'discretization-invariant' as its edge-preserving property is lost when the discretization becomes finer. This

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

prompted the development of other edge-preserving priors, such as Besov space priors, which are discretization-invariant (Lassas *et al.* 2009, Kolehmainen, Lassas, Niinimäki and Siltanen 2012). See also Dashti and Stuart (2017, Theorem 5), which characterizes Besov priors that generate random functions contained in separable Banach spaces. However, edge-preserving inversion using a Besov space prior often relies on the Haar wavelet basis. Due to the structure of the Haar basis, discontinuities are preferred on an underlying dyadic grid given by the discontinuities of the basis functions. As an example, on $[0, 1]$ a Besov space prior prefers a discontinuity at $x = 1/4$ over $x = 1/3$. Thus, in most practical cases, Besov priors rely on both a strong and unrealistic assumption. For this reason, Markkanen, Roininen, Huttunen and Lasanen (2019) propose another class of priors for edge-preserving Bayesian inversion, the Cauchy difference priors. Starting from continuous one-dimensional Cauchy motion, its discretized version, Cauchy random walk, can be used as a non-Gaussian prior for edge-preserving Bayesian inversion. As shown by Markkanen *et al.* (2019), one can also develop a suitable posterior distribution sampling algorithm for computing the conditional mean estimates with single-component Metropolis–Hastings. The approach is applied to CT reconstruction problems in materials science.

The above constructions of random functions through randomized series can be linked to each other through the notion of random fields as shown in Dashti and Stuart (2017, Section 2.5); see also Ghosal and van der Vaart (2017, Chapters 2 and 10) for further examples. These constructions also extend straightforwardly to $\mathbb{R}^n$- or $\mathbb{C}^n$-valued functions.

3.5. Challenges

The statistical view of an inverse problem in Bayesian inversion extends the functional analytic one, since the output is ideally the posterior that describes all possible solutions. This is very tractable and fits well within the scientific tradition of presenting data and inferred quantities with error bars.

Most priors are chosen to regularize the problem rather than improving the output quality. Next, algorithmic advances (Section 3.5.2) have resulted in methods that can sample in a computationally feasible manner from a posterior distribution in a high-dimensional setting, say up to 10^6 dimensions. This is still not sufficient for large-scale two-dimensional imaging or regular three-dimensional imaging applications. Furthermore, these methods require an explicit prior, which may not be feasible if one uses learning to obtain it. They may also make use of analytic approximations such as those given in Section 3.2.5, which restricts the priors that can come into question. For these reasons, most applications of Bayesian inversion on large-scale problems only compute a MAP estimator, whereas estimators requiring integration over the model parameter space remain computationally

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

unfeasible. These include Bayes estimators and the conditional mean as well as estimators relevant for uncertainty quantification.

In conclusion, the above *difficulties in specifying a 'good' prior and in meeting the computational requirements have seriously limited the dissemination of Bayes inversion in large-scale inverse problems, such as those arising in imaging.* Before providing further remarks on this, let us mention that Section 5.1 shows how techniques from deep learning can be used to address the above challenges when computing a wide range of estimators. Likewise, in Section 5.2 we show how deep learning can be used to efficiently sample from the posterior.

3.5.1. Choosing a good prior

Current theory for choosing a 'good' prior mainly emphasizes the regularizing function of the prior (Section 3.4). In particular, one seeks a prior that ensures posterior consistency (Section 3.2.3) and good contraction rates (Section 3.2.4) and, if possible, also allows for an asymptotic characterization of the posterior (Section 3.2.5).

This theory, however, is for the asymptotic setting as the noise level in data tends to zero, whereas there seems to be no theory for Bayesian inversion that deals with the setting when the data have a fixed noise level. Here, the prior has a role that goes beyond acting as a regularizer, and its choice may have a large influence on the posterior. For example, it is far from clear whether a prior that provides 'optimal' contraction rates is the most suitable one when the data are fixed with a given noise level. Another difficulty is that norms used to quantify distance in the mathematical theorems have little to do with what is meant by a 'good' estimate of a model parameter. An archetypal example is the difficulty in quantifying the notion of 'image quality' in radiology. This is very difficult since the notion of image quality depends on the task motivating the imaging application. Hand-crafted priors surveyed in Section 3.4 have limitations in this regard, and in Sections 4.3, 4.4, 4.7 and 4.10 we survey work that uses data-driven modelling to obtain a prior.

3.5.2. Computational feasibility

The focus here is on techniques that are not based on deep learning for sampling from a very high-dimensional posterior distribution that lacks an explicit expression.

A well-known class of methods is based on Markov chain Monte Carlo (MCMC), where the aim is to define an iterative process whose stationary distribution coincides with the target distribution, which in Bayesian inversion is the posterior. MCMC techniques come in many variants, and one common variant is MCMC sampling with Metropolis–Hastings dynamics (Minh and Le Minh 2015), which generates a Markov chain with equilibrium

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

distribution that coincides with the posterior in the limit. Other variants use Gibbs sampling, which reduces the autocorrelation between samples. Technically, Gibbs sampling can be seen as a special case of Metropolis–Hastings dynamics and it requires computation of conditional distributions. Further variants are auxiliary variable MCMC methods, such as slice sampling (Neal 2003), proximal MCMC (Green, Łatuszyśński, Pereyra and Robert 2015, Durmus, Moulines and Pereyra 2018, Repetti, Pereyra and Wiaux 2019) and Hamiltonian Monte Carlo (Girolami and Calderhead 2011, Betancourt 2017). See also Dashti and Stuart (2017, Section 5) for a nice abstract description of MCMC in the context of infinite-dimensional Bayesian inversion.

An alternative approach to MCMC seeks to approximate the posterior with more tractable distributions (deterministic inference), for example in variational Bayes inference (Fox and Roberts 2012, Blei, Küçükelbir and McAuliffe 2017) and expectation propagation (Minka 2001). Variational Bayes inference has indeed emerged as a popular alternative to the classical MCMC methods for sampling from a difficult-to-compute probability distribution, which in Bayesian inversion is the posterior distribution. The idea is to start from a fixed family of probability distributions (variational family) and select the one that best approximates the target distribution under some similarity measure, such as the Kullback–Leibler divergence.

Blei *et al.* (2017, p. 860) try to provide some guidance on when to use MCMC and when to use variational Bayes. MCMC methods tend to be more computationally intensive than variational inference, but they also provide guarantees of producing (asymptotically) exact samples from the target density (Robert and Casella 2004). Variational inference does not enjoy such guarantees: it can only find a density close to the target but tends to be faster than MCMC. A recent development is the proof of a Bernstein–von Mises theorem (Wang and Blei 2017, Theorem 5), which shows that the variational Bayes posterior is asymptotically normal around the variational frequentist estimate. Hence, if the variational frequentist estimate is consistent, then the variational Bayes posterior converges to a Gaussian with a mean centred at the true model parameter. Furthermore, since variational Bayes rests on optimization, variational inference easily takes advantage of methods such as stochastic optimization (Robbins and Monro 1951, Kushner and Yin 1997) and distributed optimization (though some MCMC methods can also exploit these innovations (Welling and Teh 2011, Ahmed *et al.* 2012)). Thus, variational inference is suited to large data sets and scenarios where we want to quickly explore many models; MCMC is suited to smaller data sets and scenarios where we are happy to pay a heavier computational cost for more precise samples. Another factor is the geometry of the posterior distribution. For example, the posterior of a mixture model admits multiple modes, each corresponding to label

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

permutations of the components. Gibbs sampling, if the model permits, is a powerful approach to sampling from such target distributions; it quickly focuses on one of the modes. For mixture models where Gibbs sampling is not an option, variational inference may perform better than a more general MCMC technique (*e.g.* Hamiltonian Monte Carlo), even for small datasets (Küçükelbir, Ranganath, Gelman and Blei 2017).

4. Learning in functional analytic regularization

There have been two main ways to incorporate learning into functional analytic regularization. The first relates to the 'evolution' of regularizing functionals, primarily within variational regularization (Sections 4.3, 4.4, 4.6, 4.7 and 4.10). Early approaches focused on using measured data to determine the regularization parameter(s), but as prior models became increasingly complex, this blended into approaches where one learns a highly parametrized regularizer from data. The second category uses learning to address the computational challenge associated with variational regularization. The idea is to 'learn how to optimize' in variational regularization given an *a priori* bound on the computational budget (Section 4.9).

4.1. Choosing the regularization parameter

To introduce this topic we first consider the simplified case where only one parameter is used for characterizing the scale of regularization as in (2.10) and in the functional analytic regularization methods encountered in Sections 2.5–2.7. Thus we use the notation of Remarks 2.3 and 2.6, that is, f_{true} denotes the true (unknown) model parameter and $f_\lambda := \mathcal{R}_\lambda(g)$ is the regularized solution obtained from applying a reconstruction operator $\mathcal{R}_\lambda \colon Y \to X$ on data g satisfying (2.1).

In this context, and recalling some historical classification, the three main types of parameter choice rules are characterized as *a posteriori*, *a priori* and error-free parameter choice rules: see Bertero and Boccacci (1998, Section 5.6) and Engl *et al.* (2000). With hindsight, many of these parameter choice rules can be seen as early attempts to use 'learning' from data in the context of reconstruction.

***A posteriori* rules.** This class of rules is based on the assumption that a reasonably tight estimate of the data discrepancy and/or value of regularizer at the true solution can be accessed. That is, one knows $\epsilon > 0$ and/or $S > 0$ such that

$$\mathcal{L}(\mathcal{A}(f_{\text{true}}), g) \leq \epsilon \text{ for } g = \mathcal{A}(f_{\text{true}}) + e \quad \text{and/or} \quad \mathcal{S}(f_{\text{true}}) \leq S.$$

A prominent example of an *a posteriori* parameter choice rule is the Morozov discrepancy principle (Morozov 1966). Here, the regularization parameter

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

λ is chosen so that

$$\mathcal{L}(\mathcal{A}(f_\lambda), g) \leq \epsilon. \tag{4.1}$$

Another example is Miller's method (Miller 1970), where the regularization parameter λ is chosen so that

$$\mathcal{L}(\mathcal{A}(f_\lambda), g) \leq \epsilon \quad \text{and} \quad \mathcal{S}(f_\lambda) \leq S.$$

***A priori* rules.** These methods combine an estimate of the noise level in the data with some knowledge of the smoothness of the solution as *a priori* information. Hence, the choice of regularization parameter can be made before computing f_λ. The choice of λ is ideally guided by a theorem that ensures the parameter choice rule yields an optimal convergence rate, for example as in Theorem 2.7 where the (scalar) regularization parameter is chosen in proportion to the noise level. For more detailed references, see Engl *et al.* (2000) and Kindermann (2011).

Error-free parameter choice rules. Use data to guide the choice of parameter, for example by balancing principles between the error in the fidelity and the regularization terms. An important example in this context is *generalized cross-validation* (Golub, Heat and Wahba 1979). Let $f_\lambda^{[k]} \in X$ denote the regularized solution obtained from data when we have removed the kth component g_k of g. Then the regularization parameter λ is chosen to predict the missing data values:

$$\text{minimize } \lambda \mapsto \sum_{k=1}^{m} |\, \mathcal{A}(f_\lambda^{[k]})_k - g_k| \quad \text{subject to } \mathcal{A}(f_\lambda^{[k]})_k \simeq g_k.$$

Another method within this class is the *L-curve* (Hansen 1992). Here the regularization parameter λ is chosen where the log-log plot of $\lambda \mapsto (\mathcal{L}(\mathcal{A}(f_\lambda), g) + \lambda\, \mathcal{S}(f_\lambda))$ has the highest curvature (*i.e.* it exhibits a corner).

Most of the work on parameter choice techniques addresses the case of a single scalar parameter. Much of the theory is developed for additive Gaussian noise, that is, when the data discrepancy $\mathcal{L}$ is a squared (possibly weighted) 2-norm. For error-free parameter choice rules, convergence $f_{\lambda(\delta)} \to f_{\text{true}}$ as $\delta \to 0$ cannot be guaranteed (Bakushinskii 1984). Error-free parameter choice rules are computationally very demanding as they require solutions for varying values of the regularization parameter. Although many rules have been proposed, very few of them are used in practice. In Section 4.3.1 we will encounter another instance of parameter choice rules for TV-type regularization problems via bilevel learning.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

4.2. Learning in approximate analytic inversion

Early approaches to using learning in approximate analytic inversion (Section 2.4) mostly dealt with the FBP method for tomographic image reconstruction.

One of the first examples of the above is that of Floyd (1991), which learns the reconstruction filter of an analytic reconstruction algorithm. The same principle is considered by Würfl *et al.* (2018), who have designed a convolutional neural network (CNN) architecture specifically adapted to encode an analytic reconstruction algorithm for inverting the ray transform. A key observation is that the normal operator $\mathcal{A}^*\mathcal{A}$ is of convolutional type when $\mathcal{A}$ is the ray transform, and the reconstruction filter in FBP acts by a convolution as well. Hence, both of these are easily representable in a CNN. The paper gives explicit constructions of such CNNs for FBP in parallel-beam geometry and fan-beam geometry and the Feldkamp–Davis–Kress method in cone-beam geometry. Having the analytic reconstruction operator encoded as a CNN allows one to learn every other possible step in it, so the approach in Würfl *et al.* (2018) actually goes beyond learning the reconstruction filter of an analytic reconstruction algorithm. Finally, we mention Janssens *et al.* (2018), who consider a fan-beam reconstruction algorithm based on the Hilbert transform FBP (You and Zeng 2007) for which the filter is trained by a neural network.

4.3. Bilevel optimization

In variational methods (see Section 2.6), we define the reconstruction operator $\mathcal{R}_\theta \colon Y \to X$ by

$$\mathcal{R}_\theta(g) := \arg\min_{f \in X} \{\mathcal{L}(\mathcal{A}(f), g) + \mathcal{S}_\theta(f)\} \quad \text{for } g \in Y. \tag{4.2}$$

As already mentioned, ideally $f \mapsto \mathcal{L}(\mathcal{A}(f), g)$ corresponds to an affine transformation of the negative log-likelihood of the data. However, it is less clear how to choose the regularizer $\mathcal{S}_\theta \colon X \to \mathbb{R}$ and/or the value for the parameter θ.

The focus here is on formulating a generic set-up for learning selected components of (4.2) from supervised training data given a loss function $\ell_X \colon X \times X \to \mathbb{R}$. This set-up can be tailored towards learning the regularization functional $\mathcal{S}$, or the data fidelity term $\mathcal{L}$, or an appropriate component in a forward operator $\mathcal{A}$, *e.g.* in blind image deconvolution (Hintermüller and Wu 2015). The starting point is to have access to supervised training data $(f_i, g_i) \in X \times Y$ that are generated by a $(X \times Y)$-valued random variable $(\mathbb{f}, \mathbb{g}) \sim \mu$. We can then form the following *bilevel optimization*

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

formulation:

$$\begin{cases} \widehat{\theta} \in \arg\min_{\theta} \mathbb{E}_{(\mathbb{f},\mathbb{g})\sim\mu}[\ell_X(\mathcal{R}_\theta(\mathbb{g}), \mathbb{f})], \\ \mathcal{R}_\theta(g) := \arg\min_{f\in X} \{\mathcal{L}(\mathcal{A}(f), g) + \mathcal{S}_\theta(f)\}. \end{cases} \tag{4.3}$$

Note here that $\widehat{\theta}$ is a Bayes estimator, but μ is not fully known. Instead it is replaced by its empirical counterpart given by the supervised training data, in which case $\widehat{\theta}$ corresponds to *empirical risk minimization* Section 3.3.

In the bilevel optimization literature, as in the optimization literature as a whole, there are two main and mostly distinct approaches. The first one is the discrete approach that first discretizes the problem (4.2) and subsequently optimizes its parameters. In this way, optimality conditions and their well-posedness are derived in finite dimensions, which circumvents often difficult topological considerations related to convergence in infinite-dimensional function spaces, but also jeopardizes preservation of continuous structure (*i.e.* optimizing the discrete problem is not automatically equivalent to discretizing the optimality conditions of the continuous problem (De los Reyes 2015)) and dimension-invariant convergence properties.

Alternatively, (4.2) and its parameter θ are optimized in the continuum (*i.e.* appropriate infinite-dimensional function spaces) and then discretized. The resulting problems present several difficulties due to the frequent non-smoothness of the lower-level problem (think of TV regularization), which, in general, makes it impossible to verify Karush–Kuhn–Tucker constraint qualification conditions. This issue has led to the development of alternative analytical approaches in order to obtain first-order necessary optimality conditions (Bonnans and Tiba 1991, De los Reyes 2011, Hintermüller, Laurain, Löbhard, Rautenberg and Surowiec 2014). The bilevel problems under consideration are also related to generalized mathematical programs with equilibrium constraints in function spaces (Luo, Pang and Ralph 1996, Outrata 2000).

One of the first examples of the above is the paper by Haber and Tenorio (2003), who considered a regularization functional $\mathcal{S}_\theta \colon X \to \mathbb{R}$ that can depend on location and involves derivatives or other filters. Concrete examples are anisotropic weighted Dirichlet energy where θ is a function, that is,

$$\mathcal{S}_\theta(f) := \|\theta(\,\cdot\,)\nabla f(\,\cdot\,)\|_2^2 \quad \text{for } \theta \colon \Omega \to \mathbb{R},$$

and anisotropic weighted TV,

$$\mathcal{S}_\theta(f) := \|\theta(|\nabla f(\,\cdot\,)|)\|_1 \quad \text{for } \theta \colon \mathbb{R} \to \mathbb{R}.$$

The paper contains no formal mathematical statements or proofs, but there are many numerical examples showing how to use supervised learning techniques to determine a regularization functional given a training set of feasible solutions.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Another early example of learning a regularizer is the paper by Tappen (2007), who considered bilevel optimization for finding optimal regularizers parametrized by finite-dimensional Markov random field models. Bilevel optimization for optimal model design for inverse problems has also been discussed by Haber, Horesh and Tenorio (2010), Bui-Thanh, Willcox and Ghattas (2008) and Biegler *et al.* (2011).

A revival of bilevel learning in the context of non-smooth regularizers took place in 2013 with a series of papers: De los Reyes and Schönlieb (2013), Calatroni, De los Reyes and Schönlieb (2014), De los Reyes, Schönlieb and Valkonen (2016, 2017), Calatroni, De los Reyes and Schönlieb (2017), Van Chung, De los Reyes and Schönlieb (2017), Kunisch and Pock (2013), Chen, Pock and Bischof (2012), Chen, Yu and Pock (2015), Chung and Espanol (2017), Hintermüller and Wu (2015), Hintermüller and Rautenberg (2017), Hintermüller, Rautenberg, Wu and Langer (2017), Baus, Nikolova and Steidl (2014) and Schmidt and Roth (2014). A critical theoretical issue is the well-posedness of the learning; another is to derive a characterization of the optimal solutions that can be used in the design of computational methods. Such results were first derived by De los Reyes and Schönlieb (2013), Calatroni *et al.* (2014), De los Reyes, Schönlieb and Valkonen (2016, 2017), Van Chung *et al.* (2017) and Hintermüller and Rautenberg (2017), with applications to inverse problems and image processing (*e.g.* bilevel learning for image segmentation (Ranftl and Pock 2014)) as well as classification (*e.g.* learning an optimal set-up of support vector machines (Klatzer and Pock 2015)).

In what follows, we survey the main mathematical properties of bilevel learning and review the main parametrizations of regularizers in (4.2) that are considered in the literature.

4.3.1. Learning of TV-type regularizers and data fidelity terms

We start with a simple but theoretically and conceptually important example, namely the learning of total variation (TV)-type regularization models as proposed in De los Reyes and Schönlieb (2013), Kunisch and Pock (2013) and De los Reyes, Schönlieb and Valkonen (2017).

Let $X = BV(\Omega)$, where $\Omega \subset \mathbb{R}^n$ is a fixed open bounded set with Lipschitz boundary, and $Y = L^2(\mathbb{M}, \mathbb{R})$, where $\mathbb{M}$ is a manifold given by the data acquisition geometry. Next, let $\theta = (\lambda, \alpha)$, where $\lambda = (\lambda_1, \ldots, \lambda_M)$ and $\alpha = (\alpha_1, \ldots, \alpha_N)$ are non-negative scalar parameters. We also assume that the loss $\ell_X \colon X \times X \to \mathbb{R}$ is convex, proper and weak* lower semicontinuous. We next study the bilevel problem

$$\begin{cases} \widehat{\theta} \in \arg\min_{\theta} [\ell_X(\mathcal{R}_\theta(g), f)], \\ \mathcal{R}_\theta(g) = \arg\min_{f \in X} \{\mathcal{L}_\theta(\mathcal{A}(f), g) + \mathcal{S}_\theta(f)\}. \end{cases} \tag{4.4}$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

In the above, $\mathcal{L}_\theta\colon Y \times Y \to \mathbb{R}$ and $\mathcal{S}_\theta\colon X \to \mathbb{R}$ are defined for $\theta = (\lambda_i, \alpha_i)$ as

$$\mathcal{L}_\theta(\mathcal{A}(f), g) = \sum_{i=1}^{M} \lambda_i \mathcal{L}_i(\mathcal{A}(f), g) \quad \text{and} \quad \mathcal{S}_\theta(f) = \sum_{j=1}^{N} \alpha_j \| \mathcal{J}_j(f)\|_{\mathcal{M}(\Omega;\mathbb{R}^{m_j})} \tag{4.5}$$

where $\mathcal{L}_i\colon Y \times Y \to \mathbb{R}$ and $\mathcal{J}_j \in \mathcal{M}(\Omega;\mathbb{R}^{m_j})$ are given. Hence, in (4.4) the variational model is parametrized in terms of sums of different fidelity terms $\mathcal{L}_i$ and TV-type regularizers $\| \mathcal{J}_j(f)\|_{\mathcal{M}(\Omega;\mathbb{R}^{m_j})}$, weighted against each other with parameters λ_i and α_j (respectively). For $N = 1$ and $S_1 = D$ the distributional derivative, then

$$\|S_1(f)\|_{\mathcal{M}(\Omega;\mathbb{R}^{m_1})} = \mathrm{TV}(f).$$

This framework is the basis for the analysis of the learning model, in which convexity of the variational model and compactness properties in the space of functions of bounded variation are crucial for proving existence of an optimal solution: see De los Reyes, Schönlieb and Valkonen (2016). Richer parametrizations for bilevel learning are discussed in Chen *et al.* (2012, 2015), for example, where non-linear functions and convolution kernels are learned. Chen *et al.*, however, treat the learning model in finite dimensions, and a theoretical investigation of these more general bilevel learning models in a function space setting is a matter for future research.

In order to derive sharp optimality conditions for optimal parameters of (4.4) more regularity on the lower-level problem is needed. For shifting the problem (4.4) into a more regular setting, the Radon norms are regularized with Huber regularization and a convex, proper and weak* lower-semicontinuous smoothing functional $\mathrm{H}\colon X \to [0, \infty]$ is added to the lower-level problem, typically $\mathrm{H}(f) = \frac{1}{2}\| \nabla f\|^2$. In particular, the former is required for the single-valued differentiability of the solution map $(\lambda, \alpha) \mapsto f_{\alpha,\lambda}$, required by current numerical methods, irrespective of whether we are in a function space setting (see *e.g.* Rockafellar and Wets 1998, Theorem 9.56, for the finite-dimensional case). For parameters $\mu \geq 0$ and $\gamma \in (0, \infty]$, the lower-level problem in (4.4) is then replaced by

$$\mathcal{R}_\theta := \operatorname*{arg\,min}_{f \in X} \{\mu\, \mathrm{H}(f) + \mathcal{L}(\mathcal{A}(f), g) + \mathcal{S}_\theta^\gamma(f)\}, \tag{4.6}$$

with

$$\mathcal{S}_\theta^\gamma(f) := \sum_{j=1}^{N} \alpha_j \| \mathcal{J}_j(f)\|^\gamma_{\mathcal{M}(\Omega;\mathbb{R}^{m_j})}.$$

Here, $f \mapsto \| \mathcal{J}_j(f)\|^\gamma_{\mathcal{M}(\Omega;\mathbb{R}^{m_j})}$ is the Huberized TV measure, given as follows.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Definition 4.1. Given $\gamma \in (0, \infty]$, the Huber regularization for the norm $\|\cdot\|_2$ on $\mathbb{R}^n$ is defined by

$$\|g\|_\gamma = \begin{cases} \|g\|_2 - \dfrac{1}{2\gamma}, & \|g\|_2 \geq \dfrac{1}{\gamma}, \\ \dfrac{\gamma}{2}\|g\|_2^2, & \|g\|_2 < \dfrac{1}{\gamma}. \end{cases}$$

Then, for $\mu \in \mathcal{M}(\Omega; \mathbb{R}^{m_j})$ with Lebesgue decomposition $\mu = \nu L^n + \mu^s$ we have the Huber-regularized total variation measure,

$$|\mu|_\gamma(V) := \int_v |\nu(x)|_\gamma \, \mathrm{d}x + |\mu^s|(V) \quad (V \subset \Omega \ \text{ Borel-measurable}),$$

and finally its Radon norm,

$$\|\mu\|^\gamma_{\mathcal{M}(\Omega;\mathbb{R}^{m_j})} := \||\mu|_\gamma\|_{\mathcal{M}(\Omega;\mathbb{R}^{m_j})}.$$

In all of these, we interpret the choice $\gamma = \infty$ to give back the standard unregularized total variation measure or norm. In this setting existence of optimal parameters and differentiability of the solution operator can be proved, and with this an optimality system can be derived: see De los Reyes *et al.* (2016). More precisely, for the special case of the TV-denoising model the following theorem holds.

Theorem 4.2 (TV denoising (De los Reyes *et al.* 2016)). Consider the denoising problem (2.2) where $g, f_{\text{true}} \in BV(\Omega) \bigcap L^2(\Omega)$, and assume $\mathrm{TV}(g) > \mathrm{TV}(f_{\text{true}})$. Also, let $\mathrm{TV}^\gamma(f) := \|Df\|^\gamma_{\mathcal{M}(\Omega;\mathbb{R}^n)}$. Then there exist $\bar{\mu}, \bar{\gamma} > 0$ such that any optimal solution $\alpha_{\gamma,\mu} \in [0, \infty]$ to the problem

$$\begin{cases} \min\limits_{\alpha \in [0,\infty]} \dfrac{1}{2}\|f_{\text{true}} - f_\alpha\|^2_{L^2(\Omega)}, \\ f_\alpha = \underset{f \in BV(\Omega)}{\arg\min} \left\{ \dfrac{1}{2}\|g - f\|^2_{L^2(\Omega)} + \alpha \, \mathrm{TV}^\gamma(f) + \dfrac{\mu}{2}\|\nabla f\|^2_{L^2(\Omega;\mathbb{R}^n)} \right\} \end{cases}$$

satisfies $\alpha_{\gamma,\mu} > 0$ whenever $\mu \in [0, \bar{\mu}]$ and $\gamma \in [\bar{\gamma}, \infty]$.

Theorem 4.2 states that if g is a noisy image which oscillates more than the noise-free image f_{true}, then the optimal parameter is strictly positive, which is exactly what we would naturally expect. De los Reyes *et al.* (2016) proved a similar result for second-order TGV and ICTV regularization for the case when $X = Y$. The result was not extended to data with a general Y, but it is possible with additional assumptions on the parameter space.

Moreover, in much of the analysis for (4.4) we could allow for spatially dependent parameters α and λ. However, the parameters would then need to lie in a finite-dimensional subspace of $C_0(\Omega; \mathbb{R}^N)$: see De los Reyes and Schönlieb (2013) and Van Chung *et al.* (2017). Observe that Theorem 4.2 allows for infinite parameters α. Indeed, for regularization parameter learning

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

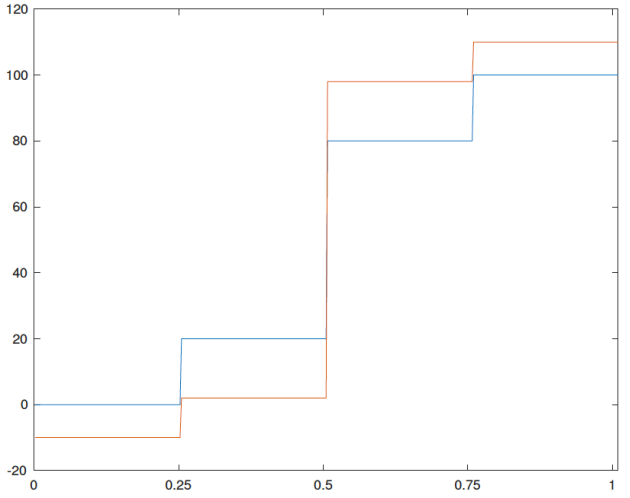
it is important not to restrict the parameters to be finite, as this allows us to decide between TGV^2, TV and TV^2 regularization. De los Reyes *et al.* (2016) also prove a result on the approximation properties of the bilevel scheme with the smoothed variational model (4.6) as $\gamma \nearrow \infty$ and $\mu \searrow 0$. In particular, they prove that as the numerical regularization vanishes, any optimal parameters for the regularized models tend to optimal parameters of the original model (4.4) in the sense of Γ-convergence. Moreover, De los Reyes and Schönlieb (2013) take the limit as $\gamma \nearrow \infty$ but $\mu > 0$ fixed for the optimality system and derive an optimality system for the non-smooth problem. Recently Davoli and Liu (2018) expanded the analysis for bilevel optimization of total variation regularizers beyond (4.4) to a richer family of anisotropic total variation regularizers in which the parameter of the dual norm and the (fractional) order of derivatives becomes part of the parameter space that (4.4) is optimized over.

Remark 4.3. Let us note here that despite the apparent simplicity of only one parameter to optimize over in Theorem 4.2, even in the case of the forward operator $\mathcal{A} = \mathrm{id}$ being the identity, the bilevel optimization problem is non-convex in α (against common hypotheses previously stated in publications). Evidence for this provides the counter-example constructed by Pan Liu (private communication) in Figure 4.1. Here, the one-dimensional TV regularization problem for signal denoising is considered. Fed with a piecewise constant input g, the one-dimensional TV problem can be solved analytically (Strong *et al.* 1996) and its solution is denoted by $f(\alpha)$. Figure 4.1(b) shows the non-convexity of the associated L^2-loss function $I(\alpha) = \|f_{\mathrm{true}} - f(\alpha)\|^2$.

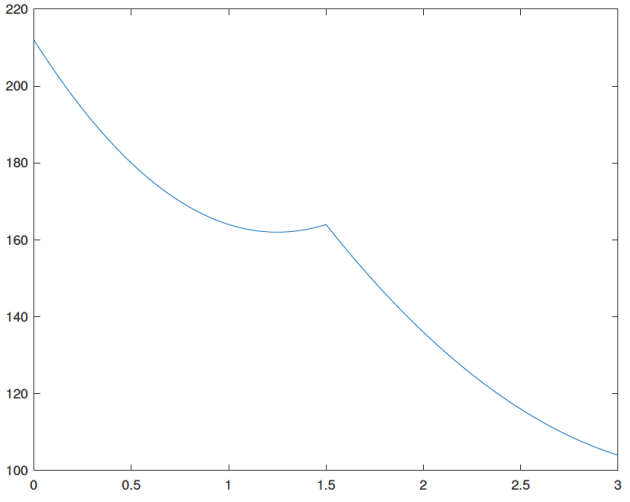
Computing optimal solutions to the bilevel learning problems requires a proper characterization of optimal solutions in terms of a first-order optimality system. Since (4.4)–(4.6) constitutes a PDE-constrained optimization problem, suitable techniques from this field may be utilized. For the limit cases, an additional asymptotic analysis needs to be performed in order to get a sharp characterization of the solutions as $\gamma \to \infty$ or $\mu \to 0$, or both.

Several instances of the abstract problem (4.4) have been considered in the literature. De los Reyes and Schönlieb (2013) considered the case with TV regularization in the presence of several noise models. They proved the Gâteaux differentiability of the solution operator, which led to the derivation of an optimality system. Thereafter they carried out an asymptotic analysis with respect to $\gamma \to \infty$ (with $\mu > 0$), obtaining an optimality system for the corresponding problem. In that case the optimization problem corresponds to one with variational inequality constraints and the characterization concerns C-stationary points. Also, De los Reyes *et al.* (2017) have investigated differentiability properties of higher-order regularization solution operators. They proved a stronger Fréchet-differentiability result

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press



(a) data g (red) against ground truth f_{true} (blue)



(b) plot of $I(\alpha) = \|f_{\text{true}} - f(\alpha)\|^2$

Figure 4.1. Parameter optimality for TV denoising in Theorem 4.2. The non-convexity of the loss function, even for this one-parameter optimization problem, is clearly visible. Courtesy of Pan Liu.

for the TGV^2 case, which also holds for TV. In particular, these stronger results open the door to further necessary and sufficient optimality conditions. Further, using the adjoint optimality condition gradient, formulas for the reduced cost functional can be derived, which in turn feed into the design of numerical algorithms for solving (4.4): see Calatroni *et al.* (2016, Section 3).

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

4.3.2. From fields of experts to variational networks

The TV-type regularization approaches discussed above used popular handcrafted variational regularizers in a first attempt to make them more data-driven, staying in the framework of optimizing proposed regularization models in infinite-dimensional function space. In what follows, we change our perspective to discrete variational regularization models (4.2) in which $X = \mathbb{R}^n$, and the functional $\mathcal{S}(\cdot)$ is equipped with parametrizations that go beyond TV-type regularizers; however, these are not covered by the theory in the previous section.

MRF parametrizations. A prominent example of such richer parametrizations of regularizers has its origins in MRF theory. MRFs, first introduced by Besag (1974) and then used by Cross and Jain (1983) for texture modelling, provide a probabilistic framework for learning image priors: see *e.g.* Zhu, Wu and Mumford (1998). Higher-order MRF models such as the celebrated Field of Experts (FoE) model (Roth and Black 2005) are the most popular ones, as they are able to express very complex image structures and yield good performance for various image processing tasks. The FoE model is learning a rich MRF image prior by using ideas from sparse coding (Olshausen and Field 1997), where image patches are approximated by a linear combination of learned filters, variations of which lead to the well-known principal component analysis (PCA) and independent component analysis (ICA), and the Product of Experts (PoE) model (Welling, Osindero and Hinton 2003).

In the context of bilevel learning, the FoE model has been used by Chen, Pock, Ranftl and Bischof (2013) and Chen, Ranftl and Pock (2014). Here, the regularizer is parametrized in the form

$$\mathcal{S}_\theta(f) := \sum_{i=1}^{N} \left[\alpha_i \sum_{k=1}^{n} \rho((J_i f)_k) \right] \quad \text{with } \theta = (\alpha_1, \ldots, \alpha_N, J_1, \ldots, J_N), \quad (4.7)$$

where N is the number of filters J_i which are sparse and implemented as a two-dimensional convolution with a kernel k_i, that is, $J_i f$ is the digitized version of $k_i * f$. In the FoE model we use the parametrization $J_i = \sum_{j=1}^{m} \beta_{i,j} B_j$, for a given set of basis filters $\{B_1, \ldots, B_m\}$ with $B_j \in \mathbb{R}^{n \times n}$. Moreover, the α_i are non-negative parameters and the non-linear function $\rho(z) = \log(1 + z^2)$. The shape of ρ is motivated by statistics over natural images that were shown to roughly follow a Student t-distribution (Huang and Mumford 1999). With these parametrizations, θ in (4.7) can be seen as $\theta = (\alpha_i, \beta_{i,j})$ for $i = 1, \ldots, n$ and $j = 1, \ldots, m$, and one learns the optimal θ from supervised training data (f_i, g_i) by minimizing the squared L^2-loss:

$$\ell_X(\mathcal{R}_\theta(g), f) := \| \mathcal{R}_\theta(g) - f \|_2^2 \quad \text{with } \mathcal{R}_\theta \colon Y \to X \text{ given by (4.4).}$$

Chen *et al.* (2014) investigated other non-linearities such as the non-smooth

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

and non-convex $\rho(z) = \log(1 + |z|)$, and the non-smooth but convex ℓ_1-norm $\rho(z) = |z|$. Their experiments in particular suggested that, while the two log-type non-linearities gave very similar regularization performance, the convex ℓ_1-norm regularizer clearly did worse. Moreover, Chen *et al.* (2015) parametrized the non-linearities ρ with radial basis functions, whose coefficients are learned as well. Earlier MRF-based bilevel learning schemes exist: see *e.g.* Samuel and Tappen (2009).

A further development is the variational networks introduced by Kobler, Klatzer, Hammernik and Pock (2017) and Hammernik *et al.* (2018). The idea was to replace the above bilevel scheme with learning to optimize (Section 4.9) using a supervised loss, thereby leading to a learned iterative method. This will be discussed in more detail in Section 5.1.4, where the variational networks re-emerge in the framework of learned iterative schemes.

4.4. Dictionary learning in sparse models

In Section 2.7 we discussed sparsity as an important concept for modelling prior knowledge in inverse problems. Assuming that the model parameter possesses a sparse or compressible representation in a given dictionary $\mathbb{D}$ sparse recovery strategies, associated computational approaches and error estimates for the model parameter can be derived. In what follows, we turn to approaches which, rather than working with a given dictionary, learn the dictionary before or jointly with the recovery of the model parameter. Note that almost all work on dictionary learning in sparse models has been carried out in the context of denoising, *i.e.* $\mathcal{A} = \mathrm{id}$ only.

Patch-based local models. A particular class of sparse coding models are those which impose sparsity on patches (segmented) that are extracted from the signal. Let $g = \mathcal{A}(f_{\text{true}}) + e$ and $\hat{f} \in X$. Further, let $\mathbb{D} = \{\phi_k\} \subset X$ be a dictionary, and $P_j : X \to X$ with $P_j(f) = f|_{\Omega_j}$ for $j = 1, \dots N$ patches of f. Assume that

$$\hat{f} = \sum_{j=1}^{N} P_j(f_{\text{true}}), \tag{4.8}$$

where $P_j(f_{\text{true}}) \in X$ is compressible with respect to $\mathbb{D}$.

Some of the currently leading denoising methods are based on patch-based local models. Examples include K-SVD, which is a sparse coding approach on image patches (Elad and Aharon 2006), BM3D, which combines sparsity and self-similarity (Dabov, Foi, Katkovnik and Egiazarian 2007), and EPLL, which is a Gaussian mixture model of image patches (Zoran and Weiss 2011). Other examples are NCSR (Dong, Zhang, Shi and Li 2013), weighted nuclear norm regularization of image patches (WNNM) (Gu, Zhang, Zuo and Feng

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

2014) and SSC-GSM, a non-local sparsity approach with a Gaussian scale mixture (Dong, Shi, Ma and Li 2015).

Sparse-land models are a particular subclass of patch-based models. Here, each patch is sparse with respect to some global dictionary. Examples include sparse coding approaches that are applied patch-wise (Elad and Aharon 2006, Dong, Zhang, Shi and Wu 2011, Mairal and Ponce 2014, Romano and Elad 2015, Sulam and Elad 2015). In sparse-land models $\hat{f} \in X$ is reconstructed by solving

$$\min_{f \in X, \xi_i \in \Xi} \{\mathcal{L}(\mathcal{A}(f), g) + \mathcal{S}_\theta(f, \xi_1, \ldots, \xi_N)\}, \tag{4.9}$$

where

$$\mathcal{S}_\theta(f, \xi_1, \ldots, \xi_N) := \sum_{j=1}^{N} [\lambda_j \|P_j(f) - \mathcal{E}^*_{\mathbb{D}}(\xi_j)\|_2^2 + \mu_j \|\xi_j\|_p^p], \tag{4.10}$$

with $\theta = (\lambda_j, \mu_j)_{j=1}^N \in (\mathbb{R}^2)^N$ and $\mathcal{E}^*_{\mathbb{D}} \colon \Xi \to X$ denoting the synthesis operator associated with the given dictionary $\mathbb{D}$. Bai *et al.* (2017) propose the following alternating scheme to solve the sparse-land reconstruction problem:

$$\begin{cases} f^{i+1} := \arg\min_{f \in X} \left\{ \mathcal{L}(\mathcal{A}(f), g) + \sum_{j=1}^{N} \lambda_j \|P_j(f) - \mathcal{E}^*_{\mathbb{D}}(\xi_j^j)\|_2^2 \right\} \\ \xi_j^{i+1} := \arg\min_{\xi_i \in \Xi} \{\lambda_j \|P_j(f^{i+1}) - \mathcal{E}^*_{\mathbb{D}}(\xi_j)\|_2^2 + \mu_j \|\xi_j\|_p^p\} \quad \text{for } j = 1, \ldots, N. \end{cases} \tag{4.11}$$

The advantage of these approaches over plain-vanilla dictionary learning is that sparse-land models are computationally more feasible. Sparse-land models are one example of a dictionary learning approach, which will be discussed in the next section.

4.4.1. Reconstruction and dictionary learning

In sparse models, there are three ways to determine the dictionary. First, the dictionary is specified analytically. Second, the dictionary is determined from example data (dictionary learning). Third (as in the sparse-land models discussed above) the dictionary is determined jointly while performing reconstruction (joint reconstruction and dictionary learning). In what follows, we particularly focus on the second and third options for determining the dictionary.

Joint reconstruction and dictionary learning. The recent paper by Chambolle, Holler and Pock (2018) proposes a convex variational model for joint reconstruction and (convolutional) dictionary learning that applies to inverse problems where data are corrupted and/or incomplete. Chambolle

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

et al. (2018) prove rigorous results on well-posedness and stability stated in the infinite-dimensional (non-digitized) setting.

Earlier approaches were less rigorous and considered the finite-dimensional setting with a patch-based dictionary (sparse-land model). One such example is adaptive dictionary-based statistical iterative reconstruction (ADSIR) (Zhang *et al.* 2016), which adds the dictionary $\mathbb{D} = \{\phi_i\} \subset X$ as an unknown to the sparse-land model as in global dictionary-based statistical iterative reconstruction (GDSIR) (Xu *et al.* 2012; see also Chun, Zheng, Long and Fessler 2017). This results in the following problem:

$$\min_{f\in X,\xi_i\in\Xi,\mathbb{D}} \{\mathcal{L}(\mathcal{A}(f), g) + \mathcal{S}_\theta(f, \xi_1, \dots, \xi_N, \mathbb{D})\}, \tag{4.12}$$

where

$$\mathcal{S}_\theta(f, \xi_1, \dots, \xi_N, \mathbb{D}) := \sum_{j=1}^{N} [\lambda_j \|P_j(f) - \mathcal{E}^*_{\mathbb{D}}(\xi_j)\|_2^2 + \mu_j \|\xi_j\|_p^p], \tag{4.13}$$

with $\theta = ((\lambda_j, \mu_j)_{j=1}^N) \in (\mathbb{R}^2)^N$, and $\mathcal{E}^*_{\mathbb{D}} : \Xi \to X$ denotes the synthesis operator associated with the dictionary $\mathbb{D}$. Usually an alternating minimization scheme is used to optimize over the three variables in (4.12).

Dictionary learning. Let $\ell_X : X \times X \to \mathbb{R}$ be a given loss function (*e.g.* the ℓ_2- or ℓ_1-norm). Further, let $f_1 \dots f_N \in X$ be a given unsupervised training data, $\mathbb{D} = \{\phi_i\} \subset X$ a dictionary, and $\mathcal{E}^*_{\mathbb{D}} : \Xi \to X$ the synthesis operator given as $\mathcal{E}^*_{\mathbb{D}}(\xi) = \sum_i \xi_i \phi_i$ for $\xi \in \Xi$. One approach in dictionary learning is based on a sparsity assumption and solves

$$(\widehat{\mathbb{D}}, \hat{\xi}_i) \in \operatorname*{arg\,min}_{\xi_i\in\Xi,\mathbb{D}\subset X} \sum_{i=1}^{N} \ell_X(f_i, \mathcal{E}^*_{\mathbb{D}}(\xi_i)), \tag{4.14}$$

such that $\|\xi_i\|_0 \le s$ for $i = 1, \dots, N$, for a given sparsity level s. Alternatively, one looks for a dictionary that minimizes the total cost for representing signals in the training data while enforcing a constraint on the precision in the following way:

$$(\widehat{\mathbb{D}}, \hat{\xi}_i) = \operatorname*{arg\,min}_{\xi_i\in\Xi,\mathbb{D}\subset X} \sum_{i=1}^{N} \|\xi_i\|_0, \tag{4.15}$$

such that $\ell_X(f_i, \mathcal{E}^*_{\mathbb{D}}(\xi_i)) \le \epsilon$ for $i = 1, \dots, N$. A unified formulation is given by the unconstrained problem

$$(\widehat{\mathbb{D}}, \hat{\xi}_i) = \operatorname*{arg\,min}_{\xi_i\in\Xi,\mathbb{D}\subset X} \sum_{i=1}^{N} [\ell_X(f_i, \mathcal{E}^*_{\mathbb{D}}(\xi_i)) + \theta\|\xi_i\|_0]. \tag{4.16}$$

All three formulations are posed in terms of the ℓ_0-norm and are NP-hard

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

to compute. This suggests the use of *convex relaxation*, by which (4.16) becomes

$$(\widehat{\mathbb{D}}, \hat{\xi}_i) = \underset{\xi_i \in \Xi, \mathbb{D} \subset X}{\arg\min} \sum_{i=1}^{N} [\ell_X(f_i, \mathcal{E}^*_{\mathbb{D}}(\xi_i)) + \theta \|\xi_i\|_1]. \tag{4.17}$$

If $\mathbb{D}$ is fixed then the sum in (4.17) decouples and leads to the convex relaxation of the *sparse coding* problem in (2.20) for $\mathcal{A} = \mathrm{id}$.

In the finite-dimensional setting $X = \mathbb{R}^m$ and Ξ is replaced by $\mathbb{R}^n$ for some n. Then the dictionary is $\mathbb{D} := \{\phi_k\}_{k=1\ldots n} \subset \mathbb{R}^m$ represented by an $n \times m$ matrix $\mathbf{D}$ leading to the synthesis operator becoming a mapping $\mathcal{E}^*_{\mathbf{D}} \colon \mathbb{R}^m \to \mathbb{R}^n$ with $\mathcal{E}^*_{\mathbb{D}}(\xi) = \mathbf{D} \cdot \xi$. In this setting (4.17) becomes

$$(\widehat{\mathbf{D}}, \hat{\xi}_i) := \underset{\xi_i \in \mathbb{R}^m, \mathbf{D} \in \mathbb{R}^{n \times m}}{\arg\min} \sum_{i=1}^{N} [\ell_X(f_i, \mathbf{D} \cdot \xi_i) + \theta \|\xi_i\|_1]. \tag{4.18}$$

Here, if $\mathbf{D}$ satisfies the RIP then the convex relaxation preserves the sparse solution (Candès *et al.* 2006). State-of-the-art dictionary learning algorithms are K-SVD (Aharon, Elad and Bruckstein 2006), geometric multi-resolution analysis (GRMA) (Allard, Chen and Maggioni 2012) and online dictionary learning (Mairal, Bach, Ponce and Sapiro 2010). Most work on dictionary learning to date has been done in the context of denoising, *i.e.* $\mathcal{A} = \mathrm{id}$; see also Rubinstein *et al.* (2010).

4.4.2. Convolutional sparse coding and convolutional dictionary learning
Dictionary learning in the context of sparse coding has been very popular and successful, with several seminal approaches arising in this field, as outlined in the previous section. However, there are still several issues with sparse coding related to the locality of learned structures and the computational effort needed. Sparse-land models (4.9), for instance, perform sparse coding over all the patches, and this tends to be a slow process.

The computational performance can be addressed by using learning to improve upon an optimization scheme (Section 4.9), for example the Learned Iterative Soft-Thresholding Algorithm (LISTA) (see Section 4.9.2) learns a finite number of unrolled Iterative Soft-Thresholding Algorithm (ISTA) iterates using unsupervised training data to match ISTA solutions (Gregor and LeCun 2010). Moreover, learning a dictionary over each patch independently as in (4.12) cannot account for global information, *e.g.* shift-invariance in images. What is needed is a computational feasible approach that introduces further structure and invariances in the dictionary, *e.g.* shift-invariance, and that makes each atom dependent on the whole signal instead of just individual patches. In this realm convolutional dictionaries have been introduced. Here atoms are given by convolution kernels and act on signal features via convolution, that is, $\mathbf{D}$ is a concatenation of Toeplitz

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

matrices (a union of banded and circulant matrices). Set up like this, convolutional dictionaries render computationally feasible shift-invariant dictionaries, where atoms depend on the entire signal.

Convolutional sparse coding. Consider now the inverse problem of recovering $f_{\text{true}} \in X$ from (2.1) with the assumption that f_{true} is compressible with respect to convolution dictionary $\mathbb{D} := \{\phi_i\} \subset X$.

In convolutional sparse coding (CSC), this is done by performing a synthesis using convolutional dictionaries, that is, atoms act by convolutions. More precisely, the reconstruction operator $\mathcal{R}_\theta \colon Y \to X$ is given as

$$\mathcal{R}(g) := \sum_i \hat{\xi}_i * \phi_i, \tag{4.19}$$

where

$$\hat{\xi}_i \in \operatorname*{arg\,min}_{\xi_i \in X} \left\{ \mathcal{L}\left(\mathcal{A}\left(\sum_i \xi_i * \phi_i \right), g \right) + \lambda \|\xi_i\|_0 \right\}.$$

Computational methods for solving (4.19) for denoising use convex relaxation followed by the alternating direction method of multipliers (ADMM) in frequency space (Bristow, Eriksson and Lucey 2013) and its variants. See also Sreter and Giryes (2017) on using LISTA in this context. So far CSC has only been analysed in the context of denoising (Bristow *et al.* 2013, Wohlberg 2014, Gu *et al.* 2015, Garcia-Cardona and Wohlberg 2017) with theoretical properties given in Papyan, Sulam and Elad (2016*a*, 2016*b*).

Convolutional dictionary learning. Learning a dictionary in the context of CSC is called convolutional dictionary learning. Here, given unsupervised training data $f_1, \ldots, f_m \in X$ and a loss function $\ell_X : X \colon X \to X$, one solves

$$\operatorname*{arg\,min}_{\phi_i, \xi_{j,i} \in X} \left\{ \sum_{j=1}^{m} \ell_X \left(f_j, \sum_i \xi_{j,i} * \phi_i \right) + \lambda \sum_{j=1}^{m} \sum_i \|\xi_{j,i}\|_1 \right\}, \tag{4.20}$$

where $\|\phi_i\|_2 = 1$. For instance, Garcia-Cardona and Wohlberg (2017) solved (4.20) with a squared L_2-loss using an ADMM-type scheme. Extension of convolutional dictionary learning to a supervised data setting has been considered by Affara, Ghanem and Wonka (2018), for instance. Here, discriminative dictionaries instead of purely reconstructive ones are learned by introducing a supervised regularization term in the usual CSC objective that encourages the final dictionary elements to be discriminative.

Multi-layer convolutional sparse coding. A multi-layer extension of CSC, referred to as multi-layer convolutional sparse coding (ML-CSC), is proposed by Sulam, Papyan, Romano and Elad (2017). Given L convolutional dictionaries $\mathbb{D}_1, \ldots, \mathbb{D}_L \subset X$ with atoms $\mathbb{D}_k := \{\phi_j^k\}_j$, a model parameter $f \in X$ admits a representation in terms of the corresponding multi-layer

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

convolutional sparse coding (ML-CSC) model if there are $s_1, \dots, s_L \in \mathbb{R}$ such that

$$\begin{cases} f = \sum_j (\xi_j^1 * \phi_j^1), \\ \boldsymbol{\xi}^k = \sum_j (\xi_j^{k+1} * \phi_j^{k+1}) \quad \text{for } k = 1, \dots, L-1, \end{cases}$$

and $\|\boldsymbol{\xi}^k\|_{0,\infty} \leq s_k$ for $k = 1, \dots, L$. Hence, atoms $\phi_{k,i} \in \mathbb{D}_k$ in the kth convolution dictionary are compressible in the $(k+1)$th dictionary $\mathbb{D}_{k+1}$ for $k = 1, \dots, L-1$.

The ML-CSC model is a special case of CSC where intermediate representations have a specific structure (Sulam *et al.* 2017, Lemma 1). Building on the theory for CSC, Sulam *et al.* (2017) provide a theoretical study of this novel model and its associated pursuits for dictionary learning and sparse coding in the context of denoising. Further, consequences for the theoretical analysis of CNNs can be extracted from ML-CSC using the fact that the resulting layered thresholding algorithm and the layered basis pursuit share many similarities with a forward pass of a deep CNN.

Indeed, Papyan, Romano and Elad (2017) show that ML-CSC yields a Bayesian model that is implicitly imposed on $\hat{f}$ when deploying a CNN, and that consequently characterizes signals belonging to the model behind a deep CNN. Among other properties, one can show that the CNN is guaranteed to recover an estimate of the underlying representations of an input signal, assuming these are sparse in a local sense (Papyan *et al.* 2017, Theorem 4) and the recovery is stable (Papyan *et al.* 2017, Theorems 8 and 10). Many of these results also hold for fully connected networks, and they can be used to formulate new algorithms for CNNs, for example to propose an alternative to the commonly used forward pass algorithm in CNN. This is related to both deconvolutional (Zeiler, Krishnan, Taylor and Fergus 2010, Pu *et al.* 2016) and recurrent networks (Bengio, Simard and Frasconi 1994).

An essential technique for proving the key results in the cited references for ML-CSC is based on unrolling, which establishes a link between sparsity-promoting regularization (compressed sensing) and deep neural networks. More precisely, one starts with a variational formulation like that in (4.20) and specifies a suitable iterative optimization scheme. In this setting one can prove several theoretical results, such as convergence, stability and error estimates. Next, one unrolls the truncated optimization iterates and identifies the updating between iterates as layers in a deep neural network (Section 4.9.1). The properties of this network, such as stability and convergence, can now be analysed using methods from compressed sensing.

Deep dictionary learning. Another recent approach in the context of dictionary learning is deep dictionary learning. Here, the two popular representation learning paradigms – dictionary learning and deep learning – come together. Conceptually, while dictionary learning focuses on learning

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

a 'basis' and 'features' by matrix factorization, deep learning focuses on extracting features via learning 'weights' or 'filters' in a greedy layer-by-layer fashion. Deep dictionary learning, in turn, builds deeper architectures by using the layers of dictionary learning. See Tariyal, Majumdar, Singh and Vatsa (2016), who show that their approach is competitive against other deep learning approaches, such as stacked auto-encoders, deep belief networks and convolutional neural networks, regarding classification and clustering accuracies.

4.4.3. TV-type regularizer with learned sparsifying transforms.
The idea here is to learn the underlying sparsifying transform in a TV-type regularizer. One such approach is presented by Chun, Zheng, Long and Fessler (2017), who consider the case of a single sparsifying transform. This idea is further developed by Zheng, Ravishankar, Long and Fessler (2018), who consider a regularizer that is given by a union of sparsifying transforms (ULTRA), which act on image patches and quantify the sparsification error of each patch using its best-matched sparsifying transform: see Zheng *et al.* (2018, equation (3)). The resulting optimization problem is solved by an intertwined scheme that alternates between a CT image reconstruction step, calculated via a relaxed linearized augmented Lagrangian method with ordered subsets, and a sparse coding and clustering step, which simultaneously groups the training patches into a fixed number of clusters and learns a transform in each cluster along with the sparse coefficients (in the transform domain) of the patches.

A variant of Zheng *et al.* (2018) is considered by Ye, Ravishankar, Long and Fessler (2018*b*), who use the same pre-learned union of sparsifying transforms as in Zheng *et al.* (2018), but the alternating scheme also includes iteratively updating a quadratic surrogate functions for the data discrepancy. These variants of learned sparsifying transforms are all applied to low-dose CT image reconstruction. Finally, Chen *et al.* (2018) use essentially the method of Zheng *et al.* (2018) applied to cone-beam CT.

4.5. Scattering networks

Scattering networks refer to networks that share the hierarchical structure of deep neural networks but replace data-driven filters with wavelets (Mallat 2012, Bruna and Mallat 2013, Mallat 2016), thus providing an alternative method for parametrizing a regularizer (Dokmanić, Bruna, Mallat and de Hoop 2016). The networks can be made globally invariant to the translation group and locally invariant to small deformations, and they have many other desirable properties, such as stability and regularity. They also manifest better performance than regular CNNs for image classification problems and small-sample training data.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

A key component in defining scattering networks is to have access to wavelet transforms at different scales, $\mathcal{W}_{s_k}: X \to X_{\mathbb{C}}$, where X is the set of real-valued functions on Ω and $X_{\mathbb{C}}$ is the set of complex-valued functions on Ω. We also have $\rho: X_{\mathbb{C}} \to X$, where $\rho(f) = |f|$. For N levels, we now define $\Gamma^N_{s_1,\ldots,s_N}: X \to X$ by

$$\Gamma^N_{s_1,\ldots,s_N}(f) = [\phi * (\rho \circ \mathcal{W}_{s_N}) \circ \ldots \circ (\rho \circ \mathcal{W}_{s_1})](f),$$

and the scattering transform of order N is defined as

$$\Lambda^N(f) := (\phi * f, \Gamma^1_{s_1}(f), \ldots, \Gamma^N_{s_1,\ldots,s_N}(f))_{s_1,\ldots,s_N \in \mathbb{Z}}.$$

The output of $\Lambda^N(f)$ is a hierarchically ordered sequence that forms the 'tree' of scattering coefficients up to order N.

Dokmanić *et al.* (2016) combine the Central Limit Theorem with some additional assumptions to conclude that $\Lambda^N(\mathbb{f})$ with $\mathbb{f} \sim \Pi_{\text{prior}}$ is approximately Gaussian with mean $\bar{f}$ and covariance Σ, where

$$\bar{f} := \mathbb{E}_{\mathbb{f} \sim \widehat{\Pi}_{\text{prior}}}[\Lambda^N(\mathbb{f})] \quad \text{and} \quad \Sigma = \text{Cov}_{\mathbb{f} \sim \widehat{\Pi}_{\text{prior}}}[\Lambda^N(\mathbb{f})].$$

The regularizer $\mathcal{S}$ in (2.7) is now defined as

$$\mathcal{S} := \frac{1}{2}\|\bar{f} - \Lambda^N(f)\|^2_{\Sigma^{-1/2}}.$$

4.6. Black-box denoisers

Several approaches for solving (2.7) explicitly decouple the data discrepancy and regularization terms so that the latter can be treated with stand-alone methods. This is especially useful for cases where the data discrepancy term is differentiable but the regularizer is not.

The Plug-and-Play Prior (P^3) method. This approach, which was introduced in Venkatakrishnan, Bouman and Wohlberg (2013), is based on the observation that a split operator method for minimizing the objective in (2.7) can be posed as an equality constrained problem, that is,

$$\hat{f} := \arg\min_f \{\mathcal{L}(\mathcal{A}(f), g) + \lambda \mathcal{S}(f)\}$$

is equivalent to

$$(\hat{f}, \hat{h}) := \arg\min_{f,h} \{\mathcal{L}(\mathcal{A}(f), g) + \lambda \mathcal{S}(h)\} \quad \text{subject to } f = h.$$

The latter can be solved using ADMM (Section 8.2.7), where the update in h is computed using a proximal operator,

$$h^{k+1)} = \text{prox}_{\tau\lambda\mathcal{S}}(h^{(k)} - f + u), \tag{4.21}$$

where u is a Lagrange (dual) variable.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

The idea is now to replace the proximal operator with a generic denoising operator, which implies that *the regularization functional $\mathcal{S}$ is not necessarily explicitly defined*. This opens the door to switching in any demonstrably successful denoising algorithms without redesigning a reconstruction algorithm – hence the name 'Plug-and-Play'. However, the method comes with some disadvantages. The lack of an explicit representation of the regularizer detracts from a strict Bayesian interpretation of the regularizer as a prior probability distribution in MAP estimation and prevents explicit monitoring of the change in posterior probability of the iterative estimates of the solution. Next, the method is by design tied to the ADMM iterative scheme, which may not be optimal and which requires a non-trivial tuning of parameters of the ADMM algorithm itself (*e.g.* the Lagrangian penalty weighting term). Finally, it is not provably convergent for arbitrary denoising 'engines'.

Regularization by denoising (RED). The RED method (Romano, Elad and Milanfar 2017*a*) is motivated by the P^3 method. It is a variational method where the reconstruction operator is given as in (2.7), where the regularization functional is *explicitly* given as

$$\mathcal{S}(f) := \langle f, f - \Lambda(f) \rangle \quad \text{for some } \Lambda \colon X \to X. \tag{4.22}$$

The Λ operator above is a general (non-linear) *denoising operator*: it can for example be a trained deep neural network. It does, however, need to satisfy two two key properties, which are justifiable in terms of the desirable features of an image denoiser.

Local homogeneity. The denoising operator should locally commute with scaling, that is,

$$\Lambda(cf) = c\Lambda(f)$$

for all $f \in X$ and $|c - 1| \leq \epsilon$ with ϵ small.[5]

Strong passivity. The derivative of Λ should have a spectral radius less than unity:

$$\rho(\partial\Lambda(f)) \leq 1.$$

This is justified by imposing a condition that the effect of the denoiser should not increase the norm of the model parameter:

$$\|\Lambda(f)\| = \|\partial\Lambda(f)f\| \leq \|\rho(\partial\Lambda(f))\| \, \|f\| \leq \|f\|.$$

The key implication from local homogeneity is that the directional derivative of Λ along f is just the application of the denoiser to the f itself:

$$\partial\Lambda(f)f = \Lambda(f).$$

[5] This is a less restrictive condition than requiring equality for all $c \geq 0$.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

The key implication from strong passivity is that it allows for convergence of the proposed RED methods by ensuring convexity of their associated regularization functionals.

Defining $W\colon X \to \mathbb{R}$ implicitly through the relation $W(f)f = \Lambda(f)$ and assuming local homogeneity and strong passivity yields the following computationally feasible expression for the gradient of the regularizer:

$$\nabla \mathcal{S}(f) = f - \Lambda(f) = (\mathrm{id} - W(f))(f). \tag{4.23}$$

Here, the operator $(\mathrm{id} - W(f))$ is further interpreted as an 'image adaptive Laplacian-based' regularizer. The above allows one to implement the RED framework in any optimization such as gradient-descent, fixed-point or ADMM in contrast to the P^3 approach, which is coupled to the ADMM scheme. See also Reehorst and Schniter (2018) for further clarifications and new interpretations of the regularizing properties of the RED method.

4.7. Deep neural networks as regularization functionals

In this section we review two recent approaches (Lunz, Öktem and Schönlieb 2018, Li, Schwab, Antholzer and Haltmeier 2018*b*) to using deep learning to train a regularizer $\mathcal{S}_\theta\colon X \to \mathbb{R}$ in (2.7).

4.7.1. Adversarial regularizer

A recent proposal by Lunz *et al.* (2018) for the construction of data-driven regularizers is inspired by how discriminative networks are trained using modern generative adversarial network (GAN) architectures. Our aim is to learn a regularizer $\mathcal{S}_\theta$, which in some cases (Section 3.3.2) can be interpreted as being proportional to the negative log-likelihood of the prior Π_{prior} in a MAP estimator.

Consider the statistical setting in (3.4) and let $f_i \in X$ be samples of the X-valued random variable $\mathbb{f} \sim \Pi_{\text{prior}}$. Likewise, let $g_i \in X$ be samples of the Y-valued random variable $\mathrm{g} \sim \sigma$ that are independent of the samples f_i, that is, we have unmatched training data. We also assume there exists a (potentially regularizing) pseudo-inverse $\mathcal{A}^\dagger\colon Y \to X$ to the forward operator $\mathcal{A}$ in (3.2) and define the measure $\rho \in \mathscr{P}_X$ as $\rho := \mathcal{A}^\dagger_\#(\sigma)$ for $\sigma \in \mathscr{P}_(Y)$. Note that both Π_{prior} and σ are replaced by their empirical counterparts given by the training data $f_i \in X$ and $g_i \in Y$, respectively.

The idea is to train the regularizer $\mathcal{S}_\theta$ parametrized by a neural network (see Section 7.4 for an exemplar architecture and application of this approach) in order to discriminate between the distributions Π_{prior} and ρ, the latter representing the distribution of imperfect solutions $\mathcal{A}^\dagger(g_i)$. More specifically, we compute

$$\mathcal{S}_{\widehat{\theta}}\colon X \to \mathbb{R}, \tag{4.24}$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

where $\widehat{\theta} \in \arg\min_\theta L(\theta)$, with the loss function $\theta \mapsto L(\theta)$ defined as

$$L(\theta) := \mathbb{E}_{\mathbb{f}\sim\Pi_{\text{prior}}}[G_1(\mathcal{S}_\theta(\mathbb{f}))] - \mathbb{E}_{\mathbb{f}\sim\rho}[G_2(\mathcal{S}_\theta(\mathbb{f}))]. \tag{4.25}$$

Here, $G_1, G_2 \colon \mathbb{R} \to \mathbb{R}$ are monotone functions that have to be chosen. Popular choices for G_i are logarithms, as in the original GAN paper (Goodfellow *et al.* 2014), and the G_i associated with the Wasserstein loss, as in Arjovsky, Chintala and Bottou (2017) and Gulrajani *et al.* (2017). The heuristic behind this choice for the loss function for training a regularizer is that a network trained with (4.25) will penalize noise and artefacts generated by the pseudo-inverse (and contained in ρ). When used as a regularizer, it will hence prevent these undesirable features from occurring. Note also that in practical applications, the measures $\Pi_{\text{prior}}, \rho \in \mathscr{P}_X$ are replaced with their empirical counterparts $\widehat{\Pi}_{\text{prior}}$ and $\hat{\rho}$, given by training data f_i and $\mathcal{A}^\dagger(g_i)$, respectively. The training problem in (4.24) for computing $\widehat{\theta}$ then reads as

$$\widehat{\theta} \in \arg\min_\theta \left\{ \frac{1}{m}\sum_{i=1}^{m} G_1(\mathcal{S}_\theta(f_i)) - \frac{1}{n}\sum_{i=1}^{n} G_2(\mathcal{S}_\theta(\mathcal{A}^\dagger(g_i))) \right\}. \tag{4.26}$$

We also point out that, unlike other data-driven approaches for inverse problems, the above method can be adapted to work with only unsupervised training data. A special case is to have $g_i \approx \mathcal{A}(f_i)$, which gives a unsupervised formulation of (4.26).

Lunz *et al.* (2018) chose a Wasserstein-flavoured loss functional (Gulrajani *et al.* 2017) to train the regularizer, that is, one solves (4.24) with the loss function

$$L(\theta) := \mathbb{E}_{\mathbb{f}\sim\Pi_{\text{prior}}}[\mathcal{S}_\theta(\mathbb{f})] - \mathbb{E}_{\mathbb{f}\sim\rho}[\mathcal{S}_\theta(\mathbb{f})] + \lambda\mathbb{E}[(\|\nabla\mathcal{S}_\theta(\mathbb{f})\| - 1)_+^2]. \tag{4.27}$$

The last term in the loss function serves to enforce the trained regularizer $\mathcal{S}_\theta$ to be Lipschitz-continuous with constant one (Gulrajani *et al.* 2017). Under appropriate assumptions on ρ and π (see Assumptions 4.4 and 4.5) and for the asymptotic case of $\mathcal{S}_{\widehat{\theta}}$ having been trained to perfection, the loss (4.27) coincides with the 1-Wasserstein distance defined in (B.2).

A list of qualitative properties of $\mathcal{S}_{\widehat{\theta}}$ can be proved: for example, Theorem 1 of Lunz *et al.* (2018) shows that under appropriate regularity assumptions on the Wasserstein distance between ρ and π, starting from elements in ρ and taking a gradient descent step of $\mathcal{S}_{\widehat{\theta}}$ (which results in a new distribution ρ_η) strictly decreases the Wasserstein distance between the new distribution ρ_η and π. This is a good indicator that using $\mathcal{S}_{\widehat{\theta}}$ as a variational regularization term, and consequently penalizing it, indeed introduces the highly desirable incentive to align the distribution of regularized solutions with the distribution of ground truth samples Π_{prior}. Another characterization of such a trained regularizer $\mathcal{S}_{\widehat{\theta}}$ using the Wasserstein loss in (B.2) is

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

in terms of a distance function to a manifold of desirable solutions $\mathcal{M}$. To do so, we make the following assumptions.

Assumption 4.4 (weak data manifold assumption). Assume that the measure ρ is supported on a weakly compact set $\mathcal{M}$, *i.e.* $\rho(\mathcal{M}^c) = 0$

This assumption captures the intuition that real data lie in a lower-dimensional submanifold of X, which is a common assumption for analysing adversarial networks (Goodfellow *et al.* 2014). Moreover, it is assumed that the distributions π can be recovered from the distribution π through an appropriate projection onto the data manifold $\mathcal{M}$.

Assumption 4.5. Assume that π and ρ satisfy $(P_{\mathcal{M}})_{\#}(\rho) = \pi$, where $P_{\mathcal{M}} \colon D \to \mathcal{M}$ is the mapping $x \mapsto \arg\min_{y \in \mathcal{M}} \|x - y\|$. Here D denotes the set of points for which such a projection exists (which under weak assumptions on $\mathcal{M}$ and ρ can be assumed to cover all of ρ, *i.e.* $\rho(D) = 1$)

Note that Assumption 4.5 is weaker than assuming that any given f can be recovered by projecting the pseudo-inverse of the corresponding g back onto the data manifold. Assumption 4.5 can instead be considered as a low-noise assumption. These assumptions yield the following theorem.

Theorem 4.6 (Lunz, Öktem and Schönlieb 2018). The distance function to the data manifold $d_{\mathcal{M}}(x) := \min_{y \in \mathcal{M}} \|x - y\|$ is a maximizer to (B.2) under Assumptions 4.4 and 4.5.

Theorem 4.6 shows that, if $\mathcal{S}_\theta$ were trained to perfection, *i.e.* trained so that $\mathcal{S}_\theta$ solves (B.2), then $\mathcal{S}_\theta$ would be given by the L^2-distance function to $\mathcal{M}$. This is implicitly also done in the RED approach described in Section 4.6. Similarly, Wu, Kim, Fakhri and Li (2017) learn a regularizer in a variational model given as in Wu *et al.* (2017, equation (3)) from unsupervised data by means of a K-sparse auto-encoder. This yields a regularizer that minimizes the distance of the image to the data manifold.

Finally, a weak stability result for $\mathcal{S}_\theta$ is proved in the spirit of the classical theory provided in Engl *et al.* (2000). Since $\mathcal{S}_\theta$ is not necessarily bounded from below, the 1-Lipschitz property of $\mathcal{S}_\theta$ is used instead to prove this stability result.

Theorem 4.7 (Lunz, Öktem and Schönlieb 2018). Under appropriate assumptions on $\mathcal{A} \colon X \to Y$ given in Lunz *et al.* (2018, Appendix A), the following holds. Consider a sequence $\{g_n\}_n \subset Y$ with $g_n \to g$ in the norm topology in Y and let $\{f_n\} \subset X$ denote a sequence of corresponding minimizers of (2.7), that is,

$$f_n \in \arg\min_{f \in X} \{\| \mathcal{A}(f) - g_n\|^2 + \lambda \mathcal{S}_\theta(f)\}.$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Then f_n has a subsequence that converges weakly to a minimizer of $f \mapsto \|\mathcal{A}(f) - g\|^2 + \lambda \mathcal{S}_\theta(f)$.

Theorem 4.7 constitutes a starting point for further investigation into the regularizing properties of adversarial regularizers $\mathcal{S}_\theta$. Section 7.4 shows the application of the adversarial regularizer to CT reconstruction.

4.7.2. The neural network Tikhonov (NETT) approach

Another proposal for learning a regularizer in (2.7) by a neural network is given in Li *et al.* (2018*b*) and called the NETT approach. NETT is based on composing a pre-trained network $\Psi_\theta \colon X \to \Xi$ with a regularization functional $\mathcal{S} \colon \Xi \to [0, \infty]$, such that $\mathcal{S} \circ \Psi_\theta \colon X \to [0, \infty]$ takes small values for desired model parameters and penalizes (larger values) model parameters with artefacts or other unwanted structures.

Here, $\Psi_\theta \colon X \to \Xi$ and $\mathcal{S} \colon \Xi \to [0, \infty]$ and the deep neural network Ψ_θ is allowed to be a rather general network model, a typical choice for Ψ_θ being an auto-encoder network. Once trained, the reconstruction operator is given by

$$\mathcal{R}_\theta := \arg\min_f \mathcal{J}_\theta(f), \tag{4.28}$$

where $\mathcal{J}_\theta(f) := \mathcal{L}(\mathcal{A}(f), g) + \lambda \mathcal{S}(\Psi_\theta(f))$.

The main focus of Li *et al.* (2018*b*) is a discussion on analytic conditions, which guarantees that the NETT approach is indeed a regularization method in the sense of functional analytic regularization. In particular, Li *et al.* discuss assumptions on $\mathcal{S}$ and Ψ_θ such that the functional analytic regularization theory of Grasmair *et al.* (2008) can be applied. This theory requires a weakly lower semicontinuous and coercive regularization term, which obviously holds for many deep neural networks Ψ_θ with coercive activation functions. Accordingly, Li *et al.* (2018*b*) discuss replacing the usual ReLU activation function with leaky ReLU, defined with a small $\tau > 0$ as

$$\ell\mathrm{ReLU}_\tau(s) := \max(\tau s, s).$$

For $s \to -\infty$, leaky ReLU also tends to $-\infty$, which in combination with the affine linear maps $\mathcal{W}$ in Ψ_θ yields a coercive and weakly lower semicontinuous regularization function $\mathcal{S} \circ \Psi_\theta$ for standard choices of $\mathcal{S}$, such as weighted ℓ_p-norms $\mathcal{S}(\xi) = \sum_i v_i |\xi_i|^p$ with uniformly positive weights v_i and $p \geq 1$. They even go beyond these classical results and, by introducing the novel concept of absolute Bregman distances (Li *et al.* 2018*b*), they obtain convergence results and convergence rates in the underlying function space norm.

Li *et al.* (2018*b*) discuss the application of NETT to PAT reconstruction from limited data. The neural network Ψ_θ is trained against supervised data $(f_i, h_i) \in X \times X$, where f_i serves as ground truth model parameter

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

and $h_i = (\mathcal{A}^\dagger \circ \mathcal{A})(f_i)$, where $\mathcal{A}^\dagger \colon Y \to X$ is some (regularized) pseudo-inverse of the forward operator $\mathcal{A}$. In tomographic applications it is taken as FBP, so h_i will typically contain sampling artefacts. The network Ψ_θ is then modelled as an auto-encoder, more specifically as the encoder part of an encoder–decoder neural network. The parameters θ are trained by minimizing a loss function that evaluates the capability of the encoder–decoder pair to reproduce desirable f as well as $h = (\mathcal{A}^\dagger \circ \mathcal{A})(f)$ with artefacts using an appropriate distance function. After training Ψ_θ as described above, the NETT functional (4.28) is minimized by a generalized gradient descent method. The numerical results obtained with artificial data confirm the theoretical findings but lack comparison or seem to be slightly less favourable when compared with results obtained with other neural network approaches: see *e.g.* Hauptmann *et al.* (2018).

4.8. Learning optimal data acquisition schemes

Learning has also been used to determine data sampling patterns. An example is that of Baldassarre *et al.* (2016), who use training signals and develop a combinatorial training procedure which efficiently and effectively learns the structure inherent in the data. Thereby it is possible to design measurement matrices that directly acquire only the relevant information during acquisition. The resulting data sampling schemes not only outperform the existing state-of-the-art compressive sensing techniques on real-world datasets (including neural signal acquisition and magnetic resonance imaging): they also come with strong theoretical guarantees. In particular, Baldassarre *et al.* (2016) describe how to optimize the samples for the standard linear acquisition model along with the use of a simple linear decoder, and build towards optimizing the samples for non-linear reconstruction algorithms.

Gözcü *et al.* (2018) apply these techniques to MRI in order to learn optimal subsampling patterns for a specific reconstruction operator and anatomy, considering both the noiseless and noisy settings. Examples are for reconstruction operators given by sparsity-promoting variational regularization. The idea is to parametrize the data sampling (data acquisition and its digitization), then learn over these parameters in a supervised setting.

4.9. Data-driven optimization

Reconstruction methods given by variational methods or by the MAP estimator give rise to an optimization problem with an objective parametrized by $g \in Y$:

$$\mathcal{R}(g) := \operatorname*{arg\,min}_{f \in X} \mathcal{J}(f, g) \tag{4.29}$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

for $g \in Y$ and $\mathcal{J}\colon X \times Y \to \mathbb{R}$. A typical example is that of reconstruction methods in (2.7), where $\mathcal{J}(f, g) := \mathcal{L}(\mathcal{A}(f), g) + \mathcal{S}(f)$.

The *objective here is to use data-driven methods for faster evaluation of* $\mathcal{R}$, which is often computationally demanding. Given a parametrized family (an architecture) $\{\mathcal{R}_\theta\}_\theta$ of candidate solution operators, the 'best' approximation to $\mathcal{R}$ is given by $\mathcal{R}_{\widehat{\theta}}\colon Y \to X$, where $\widehat{\theta}$ solves the unsupervised learning problem:

$$\widehat{\theta} \in \arg\min_\theta \mathbb{E}_{\mathrm{g}\sim\sigma}[\mathcal{J}(\mathcal{R}_\theta(\mathrm{g}), \mathrm{g})]. \tag{4.30}$$

The probability distribution σ of the random variable g generating data is unknown. In (4.30) it is therefore replaced by its empirical counterpart derived from unsupervised training data $g_1, \ldots, g_m \in Y$ that are samples of $\mathrm{g} \sim \sigma$. In such a case, the unsupervised learning problem in (4.30) reads as

$$\widehat{\theta} \in \arg\min_\theta \left\{ \frac{1}{m} \sum_{i=1}^{m} \mathcal{J}(\mathcal{R}_\theta(g_i), g_i) \right\}, \tag{4.31}$$

where $g_1, \ldots, g_m \in Y$ are samples of $\mathrm{g} \sim \sigma$.

Note that the loss in (4.30) involves evaluations of the objective $f \mapsto \mathcal{J}(f, g)$, so training can be computationally quite demanding. However, the training is an off-line batch operation, so the computational performance requirements on $\mathcal{R}$ are much more relaxed during training than when $\mathcal{R}$ is used for reconstruction. Besides having access to a sufficient amount of unsupervised training data, a central part in successfully realizing the above scheme is to select an appropriate architecture for $\mathcal{R}_\theta$. It should be computationally feasible, yet one should be able to approximate $\mathcal{R}$ with reasonable accuracy by solving (4.31).

An early example in the context of tomographic reconstruction is that of Pelt and Batenburg (2013) and Plantagie and Batenburg (2015). Here, $\mathcal{R}$ is the TV-regularized reconstruction operator, which in many imaging applications is considered computationally unfeasible. Each $\mathcal{R}_\theta$ is given as a non-linear combination of FBP reconstruction operators that are fast to compute. These non-linear combinations of the reconstruction filters in the FBP reconstruction operators are all learned by training against the outcome of the TV regularization as in (4.31). The architecture used in the above-cited publications uses FBP operators, which only makes sense for inverse problems involving inversion of the ray transform. Another line of development initiated by Gregor and LeCun (2010) considers deep neural network architectures for approximating reconstruction operators $\mathcal{R}$ given by (2.20) (sparse coding). The idea is to incorporate selected components of an iterative scheme, in this case ISTA (Section 8.2.7), to solve the optimization problem in (2.20). This is done by 'unrolling' the iterative scheme and replacing its explicit updates with learned ones, which essentially amounts

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

to optimizing over optimization solvers; see also Andrychowicz *et al.* (2016) for similar work. This has been used for more efficient solution of variational problems arising in a large-scale inverse problem. As an example, Hammernik, Knoll, Sodickson and Pock (2016) and Lee, Yoo and Ye (2017) consider unrolling for compressed sensing MRI reconstruction and Schlemper *et al.* (2018) for dynamic two-dimensional cardiac MRI, the latter aiming to solve the variational problem (Schlemper *et al.* 2018, equation (4)). Similarly, Meinhardt, Moeller, Hazirbas and Cremers (2017) consider unrolling for deconvolution and demosaicking.

Remark 4.8. The idea of optimizing over optimization solvers also appears in *reinforcement learning*. Furthermore, a similar problem is treated in Drori and Teboulle (2014), which considers the worst-case performance

$$\sup_{\theta,g}\{\mathcal{J}(\mathcal{R}_\theta(g),g)-\mathcal{J}(\mathcal{R}(g),g)\}$$

with $\mathcal{R}_\theta$ given by a gradient-based scheme that is stopped after N steps. It is assumed here that $f\mapsto\mathcal{J}(f,g)$ is continuously differentiable with Lipschitz-continuous gradients, and with a uniform upper bound on the Lipschitz constants. Subsequent work along the same lines can be found in Kim and Fessler (2016) and Taylor, Hendrickx and Glineur (2017).

4.9.1. General principle of unrolling

The aim of unrolling is to find a deep neural network architecture $\mathcal{R}_\theta\colon Y\to X$ that is especially suited to approximating an operator $\mathcal{R}\colon Y\to X$ that is implicitly defined via an iterative scheme. A typical example is when $\mathcal{R}(g)$ is implicitly defined via an iterative scheme that is designed to converge to a (local) minimum of $f\mapsto\mathcal{J}(f,g)$.

We start by giving an illustrative example of how to unroll an iterative gradient descent scheme. This is followed by an outline of the general principle of unrolling. Two specific cases are considered in Sections 4.9.2 and 4.9.3.

Unrolled gradient descent as a neural network. Consider the case when $\mathcal{J}(\cdot,g)\colon X\to\mathbb{R}$ is smooth for any $g\in Y$ and assume $\mathcal{R}(g)$ is given by a standard gradient descent algorithm, that is,

$$\mathcal{R}(g)=\lim_{k\to\infty}f^k\quad\text{where}\quad\begin{cases}f^0=f_0\quad\text{is given,}\\ f^k=f^{k-1}-\omega_k\nabla_f\mathcal{J}(f^k,g)\quad\text{for } k=1,2,\dots.\end{cases}$$

Each parameter ω_k is a step length for the kth iteration, and normally these are chosen via the Goldstein rule or backtracking line search (Armijo rule) (Bertsekas 1999), which under suitable conditions ensures convergence to a minimum.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Computational feasibility of the above scheme is directly tied to the number of iterates necessary to get sufficiently close to the desired local minima. Time limitations imposed by many applications limit the maximum number of evaluations of $f \mapsto \mathcal{J}(f, g)$, that is, we have an *a priori* bound N on the number of iterates. Unrolling the gradient descent scheme defining $\mathcal{R}$ and stopping the iterates after N steps allows us to express the Nth iterate as

$$\mathcal{R}_\theta(g) := (\Lambda_{\omega_N} \circ \ldots \circ \Lambda_{\omega_1})(f_0) \quad \text{with } \theta := (\omega_1, \ldots, \omega_N), \tag{4.32}$$

where $\Lambda_{\omega_k} \colon X \to X$ are updating operators given by

$$\Lambda_{\omega_k} := \mathrm{id} - \omega_k \nabla \mathcal{J}(\,\cdot\,, g) \quad \text{for } k = 1, \ldots, N. \tag{4.33}$$

Now, note that $\mathcal{R}_\theta$ can be seen as a *feed-forward neural network* where each layer in the network evaluates Λ_{ω_k} and the parameters of the network are $\theta = (\omega_1, \ldots, \omega_N)$. Moreover, if the step length is fixed, *i.e.* $\omega_1 = \cdots = \omega_N = \omega$ for some ω, the gradient descent algorithm can in fact be interpreted as a *recurrent neural network*. For both cases, the best choice of step lengths $\omega_1, \ldots, \omega_N$ for approximating $\mathcal{R} \colon Y \to X$ with N gradient descent iterates can be obtained by unsupervised learning: simply solve (4.31) with $\mathcal{R}_\theta$ as in (4.32).

A further option is to replace the explicit updating in (4.33), used to define $\mathcal{R}_\theta$ in (4.32), with generic deep neural networks:

$$\Lambda_{\theta_k} := \mathrm{id} + \Gamma_{\theta_k}(\,\cdot\,, \nabla \mathcal{J}(\,\cdot\,, g)) \quad \text{where } \Gamma_{\theta_k} \colon X \times X \to X. \tag{4.34}$$

Each Γ_{θ_k} is now a deep neural network, for example a CNN, that is trained against unsupervised data by solving (4.31) with $\mathcal{R}_\theta$ as in (4.34).

Abstract unrolled schemes. The above example of unrolling a gradient descent and replacing its explicit updates with a neural network trained by unsupervised learning applies to many other iterative schemes. It essentially amounts to optimizing over optimization solvers. Since we seek $g \mapsto \mathcal{R}(g)$ given by (4.29), we are not only interested in optimizing a single objective, but rather an infinite family $\{\mathcal{J}(\,\cdot\,, g)\}_g$ of objectives parametrized by data $g \in Y$.

More precisely, if $f \mapsto \mathcal{J}(f, g)$ in (4.29) is smooth, then it is natural to consider an iterative scheme that makes use of the gradient of the objective. Given an initial model parameter $f^0 \in X$ (usually set to zero), such schemes can be written abstractly as

$$\begin{cases} f^0 = f_0 \in X \text{ chosen}, \\ f^{k+1} := \Gamma_{\theta_k}(f^k, \boldsymbol{f}_m^k, \nabla_f \mathcal{J}(f^k, g)), \end{cases}$$

for some updating operator $\Gamma_{\theta_k} \colon X \times X^m \times X \to X$. The above formulation includes accelerated schemes, like fast gradient methods (Nesterov 2004) and quasi-Newton methods (Nocedal and Wright 2006). Now, stopping and

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

unrolling the above scheme after N iterates amounts to defining $\mathcal{R}_\theta : Y \to X$ with $\theta = (\theta_1, \ldots, \theta_N)$ as

$$\mathcal{R}_\theta(g) = (\Lambda_{\theta_N} \circ \ldots \circ \Lambda_{\theta_1})(f^0),$$

where $\Lambda_{\theta_k} : X \times X^m \to X \times X^m$ is defined as

$$\Lambda_{\theta_k}(f, \boldsymbol{h}) := (\Gamma_{\theta_k}(f, \boldsymbol{h}, \nabla_f \mathcal{J}(f, g)), \boldsymbol{h}') \quad \text{with } \boldsymbol{h}' := (h_2, \ldots, h_m) \in X^{m-1}$$

and $\mathcal{P}_X : X \times X^m \to X$ is the usual projection. The final step is to replace the hand-crafted updating operators $\Gamma_{\theta_k} : X \times X^m \times X \to X$ with deep neural networks that then train the resulting $\mathcal{R}_\theta : Y \to X$ against unsupervised data as in (4.31).

In inverse problem applications, the objective in (4.29) often has further structure that can be utilized. A typical example is (2.7), where the objective is of the form

$$\mathcal{J}(f, g) := \mathcal{L}(\mathcal{A}(f), g) + \mathcal{S}_\lambda(f). \tag{4.35}$$

A wide range of iterative schemes that better utilize such a structure include updating in both X (primal) and Y (dual) spaces. These can be written abstractly as

$$\begin{cases} (f^0, g^0) = (f_0, g) \in X \times Y \text{ with } f_0 \text{ chosen}, \\ (f^{k+1}, g^{k+1}) := \Gamma_{\theta_k}(f^k, \boldsymbol{f}^k_{m_1}, g^k, \boldsymbol{g}^k_{m_2}, [\partial \mathcal{A}(f^k)]^*(g^k), \mathcal{A}(f^k), \nabla \mathcal{S}_\lambda(f^k),) \end{cases} \tag{4.36}$$

for some updating operator

$$\Gamma_{\theta_k} : X \times X^{m_1} \times Y \times Y^{m_2} \times X \times Y \times X \to X \times Y. \tag{4.37}$$

Here, m_1, m_2 is the memory in the X- and Y-iterates, so

$$\boldsymbol{f}^k_{m_1} = (f^{k-1}, \ldots, f^{k-m_1}) \in X^{m_1} \quad \text{and} \quad \boldsymbol{g}^k_{m_2} = (g^{k-1}, \ldots, g^{k-m_2}) \in Y^{m_2},$$

with the convention that $f^{k-l} = f^0$ and $g^{k-l} = g$ whenever $k - l < 0$. Stopping and unrolling such a scheme after N iterates amounts to defining $\mathcal{R}_\theta : Y \to X$ with $\theta = (\theta_1, \ldots, \theta_N)$ as

$$\mathcal{R}_\theta(g) := (\mathcal{P}_X \circ \Lambda_{\theta_N} \circ \ldots \circ \Lambda_{\theta_1})(f^0, \boldsymbol{f}^0_{m_1}, g, \boldsymbol{g}^0_{m_2}), \tag{4.38}$$

where $\Lambda_{\theta_k} : X \times X^{m_1} \times Y \times Y^{m_2} \to X \times X^{m_1} \times Y \times Y^{m_2}$ for $k = 1, \ldots, N$ is defined as

$$\Lambda_{\theta_k}(f, \boldsymbol{h}, v, \boldsymbol{v}) := (\hat{f}_{\theta_k}, \boldsymbol{h}', \hat{v}_{\theta_k}, \boldsymbol{v}'),$$

with $\boldsymbol{h}' := (h_2, \ldots, h_{m_1}) \in X^{m_1 - 1}$, $\boldsymbol{v}' := (v_2, \ldots, v_{m_2}) \in X^{m_2 - 1}$, and

$$(\hat{f}_{\theta_k}, \hat{v}_{\theta_k}) := \Gamma_{\theta_k}(f, \boldsymbol{f}, v, \boldsymbol{v}, [\partial \mathcal{A}(f)]^*(v), \mathcal{A}(f), \nabla \mathcal{S}_\lambda(f)) \in X \times Y.$$

In the above, $\mathcal{P}_X : X \times X^{m_1} \times Y \times Y^{m_2} \to X$ is the usual projection and, just as before, these hand-crafted updating operators are replaced with deep

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

neural networks and the resulting deep neural network $\mathcal{R}_\theta \colon Y \to X$ is trained against training data, for example as in (4.31) in the case with unsupervised data.

Likewise, in the setting where $\mathcal{S}_\lambda \colon X \to \mathbb{R}$ in (4.35) is non-smooth, the prototype proximal-gradient scheme without memory reads as

$$\begin{cases} f^{k+1/2} := \Lambda_{\theta_k^1}(f^k, \nabla_f \mathcal{L}(\mathcal{A}(f^k), g)), \\ f^{k+1} := \Lambda_{\theta_k^2}(f^k, f^{k+1/2}, \operatorname{prox}_{\gamma \mathcal{S}_\lambda}(f^{k+1/2})), \end{cases} \tag{4.39}$$

for updating operators $\Lambda_{\theta_k^1} \colon X \times X \to X$ and $\Lambda_{\theta_k^2} \colon X \times X \times X \to X$. Now, just as before, one can terminate and unroll the above scheme after N iterates and replace the updating operators with deep neural networks. The resulting deep neural network $\mathcal{R}_\theta \colon Y \to X$, where $\theta = (\theta_1^1, \ldots, \theta_N^1, \theta_1^2, \ldots, \theta_N^2)$, is then trained against unsupervised data as in (4.31). Just as in the smooth case, it is furthermore possible to add memory and/or 'break up' the objective even further, the latter in order to better account for the structure in the problem. For example, variational methods for linear inverse problems often result in minimizing an objective of the form

$$\mathcal{J}(f, g) := \mathcal{S}_\lambda(f) + \sum_{i=1}^{m} \mathcal{L}_i(\mathcal{A}_i(f), g_i)$$

where $\mathcal{A}_i \colon X \to Y_i$ are linear and $\mathcal{S}_\lambda \colon X \to [-\infty, \infty]$ and $\mathcal{L}_i \colon Y_i \times Y_i \to [-\infty, \infty]$ are proper, convex and lower semicontinuous. One can then consider unrolling iterative schemes that utilize this structure, such as operator splitting techniques: see Eckstein and Bertsekas (1992), Beck and Teboulle (2009), Chambolle and Pock (2011), Boyd *et al.* (2011), Combettes and Pesquet (2011, 2012), He and Yuan (2012), Boţ and Hendrich (2013), Boţ and Csetnek (2015), Ko, Yu and Won (2017), Latafat and Patrinos (2017) and Bauschke and Combettes (2017).

A natural question is to investigate the error in using $\mathcal{R}_\theta$ to approximate $\mathcal{R}$ as a function of N and properties of the unsupervised training data. Such estimates, which are referred to as time–accuracy trade-offs, have been proved for LISTA (Section 4.9.2) by Giryes, Eldar, Bronstein and Sapiro (2017); see also Oymak and Soltanolkotabi (2017) for further development along these lines. Another related type of investigation has been pursued by Banert *et al.* (2018), who derive conditions on the unrolled iterative scheme that is terminated after N iterates, so that training this yields a scheme that is convergent in the limit. It turns out that imposing such convergence constraints has only a minor impact on the performance at N iterates. Furthermore, it improves the generalization properties, that is, one can use the same trained solver even when the objective is slightly changed, for example by considering a different forward operator.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

To summarize, the iterative update in most schemes for solving optimization problems has the structure of a composition of (possibly multiple) affine operations followed by a non-linear operation. If each iterate is identified with a layer and the non-linear operation plays the role of an activation function, *e.g.* $\mathcal{S}_\lambda(f) = \lambda\|f\|$, then $\mathrm{prox}_{\lambda\,\mathcal{S}}$ corresponds to soft thresholding, which in turn is very close to the well-known ReLU activation function. In this way the iterative scheme stopped after N steps can be represented via a neural network with an architecture that is adapted to that iterative scheme.

4.9.2. Learned Iterative Soft-Thresholding Algorithm (LISTA)

LISTA is the earliest example of unrolling an optimization scheme and it was first introduced in Gregor and LeCun (2010). It is the abstract unrolling scheme in Section 4.9.1 adapted to the specific case of ISTA iterates (Section 8.2.7). This results in a fully connected network with N-internal layers of identical size that is adapted to evaluating the solution operator for the following convex non-smooth optimization problem:

$$\mathcal{R}(g) := \mathcal{E}^*(\hat{\xi}), \tag{4.40}$$

where

$$\hat{\xi} \in \arg\min_{\xi\in\Xi}\{\|g - \mathcal{A}\circ\mathcal{E}^*(\xi)\|_2^2 + \lambda\|\xi\|_1\}.$$

Such optimization problems arise as the convex relaxation of (2.20), which is sparsity-promoting regularization of an ill-posed linear inverse problem (Section 2.7).

The ISTA iterative scheme for evaluating $\mathcal{R}$ reads as

$$\xi^{n+1} = \mathrm{prox}_{\lambda\tau\|\cdot\|_1} = S_{\lambda\tau}[\tau\,\mathcal{A}_{\mathcal{E}^*}^*\,g + (\mathrm{id} - \tau\,\mathcal{A}_{\mathcal{E}^*}^*\,\mathcal{A}_{\mathcal{E}^*})\xi^n], \tag{4.41}$$

where $0 < \tau < 2/L$ is the step length approximated by the reciprocal of the Lipschitz constant of $\mathcal{A}_{\mathcal{E}^*}$, and S is the shrinkage operator (see Section 8.2.7). Now the insight is to recognize $b_k(g) := \tau_k\,\mathcal{A}_{\mathcal{E}^*}^*\,g$ as a *bias*, $W_k := (\mathrm{id} - \tau_k\,\mathcal{A}_{\mathcal{E}^*}^*\,\mathcal{A}_{\mathcal{E}^*})$ as a fixed linear operator and $\psi_k = S_{\lambda\tau_k}$ as a *pointwise non-linearity*. Note that W_k is symmetric positive definite by construction. Assuming a fixed iteration number N, we may write the approximation to (4.40) as

$$\begin{cases} \xi^{(0)} \in \Xi \text{ given} \\ \xi^{(k)} = \psi(W_k\xi^{(k-1)} + b_k(g)) \end{cases} \qquad \text{for } k = 1,\ldots,N.$$

In unrolled form, the above defines $\mathcal{R}_\theta\colon Y \to \Xi$ with $\theta_k := (\psi, \tau_k, W_k, b_k)$ as

$$\mathcal{R}_\theta := (\Lambda_{\theta_N}\circ\ldots\circ\Lambda_{\theta_1})(\xi^{(0)}) \tag{4.42}$$

where $\Lambda_{\theta_k}\colon \Xi \to \Xi$ is $\Lambda_{\theta_k} := \psi_k \circ (W_k + b_k(g)\,\mathrm{id})$. This is precisely the form

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

of a feed-forward network with N layers, each with identical fixed affine map $\mathcal{W}_k(\xi) := W_k\xi + b_k$; the activation function $\psi_k := S_{\lambda\tau_k}$ is given by the shrinkage operator (see Section 8.2.7).

Remark 4.9. The expression given in (4.42) has the form of an *encoder* $\Psi_W : Y \to \Xi$ for a fixed dictionary given by W.

The affine maps between the layers are assumed to be identical. LISTA then trains W and potentially ψ on some given unsupervised training data by solving (4.31). The derivation of Gregor and LeCun (2010) is framed in terms of learning a sparse encoder. Here we rephrase the idea in terms of a reconstruction problem and a learned *decoder* as follows.

Lemma 4.10. Let $\Psi_W^\dagger : \Xi \to X$ denote a fully connected network with input ξ^0 and N-internal layers. Furthermore, assume that the activation function is identical to a proximal mapping for a convex functional $\tau\lambda\,\mathcal{S}\colon \Xi \to \mathbb{R}$. Also, let W be restricted so that $\mathrm{id} - W$ is positive definite, *i.e.* there exists a matrix B such that

$$\mathrm{id} - W = \tau B^* B.$$

Finally, fix the bias term as $b = \tau B^* g$. Then $\Psi_W^\dagger(\xi)$ is the Nth iterate of an ISTA scheme with starting value $\xi^{(0)}$ for minimizing

$$\mathcal{J}_B(\xi) = \frac{1}{2}\|B\xi - g\|^2 + \lambda\,\mathcal{S}(\xi). \tag{4.43}$$

Note that the connection to ISTA is only given when the weights are the same across layers. The conclusion in the lemma follows directly from (A.10). In this sense, training a decoder network $\Psi_W^\dagger$ by minimizing

$$\frac{1}{2}\|\,\mathcal{A}\,\Psi_W^\dagger(\xi) - g\|^2$$

with respect to W, and computing $\hat{f} = \Psi_W^\dagger(\xi)$ is equivalent to computing $\hat{B}$ by minimizing a Tikhonov functional of the form (4.43) with respect to B, then computing $\hat{f} := \Psi_W^\dagger(\hat{B})$ as a solution to the inverse problem. Following these arguments, one can rephrase LISTA as a concept for learning the discrepancy term in a classical Tikhonov functional.

Remark 4.11. The restriction of activation functions to proximal mappings is not as severe as it might look at first glance. For example, as already mentioned in Section 4.9.1, ReLU is the proximal mapping for the indicator function of positive real numbers and soft shrinkage is the proximal mapping for the modulus function.

4.9.3. Learned proximal operators

Just as with LISTA, the *learned proximal operator* approach is the abstract unrolling scheme in Section 4.9.1 adapted to the specific case of unrolling

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

operator-splitting approaches, such as proximal gradient (PG), primal–dual hybrid gradient (PDHG) or ADMM; see also Section 8.2.7 and Chambolle and Pock (2016).

More precisely, consider the variational regularization that minimizes the functional in (4.35) under the Gaussian noise log-likelihood model with covariance Σ and a non-smooth regularizer, that is,

$$\mathcal{R}(g) := \hat{f}, \tag{4.44}$$

where

$$\hat{f} := \arg\min\{\lambda\,\mathcal{S}(f) + \|\,\mathcal{A}(f) - g\|^2_{\Sigma^{-1}}\}$$

and $\mathcal{S}\colon X \to \mathbb{R}$ is non-smooth. Our aim is to approximate $\mathcal{R}(g)$ in (4.44) by a deep neural network that is obtained by unrolling a suitable scheme.

Unrolling iterates of a PG method (Section 8.2.7) and replacing the proximal operator with a learned operator yields

$$\mathcal{R}_\theta(g) := f^N, \tag{4.45}$$

where

$$f^{k+1} := \Lambda_\theta(f^k + \tau\,\mathcal{A}^*\,\Sigma^{-1}(\mathcal{A}\,f^k - g)).$$

In the above, $\Lambda_\theta\colon X \to X$ is a deep neural network that replaces the proximal of $\mathcal{S}$ in (4.44). Similarly, the ADMM framework (Section 8.2.7) has the augmented Lagrangian

$$\mathcal{L}(f,h,u) = \frac{1}{2}\|\,\mathcal{A}(f) - g\|^2_{\Sigma^{-1}} + \lambda\,\mathcal{S}(h) + \frac{\beta}{2}\left\|f - h + \frac{1}{\beta}u\right\|^2_2 - \frac{1}{2\beta}\|u\|^2_2, \tag{4.46}$$

which results in $\mathcal{R}_\theta(g) := f^N$, where

$$f^{k+1} := (\mathcal{A}^*\,\Sigma^{-1}\,\mathcal{A} + \beta\,\mathrm{id})^{-1}(\mathcal{A}^*\,\Sigma^{-1}(\mathcal{A}(f^k) - g) + \beta(f^k - h^k) + u^k), \tag{4.47}$$

$$h^{k+1} := \Lambda_\theta\left(f^{k+1} + \frac{1}{\beta}u^k\right), \tag{4.48}$$

$$u^{k+1} := u^k + \beta(f^{k+1} - h^{k+1}), \tag{4.49}$$

where the general expression for the proximal operator of the log-likelihood from (A.12) has the direct form (4.47), and the general expression for the proximal operator of $\mathcal{S}$ from (A.13) is replaced with a network $\Lambda_\theta\colon X \to X$ in (4.48).

This compares to other P^3 approaches (Section 4.6) using a denoising method such as BM3D in place of the proximal operator. An important point noted in Meinhardt *et al.* (2017) is that by choosing $\beta = 1$ in (4.46), the networks do not need to be retrained on different noise levels if they can be considered as a scaling of the likelihood term. Related work includes

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

'deep image demosaicing' using a cascade of convolutional residual denoising networks (Kokkinos and Lefkimmiatis 2018, Lefkimmiatis 2017), which includes learning of the activation function.

A related method is that of Chang *et al.* (2017), who propose a general framework that implicitly learns a signal prior and a projection operator from a large image dataset and is predicated on the ADMM framework (Section 8.2.7). In place of an 'ideal' prior defined as the indicator function $\mathbb{I}_{X_0}\colon X \to \mathbb{R}$ on the set of natural images $X_0 \subset X$ and its proximal operator, they make use of a trained classifier $\mathcal{D}\colon X \to [0,1]$ and a learned projection function $\mathcal{P}$ that maps an estimated $\hat{f} \in X$ to the set defined by the classifier; the learned $\mathcal{P}$ then replaces the proximal operator within the ADMM updates. They identify sufficient conditions for the convergence of the non-convex ADMM with the proposed projection operator, and use these conditions as guidelines to design the proposed projection network. They show that it is inefficient at solving generic linear inverse problems with state-of-the-art methods using specially trained networks. Experimental results also show that these are prone to being affected by changes in the linear operators and noise in the linear measurements. In contrast, the proposed method is more robust to these factors. Results are shown for the trained network applied to several problems including compressive sensing, denoising, in-painting (random pixels and block-wise) and super-resolution, and to different databases including MNIST, MS-Celeb-1M dataset and the ImageNet dataset.

4.9.4. Summary and concluding remarks on unrolling

The idea of unrolling can be seen as constructing a deep neural network architecture. Here we used it to approximate solution operators to optimization problems, which are defined implicitly through an iterative optimization solver. This principle can, however, be applied in a much wider context and it also establishes a link between numerical analysis and deep learning.

For example, the learned iterative scheme (Section 5.1.4) uses unrolling to construct a deep learning architecture for the solution of an inverse problem that incorporates a forward operator and the adjoint of its derivative. Another example of using unrolling is that of Gilton, Ongie and Willett (2019), who construct a deep neural network by unrolling a truncated Neumann series for the inverse of a linear operator. This is used to successfully solve inverse problems with a linear forward operator. Gilton *et al.* claim that the resulting Neumann network architecture outperforms functional analytic approaches (Section 2), model-free deep learning approaches (Section 5.1.3) and state-of-the-art learned iterative schemes (Section 5.1.4) on standard datasets.

One can also unroll iterative schemes for solving PDEs, as shown by Hsieh *et al.* (2019), who unroll an iterative solver tailored to a PDE and modify

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

the updates using a deep neural network. A key part of their approach is ensuring that the learned scheme has convergence guarantees. See also Rizzuti, Siahkoohi and Herrmann (2019) for a similar approach to solving the Helmholtz equation.

Finally, we also mention Ingraham, Riesselman, Sander and Marks (2019), who unroll a Monte Carlo simulation as a model for protein folding. They compose a neural energy function with a novel and efficient simulator based on Langevin dynamics to build an end-to-end-differentiable model of atomic protein structure given amino acid sequence information.

4.10. Deep inverse priors

Deep inverse priors (DIPs) generalize deep image priors that were recently introduced in Ulyanov, Vedaldi and Lempitsky (2018) for some image processing tasks. We emphasize that deep inverse priors (DIPs) do not use a learned prior in the sense used so far in this paper. Deep inverse priors (DIPs), as we will see, more closely resemble non-linear Landweber schemes, but with a partly learned likelihood given by a trained neural network.

The choice of network design is crucial: it is assumed to provide a structural prior for the parameters or images to be reconstructed. More precisely, one assumes that the structure of a generator network is sufficient to capture most of the low-level image statistics prior to any learning. During the training of the network, more and more detailed information is added and an appropriate stopping criterion is essential to avoiding overfitting. In particular, a randomly initialized neural network can be used with excellent results in standard inverse problems such as denoising, super-resolution and in-painting.

For linear operators in a finite-dimensional setting, the task is to train a decoder network $\Psi_\theta^\dagger \colon \Xi \to X$ with fixed input $\xi_0 \in \Xi$. The reconstruction operator $\mathcal{R} \colon Y \to X$ is now given as the output of this decoder, that is,

$$\mathcal{R}(g) := \Psi^\dagger_{\widehat{\theta}(g)}(\xi_0), \tag{4.50}$$

where

$$\widehat{\theta}(g) \in \arg\min_\theta \| \mathcal{A} \circ \Psi_\theta^\dagger(\xi_0) - g\|^2.$$

At the core of the DIP approach is the assumption that one can construct a (decoder) network $\Psi_\theta^\dagger \colon \Xi \to X$ which outputs elements in X, which are close to or have a high probability of belonging to the set of feasible parameters.

We emphasize that the training is with respect to θ: the input ξ_0 is kept fixed. Furthermore, machine learning approaches generally use large sets of training data, so it is somewhat surprising that deep image priors are trained on a single data set g. In summary, the main ingredients of deep inverse

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

priors (DIPs) are a single data set, a well-chosen network architecture and a stopping criterion for terminating the training process (Ulyanov *et al.* 2018).

One might assume that the network architecture $\Psi_\theta^\dagger$ would need to incorporate rather specific details about the forward operator $\mathcal{A}$, or even more importantly about the prior distribution Π_{prior} of feasible model parameters. This seems not to be the case: empirical evidence suggests that rather generic network architectures work for different inverse problems, and the obtained numerical results demonstrate the potential of DIP approaches for large-scale inverse problems such as MPI.

So far, investigations related to DIP have been predominantly experimental and mostly restricted to problems that do not fall within the class of ill-posed inverse problems. However, some work has been done in the context of inverse problems: for example, Van Veen *et al.* (2018) consider the DIP approach to solve an inverse problem with a linear forward operator. They introduce a novel learned regularization technique which further reduces the number of measurements required to achieve a given reconstruction error. An approach similar to DIP is considered by Gupta *et al.* (2018) for CT reconstruction. Here one regularizes by projection onto a convex set (see Gupta *et al.* 2018, equation (3)) and the projector is constructed by training a U-Net against unsupervised data. Gupta *et al.* (2018, Theorem 3) also provide guarantees for convergence to a local minimum.

We now briefly summarize the known theoretical foundations of DIP for inverse problems based on the recent paper by Dittmer, Kluth, Maass and Baguer (2018), who analyse and prove that certain network architectures in combination with suitable stopping rules do indeed lead to regularization schemes, which lead to the notion of 'regularization by architecture'. We also include numerical results for the integration operator; more complex results for MPI are presented in Section 7.5.

4.10.1. DIP with a trivial generator

To better understand the regularizing properties of DIP, we begin by considering a trivial generator that simply takes a scalar value of unity on a single node (*i.e.* $\xi_0 = 1$) and outputs an element $h \in X$, that is, $\Psi_\theta^\dagger(\xi_0) = h$ independently of ξ_0. This implies simply that the network is a single layer with the value $\theta = h$, without any bias term or non-linear activation function. Then, the reconstruction operator in (4.50) reads as $\mathcal{R}(g) = \widehat{\theta}$, where

$$\widehat{\theta} \in \arg\min_\theta \| \mathcal{A}(\theta) - g\|^2. \tag{4.51}$$

In this setting θ can be directly identified with an element in X (and $\Xi \equiv X$), so training this network by gradient descent that seeks to minimize the objective in (4.51) is equivalent to the classical Landweber iteration. Despite its obvious trivialization of the neural network approach, this shows that

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

there is potential for training networks with a single data point. Also, Landweber iterations converge from rather arbitrary starting points, indicating that the choice of ξ_0 in the general case is indeed of minor importance.

4.10.2. Analytic deep priors

The idea is based on the perspective that training a DIP network is, for certain network architectures, equivalent to minimizing a Tikhonov functional as in (2.10), that is,

$$\min_{f\in X}\left\{\frac{1}{2}\|\mathcal{B}(f)-g\|^2+\lambda\,\mathcal{S}(f)\right\}. \tag{4.52}$$

This can be seen by considering networks similar to the unrolled schemes (see Section 4.9.2); that is, we consider a fully connected feed-forward network with N layers. We impose the further restriction that (i) the nonlinearity (activation function) is identical to a proximal mapping $\mathrm{prox}_{\lambda\mathcal{S}}$ with respect to a convex functional $\mathcal{S}\colon X\to\mathbb{R}$, (ii) the affine linear mapping between layers allows the decomposition $\mathrm{id}-\mathcal{W}=\lambda\mathcal{B}^*\circ\mathcal{B}$ for some linear operator $\mathcal{B}\colon X\to Y$, and (iii) the bias term is fixed as $b=\lambda\mathcal{B}^*g$.

As described in Section 4.9.2, the output of a network using this architecture is equivalent to the Nth iterate of an ISTA scheme for approximating a minimizer of (4.52). With $\mathcal{W}=\mathrm{id}-\lambda\mathcal{B}^*\circ\mathcal{B}$, the network $\Psi^\dagger_{\mathcal{W}}(\xi_0)$ is given by the unrolled ISTA scheme (Section 4.9.1)

$$\begin{cases} f^0=\xi_0,\\ f^{k+1}=\mathrm{prox}_{\lambda\mathcal{S}}(\mathcal{W}(f^k)+b),\\ \Psi^\dagger_{\mathcal{W}}(\xi_0)=f^N. \end{cases} \tag{4.53}$$

Such ISTA schemes converge as $N\to\infty$ for rather arbitrary starting points; hence, as pointed out above, the particular choice of ξ_0 in (4.50) is indeed of minor importance.

The starting point for a more in-depth mathematical analysis is the assumption that the above unrolled ISTA scheme has fully converged, that is,

$$\Psi^\dagger_{\mathcal{W}}(\xi_0)=\hat{f}:=\arg\min_f \mathcal{J}_{\mathcal{B}}(f). \tag{4.54}$$

Using this characterization of $\hat{f}$, we define the *analytic deep prior* as the network, which is obtained by a gradient descent method with respect to $\mathcal{B}$ for

$$\mathcal{L}(\mathcal{B},g)=\frac{1}{2}\|\mathcal{A}(\hat{f})-g\|^2. \tag{4.55}$$

The resulting deep prior network has $\mathrm{prox}_{\lambda\mathcal{S}}$ as activation function, and the linear map $\mathcal{W}$ and its bias b are as described in (ii) and (iii). This allows us to obtain an explicit description of the gradient descent for $\mathcal{B}$, which in turn leads to an iteration of functionals $\mathcal{J}_{\mathcal{B}}$.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Below, we provide this derivation for a toy example that nevertheless highlights the differences between a classical Tikhonov minimizer and the solution of the DIP approach.

Simple example. We here examine analytic deep priors for linear inverse problems, *i.e.* $\mathcal{A}\colon X \to Y$ is linear, and compare them to classical Tikhonov regularization with $\mathcal{S}(f) = \frac{1}{2}\|f\|^2$. Let $f_\lambda \in X$ denote the solution obtained with the classical Tikhonov regularization, which by (2.11) can be expressed as

$$f_\lambda = (\mathcal{A}^* \circ \mathcal{A} + \lambda \,\mathrm{id})^{-1} \circ \mathcal{A}^*(g).$$

This is equivalent to the solution obtained by the analytic deep prior approach, with $\mathcal{B} = \mathcal{A}$ without any iteration. Now, take $\mathcal{B} = \mathcal{A}$ as a starting point for computing a gradient descent with respect to $\mathcal{B}$ using the DIP approach, and compare the resulting $\hat{f}$ with f_λ.

The proximal mapping for the functional R above is given by

$$\mathrm{prox}_{\lambda \mathcal{S}}(z) = \frac{1}{1+\lambda} z.$$

A rather lengthy calculation (see Dittmer, Kluth, Maass and Baguer 2018) yields an explicit formula for the derivative of F with respect to $\mathcal{B}$ in the iteration

$$\mathcal{B}^{k+1} = \mathcal{B}^k - \eta \partial F(\mathcal{B}^k).$$

The expression stated there can be made explicit for special settings. For illustration we assume the rather unrealistic case that $f^+ = h$, where $h \in X$ is a singular function for $\mathcal{A}$ with singular value σ. The dual singular function is denoted by $v \in Y$, *i.e.* $\mathcal{A}\, h = \sigma v$ and $\mathcal{A}^*\, v = \sigma h$, and we further assume that the measurement noise in g is in the direction of this singular function, *i.e.* $g = (\sigma + \delta)v$. In this case, the problem is indeed restricted to the span of h and the span of v, respectively. The iterates $\mathcal{B}^k$ only change the singular value β_k of h, that is,

$$\mathcal{B}^{k+1} = \mathcal{B}^k - c_k \langle \,\cdot\, , h \rangle v,$$

with a suitable $c_k = c(\lambda, \delta, \sigma, \eta)$.

Deep inverse priors for the integration operator. We now illustrate the use of deep inverse prior approaches for solving an inverse problem with the integration operator $\mathcal{A}\colon L^2([0,1]) \to L^2([0,1])$, defined by

$$\mathcal{A}(f)(t) = \int_0^t f(s)\,\mathrm{d}s. \tag{4.56}$$

Here $\mathcal{A}$ is linear and compact, hence the task of evaluating its inverse is an ill-posed inverse problem.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

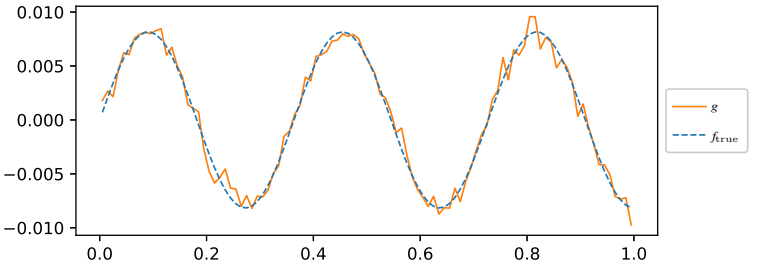
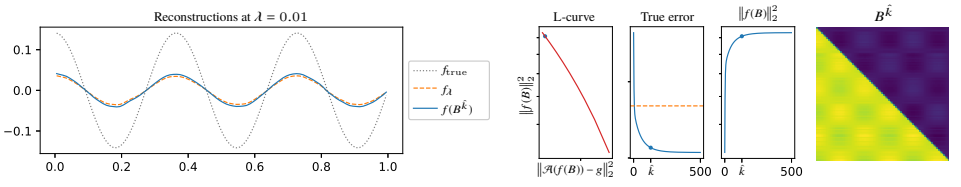
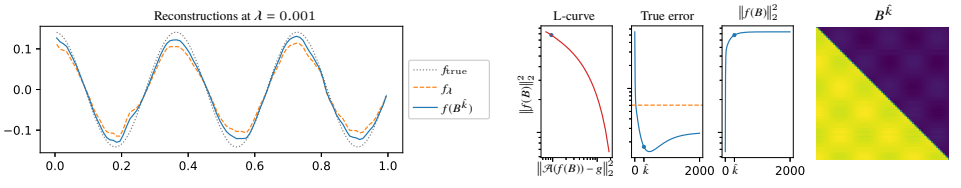


Figure 4.2. Example of g for $f_{\text{true}} = u_5$ and 10% of noise.



(a) $\lambda = 0.01$



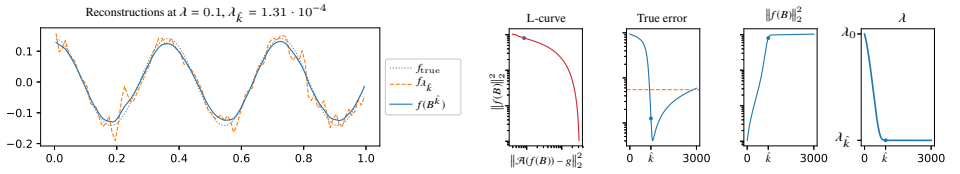
(b) $\lambda = 0.001$

Figure 4.3. Comparison of Tikhonov reconstructions and results obtained with DIP. Reconstructions are shown for different fixed values of λ. The network was trained with the standard gradient descent method and a learning rate of 0.05. In (a) 500 epochs were used whereas in (b) 2000 were used.

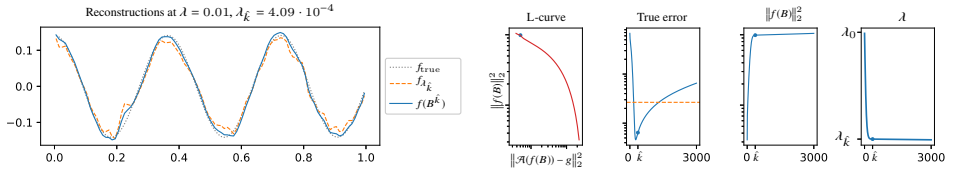
Discretizing this operator with $n = 100$ yields a matrix $\mathbf{A}_n \in \mathbb{R}^{n\times n}$, which has $h/2$ on the main diagonal, h everywhere under the main diagonal and 0 above (here $h = 1/n$). We choose f_{true} to be one of the singular vectors of $\mathbf{A}$ and determine noisy data $g = \mathbf{A}_n f_{\text{true}} + e$ with $e \sim \mathcal{N}(0, \sigma^2)$, where σ^2 is chosen as 10% of the largest coefficient of g; see Figure 4.2.

Figure 4.3 shows some reconstruction results with $N = 10$ layers. The first plot contains the true solution f_{true}, the standard Tikhonov solution f_λ, and the reconstruction obtained with the analytic deep inverse approach $f(\mathbf{B}_{\text{opt}})$ after 2000 iterations for updating $\mathbf{B}$. For both choices of λ the training of $\mathbf{B}$ converges to a matrix $\mathbf{B}_{\text{opt}}$, such that $f(\mathbf{B}_{\text{opt}})$ has a smaller true error than f_λ. As can be observed in the last plot, the resulting matrix $\mathbf{B}_{\text{opt}}$ contains some patterns that reflect what was predicted by the analytic deep prior.

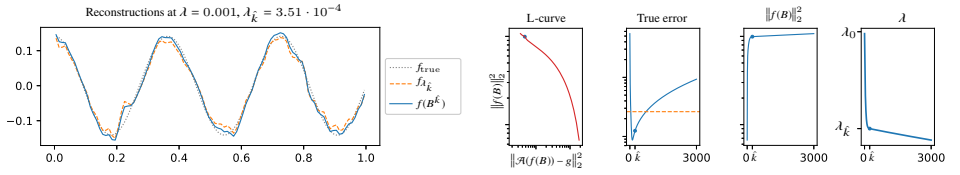
https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press



(a) $\lambda = 0.1$, $\lambda_{\hat{k}} = 1.31 \times 10^{-4}$



(b) $\lambda = 0.01$, $\lambda_{\hat{k}} = 4.09 \times 10^{-4}$



(c) $\lambda = 0.001$, $\lambda_{\hat{k}} = 3.51 \times 10^{-4}$

Figure 4.4. Reconstructions with an adaptive λ for different starting values λ_0. The networks were trained with gradient descent using 0.1 as learning rate. In all cases 3000 epochs were used.

So far, the regularization parameter λ has been assumed to be fixed. In a real application one needs to choose it via a technique such as the L-curve (Hansen 1992) or the discrepancy principle (4.1) in Section 4.1. However, they usually involve finding reconstructions for many different values of λ. In our case, that would mean retraining the network each time, which would lead to a really high computational cost. This motivates an adaptive choice of λ during the training, which could be achieved by letting λ also be a trainable weight of the network. The results for the same example and different starting values λ_0 are shown in Figure 4.4.

5. Learning in statistical regularization

The focus here is on various approaches for combining techniques from deep learning with Bayesian inversion and we begin by recapitulating the statistical setting in Section 3.1.2.

To recapitulate, we assume there exists a $(X \times Y)$-valued random variable $(\mathbb{f}, \mathbb{g}) \sim \mu$ that generates model parameters with associated data. The aim

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

is to compute various estimators from the posterior Π^g_{post} where $g \in Y$ is single sample of $(\mathbb{g} \mid \mathbb{f} = f_{\text{true}})$ with $f_{\text{true}} \in X$ unknown. The data likelihood Π^f_{data}, which is the distribution of $(\mathbb{g} \mid \mathbb{f} = f)$, is here known for any $f \in X$ via (3.3), that is,

$$\mathbb{g} = \mathcal{A}(\mathbb{f}) + \mathbb{e},$$

with $\mathbb{e} \sim \Pi_{\text{noise}}$ independent of $\mathbb{f}$.

Ideally one would like to recover the entire posterior distribution $f \mapsto \Pi^g_{\text{post}}$ for the measured data g. However, this is very challenging (Section 3.5), so many approaches settle for computing a selected estimator (Section 5.1). Alternatively, one may use deep neural nets to sample from the posterior, as surveyed in Section 5.2.

5.1. Learning an estimator

As outlined in Section 3.3, any reconstruction operator that can be represented by a deterministic measurable map $\mathcal{R}\colon Y \to X$ formally corresponds to a point estimator (also called a non-randomized decision rule). One can now use techniques from deep learning in computing such estimators.

5.1.1. Overview

There are various ways of combining techniques from deep learning with statistical regularization. The statistical characteristics of training data together with the choice of loss function determines the training problem one seeks to solve during learning. This in turn determines the type of estimator (reconstruction operator) one is approximating.

Supervised learning. The training data are given as samples $(f_i, g_i) \in X \times Y$ generated by $(\mathbb{f}, \mathbb{g}) \sim \mu$. One can then approximate the Bayes estimator, that is, we seek $\mathcal{R}_{\widehat{\theta}}\colon Y \to X$, where $\widehat{\theta}$ solves

$$\widehat{\theta} \in \arg\min_{\theta} \mathbb{E}_{(\mathbb{f},\mathbb{g})\sim\mu}[\ell_X(\mathcal{R}_\theta(\mathbb{g}), \mathbb{f})]. \tag{5.1}$$

The actual training involves replacing the joint law μ with its empirical counterpart induced by the supervised training data. Examples of methods that build on the above are surveyed in Section 5.1.2.

Learned prior. The training data $f_i \in X$ are samples generated by a $\mu_{\mathbb{f}}$-distributed random variable, where $\mu_{\mathbb{f}} \in \mathscr{P}_X$ is the $\mathbb{f}$-marginal of μ. One can then learn the negative log prior density in a MAP estimator, that is, $\mathcal{R}_{\widehat{\theta}}\colon Y \to X$ is given by

$$\mathcal{R}_{\widehat{\theta}}(g) := \arg\min_{f\in X}\{-\log \pi_{\text{data}}(g \mid f) + \mathcal{S}_{\widehat{\theta}}(f)\}.$$

Here $\pi_{\text{data}}(\,\cdot \mid f)$ is the density for the data likelihood $\Pi^f_{\text{data}} \in \mathscr{P}_Y$ and $\widehat{\theta}$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

is learned such that $\mathcal{S}_{\widehat{\theta}}(f) \approx -\log(\pi_{\mathbb{f}}(f))$, with $\pi_{\mathbb{f}}$ denoting the density for $\mu_{\mathbb{f}} \in \mathscr{P}_X$, which is the $\mathbb{f}$-marginal of μ. The actual training involves replacing $\mu_{\mathbb{f}}$ with its empirical counterpart induced by the training data. Examples of methods that build on the above are surveyed in Section 4.7.

Unsupervised learning. The training data $\mathbb{g}_i \in Y$ are samples generated by a $\mu_{\mathbb{g}}$-distributed random variable where $\mu_{\mathbb{g}} \in \mathscr{P}_Y$ is the $\mathbb{g}$-marginal of μ. It is not possible to learn a prior in a MAP estimator from such training data, but one can improve upon the computational feasibility for evaluating a given MAP estimator. We do that by considering $\mathcal{R}_{\widehat{\theta}}\colon Y \to X$, where $\widehat{\theta}$ solves

$$\widehat{\theta} := \arg\min_{\theta} \mathbb{E}_{\mathbb{g}\sim\mu_{\mathbb{g}}}[-\log \pi_{\text{data}}(\mathbb{g} \mid \mathcal{R}_\theta(\mathbb{g})) + \mathcal{S}_\lambda(\mathcal{R}_\theta(\mathbb{g}))]. \tag{5.2}$$

In the above, both the density $\pi_{\text{data}}(\,\cdot \mid f)$ for the data likelihood $\Pi^f_{\text{data}} \in \mathscr{P}_Y$ and the negative log density $\mathcal{S}_\lambda\colon X \to \mathbb{R}$ of the prior are handcrafted. The actual training involves replacing $\mu_{\mathbb{g}} \in \mathscr{P}_Y$ with its empirical counterpart induced by the training data. In the above, $\widehat{\mu}_{\mathbb{g}}$ is the empirical counterpart of $\widehat{\mu}_{\mathbb{g}}$ given by training data, $\mathcal{L}\colon Y \times Y \to \mathbb{R}$ is the negative data log-likelihood, and $\mathcal{S}_\lambda\colon X \to \mathbb{R}$ is the negative log-prior. The latter two are not learned. Examples of methods that build on the above are surveyed in Section 4.9.

Semi-supervised learning. The training data $\mathbb{f}_i \in X$ and $\mathbb{g}_i \in Y$ are semi-supervised, *i.e.* unpaired samples from the marginal distributions $\mu_{\mathbb{f}}$ and $\mu_{\mathbb{g}}$ of μ, respectively. One can then compute an estimator $\mathcal{R}_{\widehat{\theta}}\colon Y \to X$, where $\widehat{\theta}$ solves

$$\begin{aligned}\widehat{\theta} \in \arg\min_{\theta}\{&\mathbb{E}_{(\mathbb{f},\mathbb{g})\sim\mu_{\mathbb{f}}\otimes\mu_{\mathbb{g}}}[\ell_Y(\mathcal{A}(\mathcal{R}_\theta(\mathbb{g})),\mathbb{g}) + \ell_X(\mathcal{R}_\theta(\mathbb{g}),\mathbb{f})] \\ &+ \lambda \ell_{\mathscr{P}_X}((\mathcal{R}_\theta)_{\#}(\mu_{\mathbb{g}}),\mu_{\mathbb{f}})\}.\end{aligned} \tag{5.3}$$

In the above, $\ell_X\colon X \times X \to \mathbb{R}$ and $\ell_Y\colon Y \times Y \to \mathbb{R}$ are loss functions on X and Y, respectively. Next, $\ell_{\mathscr{P}_X}\colon \mathscr{P}_X \times \mathscr{P}_X \to \mathbb{R}$ is a distance notion between probability distributions on X and $(\mathcal{R}_\theta)_{\#}(\mu_{\mathbb{g}}) \in \mathscr{P}_X$ denotes the pushforward of the measure $\mu_{\mathbb{g}} \in \mathscr{P}_Y$ by $\mathcal{R}_\theta\colon Y \to X$. It is common to evaluate $\ell_{\mathscr{P}_X}$ using techniques from GANs, which introduce a separate deep neural network (discriminator/critic). Finally, the parameter λ controls the balance between the distributional consistency, noise suppression and data consistency. One can also consider further variants of the above, for example when there is access to a large sample of unpaired data combined with a small amount of paired data, or when parts of the probability distributions involved are known.

The choice of neural network architecture for the reconstruction operator $\mathcal{R}_{\widehat{\theta}}\colon Y \to X$ is formally independent of the choice of loss function and the

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

set-up of the training problem. The choice does, however, impact the trainability of the learning, especially when there is little training data. In such cases, it is important to make use of all the information. In inverse problems one has explicit knowledge about how data are generated that comes in the form of a forward operator, or one might have an expression for the entire data likelihood. Architectures that embed such explicit knowledge, *e.g.* the forward operator and the adjoint of its derivative, perform better when there is little training data. They also have better generalization properties, and against adversarial attacks (Chakraborty *et al.* 2018, Akhtar and Mian 2018) they are more difficult to design since a successful attack needs to be consistent with how data are generated. Architectures that account for such information can be defined by unrolling (Section 4.9.4).

The remaining sections survey various approaches from the literature in computing the above estimators.

5.1.2. Deep direct Bayes estimation

The aim is to compute a Bayes estimator, which by the definition given in (3.14) amounts to finding a reconstruction operator $\mathcal{R}_\mu : Y \to X$ that solves

$$\mathcal{R}_\mu \in \operatorname*{arg\,min}_{\mathcal{R}:\, Y \to X} \mathbb{E}_{(\mathbb{f},\mathbb{g})\sim\mu}[\ell_X(\mathbb{f}, \mathcal{R}(\mathbb{g}))], \tag{5.4}$$

where $\ell_X : X \times X \to \mathbb{R}$ is a fixed loss function. The data likelihood is often known, and by the law of total probability we have

$$\mu(f, g) = \Pi_{\text{prior}}(f) \otimes \Pi^f_{\text{data}}, \tag{5.5}$$

so the joint law μ is known as soon as a prior has been selected.

As already mentioned (Section 3.5.1), selecting an appropriate prior that reflects the probability distribution of natural model parameters is very challenging, and current hand-crafted choices (Section 3.4) do not capture the full extent of the available *a priori* information about the true unknown model parameter $f_{\text{true}} \in X$. Next, the expression in (5.4) involves taking an expectation over $X \times Y$ as well as an optimization over all possible non-randomized decision rules. Both these operations easily become computationally overwhelming in large-scale inverse problem, such as those that arise in imaging.

These issues can be addressed by using techniques from supervised training. To start with, one can restrict the minimization in (5.4) to a family of reconstruction methods parametrized by a deep neural network architecture $\mathcal{R}_\theta : Y \to X$. Next, the unknown joint law μ can be replaced with its empirical counterpart given by supervised training data

$$\Sigma_m := \{(f_1, g_1), \ldots, (f_m, g_m)\} \subset X \times Y, \tag{5.6}$$

where (f_i, g_i) are generated by $(\mathbb{f}, \mathbb{g}) \sim \mu$. If there is a sufficient amount of

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

such training data, then one can approximate the Bayes estimator in (5.4) by the neural network $\mathcal{R}_{\widehat{\theta}}\colon Y \to X$, where the finite-dimensional network parameter $\widehat{\theta} \in \Theta$ is learned from data by solving the following empirical risk minimization problem:

$$\widehat{\theta} \in \arg\min_{\theta\in\Theta} \frac{1}{m}\sum_{i=1}^{m} \ell_X(f_i, \mathcal{R}_\theta(g_i)), \tag{5.7}$$

where $(f_i, g_i) \in \Sigma_m$ as in (5.6). Note now that (5.7) does not explicitly require specifying a prior $f \mapsto \Pi_{\text{prior}}(f)$ or a data likelihood $g \mapsto \Pi^f_{\text{data}}$ that models how data are generated given a model parameter. Information about both of these is implicitly contained in supervised training data $\Sigma_m \subset X \times Y$.

Fully learned Bayes estimation (Section 5.1.3) refers to approaches where one assumes there is enough supervised training data to learn the joint law μ, that is, one disregards the explicit knowledge about the data likelihood. In contrast, *learned iterative schemes* (Section 5.1.4) include the information about the data likelihood by using an appropriate architecture of $\mathcal{R}_\theta\colon Y \to X$. *Learned post-processing* methods (Section 5.1.5) offer an alternative way to account for the data likelihood since these methods apply an initial reconstruction operator that maps data to a model parameter. This is actually an estimator different from the above Bayes estimator, but if the loss is the squared L^2-norm and the initial reconstruction operator is a linear sufficient statistic, then these estimators coincide in the 'large-sample' or 'small-noise' limit.

Regularizing the learning. The problem in (5.7) is ill-posed in itself, so one should not try to solve it in the formal sense. A wide range of techniques have been developed within supervised learning for implicitly or explicitly regularizing the empirical risk minimization problem in (5.7) as surveyed and categorized by Kukačka, Golkov and Cremers (2017). A key challenge is to handle the non-convexity, and the energy landscape for the objective in (5.7) typically has many local minima: for example, for binary classification there is an exponential number (in terms of network parameters) of distinct local minima (Auer, Herbster and Warmuth 1996).

Similar to Shai and Shai (2014, Section 2.1), we define a *training algorithm* for (5.7) as an operator mapping a probability measure on $X \times Y$ to a parameter in Θ that approximately solves (5.7):

$$\mathcal{T}\colon \mathscr{P}_{X\times Y} \to \Theta, \tag{5.8}$$

where $\mathcal{T}(\widehat{\mu}) \approx \widehat{\theta}$ with $\widehat{\theta} \in \Theta$ denoting a solution to (5.7). Thus, the training algorithm is a method for approximately solving (5.7) given a fixed neural network architecture. This also includes necessary regularization techniques: for example, a common strategy for solving (5.7) is to use some variant of stochastic gradient descent that is cleverly initialized (often at

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

random from a specific distribution) along with techniques to ensure that the value of the objective (training error) decreases sufficiently rapidly. This, combined with early stopping (*i.e.* not fully solving (5.7)), warm-start, use of mini-batches, adding a regularization term to the objective, and so on, acts as a regularization (Ruder 2016, Kukačka *et al.* 2017, Bottou, Curtis and Nocedal 2018).

Concerning the model architecture, there is strong empirical evidence that the choice of neural network architecture has an influence on the ill-posedness of (5.7) (Li, Xu, Taylor and Goldstein 2018*c*, Draxler, Veschgini, Salmhofer and Hamprecht 2018). Many tasks that are successfully solved by supervised learning rely on (deep) neural networks, which can approximate a wide range of non-linear phenomena (large model capacity) without impairing computability. For this reason we consider deep neural networks to parametrize $\mathcal{R}_\theta \colon Y \to X$. Furthermore, empirical evidence indicates that deep neural network architectures yield a more favourable energy landscape for the objective of (5.7) than shallow ones: for example, most local minima are almost global (Choromanska *et al.* 2015, Becker, Zhang and Lee 2018). This intricate interplay between the choice of architecture and avoiding getting trapped in 'bad' local minima is poorly understood, and it is an active area of research within the machine learning community. Despite the lack of a theory, there is a consensus that an appropriate model architecture not only ensures computational feasibility but also acts as a kind of implicit regularization for (5.7).

To summarize, a training algorithm $\mathcal{T}$ as in (5.8) together with a specific model architecture regularizes (5.7), thereby resulting in the following approximation to the Bayes estimator (5.4):

$$\mathcal{R}_{\mathcal{T}(\widehat{\mu})} \colon Y \to X \tag{5.9}$$

for the empirical measure $\widehat{\mu}$ given by $\Sigma_m \subset X \times Y$ as in (5.6).

5.1.3. Fully learned Bayes estimation

Here the reconstruction operator $\mathcal{R}_\theta \colon Y \to X$ has a generic parametrization given by a deep neural network that does not explicitly account for the data likelihood.

An obvious difficulty with this approach is that the data space Y and model parameter space X are mathematically different. Without an explicit mapping from Y to X the action of convolutional layer operators cannot be properly defined. Therefore, fully learned approaches usually involve one or more 'fully connected layers' that represent a pseudo-inverse operator $\mathcal{B}_{\theta_1} \colon Y \to X$ mapping elements in Y to elements in X followed by a conventional neural network for $\mathcal{F}_{\theta_2} \colon X \to X$, that is, we get

$$\mathcal{R}_\theta := \mathcal{F}_{\theta_2} \circ \mathcal{B}_{\theta_1} \quad \text{with } \theta = (\theta_1, \theta_2). \tag{5.10}$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

In the discrete setting, where $Y = \mathbb{R}^n$ and $X = \mathbb{R}^m$, the simplest representation for $\mathcal{B}$ is a dense matrix $\mathbf{B} \in \mathbb{R}^{m\times n}$; the inclusion of an activation function makes this a non-linear mapping.

Initial examples of fully learned reconstruction in tomographic imaging include Paschalis *et al.* (2004) for SPECT imaging and Argyrou, Maintas, Tsoumpas and Stiliaris (2012) for transmission tomography. Both papers consider small-scale problems: for example, Paschalis *et al.* (2004) consider recovering 27×27 pixel SPECT images, and Argyrou *et al.* (2012) consider recovering 64×64 pixel images from tomographic data.

A more recent example is the automated transform by manifold approximation (AutoMap) method introduced in Zhu *et al.* (2018) as a tool for fully data-driven image reconstruction. Here, $\mathcal{R}_\theta \colon Y \to X$ is represented by a feed-forward deep neural network with fully connected layers followed by a sparse convolutional auto-encoder. This is in some sense similar to the Deep Cascade architecture in Schlemper *et al.* (2017), as it has one portion of the network for data consistency and the other for super-resolution/refinement of image quality. The encoder from data space to the model parameter space is implemented using three consecutive fully connected networks with sinh activation functions followed by two CNN layers with ReLU activation. We interpret this as a combination of a pseudo-inverse $\mathcal{R}^\dagger_{\theta_3} : Y \to X$ with a conventional convolutional auto-encoder:

$$\mathcal{R}_\theta = \underbrace{\Psi^\dagger_{\theta_1} \circ \Psi_{\theta_2}}_{\text{auto-encoder}} \circ \mathcal{R}^\dagger_{\theta_3} \quad \text{for } \theta = (\theta_1, \theta_2.\theta_3).$$

AutoMap was used to reconstruct 128×128 pixel images from MRI and PET imaging data. The dependence on fully connected layers results in a large number of neural network parameters that have to be trained. Primarily motivated by this difficulty, a further development of AutoMap is ETER-net (Oh *et al.* 2018), which uses a recurrent neural network architecture in place of the fully connected/convolutional auto-encoder architecture. Also addressing 128×128 pixel images from MRI, Oh *et al.* (2018) found a reduction in required parameters by over 80%. A method similar to AutoMap is used by Yoo *et al.* (2017) to solve the non-linear reconstruction problem in diffuse optical tomography. Here the forward problem is the Lippman–Schwinger equation but only a single fully connected layer is used in the backprojection step. Yoo *et al.* (2017) exploit the intrinsically ill-posed nature of the forward problem to argue that the mapping induced by the auto-encoder step is low-rank and therefore sets an upper bound on the dimension of the hidden convolution layers.

The advantage of fully learned Bayes estimation lies in its simplicity, since one avoids making use of an explicit forward operator (or data likelihood). On the other hand, any generic approach to reconstruction by deep

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

neural networks requires having connected layers that represent the relation between model parameters and data. For this reason, generic fully learned Bayes estimation will always scale badly: for example, in three-dimensional tomographic reconstruction it is common to have an inverse problem which, when discretized, involves recovering a $(512 \times 512 \times 512 \approx 10^8)$-dimensional model parameter from data of the same order of magnitude. Hence, a fully learned generic approach would involve learning at least 10^{16} weights from supervised data! There have been several attempts to address the above issue by considering neural network architectures that are adapted to specific direct and inverse problems. One example is that of Khoo and Ying (2018), who provide a novel neural network architecture (SwitchNet) for solving inverse scattering problems involving the wave equation. By leveraging the inherent low-rank structure of the scattering problems and introducing a novel switching layer with sparse connections, the SwitchNet architecture uses far fewer parameters than a U-Net architecture for such problems. Another example is that of Ardizzone *et al.* (2018), who propose encoding the forward operator using a invertible neural network, also called a reversible residual network (Gomez, Ren, Urtasun and Grosse 2017). The reconstruction operator is then obtained as the inverse of the invertible neural network for the forward operator. However, it is unclear whether this is a clever approach to problems that are ill-posed, since an inverse of the forward operator is not stable. Another approach is that of Yoo *et al.* (2017), who apply an AutoMap-like architecture for non-linear reconstruction problems in diffuse optical tomography. Here the forward problem is the Lippman–Schwinger equation, but only a single fully connected layer is used in the backprojection step. Yoo *et al.* (2017) exploit the intrinsically ill-posed nature of the forward problem to argue that the mapping induced by the auto-encoder step is low-rank and therefore sets an upper bound on the dimension of the hidden convolution layers. The above approaches can also to some extent be seen as further refinements of methods in Section 4.2.

However, neither of the above efforts address the challenge of finding sufficient supervised training data necessary for the training. Furthermore, any changes to the acquisition protocol or instrumentation may require retraining, making the method impractical. In particular, due to the lack of training data, fully learned Bayes estimation is inapplicable to cases when data are acquired using novel instrumentation. A practical case would be spectral CT, where novel direct counting energy resolving detectors are being developed.

5.1.4. Learned iterative schemes

The idea here is to choose an architecture for $\mathcal{R}_\theta \colon Y \to X$ in (5.7) that contains an explicit expression for the data likelihood, which accounts for how a model parameter gives rise to data. This requires us to embed an

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

explicit forward operator $\mathcal{A}\colon X \to Y$ into the architecture for $\mathcal{R}_\theta$, which is somewhat tricky since $\mathcal{R}_\theta$ and $\mathcal{A}$ are mappings that go in the reverse direction compared to each other.

One approach is presented by Mousavi and Baraniuk (2017), who suggest a CNN architecture (DeepInverse) adapted for solving a linear inverse problem. The architecture involves a fully connected layer (that is not learned) to represent the normal operator $\mathcal{A}^* \circ \mathcal{A}\colon X \to X$ followed by convolutional layers as in a regular CNN with ReLU activation, but here one dispenses with the downsampling (max-pooling operation) that is common in a CNN. The usefulness is limited, however, since the normal operator needs to have a certain structure for the sake of computational efficiency, for example when inverting the Fourier transform, which results in a block-circulant matrix.

Another class of methods is that of *learned iterative schemes*, which include a handcrafted forward operator and the adjoint of its derivative into the architecture by unrolling a suitable iterative scheme (Section 4.9.4). An early variant was presented by Yang, Sun, Li and Xu (2016), who define a learned iterative method based on unrolling an ADMM-type scheme. The network is trained against supervised data using a somewhat unusual asymmetric loss, namely

$$\ell_X(f,h) := \sqrt{\|f-h\|_2^2/\|f\|_2^2}.$$

The trained network is used to invert the Fourier transform (MRI image reconstruction). However, the whole approach is unnecessarily complex, and it is now surpassed by learned iterative methods that have a more transparent logic. The survey will therefore focus on these latter variants.

Learned iterative in model parameter space. In the simplest setting, $\mathcal{R}_\theta$ in (5.7) is given as in (4.32) with an updating operator as in (4.34) where $\mathcal{J} := \mathcal{L}(\mathcal{A}(\,\cdot\,), g)$. Hence, given an initial model parameter $f^0 \in X$, we define $\mathcal{R}_\theta\colon Y \to X$ with $\theta = (\theta_1, \dots, \theta_N)$ as

$$\mathcal{R}_\theta(g) := (\Lambda_{\theta_N} \circ \cdots \circ \Lambda_{\theta_1})(f^0), \tag{5.11}$$

where $\Lambda_{\theta_k} := \mathrm{id} + \Gamma_{\theta_k} \circ \nabla\mathcal{L}(\mathcal{A}(\,\cdot\,), g)$. In the above, $\Gamma_{\theta_k}\colon X \to X$ is learned from supervised data (5.6) by approximately solving (5.7) using a training algorithm as in (5.8). In contrast, $\nabla\mathcal{L}(\mathcal{A}(\,\cdot\,), g)\colon X \to X$ is not learned: it is derived from an explicit expression for the data likelihood. For example, a common choice for data where the observational noise is Gaussian is

$$\mathcal{L}(v,g) := \frac{1}{2}\|v-g\|_2^2 \implies \nabla\mathcal{L}(\mathcal{A}(f),g) = [\partial\mathcal{A}(f)]^*(\mathcal{A}(f)-g). \tag{5.12}$$

The reconstruction operator $\mathcal{R}_\theta\colon Y \to X$ in (5.11) can now be interpreted as a *residual neural network*, as popularized by He, Zhang, Ren and Sun (2016) for image classification. Furthermore, the operators $\Gamma_{\theta_k}\colon X \to X$ for

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

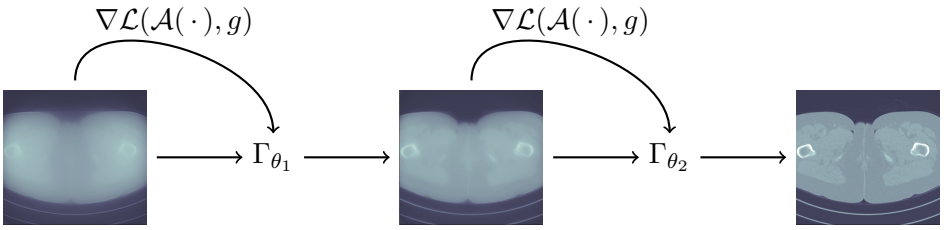


Figure 5.1. Learned iterative method in model parameter space. Illustration of the unrolled scheme in (5.11) for $N = 2$ in the context of CT image reconstruction (Section 7.3.1). Each $\Gamma_{\theta_1} : X \to X$ is a CNN, $g \in X$ is the measured data, and f^0 is an initial image, usually taken as zero.

$k = 1, \ldots, N$ are represented by deep neural networks with an architecture that is usually fixed: for example, in imaging problems one selects a suitable CNN architecture. See Figure 5.1 for an illustration of (5.11).

Following Section 4.9.1, the next level of complexity comes when the learned component is allowed to be less constrained by removing the explicit expression for $f \mapsto \nabla\mathcal{L}(\mathcal{A}(f), g)$ in (5.11) while keeping expressions for the forward operator and the adjoint of its derivative. This corresponds to defining $\mathcal{R}_\theta : Y \to X$ as in (4.38) but with no memory, that is,

$$\mathcal{R}_\theta(g) := (\mathcal{P}_X \circ \Lambda_{\theta_N} \circ \ldots \circ \Lambda_{\theta_1})(f^0, g), \tag{5.13}$$

where $\mathcal{P}_X : X \times Y \to X$ is the usual projection and $\Lambda_{\theta_k} : X \times Y \to X \times Y$ for $k = 1, \ldots, N$ is

$$\Lambda_{\theta_k}(f, v) := \Gamma_{\theta_k}(f, v, [\partial\mathcal{A}(f)]^*(v), \mathcal{A}(f), \nabla\mathcal{S}_\lambda(f)) \quad \text{for } (f, v) \in X \times Y.$$

Here, $\Gamma_{\theta_k} : X \times Y \times X \times Y \times X \to X \times Y$ is the updating operator that is given by a deep neural network. The resulting deep neural network $\mathcal{R}_\theta : Y \to X$ is learned from supervised data (5.6) by approximately solving (5.7) using a training algorithm as in (5.8).

The constraints on the learning can be further decreased, at the expense of increased memory footprint and computational complexity, by allowing for some memory $l > 0$, that is, each of the learned updating operators account for more than the previous iterate. This leads to an architecture for $\mathcal{R}_\theta : Y \to X$ of the form (4.38) with updating operators as in (4.37). A special case of this formulation is the learned gradient method of Adler and Öktem (2017), who in turn present the recurrent inference machines of Putzky and Welling (2017) as special case.

Another special case is variational networks. These are defined by unrolling an iterative scheme for minimizing an explicit objective that has a data discrepancy component and a regularizer. The idea was introduced by Hammernik *et al.* (2016) for two-dimensional Fourier inversion, where

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

the objective has a regularizer based on a reaction–diffusion model. Hammernik *et al.* (2018) develop it further: they consider a variational network (outlined in their Figure 1) obtained by unrolling the iterations of their equation (6). This corresponds to (5.11) with $\Lambda_{\theta_k} : X \to X$ given as

$$\Lambda_{\theta_k} := \mathrm{id} + \omega_k \nabla[\mathcal{L}(\mathcal{A}(\,\cdot\,), g) + \mathcal{S}_{\phi_k}(\,\cdot\,)] \quad \text{for } \theta_k = (\omega_k, \phi_k).$$

The regularizer $\mathcal{S}_{\phi_k} : X \to \mathbb{R}$ is chosen as the FoE model (see also Section 4.3.2):

$$\mathcal{S}_{\phi_k}(f) := \sum_j \Phi_{k,j}(f * K_{k,j}) \quad \text{for } \phi_k = (\Phi_{k,j}, K_{k,j}),$$

where the (potential) functionals $\Phi_{k,j} : X \to \mathbb{R}$ and the convolution kernels $K_{k,j} \in X$ are all parametrized by finite-dimensional parameters, so ϕ_k is a finite-dimensional parameter. See also Chen *et al.* (2019), who essentially apply the approach of Hammernik *et al.* (2018) to CT reconstruction. Another variant of variational networks is that of Bostan, Kamilov and Waller (2018), who unroll a proximal algorithm for an objective with a TV regularizer and replace the scalar soft-thresholding function with a parametrized variant (see Bostan, Kamilov and Waller 2018, equation (8)). This yields a proximal algorithm that uses a sequence of adjustable shrinkage functions in addition to self-tuning the step-size. In particular, unlike Mousavi and Baraniuk (2017), the method presented here does not rely on having a structured normal operator. A further variant of a variational network is given by Aggarwal, Mani and Jacob (2019), who unroll a gradient descent scheme for minimizing an objective whose regularizer is given by a CNN (see Aggarwal, Mani and Jacob 2019, equation (7)). A similar approach is also considered by Zhao, Zhang, Wang and Gao (2018*a*), who unroll an ADMM scheme and stop iterates according to a Morozov-type stopping criterion (not a fixed number of iterates), so the number of layers depends on the noise level in data.

Remark 5.1. An interesting aspect of the variational network of Hammernik *et al.* (2016) is that Λ_{θ_k} can be interpreted as a gradient descent step in an optimization scheme that minimizes an objective functional $f \mapsto \mathcal{L}(\mathcal{A}(f), g) + \mathcal{S}_{\phi_k}(f)$. In particular, if ϕ_k is the same for all k, then increasing the number of layers by $N \to \infty$ will in the limit yield a MAP estimator instead of a Bayes estimator.

Other applications of the learned gradient method include those of Gong *et al.* (2018) for deconvolution, Qin *et al.* (2019) for dynamic cardiac two-dimensional MRI (here one needs to exploit the temporal dependence), Hauptmann *et al.* (2018) for three-dimensional PAT, and Wu, Kim and Li (2018*a*) for three-dimensional CT reconstruction. The challenge in three-dimensional applications is to manage the high computational and memory

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

cost for the training of the unrolled network. One approach is to replace the end-to-end training of the entire neural network and instead break down each iteration and train the sub-networks sequentially (gradient boosting). This is the approach taken by Hauptmann *et al.* (2018) and Wu *et al.* (2018*a*), but as shown by Wu *et al.* (2018*a*), the output quality has minor improvements over learned post-processing (Section 5.1.5), which also scales to the three-dimensional setting. A better alternative could be to use a reversible residual network architecture (Gomez *et al.* 2017, Ardizzone *et al.* 2018) for a learned iterative method, since these are much better at managing memory consumption in training (mainly when calculating gradients using backpropagation) as networks grow deeper and wider. However, this is yet to be done.

Learned iterative in both model parameter and data spaces. The final enhancement to the learned iterative schemes is to introduce an explicit learned updating in the data space as well. To see how this can be achieved, one can unroll a primal–dual-type scheme of the form

$$\begin{cases} v^0 = g \text{ and } f^0 \in X \text{ given} \\ v^{k+1} = \Gamma^{\mathrm{d}}_{\theta^{\mathrm{d}}_k}(v^k, \mathcal{A}(f^k), g)) \\ f^{k+1} = \Gamma^{\mathrm{m}}_{\theta^{\mathrm{m}}_k}(f^k, [\partial\,\mathcal{A}(f)]^*(v^{k+1})) \end{cases} \qquad \text{for } k = 0, \ldots, N-1. \tag{5.14}$$

Here,

$$\Gamma^{\mathrm{m}}_{\theta^{\mathrm{m}}_k} \colon X \times X \to X \quad \text{and} \quad \Gamma^{\mathrm{d}}_{\theta^{\mathrm{d}}_k} \colon Y \times Y \times Y \to Y$$

are the updating operators. This corresponds to defining $\mathcal{R}_\theta \colon Y \to X$ with $\theta_k = (\theta^{\mathrm{m}}_k, \theta^{\mathrm{d}}_k)$ as in (5.13), where $\Lambda_{\theta_k} \colon X \times Y \to X \times Y$ is given by

$$\Lambda_{\theta_k}(f, v) := (\Gamma^{\mathrm{m}}_{\theta^{\mathrm{m}}_k}(f, [\partial\,\mathcal{A}(f)]^*(\Gamma^{\mathrm{d}}_{\theta^{\mathrm{d}}_k}(v, \mathcal{A}(f), g))), \Gamma^{\mathrm{data}}_{\theta^{\mathrm{d}}_k}(v, \mathcal{A}(f), g))). \tag{5.15}$$

This is illustrated in Figure 5.2, and similar networks are also suggested by Vogel and Pock (2017) and Kobler *et al.* (2017), who extend the approach of Hammernik *et al.* (2018) by parametrizing and learning the data discrepancy $\mathcal{L}$. Applications are for inverting the two-dimensional Fourier transform (two-dimensional MRI image reconstruction). See also He *et al.* (2019), who unroll an ADMM scheme with updates in both reconstruction and data spaces and apply that to two-dimensional CT reconstruction.

Finally, allowing for some memory in both model parameter and data spaces leads to the learned primal–dual scheme of Adler and Öktem (2018*b*), which is used for low-dose two-dimensional CT reconstruction. The robustness of this approach against uncertainties in the image and uncertainties in system settings is empirically studied by Boink, van Gils, Manohar and Brune (2018). The conclusion is that learning improves pure knowledge-

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

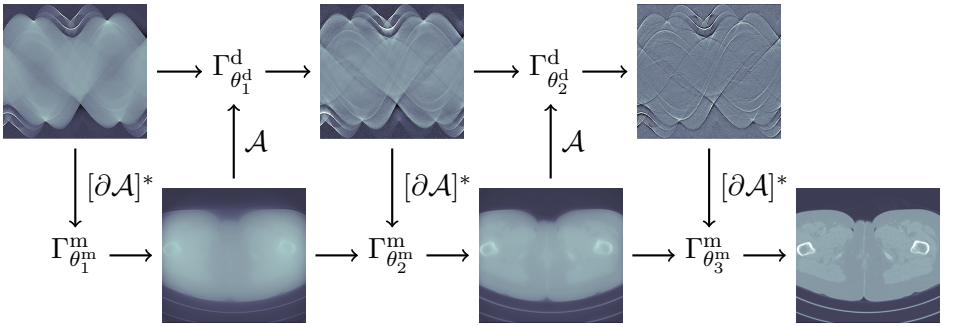


Figure 5.2. Learned iterative method in both model parameter and data spaces. Illustration of the operator obtained by unrolling the scheme in (5.14) for $N = 3$ in the context of CT image reconstruction (Section 7.3.1).

based reconstruction in terms of noise removal and background identification, and more variety in the training set increases the robustness against image uncertainty. Robustness against model uncertainty, however, is not readily obtained. An application of a learned primal dual scheme to breast tomosynthesis is given by Moriakov *et al.* (2018). To outperform existing reconstruction methods, one needs to encode the information about breast thickness into the learned updates for both primal and dual variables.

Further remarks. Successively reducing the constraints on the neural network architecture in the learned iterative scheme allows for larger model capacity, but training such a model also requires more training data. The well-known universal approximation property of deep neural networks (see Cybenko 1989, Hornik, Stinchcombe and White 1989, Hornik 1991, Barron 1994) implies that the learned iterative schemes in their most unconstrained formulation can approximate a Bayes estimator arbitrarily well given enough supervised data and neural network parameters (model capacity). This does not necessarily hold for the more constrained network architectures, such as those used by the variational networks of Hammernik *et al.* (2018). See also Section 8.2.1 for remarks on approximation properties of certain deep neural networks.

Unrolling an iterative scheme is a central theme in learned iterative methods, and it allows one to construct a deep neural network from an iterative scheme that is stopped after N steps. Here, a larger N simply means adding more layers, *i.e.* increasing the model capacity. The same idea is used to solve an optimization problem more rapidly (Section 4.9), but here the training is unsupervised and the loss is given by the objective that we seek to minimize. Such an architecture can be used for computing a Bayes estimator when trained using a supervised data or a MAP estimator when trained using unsupervised data. The same holds also for the GAN approach taken in

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Mardani *et al.* (2017*b*) (Section 5.1.6), which uses the same architecture as learned iterative methods but a different loss, that is, it computes a different estimator.

5.1.5. Learned post- and/or pre-processing

One of the earliest applications of deep learning in inverse problems in imaging was as a post-processing tool. Here, an established non-learned reconstruction method is followed by a deep neural network that is trained to 'clean up' noise and artefacts introduced by the first inversion step. The first inversion step is typically performed using a pseudo-inverse of the forward operator, *e.g.* backprojection or FBP in CT or a zero-filling solution in MRI. This initial reconstruction method can be seen as a way to account for how data are generated, whereas the learning part acts only on the model parameter instead of data.

As one might expect, a vast number of papers apply deep learning to images obtained from some kind of image reconstruction. In principle, all of these qualify as a learned post-processing scheme. We will not attempt to prove a (near-complete) survey of these since the learning part is not directly related to the inverse problem. Instead we mention some key publications from imaging that have image reconstruction as their main theme, followed by a characterization of the estimator one seeks to approximate when using learned post-processing.

Selected publications focusing on image reconstruction. We start with surveying work related to learned post-processing for CT image reconstruction. Early approaches used a CNN to map a sparse-view CT reconstruction to a full-view one, as in Zhao, Chen, Zhang and Jin (2016). A similar approach to mapping low-dose CT images to normal-dose images (denoising) is that of Chen *et al.* (2017*b*), who train a CNN on image patches. See also Chen *et al.* (2017*a*) for an approach that uses a residual encoder–decoder CNN (RED-CNN) trained on image patches for the same purpose.

Denoising low-dose CT images can also be done using a U-Net, as in Jin, McCann, Froustey and Unser (2017). Another variant is to use the U-Net on directional wavelets (AAPM-Net), as in Kang, Min and Ye (2017). This method came second in the 2016 AAPM Low Dose CT Grand Challenge.[6] It has since been further developed and refined in a series of publications: for example, Kang, Chang, Yoo and Ye (2018) and Kang and Ye (2018) modify the AAPM-Net architecture by using a wavelet residual network (WavResNet), which is a deep CNN reinterpreted as cascaded convolution framelet signal representation. Another drawback of AAPM-Net is that

[6] The method that won was a variational method with a non-local regularizer (Kim, Fakhri and Li 2017), but that approach has a run-time that scales very poorly with problem size.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

it does not satisfy the frame condition and it overly emphasizes the low-frequency component of the signal, which leads to blurring artefacts in the post-processed CT images. To address this, Han and Ye (2018) suggest a U-Net-based network architecture with directional wavelets that satisfy the frame condition. Finally, Ye *et al.* (2018) develop a mathematical framework to understand deep learning approaches for inverse problems based on these deep convolutional framelets. Such architectures represent a signal decomposition similar to using wavelets or framelets, but here the basis is learned from the training data. This idea of using techniques from applied harmonic analysis and sparse signal processing to analyse approximation properties of certain classes of deep neural networks bears similarities to Bölcskei, Grohs, Kutyniok and Petersen (2019) (see Section 8.2.1) and the scattering networks discussed in Section 4.5 as well as work related to multilayer convolutional sparse coding outlined in Section 4.4.2.

Yet another CNN architecture (Mixed-Scale Dense CNN) is proposed in Pelt, Batenburg and Sethian (2018) for denoising and removing streak artefacts from limited angle CT reconstructions. Empirical evidence shows that this architecture comes with some advantages over encoder–decoder networks. It can be trained on relatively small training sets and the same hyper-parameters in training can often be re-used across a wide variety of problems. This removes the need to perform a time-consuming trial-and-error search for hyper-parameter values.

Besides architectures, one may also consider the choice of loss function, as in Zhang *et al.* (2018), who consider CNN-based denoising of CT images using a loss function that is a linear combination of squared L^2 and multi-scale structural similarity index (SSIM). A closely related work is that of Zhang and Yu (2018), which uses a CNN trained on image patches with a loss function that is a sum of squared L^2 losses over the patches. The aim here is to reduce streak artefacts from highly scattering media, such as metal implants. A number of papers use techniques from GANs to post-process CT images. Shan *et al.* (2018) use a conveying path-based convolutional encoder–decoder network. A novel feature of this approach is that an initial three-dimensional denoising model can be directly obtained by extending a trained two-dimensional CNN, which is then fine-tuned to incorporate three-dimensional spatial information from adjacent slices (transfer learning from two to three dimensions). The paper also contains a summary of deep learning network architectures for CT post-processing listing the loss function (squared L^2, adversarial or perpetual loss). A similar approach is taken by Yang *et al.* (2018*c*), who denoise CT images via a GAN with Wasserstein distance and perceptual similarity. The perceptual loss suppresses noise by comparing the perceptual features of a denoised output against those of the ground truth in an established feature space, while the generator focuses more on migrating the data noise distribution. Another approach is

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

that of You *et al.* (2018*a*), who use a generator from a GAN to generate high-resolution CT images from low-resolution counterparts. The model is trained on semi-supervised training data and the training is regularized by enforcing a cycle-consistency expressed in terms of the Wasserstein distance. See also You *et al.* (2018*b*) for similar work, along with an investigation of the impact of different loss functions for training the GAN.

For PET image reconstruction, da Luis and Reader (2017) use a CNN to denoise PET reconstructions obtained by ML-EM. A more involved approach is presented in Yang, Ying and Tang (2018*a*), which learns a post-processing step for enhancing PET reconstructions obtained by MAP with a Green smoothness prior (see Yang, Ying and Tang 2018*a*, equation (6)). More precisely, this is supervised training on tuples consisting of ground truth and a number of MAP solutions where one varies parameters defining the prior. The training seeks to learn a mapping that takes a set of small image patches at the same location from the MAP solutions to the corresponding patch in the ground truth, thereby resulting in a learned patch-based image denoising scheme. A related approach is that of Kim *et al.* (2018), who train a CNN to map low-dose PET images to a full-dose one. Both low-dose and full-dose reconstructions are obtained using ordered subsets ML-EM. Since the resulting trained CNN denoiser produces additional bias induced by the disparity of noise levels, one considers learning a regularizer that includes the CNN denoiser (see Kim *et al.* 2018, equation (8)). The resulting variational problem is solved using the ADMM method.

Concerning MRI, most learned post-processing applications seek to train a mapping that takes a zero-filling reconstruction[7] obtained from undersampled MRI data to the MRI reconstruction that corresponds to fully sampled data. For example, Wang *et al.* (2016) use a CNN for this purpose, and the deep learning output is either used as an initialization or as a regularization term in classical compressed sensing approaches to MRI image reconstruction. Another example is that of Hyun *et al.* (2018), who use a CNN to process a zero-filling reconstruction followed by a particular k-space correction. This outperforms plain zero-filling reconstruction as well as learned post-processing where Fourier inversion is combined with a trained denoiser based on a plain U-Net architecture.

Similar to CT image processing, there has been some work on using GANs to post-process MRI reconstructions. One example is that of Quan, Member, Nguyen-Duc and Jeong (2018), who use a generator within a GAN setting to learn a post-processing operator that maps a zero-filling reconstruction image to a full reconstruction image. Training is regularized using

[7] Zero-filling reconstruction is computed by setting to zero all Fourier coefficients that are not measured in the MRI data, and then applying a normal inverse Fourier transform.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

a loss that includes a cyclic loss consistency term that promotes accurate interpolation of the given under-sampled k-space data. The generator consists of multiple end-to-end networks chained together, where the first network translates a zero-filling reconstruction image to a full reconstruction image, and the following networks improve accuracy of the full reconstruction image (refinement step). Another approach using a generator from a trained GAN is given by Yang *et al.* (2018*b*), who use a U-Net architecture with skip connections for the generator. The loss consists of an adversarial loss term, a novel content loss term considering both squared L^2 and a perceptual loss term defined by pre-trained deep convolutional networks. There is also a squared L^2 in both model parameter and data spaces, and the latter involves applying the forward operator to the training data to evaluate the squared L^2 in data space. See Yang *et al.* (2018*b*, equation (13)) for the full expression.

We conclude by mentioning some approaches that involve pre-processing. For CT imaging, deep-learning-based pre-processing targets sinogram inpainting, which is the task of mapping observed, sparsely sampled, CT projection data onto corresponding densely sampled CT projection data. Lee *et al.* (2019) achieve this via a plain CNN whereas Ghani and Karl (2018) use a generator from a trained conditional GAN. A CNN is also used by Hong *et al.* (2018) to pre-process PET data. Here one uses a deep residual CNN for PET super-resolution, that is, to map PET sinogram data from a scanner with large pixellated crystals to one with small pixellated crystals. The CNN-based method was designed and applied as an intermediate step between the projection data acquisition and the image reconstruction. Results are validated using both analytically simulated data, Monte Carlo simulated data and experimental pre-clinical data. In a similar manner, Allman, Reiter and Bell (2018) use a CNN to pre-process photoacoustic data to identify and remove noise artefacts. Finally, we cite Huizhuo, Jinzhu and Zhanxing (2018), who use a CNN to jointly pre- and post-process CT data and images. The pre-processing amounts to sinogram in-painting and the post-processing is image denoising, and the middle reconstruction step is performed using FBP. The loss function for this *joint pre- and post-processing* scheme is given in Huizhuo *et al.* (2018, equation (3)).

Characterizing the estimator. Here we consider post-processing; one can make analogous arguments for pre-processing. To understand what learned post-processing computes, consider the supervised learning setting. Learned post-processing seeks to approximate the Bayes estimator for the model parameter conditioned on the initial reconstruction, which is a different estimator from the one considered in deep direct Bayes estimation (Sections 5.1.2–5.1.4).

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Stated formally, let $\mathbb{h}$ be a X-valued random variable defined as $\mathbb{h} := \mathcal{A}^\dagger(\mathbb{g})$, where $\mathbb{g}$ is the Y-valued random variable generating data. Also, let $\mathcal{A}^\dagger : Y \to X$ denote the fixed initial reconstruction operator that is not learned. We now seek the Bayes estimator for the conditional random variable $(\mathbb{f} \mid \mathbb{h} = h)$, where $h = \mathcal{A}^\dagger(g)$ with data $g \in Y$ being a single sample of $\mathbb{g}$. This yields a reconstruction operator $\mathcal{R} : Y \to X$ given as $\mathcal{R} := \mathcal{B}_\sigma \circ \mathcal{A}^\dagger$ with $\mathcal{B}_\sigma : X \to X$ solving

$$\mathcal{B}_\sigma \in \operatorname*{arg\,min}_{\mathcal{B}:\, X \to X} \mathbb{E}_{(\mathbb{f},\mathbb{h})\sim\sigma}[\ell_X(\mathbb{f}, \mathcal{B}(\mathbb{h}))], \tag{5.16}$$

where $\ell_X : X \times X \to \mathbb{R}$ is a fixed loss function. In the above, σ denotes the joint law for $(\mathbb{f}, \mathbb{h})$, which is clearly unknown. It can, however, be replaced by its empirical counterpart given from supervised training data,

$$\Sigma_m := \{(f_1, h_1), \ldots, (f_m, h_m)\} \subset X \times X, \tag{5.17}$$

where (f_i, h_i) are generated by $(\mathbb{f}, \mathbb{h}) \sim \sigma$. Furthermore, considering all possible estimators (non-randomized decision rules) $\mathcal{B} : X \to X$ in the minimization in (5.16) is computationally unfeasible. To address this, we consider a family $\{\mathcal{B}_\theta\}_{\theta\in\Theta}$ of estimators that is parametrized by a finite-dimensional parameter in Θ. Restricting attention to such a parametrized family of estimators yields the following empirical risk minimization problem:

$$\widehat{\theta} \in \operatorname*{arg\,min}_{\theta\in\Theta} \frac{1}{m} \sum_{i=1}^{m} \ell_X(f_i, \mathcal{B}_\theta(h_i)), \tag{5.18}$$

with $(f_i, h_i) \in \Sigma_m$ as in (5.17).

Now, consider the case when the initial reconstruction method $\mathcal{A}^\dagger : Y \to X$ is a linear sufficient statistic, that is,

$$\mathbb{E}[\mathbb{f} \mid \mathbb{g} = g] = \mathbb{E}[\mathbb{f} \mid \mathcal{A}^\dagger(\mathbb{g}) = \mathcal{A}^\dagger(g)].$$

For example, the operators given by the FBP and the backprojection are both linear sufficient statistics. If in addition the loss function is the squared L^2-norm, then $\mathcal{R} := \mathcal{B}_\sigma \circ \mathcal{A}^\dagger$ is also a Bayes estimator for $(\mathbb{f} \mid \mathbb{g} = g)$, that is, the reconstruction obtained from learned post-processing coincides with the one from deep direct Bayes estimation. Note, however, that this holds in the limit of infinite amount of training data and infinite model capacity. In fact, as we shall see in Section 7.3, when applied to finite number of training data and finite model capacity, learned post-processing differs from deep direct Bayes estimation.

5.1.6. Other estimators

Learning using a supervised GAN. Some recent work uses a GAN in a supervised learning setting, which leads to a training problem of the type (5.3). For example, Mardani *et al.* (2017*b*) defines the variant of (5.3) where ℓ_X

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

is the L^1-norm, $\ell_{\mathscr{P}_X}$ is the Pearson χ^2-divergence, which can be evaluated using a least-squares GAN (Mao *et al.* 2016), and $\mathcal{R}_\theta : Y \to X$ is given by an architecture adapted to MRI. Use of a 1-norm in the generator loss motivates the reference made to 'compressed sensing' made by the authors. See also Mardani *et al.* (2017*a*) and Schwab, Antholzer and Haltmeier (2018) for further work along these lines: for example, Schwab *et al.* (2018) more formally treat the manifold projection step that in Mardani *et al.* (2017*b*) is specially tailored for MRI imaging.

It is not easy to identify what estimator the above really corresponds to, but clearly the generator (after training) resembles a MAP estimator where the prior is given implicitly by the supervised training data. To some extent, one may view the above as a supervised variant of Section 4.7 that uses a GAN to learn a regularizer (prior) from unsupervised data.

Deep direct estimation of higher-order moments. As described in Adler and Öktem (2018*a*), one can train a deep neural network to directly approximate an estimator involving higher-order moments, such as pointwise variance and correlation. The starting point is the well-known result

$$\mathbb{E}_{\mathbb{w}}[\mathbb{w} \mid \mathbb{g} = \cdot\,] = \min_{\mathrm{h}\colon Y \to W} \mathbb{E}_{(\mathbb{g},\mathbb{w})}[\|\,\mathrm{h}(\mathbb{g}) - \mathbb{w}\|_W^2]. \tag{5.19}$$

In the above, $\mathbb{w}$ is *any* random variable taking values in some measurable Banach space W and the minimization is over all W-valued measurable maps on Y. This is useful since many estimators relevant for uncertainty quantification are expressible using terms of this form for appropriate choices of $\mathbb{w}$.

Specifically, Adler and Öktem (2018*a*) consider two (deep) neural networks $\mathcal{R}_{\theta^*} : Y \to X$ and $\mathrm{h}_{\phi^*} : Y \to X$ with appropriate architectures that are trained according to

$$\begin{aligned}
\theta^* &\in \arg\min_\theta \{\mathbb{E}_{(\mathbb{f},\mathbb{g})\sim\mu}[\|\mathbb{f} - \mathcal{R}_\theta(\mathbb{g})\|_X^2]\},\\
\phi^* &\in \arg\min_\phi \{\mathbb{E}_{(\mathbb{f},\mathbb{g})\sim\mu}[\|\,\mathrm{h}_\phi(\mathbb{g}) - (\mathbb{f} - \mathcal{R}_{\theta^*}(\mathbb{g}))^2\|_X^2]\}.
\end{aligned}$$

The joint law μ above can be replaced by its empirical counterpart given from supervised training data (f_i, g_i), so the μ-expectation is replaced by an averaging over training data. The resulting networks will then approximate the conditional mean and the conditional pointwise variance, respectively.

As already shown, by using (5.19) it is possible to rewrite many estimators as minimizers of an expectation. Such estimators can then be approximated using the direct estimation approach outlined here. This should coincide with computing the same estimator by posterior sampling (Section 5.2.1). Direct estimation is significantly faster, but not as flexible as posterior sampling since each estimator requires a new neural network that

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

specifically trained for that estimator. Section 7.7 compares the outcomes of the two approaches.

5.2. Deep posterior sampling

The idea here is to use techniques from deep learning to sample from the posterior. This can then be used to perform various statistical computations relevant to solving the inverse problem.

Approaches to sampling from high-dimensional distributions that do not use neural networks are very briefly surveyed in Section 3.5.2. A drawback of these approaches is that they require access to an explicit prior, so they do not apply to cases where no explicit prior is available. Furthermore, despite significant algorithmic advances, these methods do not offer computationally feasible means for sampling from the posterior in large-scale inverse problems, such as those arising in three-dimensional imaging. Here we survey an alternative method that uses conditional GAN for the same purpose (Section 5.2.1). This approach has very desirable properties, so it does not require access to an explicit prior, and it is computationally very efficient.

Remark 5.2. Deep Gaussian mixture models (Viroli and McLachlan 2017) are multi-layered networks where the variables at each layer follow a mixture of Gaussian distributions. Hence, the resulting deep mixture model consists of a set of nested (non-linear) mixtures of linear models. Such models can be shown to be universal approximators of probability densities and they can be trained using ML-EM techniques. This is an interesting approach for sampling from the posterior in Bayesian inversion, but it is yet to be used in this context.

Recent work using conditional GAN for the same purpose (Section 5.2.1) gives very promising results. We do not cover deep Gaussian mixture models (Viroli and McLachlan 2017), which are multi-layered networks where, at each layer, the variables follow a mixture of Gaussian distributions. Thus, the deep mixture model consists of a set of nested mixtures of linear models, which globally provide a non-linear model able to describe the data in a very flexible way. These are universal approximators of probability densities that are trainable using ML-EM techniques. This is an interesting approach that is yet to be used in the context of inverse problems.

5.2.1. Conditional GAN

The approach taken was first introduced by Adler and Öktem (2018*a*), and it is a special case of variational Bayes inference (Section 3.5.2) where the variational family is parametrized via GAN. More precisely, the idea is to explore the posterior by sampling from a generator that has been trained using a conditional Wasserstein GAN discriminator.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

To describe how a Wasserstein GAN can be used for this purpose, let data $g \in Y$ be fixed and assume that Π^g_{post}, the posterior of $\mathbb{f}$ at $\mathbb{g} = g$, can be approximated by elements in a parametrized family $\{\mathcal{G}_\theta(g)\}_{\theta\in\Theta}$ of probability measures on X. The best such approximation is defined as $\mathcal{G}_{\theta^*}(g)$, where $\theta^* \in \Theta$ solves

$$\theta^* \in \arg\min_{\theta\in\Theta} \ell_{\mathscr{P}_X}(\mathcal{G}_\theta(g), \Pi^g_{\text{post}}). \tag{5.20}$$

Here, $\ell_{\mathscr{P}_X} : \mathscr{P}_X \times \mathscr{P}_X \to \mathbb{R}$ quantifies the 'distance' between two probability measures on X. We are, however, interested in the best approximation for 'all data', so we extend (5.20) by including an averaging over all possible data. The next step is to choose a distance notion ℓ that is desirable from both a theoretical and a computational point of view. For example, the distance should be finite, and computational feasibility requires it to be differentiable almost everywhere, since this opens up using stochastic gradient descent (SGD)-type schemes. The Wasserstein 1-distance $\mathcal{W}$ (Section 8.2.7) has these properties (Arjovsky *et al.* 2017), and sampling from the posterior Π^g_{post} can then be replaced by sampling from the probability distribution $\mathcal{G}_{\theta^*}(g)$, where θ^* solves

$$\theta^* \in \arg\min_{\theta\in\Theta} \mathbb{E}_{\mathbb{g}\sim\sigma}[\mathcal{W}(\mathcal{G}_\theta(\mathbb{g}), \Pi^{\mathbb{g}}_{\text{post}})]. \tag{5.21}$$

In the above, σ is the probability distribution for data and the random variable $\mathbb{g} \sim \sigma$ generates data.

Observe now that evaluating the objective in (5.21) requires access to the very posterior that we seek to approximate. Furthermore, the distribution σ of data is often unknown, so an approach based on (5.21) is essentially useless if the purpose is to sample from an unknown posterior. Finally, evaluating the Wasserstein 1-distance directly from its definition is not computationally feasible.

On the other hand, as we shall see, *all* of these drawbacks can be circumvented by rewriting (5.21) as an expectation over the joint law $(\mathbb{f}, \mathbb{g}) \sim \mu$. This makes use of the Kantorovich–Rubinstein duality for the Wasserstein 1-distance (see (B.2)), and one obtains the following approximate version of (5.21):

$$\theta^* \in \arg\min_{\theta\in\Theta}\left\{\sup_{\phi\in\Phi} \mathbb{E}_{(\mathbb{f},\mathbb{g})\sim\mu}\big[\mathrm{D}_\phi(\mathbb{f},\mathbb{g}) - \mathbb{E}_{\mathbb{z}\sim\eta}[\mathrm{D}_\phi(\mathrm{G}_\theta(\mathbb{z},\mathbb{g}),\mathbb{g})]\big]\right\}. \tag{5.22}$$

Here, $\mathrm{G}_\theta : Z \times Y \to X$ (generator) is a deterministic mapping such that $\mathrm{G}_\theta(\mathbb{z}, g) \sim \mathcal{G}_\theta(g)$, where $\mathbb{z} \sim \eta$ is a 'simple' Z-valued random variable in the sense that it can be sampled in a computationally feasible manner. Next, the mapping $\mathrm{D}_\phi : X \times Y \to \mathbb{R}$ (discriminator) is a measurable mapping that is 1-Lipschitz in the X-variable.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

At first sight, it might be unclear why (5.22) is better suited than (5.21) to sampling from the posterior, especially since the joint law μ in (5.22) is unknown. The advantage becomes clear when one has access to supervised training data for the inverse problem, *i.e.* i.i.d. samples $(f_1, g_1), \ldots, (f_m, g_m)$ generated by the random variable $(\mathbb{f}, \mathbb{g}) \sim \mu$. The μ-expectation in (5.22) can then be replaced by an averaging over training data.

To summarize, solving (5.22) given supervised training data in $X \times Y$ amounts to learning a generator $\mathrm{G}_{\theta^*}(\mathbb{z}, \cdot)\colon Y \to X$ such that $\mathrm{G}_{\theta^*}(\mathbb{z}, g)$ with $\mathbb{z} \sim \eta$ is approximately distributed as the posterior Π^g_{post}. In particular, for given $g \in Y$ we can sample from Π^g_{post} by generating values of $z \mapsto \mathrm{G}_{\theta^*}(z, g) \in X$ in which $z \in Z$ is generated by sampling from $\mathbb{z} \sim \eta$.

An important part of the implementation is the concrete parametrizations of the generator and discriminator:

$$\mathrm{G}_\theta\colon Z \times Y \to X \quad \text{and} \quad \mathrm{D}_\phi\colon X \times Y \to \mathbb{R}.$$

We use deep neural networks for this purpose, and following Gulrajani *et al.* (2017), we softly enforce the 1-Lipschitz condition on the discriminator by including a gradient penalty term in the training objective function in (5.22). Furthermore, if (5.22) is implemented as is, then in practice $\mathbb{z}$ is not used by the generator (so called mode-collapse). To solve this problem, we introduce a novel conditional Wasserstein GAN discriminator that can be used with conditional WGAN without impairing its analytical properties: see Adler and Öktem (2018*a*) for more details.

We conclude by referring to Section 7.7 for an example of how the conditional Wasserstein GAN can be used in clinical image-guided decision making.

6. Special topics

In this section we address several topics of machine learning that do not strictly fall within the previously covered contexts of functional analytic or statistical regularization. In Section 6.1 we discuss regularization methods that go beyond pure reconstructions. These reconstructions include – at least partially – the decision process, which typically follows the solution of inverse problems, for example examination of a CT reconstruction by a medical expert. Then Section 6.2.1 aims at investigating the connections between neural networks and differential equations, and Section 6.2 discusses the case where the forward operator is incorrectly known. Finally, Section 6.2.2 discusses total least-squares approaches, which are classical tools for updating the operator as well as the reconstruction based on measured data. We are well aware that this is still very much an incomplete list of topics not covered in the previous sections. As already mentioned

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

in Section 1, we apologize for our ignorance with respect to the missing material.

6.1. Task-adapted reconstruction

Estimating a model parameter in an inverse problem is often only one of many steps in a procedure where the reconstructed model parameter is used in a task. Consider a setting where the task is given by an operator $\mathcal{T}\colon X \to D$ (*task operator*) which maps a model parameter f to an element in some set D (decision space). Such tasks were introduced in Louis (2011) within the functional analytic framework (Section 2), and image segmentation served as the prime example.

A wider range of tasks can be accounted for if one adopts the statistical view as in Adler *et al.* (2018). Here we introduce a D-valued random variable $\mathbb{d}$ and interpret $\mathcal{T}$ as a non-randomized decision rule that is given as a Bayes estimator. The risk is given through the *task loss* $\ell_D\colon D \times D \to \mathbb{R}$ and the optimal task operator is the Bayes estimator with respect to the task loss, that is,

$$\mathcal{T}_{\text{opt}} \in \operatorname*{arg\,min}_{\mathcal{T}\colon X \to D} \mathbb{E}_{(\mathbb{f},\mathbb{d})}[\ell_D(\mathcal{T}(\mathbb{f}), \mathbb{d})]. \tag{6.1}$$

In practice we restrict ourselves to a parametrized family of tasks $\mathcal{T}_\phi\colon X \to D$ with $\phi \in \Phi$, typically given by deep neural networks. There are essentially three ways to combine a neural network $\mathcal{R}_\theta\colon Y \to X$ for reconstruction with a neural network $\mathcal{T}_\phi\colon X \to D$ for the task that is given by (6.1). The approaches differ in the choice of loss used for learning $(\widehat{\theta}, \widehat{\phi}) \in \Theta \times \Phi$ in

$$\mathcal{T}_{\widehat{\phi}} \circ \mathcal{R}_{\widehat{\theta}}\colon Y \to D. \tag{6.2}$$

Sequential training. The optimal parameter $(\widehat{\theta}, \widehat{\phi}) \in \Theta \times \Phi$ in (6.2) is given as

$$\begin{cases} \widehat{\theta} \in \operatorname*{arg\,min}_{\theta \in \Theta} \mathbb{E}_{\mathbb{f},\mathbb{g}}[\ell_X(\mathcal{R}_\theta(\mathbb{g}), \mathbb{f})], \\ \widehat{\phi} \in \operatorname*{arg\,min}_{\phi \in \Phi} \mathbb{E}_{\mathbb{g},\mathbb{d}}[\ell_D(\mathcal{T}_\phi \circ \mathcal{R}_{\widehat{\theta}}(\mathbb{g}), \mathbb{d})]. \end{cases}$$

The training data are samples (f_i, g_i) generated by $(\mathbb{f}, \mathbb{g})$ for computing $\widehat{\theta}$ and (g_i, d_i) generated by $(\mathbb{g}, \mathbb{d})$ for computing $\widehat{\phi}$.

End-to-end training. The optimal parameter $(\widehat{\theta}, \widehat{\phi}) \in \Theta \times \Phi$ in (6.2) is given by directly minimizing the loss for the task, that is,

$$(\widehat{\phi}, \widehat{\theta}) \in \operatorname*{arg\,min}_{(\phi,\theta) \in \Theta \times \Phi} \mathbb{E}_{\mathbb{g},\mathbb{d}}[\ell_D(\mathcal{T}_\phi \circ \mathcal{R}_\theta(\mathbb{g}), \mathbb{d})].$$

The training data are samples (g_i, d_i) generated by $(\mathbb{g}, \mathbb{d})$.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Task-adapted training. This refers to anything in between sequential and end-to-end training. More precisely, $(\widehat{\theta}, \widehat{\phi}) \in \Theta \times \Phi$ in (6.2) is given by minimizing the following *joint expected loss* (*risk*):

$$(\widehat{\theta}, \widehat{\phi}) \in \underset{(\theta,\phi)\in\Theta\times\Phi}{\arg\min}\, \mathbb{E}_{(\mathbb{f},\mathbb{g},\mathbb{d})}[(1-C)\ell_X(\mathcal{R}_\theta(\mathbb{g}), \mathbb{f}) + C\ell_D(\mathcal{T}_\phi \circ \mathcal{R}_\theta(\mathbb{g}), \mathbb{d})]. \quad (6.3)$$

The parameter $C \in [0, 1)$ above is a tuning parameter where $C \approx 0$ corresponds to sequential training and $C \to 1$ to end-to-end training. The training data are samples (f_i, g_i, d_i) generated by $(\mathbb{f}.\mathbb{g}, \mathbb{d})$.

Task-adapted training is a generic approach to adapting the reconstruction to a task with a plug-and-play structure for adapting to a specific inverse problem and a specific task. The former can be achieved by using a suitable neural network architecture, such as one given by a learned iterative method (Section 5.1.4). For the latter, note that the framework can handle *any task* that is given by a trainable neural network. This includes a wide range of tasks, such as semantic segmentation (Thoma 2016, Guo, Liu, Georgiou and Lew 2018), caption generation (Karpathy and Fei-Fei 2017, Li, Liang, Hu and Xing 2018*a*), in-painting (Xie, Xu and Chen 2012), depixelization/super-resolution (Romano, Isidoro and Milanfar 2017*b*), demosaicing (Syu, Chen and Chuang 2018), image translation (Wolterink *et al.* 2017), object recognition (Sermanet *et al.* 2013, He *et al.* 2016, Farabet, Couprie, Najman and LeCun 2013) and non-rigid image registration (Yang, Kwitt, Styner and Niethammer 2017, Ghosal and Ray 2017, Dalca, Balakrishnan, Guttag and Sabuncu 2018, Balakrishnan *et al.* 2019). Section 7.6 shows the performance of task-adapted reconstruction for joint tomographic image reconstruction and segmentation of white brain matter.

The importance of task-adapted reconstruction is also emphasized in the editorial of Wang, Ye, Mueller and Fessler (2018), which explicitly points out the potential in integrating reconstruction in an end-to-end workflow for medical imaging. They even coin the notion of 'rawdiomics', which is task-adapted reconstruction with the task corresponding to radiomics.[8] Most approaches to radiomics include some kind of classification that is performed by deep learning, and repeatability and reproducibility are among the main challenges (Rizzo *et al.* 2018, Traverso, Wee, Dekker and Gillies 2018). Typically, trained classifiers fail when confronted with images that are acquired using an acquisition protocol that is not represented in training data. This becomes especially problematic in multicentre studies where images are acquired using varying acquisition protocols and/or equipment. Clearly, the natural option is to include the information on how the images are generated, which naturally leads to task-adapted reconstruction.

[8] Radiomics seeks to identify distinctive imaging features between disease forms that may be useful for predicting prognosis and therapeutic response for various conditions.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

6.2. Non-perfect forward operators

As already discussed in Section 1, classical inverse problems are based on a mathematical formulation of the forward operator $\mathcal{A}$. Those models are typically derived from physical first principles or other well-established laws and expert descriptions. These models are never complete. In most cases these models are regarded as sufficiently accurate to capture the main properties and a more detailed model would not help the reconstruction process in the presence of noisy data. However, there are certain cases, for example emerging new technologies, where models are still underdeveloped. Here one can aim to obtain an at least partially updated operator based on sets of test data.

Secondly, such a data-driven approach to model updates might also be necessary if one has a complete but very complex forward operator. The complexity of the model might lead to numerically very costly computations, which, for example in the case of optoacoustic tomography, are beyond any limits required for routine clinical applications. In this case one might resort to a much simpler analytical model, which is then updated using data-driven approaches. A third line of motivation for using partially learned operators refers to models that use so-called measured system matrices. These system matrices determine the linear forward operator experimentally, and hence their accuracy is limited by measurement accuracy.

6.2.1. Learning physics

Several recent papers have discussed the application of deep learning in forward problems: see Khoo, Lu and Ying (2017), Raissi and Karniadakis (2017), Sirignano and Spiliopoulos (2017), Tompson, Schlachter, Sprechmann and Perlin (2017), E, Han and Jentzen (2017) and Wu, Zhang, Shen and Zhai (2018).

Several authors have drawn the comparison between neural networks and PDEs. For example 'PDE-Net' (Long, Lu, Ma and Dong 2018) proposes designing a feed-forward neural network with convolution filters representing spatial derivatives up to a certain order, and multiplied by spatially varying weights. Training this system on dynamic data obtained with accurate numerical models allowed the discovery of appropriate PDEs for different physical problems. Other examples include learning coefficients of a PDE via optimal control (Liu, Lin, Zhang and Su 2010), as well as deriving CNN architectures motivated by diffusion processes (Chen *et al.* 2015) (compare also Section 4.3.2 and in particular (4.7) for learned reaction–diffusion equations), deriving stable architectures by drawing connections to ordinary differential equations (Haber and Ruthotto 2017) and constraining CNNs (Ruthotto and Haber 2018) by the interpretation as a partial differential

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

equation. Another fascinating approach to learning first-principles physical models from data is that of Lam, Horesh, Avron and Willcox (2017).

6.2.2. Total least-squares

A widely used approach to integrating operator updates into regularization schemes for inverse problems can be formulated by generalized Tikhonov functionals. This is motivated by the total least-squares (TLS) approach (Golub and Van Loan 1980) (also known as 'errors-in-variable regression' in the statistical literature). One extension includes a regularization resulting in an approach called regularized total least-squares (R-TLS) (Golub, Hansen and O'Leary 1999, Markovsky and Van Huffel 2007), which for linear operators $\mathcal{A} = \mathbf{A}$ aims to learn an operator correction $\delta\mathbf{A}$ from the data by

$$\arg\min_{\delta\mathbf{A},f} \frac{1}{2}\|(\mathbf{A}+\delta\mathbf{A})f - g\|^2 + \frac{\alpha}{2}\|\,\mathbf{L}\,f\|^2 + \frac{\beta}{2}\|\delta\mathbf{A}\,\|_{\mathrm{F}}^2, \tag{6.4}$$

where the operator norm is the Frobenius norm (typical choice $\beta = 1$). The linear operator (matrix) $\mathbf{L}$ is included in order to allow for more general regularization terms; for simplicity one may choose the identity $\mathbf{L} = \mathbf{I}$.

In the TLS literature, the minimization in (6.4) is commonly formulated with respect to $\widetilde{\mathbf{A}}$, *i.e.* defined by $\widetilde{\mathbf{A}} := \mathbf{A}+\delta\mathbf{A}$. This formulation uses a single data point g for simultaneously computing an operator update and for computing an approximation to the inverse problem. This is a heavily under-determined problem, which, however, leads to good results, at least for some applications: see Gutta *et al.* (2019), Kluth and Maass (2017) and Hirakawa and Parks (2006). The regularized TLS approach has been analysed by Golub *et al.* (1999), for example, who prove an equivalence result to classical Tikhonov regularization (Golub *et al.* 1999, Theorem 2.1), which we restate here.

Theorem 6.1. The solution $\hat{f}_\gamma$ to the problem

$$\min_{\widetilde{\mathbf{A}},f}\{\|\widetilde{\mathbf{A}}f - g\|^2 + \|\widetilde{\mathbf{A}} - \mathbf{A}\,\|_{\mathrm{F}}^2\},\ \text{subject to}\ \|\,\mathbf{L}\,f\| = \gamma$$

is a solution $(\mathbf{A}^T\,\mathbf{A}+\lambda_I I_n + \lambda_L\,\mathbf{L}^T\,\mathbf{L})f = \mathbf{A}^T\,g$ where

$$\lambda_I = -\frac{\|g - \mathbf{A}\,f\|^2}{1+\|f\|^2} \quad \text{and} \quad \lambda_L = \mu(1+\|f\|^2).$$

In the above, μ is the Lagrange multiplier in

$$\mathcal{L}(\widetilde{\mathbf{A}}, f, \mu) = \|\widetilde{\mathbf{A}}f - g\|^2 + \|\widetilde{\mathbf{A}} - \mathbf{A}\,\|_{\mathrm{F}}^2 + \mu(\|\,\mathbf{L}\,f\|^2 - \gamma^2).$$

The two parameters are related by

$$\lambda_L\gamma^2 = y^{\delta,T}(g - \mathbf{A}\,f) + \lambda_I$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

and the residual fulfils

$$\|\widetilde{\mathbf{A}}f - g\|^2 + \|\widetilde{\mathbf{A}} - \mathbf{A}\|_{\mathrm{F}}^2 = -\lambda_I.$$

Golub *et al.* (1999) conclude that if $\|\mathbf{L}\, f_\gamma\| < \gamma$ solves the R-TLS problem, then it also solves the TLS problem without regularization. Moreover, this approach has been extended to include sparsity constrained optimization (Zhu, Leus and Giannakis 2011), which equivalent formulation then reads

$$\arg\min_{\delta\mathbf{A},f}\left\{\frac{1}{2}\|(\mathbf{A}+\delta\mathbf{A})f - g\|^2 + \alpha\|f\|_1 + \frac{\beta}{2}\|\partial\,\mathbf{A}\|_{\mathrm{F}}^2\right\}.$$

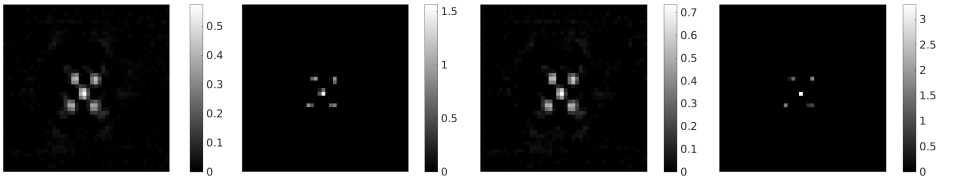
The previous sparsity-promoting approach, as well as the original R-TLS approach, can be easily extended if sets of training data (f_i, g_i) are available. One either aims for a two-stage approach to first update the operator and then solve the inverse problem with some new data point g, or one can integrate both steps at once leading to

$$\arg\min_{\delta\mathbf{A},f}\left\{\sum_i \frac{1}{2}\|(\mathbf{A}+\delta\mathbf{A})f_i - g_i\|^2 + \frac{1}{2}\|(\mathbf{A}+\delta\mathbf{A})f - g\|^2 + \frac{\alpha}{2}\|\mathbf{L}\, f\|^2 + \frac{\beta}{2}\|\delta\mathbf{A}\|_{\mathrm{F}}^2\right\}.$$

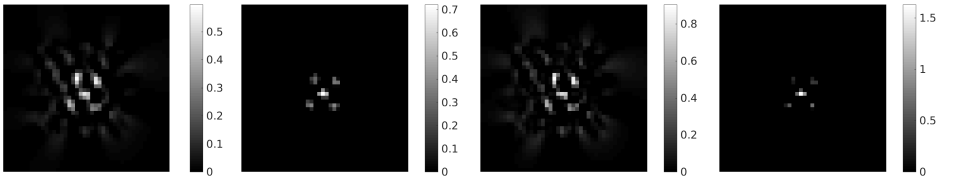
Total least-squares is still an active field of research: see *e.g.* Markovsky and Van Huffel (2007) and Beck, Sabach and Teboulle (2016). Alternative problem formulations of the R-TLS problem in terms of given error bounds $\|y-g\| \le \delta$ and $\|A-\tilde{A}\| \le \epsilon$ (instead of $\|\mathbf{L}\, f\| \le \gamma$ as choosing an appropriate γ can be challenging) were further investigated by Lu, Pereverzev and Tautenhahn (2009) and Tautenhahn (2008). One further extension of the R-TLS is to include a regularization with respect to parameters determining the operator (operator deviation). This was considered in a general Hilbert space setting (Bleyer and Ramlau 2013) for image deblurring (see Buccini, Donatelli and Ramlau 2018, who call it 'semi-blind').

For the purposes of the present review article we highlight the properties of TLS for applications in MPI: see Section 7.5, Knopp, Gdaniec and Möddel (2017) and Kluth (2018) for further information on MPI. The following is a brief summary of the results in Kluth and Maass (2017). We seek to reconstruct a five-point phantom consisting of a glass capillary with a diameter of 1.1 mm filled with tracer with a concentration of 0.5 mol/l provided by the GitHub project page of Knopp *et al.* (2016). The data-driven reconstructions (obtained by using a measured noisy forward operator) are shown in Figure 6.1(a–d). We obtain smoothed reconstructions of the five points, which is typical for Tikhonov regularization: see Figure 6.1(a). In contrast, the minimization with sparsity constraints is able to obtain a better localization of the tracer. Signal energy from regions filled with tracer which are not included in the system matrix used may cause a larger concentration value than the expected 0.5 mol/l. Using the total least-squares

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press



(a) Tikhonov, LS (b) sparsity, LS (c) Tikhonov, TLS (d) sparsity, TLS



(e) Tikhonov, LS (f) sparsity, LS (g) Tikhonov, TLS (h) sparsity, TLS

Figure 6.1. Reconstructions of a five-point phantom (pixel size 1 mm) provided by Knopp *et al.* (2016) obtained using Tikhonov (with $\alpha = 0.1 \times 10^{-6}$) and sparsity-promoting (with $\alpha = 0.1$) regularization with and without TLS. (a–d) Results from using a measured noisy forward operator. (e–h) Results from a knowledge-driven forward operator. Figure adapted from Kluth and Maass (2017).

approach further improves the localization in the sparse reconstruction for the data-based system matrices: see Figure 6.1(d).

A simplified model was fitted to measured data to obtain a knowledge-driven forward operator. As can be seen in Figure 6.1(e), using Tikhonov regularization results in a reconstruction of the five dots with additional background artefacts. Using the total least-squares approach in this set-up increases the contrast in concentration values but background artefacts are not significantly reduced. In contrast to the data-based reconstruction with Tikhonov regularization, the sparse knowledge-driven reconstruction in Figure 6.1(f) has a similar quality in terms of localization of the dots. By using the total least-squares approach, in Figure 6.1(h), the localization of the dots can be further improved such that the localization is similar in quality compared to the data-based sparse reconstruction.

6.2.3. Learned Landweber

Another approach to including a learning component into a forward operator is presented by Aspri, Banert, Öktem and Scherzer (2018). The starting point is to consider the iteratively regularized Landweber iteration (Scherzer 1998) (see also Kaltenbacher, Neubauer and Scherzer 2008), which amounts to computing the following iterative updates:

$$f^{k+1} := f^k - [\partial \mathcal{A}(f^k)]^*(\mathcal{A}(f^k) - g) - \lambda_k(f^k - f^0), \tag{6.5}$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

where $f^0 \in X$ is an initial guess that incorporates *a priori* knowledge about the unknown f_{true} we seek. Next, one can introduce a data-driven damping factor in the above Landweber iteration:

$$f^{k+1} := f^k - [\partial \mathcal{A}(f^k)]^*(\mathcal{A}(f^k) - g) - \lambda_k[\partial \mathcal{B}(f^k)]^*(\mathcal{B}(f^k) - g). \quad (6.6)$$

The (possibly non-linear) operator $\mathcal{B}\colon X \to Y$ can now be represented by a deep neural network that can be trained against supervised data by comparing the final iterate in (6.6) (iterates are stopped following the Morozov discrepancy principle (4.1)) against the ground truth for given data.

Convergence and stability for the scheme in (6.6) in infinite-dimensional Hilbert spaces is proved by Aspri *et al.* (2018). This theoretical results are complemented by several numerical experiments for solving linear inverse problems for the Radon transform and a non-linear inverse problem of Schlieren tomography. In these examples, however, Aspri *et al.* (2018) restrict attention to a linear operator $\mathcal{B}$.

6.3. Microlocal analysis

Microlocal analysis is a powerful mathematical theory for precisely describing how the singular part of a function, or more generally a distribution, is transformed under the action of an operator. Since its introduction to the mathematical community with the landmark publications by Sato (1971) and Hörmander (1971), it has proved itself useful in both pure and applied mathematical research. It is now a well-developed theory that can be used to study how singularities propagate under certain classes of operators, most notably Fourier integral operators, which include most differential and pseudo-differential operators as well as many integral operators frequently encountered in analysis, scientific computing and the physical sciences (Hörmander 1971, Candès, Demanet and Ying 2007).

The crucial underlying observation in microlocal analysis is that the information about the location of the singularities (singular support) needs to be complemented by specifying those 'directions' along which singularities may propagate. Making this precise leads to the notion of the wavefront set of the function (or distribution).

Role in inverse problems. Microlocal analysis is in particularly useful in inverse problems for a variety of reasons.

First, in many applications it is sufficient to recover the wavefront set of the model parameter from noisy data. For example, in imaging this would correspond to recovering the edges of the image from data. Such applications frequently arise when using imaging/sensing technologies where the transform is a pseudo-differential or Fourier integral operator (Krishnan and Quinto 2015). It turns out that one can use microlocal analysis to precisely describe how the wavefront set in data relates to the wavefront

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

set for the model parameter, and this explicit relation is referred to as the (microlocal) canonical relation. Using the canonical relation one can recover the wavefront set from data without solving the inverse problem, a process that can be highly non-trivial.

Second, the canonical relation also describes which part of the wavefront set one can recover from data. This was done by Quinto (1993) for the case when the two- or three-dimensional ray transform is restricted to parallel lines, and by Quinto and Öktem (2008) for an analysis in the region-of-interest limited angle setting. Faber, Katsevich and Ramm (1995) derived a related principle for the three-dimensional ray transform restricted to lines given by helical acquisition, which is common in medical imaging. Similar principles hold for transforms integrating along other types of curves, *e.g.* ellipses with foci on the x-axis and geodesics (Uhlmann and Vasy 2012).

Finally, recovering the wavefront set of the model parameter from data is a less ill-posed procedure than attempting to recover the model parameter itself. This was demonstrated in Davison (1983), where the severely ill-posed reconstruction problem in limited angle CT becomes mildly ill-posed if one settles for recovering the wavefront. See also Quinto and Öktem (2008) for an application of this principle to cryo-electron tomography.

Data-driven extraction of the wavefront set. The above motivates the inverse problems community to work with the wavefront set. One difficulty that has limited use of the wavefront set is that it is virtually impossible to extract it numerically from a digitized signal. This is due to its definition, which depends on the asymptotic behaviour of the Fourier transform after a localization procedure. An alternative possibility is to identify the wavefront set after transforming the signal using a suitable representation, *e.g.* a curvelet or shearlet transform (Candès and Donoho 2005, Kutyniok and Labate 2009). This requires analysing the rate of decay of transformed signal, which again is unfeasible in large-scale imaging applications.

A recent paper (Andrade-Loarca, Kutyniok, Öktem and Petersen 2019) uses a data-driven approach to training a wavefront set extractor applicable to noisy digitized signals. The idea is to construct a deep neural network classifier that predicts the wavefront set from the shearlet coefficients of a signal. The approach is successfully demonstrated on two-dimensional imaging examples where it outperforms all conventional edge-orientation estimators as well as alternative data-driven methods including the current state of the art. This learned wavefront set extractor can now be combined with a learned iterative method using the framework in Section 6.1.

Using the canonical relation to guide data-driven recovery. In a recent paper Bubba *et al.* (2018) consider using the aforementioned microlocal canonical relation to steer a data-driven component in limited angle CT reconstruction, which is a severely ill-posed inverse problem.

Bubba *et al.* develop a hybrid reconstruction framework that fuses a knowledge-driven sparse regularization approach with a data-driven deep learning approach. The learning part is only applied to those parts that are not possible to recover (invisible part), which in turn can be characterized *a priori* through the canonical relation. The theoretically controllable sparse regularization is thus applied to the remaining parts that can be recovered (visible part).

This decomposition into visible and invisible parts is achieved numerically via the shearlet transform, which allows us to resolve wavefront sets in phase space. The neural network is then used to infer unknown shearlet coefficients associated with the invisible part.

7. Applications

In this section we revisit some of the machine learning methods for inverse problems discussed in the previous sections, and demonstrate their applicability to prototypical examples of inverse problems.

7.1. A simple example

We start the applications part of the paper by considering the exemplar inverse problem of ill-conditioned matrix inversion. This example should highlight the particular difficulties of applying learning to solve an ill-posed inverse problem. Surprisingly, even small 2×2 examples cannot be solved reliably by straightforward neural networks! The results of this section are based on Maass (2019).

This small-scale setting allows a somewhat complete analysis of the neural network; in particular, we can prove the shortcomings of such neural nets if the condition number of the matrix and the noise level in the data are in a critical relation. To be precise, in our most basic example we set

$$\mathcal{A}_\varepsilon(f) = \mathbf{A}_\varepsilon \cdot f \quad \text{where } \mathbf{A}_\varepsilon = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 1+\varepsilon \end{pmatrix}.$$

This matrix has eigenvalues $\lambda_1 = 2 + \varepsilon/2 + O(\varepsilon^2)$ and $\lambda_2 = \varepsilon/2 + O(\varepsilon^2)$, with corresponding orthogonal eigenvectors

$$u_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + O(\varepsilon^2) \quad \text{and} \quad u_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix} + O(\varepsilon^2).$$

The ill-posedness of the problem, or rather the condition number of $\mathbf{A}_\varepsilon$, is controlled by $1/\varepsilon$. Typical values we have in mind here are $\varepsilon = 10^{-k}$ for $k = 0, \ldots, 10$.

We now compare two methods for solving the inverse problem of recovering f from $g = \mathbf{A}_\varepsilon \cdot f + e$. The first is classical Tikhonov regularization,

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

which only uses information about the operator $\mathcal{A}_\varepsilon$. Here, given data g we estimate f_{true} by $\mathcal{R}_\sigma^{\text{Tik}}(g)$, where

$$\mathcal{R}_\sigma^{\text{Tik}}(g) := (\mathcal{A}_\varepsilon^* \circ \mathcal{A}_\varepsilon + \sigma^2 \,\text{id})^{-1} \circ \mathcal{A}_\varepsilon^*(g) = (\mathbf{A}_\varepsilon^T \cdot \mathbf{A}_\varepsilon + \sigma^2 \,\mathbf{I})^{-1} \cdot \mathbf{A}_\varepsilon^T \cdot g. \quad (7.1)$$

The second inversion is based on a trained neural network, that is, given data g we estimate f_{true} by $\mathcal{R}_{\mathbf{W}^*}^{\text{NN}}(g)$ where $\mathcal{R}_{\mathbf{W}^*}^{\text{NN}} \colon Y \to X$ is a trained neural network with $\mathbf{W}^*$ given by

$$\mathbf{W}^* \in \arg\min_{\mathbf{W}} \frac{1}{m} \sum_{i=1}^{m} \| \mathcal{R}_{\mathbf{W}}^{\text{NN}}(g^{(i)}) - f^{(i)} \|^2. \quad (7.2)$$

In the above, $(f^{(i)}, g^{(i)}) \in X \times Y$ with $i = 1, \ldots, m$, where coefficients in $f^{(i)}$ are i.i.d. samples of a $N(0,1)$ distributed random variable, and $g^{(i)} := \mathcal{A}\, f^{(i)} + e^{(i)}$, where $e^{(i)} \in \mathbb{R}^2$ are i.i.d. samples of a $N(0, \sigma^2)$ distributed random variable. This approach is fully data-driven and does not use any explicit knowledge about the operator $\mathcal{A}$.

Both methods are evaluated using a different set of test data and results are compared by computing the mean error:

$$E^{\text{Tik}} := \frac{1}{n} \sum_{i=1}^{n} \| \mathcal{R}_\sigma^{\text{Tik}}(g^{(i)}) - f^{(i)} \|^2 \quad \text{and} \quad E^{\text{NN}} := \frac{1}{n} \sum_{i=1}^{n} \| \mathcal{R}_{\mathbf{W}}^{\text{NN}}(g^{(i)}) - f^{(i)} \|^2,$$

for n test pairs $(f^{(i)}, g^{(i)}) \in X \times Y$ with $g^{(i)} := \mathcal{A}\, f^{(i)} + e^{(i)}$ as in the training set above but clearly distinct from the training examples.

The design of the network is crucial. We use a minimal network which allows us to reproduce a matrix vector multiplication. Hence the network is – in principle – capable of recovering the Tikhonov regularization operator or even an improvement of it. We use a network with a single hidden layer with four nodes. We restrict the eight weights connecting the two input variables with the first layer by setting

$$w_1 = -w_3 = w_{11}, \quad w_2 = -w_4 = w_{12},$$
$$w_5 = -w_7 = w_{21}, \quad w_6 = -w_8 = w_{22},$$

as depicted in Figure 7.1. We obtain a neural network depending on four variables $w_{11}, w_{12}, w_{21}, w_{22}$ and the network acts as a multiplication of the matrix

$$\mathbf{W} = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$$

with the input vector $z = (z_1, z_2)$. We denote the output of such a neural network by $\mathcal{R}_{\mathbf{W}}^{\text{NN}}(z) = Wz$.

For later use we define $(2 \times m)$ matrices $\mathbf{X}, \mathbf{Y}$ and $\mathbf{E}$ that store the vectors $f^{(i)}, g^{(i)}$ and $e^{(i)}$ column-wise, so the training data can be summarized as

$$\mathbf{Y} = \mathbf{A}_\varepsilon \cdot \mathbf{X} + \mathbf{E}\,. \quad (7.3)$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

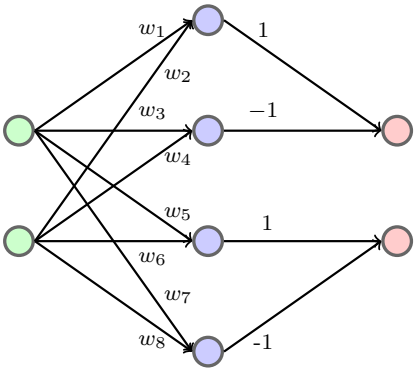


Figure 7.1. The network design with eight parameters, a setting that yields a matrix–vector multiplication of the input.

The training of such a network for modelling the forward problem is equivalent (using the Frobenius norm for matrices) to minimizing the expected mean square error

$$\min_{\mathbf{W}} \frac{1}{n}\sum_{i=1}^{n} \| \mathbf{W} f^{(i)} - g^{(i)} \|^2 = \min_{\mathbf{W}} \frac{1}{n} \| \mathbf{W}\mathbf{X} - \mathbf{Y} \|^2, \tag{7.4}$$

and the training model (7.2) for the inverse problem simplifies to

$$\min_{\mathbf{W}} \frac{1}{n}\sum_{i=1}^{n} \| \mathbf{W} g^{(i)} - f^{(i)} \|^2 = \min_{\mathbf{W}} \frac{1}{n} \| \mathbf{W}\mathbf{Y} - \mathbf{X} \|^2. \tag{7.5}$$

In the next paragraph we report some numerical examples before analysing these networks.

Testing error convergence for various values of ε. We train these networks using a set of training data $(f^{(i)}, g^{(i)})_{i=1,\dots,m}$ with $m = 10\,000$, *i.e.* $g^{(i)} = \mathcal{A}_\varepsilon f^{(i)} + e^{(i)}$. The network design with restricted coefficients as described above has four degrees of freedom $w = (w_{11}, w_{12}, w_{21}, w_{22})$. The corresponding loss function is minimized by a gradient descent algorithm, that is, the gradient of the loss function with respect to w is computed by backpropagation (Rumelhart, Hinton and Williams 1986, Martens and Sutskever 2012, Byrd, Chin, Nocedal and Wu 2012). We used 3000 iterations (epochs) of this gradient descent to minimize the loss function of a network for the forward operator using (7.4) or, respectively, for training a network for solving the inverse problem using (7.2). The MSE errors on the training data were close to zero in both cases.

After training we tested the resulting networks by drawing $n = 10\,000$ new data vectors $f^{(i)}$ as well as error vectors $e^{(i)}$. The $g^{(i)}$ were computed as above. Table 7.1 lists the resulting values using this set of test data

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Table 7.1. The errors of the inverse net with an ill-conditioned matrix $\mathbf{A}_\varepsilon$ (*i.e.* $\varepsilon \ll 1$) are large and the computed reconstructions with the test data are meaningless.

Error/choice of ε	1	0.1	0.01	0.0001
$\mathrm{NMSE}_{\mathrm{fwd}}$	0.002	0.013	0.003	0.003
$\mathrm{NMSE}_{\mathrm{inv}}$	0.012	0.8	10	10

for the network trained for the forward problem and the inverse problem, respectively:

$$\mathrm{NMSE}_{\mathrm{fwd}} := \frac{1}{n}\sum_{i=1}^{n} \| \mathbf{W}\, f^{(i)} - g^{(i)} \|^2,$$

$$\mathrm{NMSE}_{\mathrm{inv}} := \frac{1}{n}\sum_{i=1}^{n} \| \mathbf{W}\, f^{(i)} - g^{(i)} \|^2.$$

We observe that training the forward operator produces reliable results, as does the network for the inverse problem with $\varepsilon \geq 0.1$. However, training a network for the inverse problem with an ill-conditioned matrix $\mathbf{A}_\varepsilon$ with $\varepsilon \leq 0.01$ fails. This is confirmed by analysing the values of w and of the resulting matrix $\mathbf{W}$ after training. We would expect that in training the forward problem will produce values for $\mathbf{W}$ such that $\mathbf{W} \sim \mathbf{A}_\varepsilon$ and that training the inverse problems leads to $\mathbf{W} \sim \mathcal{R}_\sigma^{\mathrm{Tik}}(g)$ for some regularization parameter σ. For the forward problem, the difference between $\mathbf{W}$ and $\mathbf{A}_\varepsilon$ is of order 10^{-3} or below, but for $\varepsilon \leq 0.01$ the training of the inverse problem leads to a matrix that has no similarity to the Tikhonov regularized inverse. Using a network with a single internal layer but with more nodes and no restriction on the structure of the weights did not yield any significant improvements.

Analysis of trivial neural networks for inverse problems. In this section we analyse the case where the application of the neural network is strictly equivalent to a matrix–vector multiplication, that is, training of the network is given by (7.5). The optimal $\mathbf{W}$ in (7.5) is given by

$$\mathbf{W}^T = (\mathbf{Y}\,\mathbf{Y}^T)^{-1}\,\mathbf{Y}\,\mathbf{X}^T. \tag{7.6}$$

Standard arguments show that, together with that hypothesized in the numerical discussion above, the $\mathbf{W}$ in (7.6) coincides with the Tikhonov regularizers $\mathcal{R}_\sigma^{\mathrm{Tik}}(g)$ from (7.1). This is not a surprising result since (7.6) coincides with the classical maximum *a posteriori* (MAP) estimator of statistical inverse problems. However, analysing the variance $\mathbb{E}[\| \mathbf{W} - \mathbf{T} \|^2]$, where $\mathbf{T} = (\mathbf{A}_\varepsilon^T \cdot \mathbf{A}_\varepsilon + \sigma^2\, \mathbf{I})^{-1} \cdot \mathbf{A}_\varepsilon^T$ is the Tikhonov reconstruction matrix,

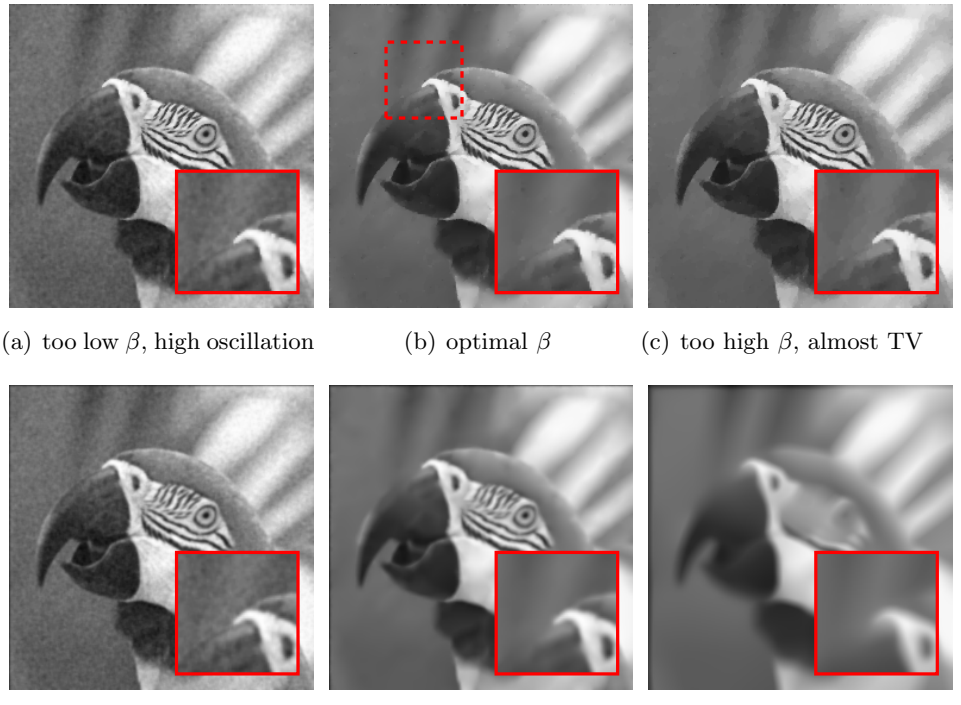
https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

reflects the ill-posedness of the problem. Indeed, as is demonstrated in a series of numerical tests in Maass (2019), the deviation of $\mathbf{W}$ from $\mathbf{T}$ will be arbitrarily large if ε and σ are both small. Of course, we can also give this a positive meaning: the noise level acts as a regularizer, and large σ yields more stable matrices $\mathbf{W}$. See Maass (2019) for details.

This small example clearly illustrates that one needs to have some insight into the nature of inverse problems for successfully applying deep learning techniques. Performance of practical examples of more targeted deep learning approaches to inverse problems will be discussed in the following Sections 7.3, 7.4, 7.6 and 7.7.

7.2. Bilevel learning for total variation image denoising

In (4.4) bilevel learning of TV-type regularizers was discussed as a way to make functional analytic regularization more data-driven. In what follows, we showcase some results of this learning approach for the case of image denoising, *i.e.* $\mathcal{A} = \mathrm{id}$.



(a) too low β, high oscillation (b) optimal β (c) too high β, almost TV

(d) too low α, low β: good match to noisy data (e) too low α, optimal β: optimal TV^2-like behaviour (f) too high α, high β: bad TV^2-like behaviour

Figure 7.2. (a–c) Effect of choosing β on total generalized variation $(\mathrm{TGV})^2$ denoising with optimal α. (d–f) Effect of choosing α too large in TGV^2 denoising.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Optimal TV-regularizers for image denoising. The regularization effect of TV and second-order TV approaches heavily depends on the choice of regularization parameters $\theta = \alpha$ (*i.e.* $= (\alpha, \beta)$ for second-order TV approaches). In Figure 7.2 we show the effect of different choices of α and β in TGV^2 denoising. In what follows we show some results from De los Reyes *et al.* (2017) applying the learning approach from (4.4) with the smoothed regularizer (4.6) to find optimal parameters in TV-type reconstruction models. The regularization effect of TV and second-order TV approaches heavily depends on the choice of regularization parameters $\theta = \alpha$ (*i.e.* $= (\alpha, \beta)$ for second-order TV approaches).

The first example is TGV denoising of an image corrupted with white Gaussian noise with PSNR of 24.72. The red dot in Figure 7.3 plots the discovered regularization parameter $\widehat{\theta} = (\hat{\alpha}, \hat{\beta})$ reported in Figure 7.4. Studying the location of the red dot, we may conclude that the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm managed to find a nearly optimal parameter in very few iterations: see Table 7.2. Although the optimization problem for (α, β) is non-convex, in all of our experiments we observed commendable convergence behaviour of the BFGS algorithm: see De los Reyes *et al.* (2017) for further examples.

To test the generalization quality of the bilevel learning model in De los Reyes *et al.* (2017), optimal parameters were cross-validated when tested for image denoising on the Berkeley segmentation data set (BSDS300) (Martin, Fowlkes, Tal and Malik 2001). A dataset of 200 images was selected and split into two halves of 100 images each. Optimal parameters were learned for each half individually, and then used to denoise the images of the other half. The results for TV denoising with L^2-loss function and data fidelity are reported in Table 7.3. The results for TGV denoising are reported in Table 7.4. In both experiments the parameters seem to be robust against cross-validation, both in terms of their optimal value, and average PSNR and SSIM (Wang, Bovik, Sheikh and Simoncelli 2004) quality measures of the denoised image.

Bilevel learning of the data discrepancy term in mixed noise scenarios. In the examples in the previous paragraph we considered bilevel parameter learning for TV-type image denoising, assuming that the noise in the image is normally distributed and consequently an L^2-data discrepancy term is the appropriate choice to take. The bilevel learning model (4.4), however, is capable of linear combinations of data discrepancy terms as in (4.5) which might be appropriate in situations of multiple noise distributions in the data: see *e.g.* Lanza, Morigi, Sgallari and Wen (2014), De los Reyes and Schönlieb (2013), Calatroni *et al.* (2017) and Calatroni (2015) and references therein. Calatroni *et al.* (2017) also considered infimal convolutions of data discrepancy functions.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

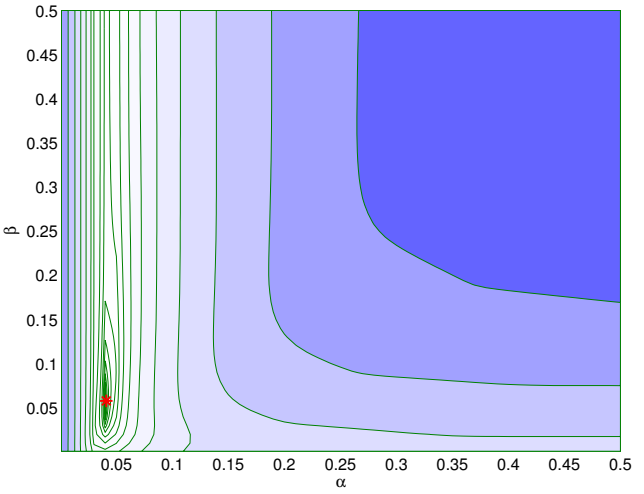


Figure 7.3. Contour plot of the objective functional in TGV^2 denoising in the (α, β)-plane.

(a) data, noisy image (b) TGV^2 (c) ICTV (d) TV

Figure 7.4. Optimal denoising results for TGV^2, ICTV and TV, all with L_2^2 as data discrepancy.

Table 7.2. Quantified results for the parrot image ($s :=$ image width/height in pixels $= 256$), using L_2^2 discrepancy.

Denoise	Initial (α, β)	Result $(\hat{\alpha}, \hat{\beta})$	Objective	SSIM	PSNR	Its	Figure
TGV^2	$(\hat{\alpha}_{\mathrm{TV}}/\mathrm{s}, \hat{\alpha}_{\mathrm{TV}})$	$(0.058/\mathrm{s}^2, 0.041/\mathrm{s})$	6.412	0.890	31.992	11	7.4(b)
ICTV	$(\hat{\alpha}/\mathrm{s}, \hat{\alpha}_{\mathrm{TV}})$	$(0.051/\mathrm{s}^2, 0.041/\mathrm{s})$	6.439	0.887	31.954	7	7.4(c)
TV	$0.1/\mathrm{s}$	$0.042/\mathrm{s}$	6.623	0.879	31.710	12	7.4(d)

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Table 7.3. Cross-validated computations on the BSDS300 data set (Martin *et al.* 2001) split into two halves of 100 images each. TV regularization with L^2-discrepancy and fidelity function. 'Learning' and 'validation' indicate the halves used for learning α and for computing the average PSNR and SSIM, respectively. Noise variance $\sigma = 10$.

Validation	Learning	α	Average PSNR	Average SSIM
1	1	0.0190	31.3679	0.8885
1	2	0.0190	31.3672	0.8884
2	1	0.0190	31.2619	0.8851
2	2	0.0190	31.2612	0.8850

Table 7.4. Cross-validated computations on the BSDS300 data set (Martin *et al.* 2001) split into two halves of 100 images each. TGV^2 regularization with L^2-discrepancy. 'Learning' and 'validation' indicate the halves used for learning α and for computing the average PSNR and SSIM, respectively. Noise variance $\sigma = 10$.

Validation	Learning	$\vec{\alpha}$	Average PSNR	Average SSIM
1	1	(0.0187, 0.0198)	31.4325	0.8901
1	2	(0.0186, 0.0191)	31.4303	0.8899
2	1	(0.0186, 0.0191)	31.3281	0.8869
2	2	(0.0187, 0.0198)	31.3301	0.8870

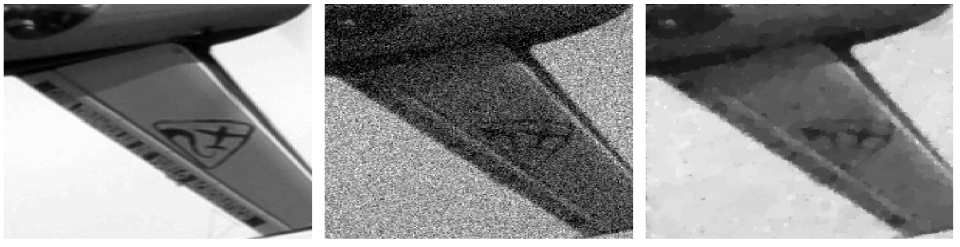
Figures 7.5 and 7.6 present denoising results with optimally learned parameters for mixed Gaussian and impulse noise and for mixed Gaussian and Poisson noise, respectively. See Calatroni *et al.* (2017) for more details. The original image has been corrupted with Gaussian noise of zero mean and variance 0.005 and then a percentage of 5% of pixels has been corrupted with impulse noise. The parameters have been chosen to be $\gamma = 10^4$, $\mu = 10^{-15}$ and the mesh step size $h = 1/312$. The computed optimal weights are $\hat{\lambda}_1 = 734.25$ and $\hat{\lambda}_2 = 3401.2$. Together with an optimal denoised image, the results show the decomposition of the noise into its sparse and Gaussian components: see Calatroni *et al.* (2017) for more details.

Remark 7.1. When optimizing only a handful of scalar parameters, as in the examples discussed above, bilevel optimization is by no means the most efficient approach for parameter learning. In fact, brute force line-search methods are in this context still computationally feasible as the dimensionality of the parameter space being explored is small. However, even in

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

(a) (b) (c) (d) (e)

Figure 7.5. Optimized impulse-Gaussian denoising: (a) original image, (b) noisy image with Gaussian noise of variance 0.005 and (c) with 5% of pixels corrupted with impulse noise, (d) impulse noise residuum, (e) Gaussian noise residuum. Optimal parameters $\hat{\lambda}_1 = 734.25$ and $\hat{\lambda}_2 = 3401.2$.



(a) (b) (c)

Figure 7.6. Optimized Poisson–Gauss denoising: (a) original image, (b) noisy image corrupted by Poisson noise and Gaussian noise with mean zero and variance 0.001, (c) denoised image. Optimal parameters $\hat{\lambda}_1 = 1847.75$ and $\hat{\lambda}_2 = 73.45$.

this small parameter example, investigating bilevel optimization methods is instructive, as it tells us something about the mathematical properties of parameter learning for the typically considered non-smooth variational regularization problems and the numerical approaches with which they can be tackled. Insight gained from this becomes particularly important when going to more advanced parametrizations, for instance when optimizing spatially varying regularization parameters (Van Chung *et al.* 2017) or different discrete parametrizations, as considered in Sections 4.3.2, 4.4 or 4.7.

7.3. Learned iterative reconstruction for computed tomography (CT) and photoacoustic tomography (PAT)

Learned iterative reconstruction schemes (Section 5.1.4) have been successfully applied to several large-scale inverse problems in imaging, such as image reconstruction in magnetic resonance imaging (MRI), CT and PAT. We will show examples from CT taken from Adler and Öktem (2017, 2018*b*) and PAT taken from Hauptmann *et al.* (2018).

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

7.3.1. CT image reconstruction

The learned primal–dual scheme in Adler and Öktem (2018*b*) is here tested on the two-dimensional CT image reconstruction problem, and its performance is evaluated in a simplified setting as well as a more realistic setting.

The forward operator for pre-log data is expressible in terms of the ray transform of (2.5), and for log data it is given by the ray transform of (2.6). The model parameter is a real-valued function defined on a domain $\Omega \subset \mathbb{R}^2$. This function represents the image we seek to recover and we assume $X \subset L^2(\Omega)$ is a suitable vector space of such functions.

This data set is used to train both the learned post-processing and learned iterative methods, where the former is filtered backprojection (FBP) reconstruction followed by a trained denoiser with a *U-Net* architecture and the latter is the learned primal–dual method in Adler and Öktem (2018*b*). Both networks were trained using the squared L^2-loss. The other two knowledge-driven reconstruction methods are the standard FBP and (isotropic) TV-regularized reconstruction. The FBP reconstructions is applied to log data using a Hann filter; the TV reconstruction was solved using 1000 iterations of the classical primal–dual hybrid gradient algorithm. The filter bandwidth in the FBP and the regularization parameter in the TV reconstruction were selected in order to maximize the PSNR.

In the simplified setting shown in Figure 7.8 (see also the summary in Table 7.5) the images are 128×128 pixel step functions, and we use supervised training data consisting of about 50 000 pairs of images and corresponding data as in Figure 7.7. Noise is 5% additive Gaussian. The images in the training data are randomly generated using a known probability distribution, and corresponding tomographic data (sinogram) are simulated with 5% additive Gaussian noise. This is a relatively small-scale problem, which allows us to also compute the conditional mean reconstruction using Markov chain Monte Carlo (MCMC) techniques (Section 3.5.2). The conditional mean reconstruction is useful since the learned iterative method approximates it. The same holds for learned post-processing since FBP is a linear sufficient statistic: see the discussion in Section 5.1.5. Hence, neither learned post-processing nor learned iterative will outperform the conditional mean, irrespective of the amount of training data and model capacity, that is, the conditional mean reconstruction serves as a theoretical limit for what one can recover.

In the realistic setting shown in Figure 7.9 (see also the summary in Table 7.5) the images are clinical CT scans. We use supervised training data consisting of about 2000 pairs of images from nine patients, and corresponding pre-log data are simulated with a Poisson noise corresponding to 10^4 incident photons per pixel before attenuation, which would correspond to a low-dose CT scan. Unfortunately we cannot compute the conditional mean as in the simplified setting, but we could compare against another

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

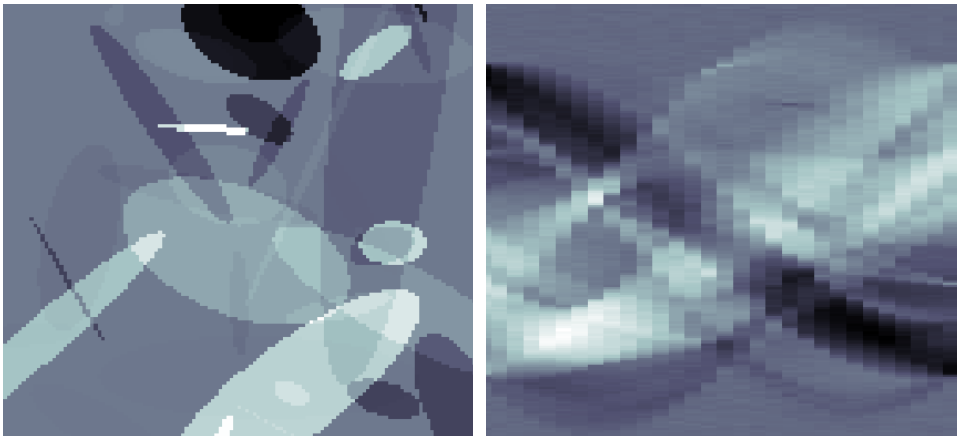
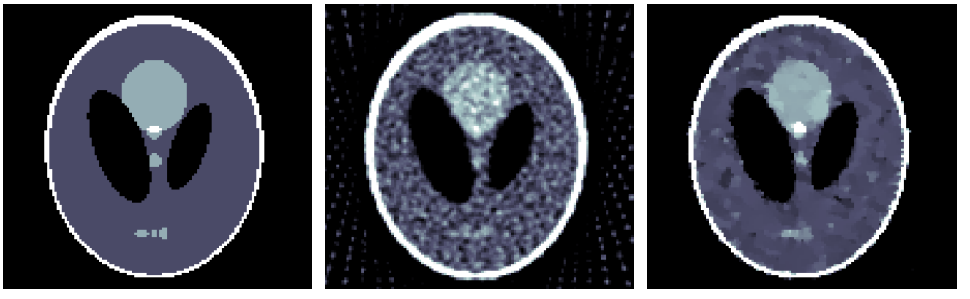


Figure 7.7. Example from supervised training data used to train the learned iterative and learned post-processing methods used in Figure 7.8.

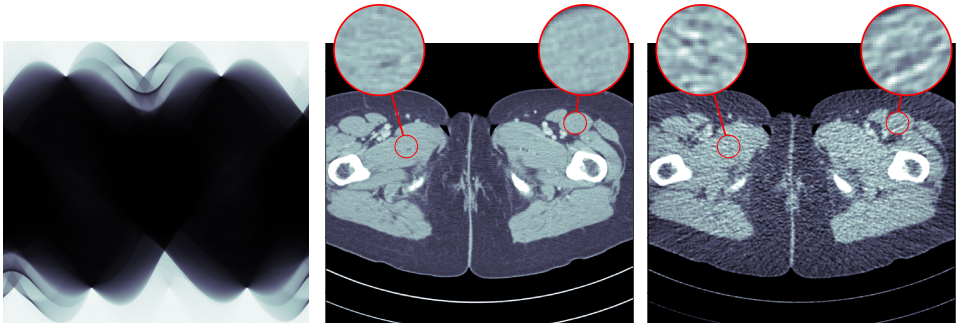


(a) ground truth (b) filtered backprojection (c) total variation

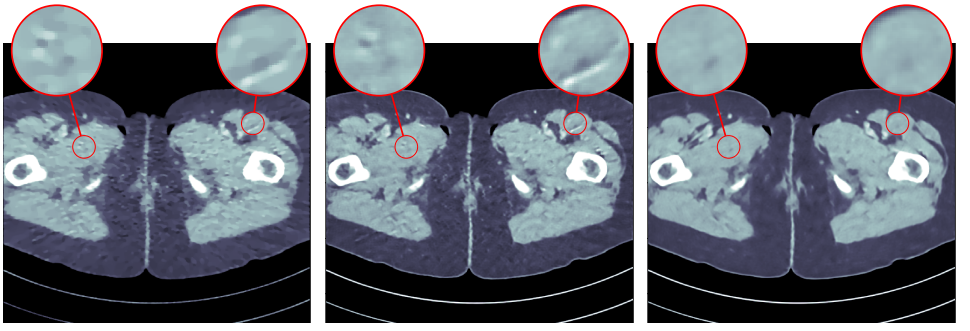
(d) learned post-processing (e) learned iterative (f) conditional mean

Figure 7.8. Reconstructions of the Shepp–Logan phantom using different methods. The window is set to $[0.1, 0.4]$, corresponding to the soft tissue of the modified Shepp–Logan phantom. We can see that the learned iterative method does indeed approximate the Bayes estimator, which here equals the conditional mean.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press



(a) data: pre-log sinogram (b) ground truth (c) filtered backprojection



(d) total variation (e) learned post-processing (f) learned iterative

Figure 7.9. Reconstructions of a 512×512 pixel human phantom along with two zoom-in regions indicated by small circles. The left zoom-in has a true feature whereas texture in the right zoom-in is uniform. The window is set to $[-200, 200]$ Hounsfield units. Among the methods tested, only the learned iterative method (learned primal–dual algorithm) correctly recovers these regions. In the others, the true feature in the left zoom-in is indistinguishable from other false features of the same size/contrast, and the right-zoom in has a streak artefact. The improvement that comes with using a learned iterative method thus translates into true clinical usefulness.

approach for computing the conditional mean, namely a sampling-based approach based on a conditional generative adversarial network (GAN) (see Section 5.2.1 and in particular Figure 7.16).

7.3.2. PAT reconstructions

Photoacoustic tomography (PAT) is a novel 'imaging from coupled physics' technique (Arridge and Scherzer 2012) that can obtain high-resolution three-dimensional *in vivo* images of absorbed optical energy by sensing laser-generated ultrasound (US) (Wang 2009, Beard 2011, Nie and Chen 2014,

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Table 7.5. Summary of results shown in Figures 7.8 and 7.9 where an SSIM score of 1 corresponds to a perfect match. Note that the learned iterative method (learned primal–dual algorithm) significantly outperforms TV regularization even when reconstructing the Shepp–Logan phantom. With respect to run-time, the learned iterative method involves calls to the forward operator, and is therefore slower than learned post-processing by a factor of ≈ 6. Compared with TV-regularized reconstruction, all learned methods are at least two orders of magnitude faster.

	Results for Figure 7.8			Results for Figure 7.9		
Method	PSNR (dB)	SSIM	Parameters	PSNR (dB)	SSIM	Parameters
Filtered backprojection	19.75	0.597	1	33.65	0.823	1
Total variation	28.06	0.928	1	37.48	0.946	1
Learned post-processing	29.20	0.943	10^7	41.92	0.941	10^7
Learned iterative	38.28	0.988	2.4×10^5	44.11	0.969	2.4×10^5
Conditional expectation	45.46	0.993	0	–	–	–

Valluru, Wilson and Willmann 2016, Zhou, Yao and Wang 2016, Xia and Wang 2014). In the setting considered here, data are collected as a time series on a two-dimensional sensor $Y = [\Gamma \subset \mathbb{R}^2] \times [0, T]$ on the surface of a domain $X = \Omega \subset \mathbb{R}^3$. Several methods exist for reconstruction, including filtered backprojection-type inversions of the spherical Radon transform and numerical techniques such as time-reversal. As in problems such as CT (Section 2.2.4) and MRI (Section 2.2.5), data subsampling may be employed to accelerate image acquisition, which leads consequently to the need for regularization to prevent noise propagation and artefact generation. The long reconstruction times ensuing from conventional iterative reconstruction algorithms have motivated consideration of machine learning methods.

The deep gradient descent (DGD) method (Hauptmann *et al.* 2018) for PAT is an example of a learned iterative method (see Section 5.1.4). The main aspects can be summarized as follows.

- Each iteration adds an update by combining measurement information delivered via the gradient $\nabla \mathcal{L}(g, \mathcal{A} f_k) = \mathcal{A}^*(\mathcal{A} f_k - g)$ with an image processing step

$$f_{k+1} = \mathrm{G}_{\theta_k}(\nabla \mathcal{L}(g, \mathcal{A} f_k), f_k), \tag{7.7}$$

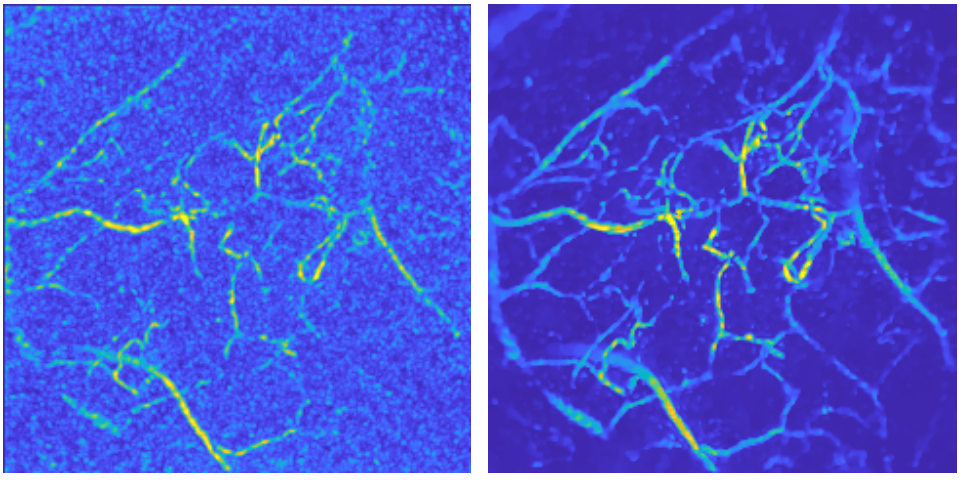
  where the layer operators G_{θ_k} correspond to convolutional neural networks (CNNs) with different, learned parameters θ_k but with the same architecture. The initialization for the iterations was the backprojection of the data $f_0 = \mathcal{A}^* g$.

- The training data were taken from the publicly available data from the ELCAP Public Lung Image Database.[9] The data set consists of 50 whole-lung CT scans, from which about 1200 volumes of vessel structures were segmented, and scaled up to the final target size of $80 \times 240 \times 240$. Out of these volumes 1024 were chosen as the ground truth f_{true} for the training and simulated limited-view, subsampled data, using the same measurement set-up as in the *in vivo* data. Precomputing the gradient information for each CNN took about 10 hours.

- Initial results from training on synthetic data showed a failure to effectively threshold the noise-like artefacts in the low absorption regions (see Figure 7.10). This effect was ameliorated by simulating the effect of the low absorbing background as a Gaussian random field with short spatial correlation length. The synthetic CT volumes with the added background were then used for the data generation, *i.e.* $g^i_{\text{back}} = \mathcal{A} f^i_{\text{back}} + \varepsilon$, whereas the clean volumes f_{true} were used as reference for the training.

[9] http://www.via.cornell.edu/databases/lungdb.html

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press



(a) DGD f_5 (b) TV reconstruction from full data, $\lambda = 2 \times 10^{-4}$

Figure 7.10. Reconstruction from real measurement data of a human palm, without adjustments of the training data. The images shown are top-down maximum intensity projections. (a) Result of the DGD trained on images without added background. (b) TV reconstruction obtained from fully sampled data.

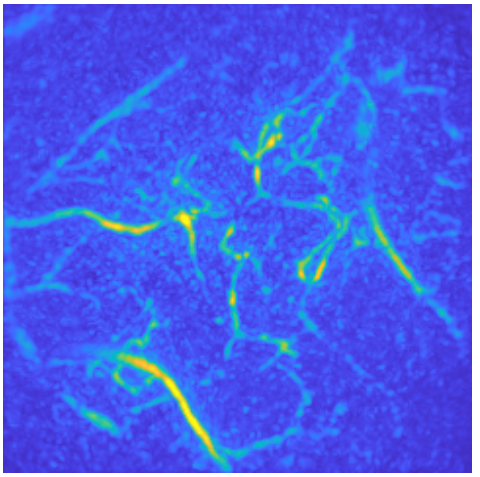
- The results were further improved using *transfer training* with a set of 20 (fully sampled) measurements of a human finger, wrist and palm from the same experimental system. To update the DGD an additional five epochs of training on the pairs $\{g_{\text{real}}, f_{\text{TV}}\}$ were performed with a reduced learning rate taking only 90 minutes. The effect of the updated DGD is shown in Figure 7.11.

7.4. Adversarial regularizer for CT

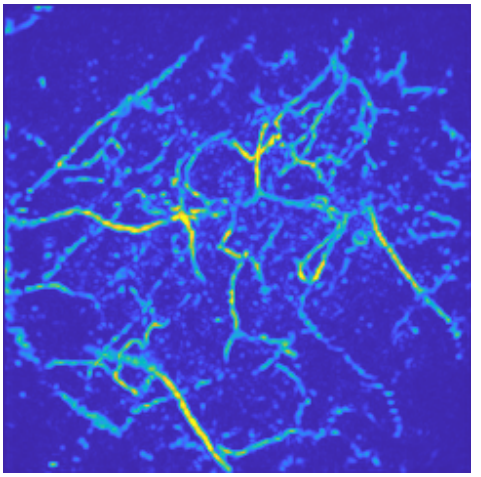
In Section 4.7 the concept of training a regularizer that is parametrized with a neural network in an adversarial manner has been presented. In what follows, we present numerical results as they are reported in Lunz *et al.* (2018). There, the performance of the adversarial regularizer for two-dimensional CT reconstruction is considered, that is, $\mathcal{A}$ is the ray transform as in (2.6). CT reconstruction is an application in which functional analytic inversion (see Section 2), and in particular the variational approach from Sections 2.5 and 2.6, is very widely used in practice. Here, it serves as a prototype inverse problem with non-trivial forward operator.

We compare the performance of TV-regularized reconstruction from Section 2.6 and (2.12), post-processing as in Section 5.1.5 (see in particular Gupta *et al.* 2018), regularization by denoising (RED) in Section 4.6 and

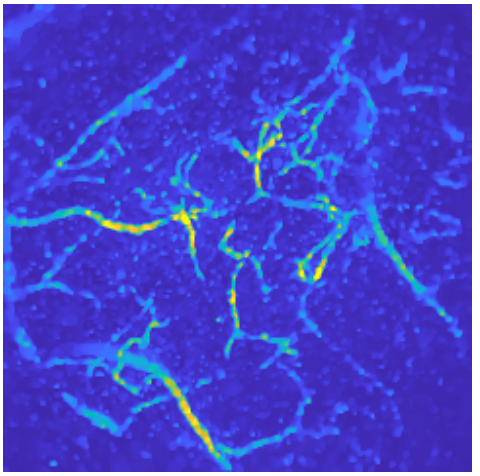
https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press



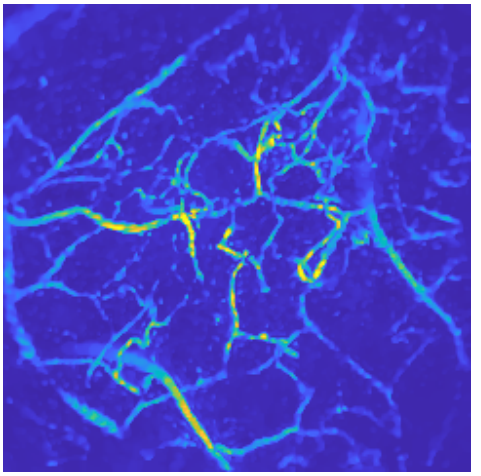
(a) initialization $f_0 = \mathcal{A}^*(g_{\text{re}})$



(b) updated DGD $\hat{f}_5$



(c) TV subsampled, $\lambda = 5 \times 10^{-5}$ and 20 iterations



(d) TV fully sampled, $\lambda = 2 \times 10^{-4}$ and 20 iterations

Figure 7.11. Example of real measurement data of a human palm. Volumetric images are shown using top-down maximum intensity projections. (a) Initialization from subsampled data, and (b) the DGD $G_{\hat{\theta}_k}$ after five iterations. (c) TV reconstruction of subsampled data with an emphasis on the data fit. (d) Reference TV reconstruction from fully sampled limited-view data. All TV reconstructions were computed with 20 iterations.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

Table 7.6. CT reconstruction on the LIDC dataset using various methods. Note that the learned post-processing and RED methods require training on supervised data, while the adversarial regularizer only requires training on unsupervised data.

	High noise		Low noise	
Method	PSNR (dB)	SSIM	PSNR (dB)	SSIM
Knowledge-driven				
Filtered backprojection	14.9	0.227	23.3	0.604
Total variation	27.7	0.890	30.0	0.924
Supervised				
Learned post-processing	31.2	0.936	33.6	0.955
RED	29.9	0.904	32.8	0.947
Unsupervised				
Adversarial regularizer	30.5	0.927	32.5	0.946

the adversarial regularizer from Section 4.7 on the LIDC/IDRI database (Armato *et al.* 2011) of lung scans.

We used a simple eight-layer convolutional neural network with a total of four average pooling layers of window size 2×2, leaky ReLU ($\alpha = 0.1$) activations and two final dense layers for all experiments with the adversarial regularizer algorithm. Training and test measurements have been simulated by taking the ray transform of the two-dimensional CT slices, adding Gaussian white noise, and under-sampling the data by storing only 30 angles in the forward operator. Results are reported in Table 7.6 and Figure 7.12.

In Table 7.6 we see that TV is outperformed by the learned regularization techniques by a large margin. The reconstructions achieved by the adversarial regularizer are at least as good in visual quality as those obtained with supervised machine learning methods, despite having used unsupervised data only. The ability of the adversarial regularizer to be trained in an unsupervised fashion could be interesting for its application to practical inverse problems, where ground truth data are often scarce or unavailable. Further results of the adversarial regularizer and discussion can be found in Lunz *et al.* (2018).

7.5. Deep learning for magnetic particle imaging (MPI)

MPI is an imaging modality based on injecting ferromagnetic nanoparticles, which are then transported by the blood flow. Reconstructing the resulting spatial distribution $c(x)$ of those nanoparticles is based on exploiting the non-linear magnetization behaviour of ferromagnetic nanoparticles (Gleich and Weizenecker 2005).

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

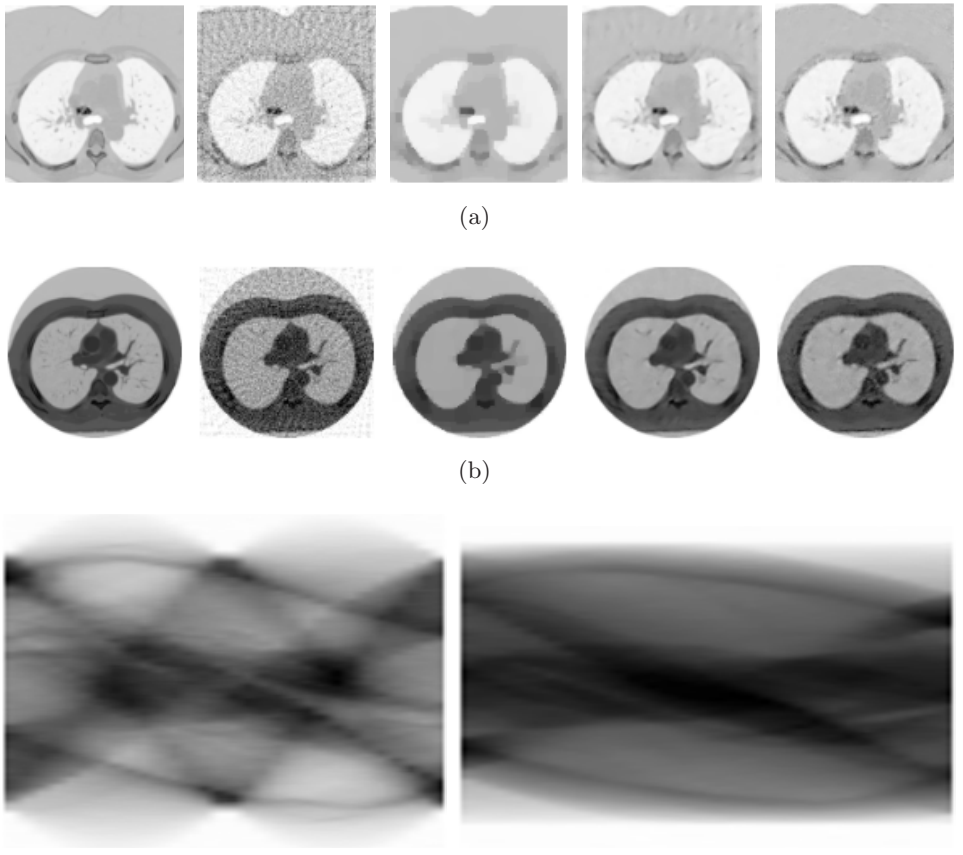


Figure 7.12. Exemplar CT reconstructions on the LIDC dataset under low-noise corruption. (a, b) Left to right: ground truth, FBP, TV, post-processing and adversarial regularization. (c,d) Data (CT sinograms): (c) data used for reconstructions in (a); (d) data used for reconstructions in (b).

More precisely, one applies a magnetic field, which is a superposition of a static gradient field, which generates a field-free point (FFP), and a highly dynamic spatially homogeneous field, which moves the FFP in space. The magnetic moment of the nanoparticles in the neighbourhood of the field-free point will oscillate, generating an electromagnetic field whose voltages can be measured by so-called receive coils. The time-dependent measurements $v_\ell(t)$ in the receive coils are the data for the inversion process, *i.e.* for reconstructing $c(x)$.

MPI benefits from a high temporal resolution and a potentially high spatial resolution which makes it suitable for several *in vivo* applications, such as imaging blood flow (Weizenecker *et al.* 2009, Khandhar *et al.* 2017),

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

instrument tracking (Haegele *et al.* 2012) and guidance (Salamon *et al.* 2016), flow estimation (Franke *et al.* 2017), cancer detection (Yu *et al.* 2017) and treatment by hyperthermia (Murase *et al.* 2015). However, real-time applications are still far from being realized; also, the mathematical foundation of such dynamic inverse problems (see Schmitt and Louis 2002, Hahn 2015, Schuster, Hahn and Burger 2018) is just developing.

Due to the non-magnetic coating of the nanoparticles, which largely suppresses particle–particle interactions, MPI is usually modelled by a linear Fredholm integral equation of the first kind describing the relationship between particle concentration and the measured voltage. After subtracting the voltage induced by the applied magnetic field one obtains a measured signal in the ℓth receive coil as

$$y_\ell(t) = S_\ell c(t) := \int_\Omega c(x)\ s_\ell(x,t)\,\mathrm{d}t,$$

where s_ℓ denotes the kernel of the linear operator. Combining the measurements of all receive coils yields – after discretization – a linear system of equations $Sc = g$. Typically, the rows of S are normalized, resulting in the final form of the linearized inverse problem denoted by

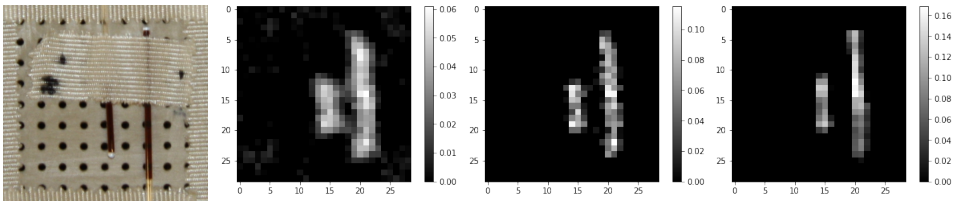
$$\mathbf{A}\, c = g. \tag{7.8}$$

This is a coarse simplification of the physical set-up, which neglects non-linear magnetization effects of the nanoparticles as well as the non-homogeneity of the spatial sensitivity of the receive coils and also the small but non-negligible particle–particle interactions. Hence this is a perfect set-up for exploiting the potential of neural networks for matching complex and high-dimensional non-linear models.

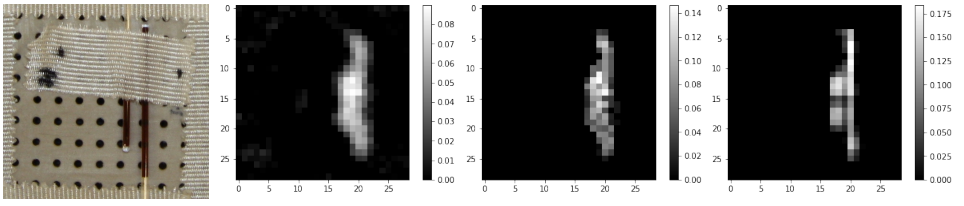
We test the capability of the deep imaging prior approach to improving image reconstruction obtained by standard Tikhonov regularization. For the experiments we use datasets generated by the Bruker preclinical MPI system at the University Medical Center, Hamburg–Eppendorf.

We use the deep image prior network introduced by Ulyanov *et al.* (2018), specifically their U-Net architecture. Our implementation is based on TensorFlow (Abadi *et al.* 2015) and Keras (Chollet *et al.* 2015), and has the following specifications. Between the encoder and decoder part of the U-Net our skip connection has four channels. The convolutional encoder goes from the input to 32, 32, 64 and 128 channels, each with strides of 2×2 and filters of size 3×3. Then the convolutional decoder has the mirrored architecture plus first a resize-nearest-neighbour layer to reach the desired output shape and second an additional ReLU convolutional layer with filters of size 1. The number of channels of this last layers is three for data set 1 (DS1) to accommodate three slices (three two-dimensional scans, one above another) of a two-dimensional phantom centred at the central slice of the three. The

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press



(a) phantom (4 mm) (b) Kaczmarz (c) ℓ_1-regularization (d) DIP



(e) phantom (2 mm) (f) Kaczmarz (g) ℓ_1-regularization (h) DIP

Figure 7.13. MPI reconstructions of two phantoms using different methods: (a)–(d) phantom with 4 mm distance between tubes containing ferromagnetic nanoparticles; (e)–(h) phantom with 2 mm distance. The methods used are Kaczmarz with L^2-discrepancy ($\tilde{\lambda} = 5 \times 10^{-4}$), ℓ_1-regularization ($\tilde{\lambda} = 5 \times 10^{-3}$) and DIP ($\eta = 5 \times 10^{-5}$) for both cases. Photos of phantoms taken by T. Kluth at the University Medical Center, Hamburg–Eppendorf.

input of the network is given by a fixed Gaussian random input of size $1 \times 32 \times 32$.

For comparison with our deep inverse prior MPI reconstructions, we also compute sparse and classical Tikhonov reconstructions. We produce the Tikhonov reconstruction, usually associated with the minimization of the functional

$$\|\mathbf{A}\, c - g\|^2 + \lambda \|c\|^2, \tag{7.9}$$

via the algebraic reconstruction technique (Kaczmarz) as generalized to allow for the constraint $x \geq 0$ by Dax (1993). We produce the sparsity reconstruction, usually associated with the minimization of the functional

$$\|\mathbf{A}\, c - g\|^2 + \lambda \|c\|_1, \tag{7.10}$$

by simply implementing this functional in TensorFlow and minimizing it via gradient descent. In the end we set all negative entries to 0.

We start by presenting direct comparisons of the Kaczmarz, sparsity and DIP reconstructions in Figure 7.13. Beneath each image we state the parameters we used for the reconstruction $\tilde{\lambda} = \|\mathbf{A}\|_F^2 \lambda$, where $\|\cdot\|_F$ denotes the

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

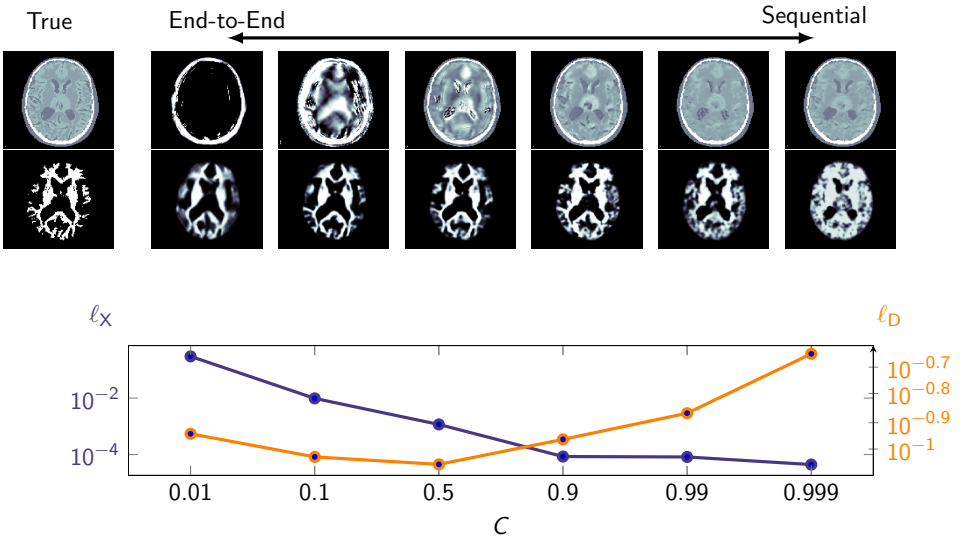


Figure 7.14. Joint tomographic reconstruction and segmentation of grey matter. Images shown using a $[-100, 100]$ HU window and segmentation using a $[0, 1]$ window. The choice $C = 0.9$ seems to be a good compromise for good reconstruction and segmentation, so clearly it helps to use a loss that includes the reconstruction and not only the task.

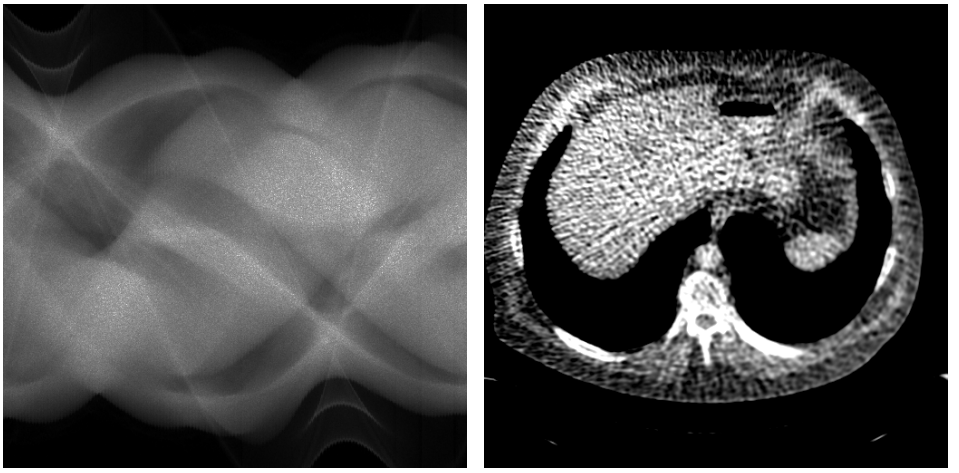
Frobenius norm and λ is the regularization parameter as used in (7.9) or (7.10) and η the learning rate used in training the network. For DIP we always used early stopping after 1000 optimization steps. The images started to deteriorate slowly for more iterations. For implementation details, as well as further numerical examples also showing the limitation of the DIP approach, see Dittmer *et al.* (2018).

7.6. Task-based reconstruction

We demonstrate the framework of Section 6.1 on joint tomographic image reconstruction and segmentation of white brain matter. $\mathcal{R}_\theta$ is given by a learned primal–dual method (Adler and Öktem 2018*b*), which incorporates a knowledge-based model for how data are generated into its architecture, and $\mathcal{T}_\phi$ is given by a U-Net (Ronneberger, Fischer and Brox 2015).

Some results are shown in Figure 7.14. Note in particular that (perhaps surprisingly) the 'best' segmentation is not obtained by a fully end-to-end approach: instead they are obtained when the reconstruction loss is included as a regularizer. Furthermore, it is clear that the reconstruction obtained for $C = 0.9$ over-emphasizes image features relevant for the task, for example white–grey matter contrast. This clearly 'helps' the task and also visually shows the image features used by the joint approach.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press



(a) (b)

Figure 7.15. Test data: (a) subset of CT data from an ultra-low-dose three-dimensional helical scan and (b) the corresponding FBP reconstruction. Images are shown using a display window set to $[-150, 200]$ Hounsfield units.

7.7. Clinical image guided decision making

We show how to compute an estimator relevant for uncertainty quantification in the context of CT image reconstruction. As a practical example, we will compute a CT reconstruction from ultra-low-dose data (Figure 7.15(a)). The aim is to identify a feature (a potential tumour) and then seek to estimate the likelihood of its presence.

Formalizing the above, let Δ denote the difference in mean intensity in the reconstructed image between a region encircling the feature and the surrounding organ, which in our example is the liver. The feature is said to 'exist' whenever Δ is bigger than a certain threshold, say 10 Hounsfield units.

To use posterior sampling, start by computing the conditional mean image (top left in Figure 7.16) by sampling from the posterior using the conditional Wasserstein GAN approach in Section 5.2.1. There is a 'dark spot' in the liver (a possible tumour) and a natural clinical question is to statistically test for the presence of this feature. To do this, compute Δ for a number of samples generated by posterior sampling, which is the same 1000 samples used to compute the conditional mean. We estimate the probability p that $\Delta > 10$ Hounsfield units from the resulting histogram in Figure 7.17 and clearly $p > 0.95$, indicating that the 'dark spot' feature exists with at least 95% significance. This is confirmed by the ground truth image (Figure 7.17(a)). The conditional mean image also under-estimates Δ, whose

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

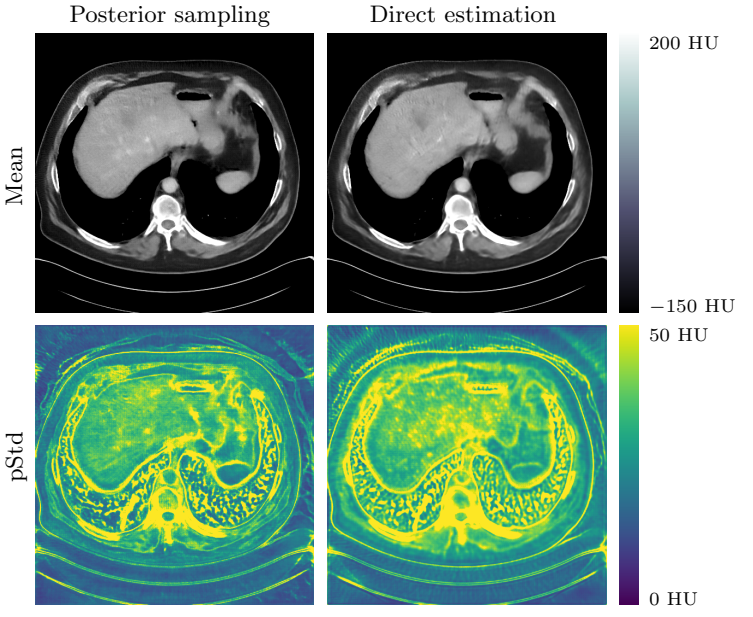
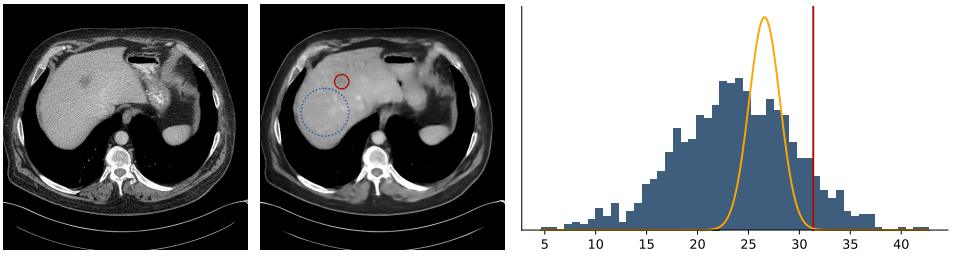


Figure 7.16. Conditional mean and pointwise standard deviation (pStd) computed from test data (Figure 7.15) using posterior sampling (Section 5.2.1) and direct estimation (Section 5.1.6).



(a) (b) (c)

Figure 7.17. (b) Suspected tumour (red) and reference region (blue) shown in the sample posterior mean image. (c) Average contrast differences between the tumour and reference region. The histogram is computed by posterior sampling applied to test data (Figure 7.15); the yellow curve is from direct estimation (Section 5.1.6), and the true value is the red threshold. (a) The normal dose image that confirms the presence of the feature.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

true value is the vertical line in Figure 7.17(c). This is to be expected since the prior introduces a bias towards homogeneous regions, a bias that decreases as the noise level decreases.

To perform the above analysis using direct estimation, start by computing the conditional mean image from the same ultra-low-dose data using direct estimation. As expected, the resulting image (top right in Figure 7.16) shows a 'dark spot' in the liver. Now, designing and training a neural network that directly estimates the distribution of Δ is unfeasible in a general setting. However, as shown in Section 5.1.6, this is possible if one assumes pixels are independent of each other. The estimated distribution of Δ is the curve in Figure 7.17 and we get $p > 0.95$, which is consistent with the result obtained using posterior sampling. The direct estimation approach is based on assuming independent pixels, so it will significantly underestimate the variance. In contrast, the approach based on posterior sampling seems to give a more realistic estimate of the variance.

8. Conclusions and outlook

8.1. Summary

In this survey we have tried to capture the state of the art in the still relatively young and fast-emerging field of machine learning approaches for solving inverse problems.

Our journey has taken us from more familiar applications of data-driven methods, such as dictionary learning (Section 4.4), bilevel learning (Section 4.3) and learning Markov random field-type regularizers (Section 4.3.2) to recent advances in using deep neural networks to solve inverse problems (Sections 4.6–4.10 and 5.1). These approaches are surveyed together with a brief account of the underlying mathematical setting (Sections 1–3), and their performance in some key applications is shown in Section 7. Taken together, we hope this will convince the reader that inverse problems can profit from data-driven methods. This claim is further strengthened by showing how data-driven methods can be used to compensate for inaccuracies in the forward model (Section 6.2) and how one can adapt the reconstruction to a specific task (Section 6.1).

The examples in Section 7 clearly show that some of these methods are very promising, regarding both output quality and computational feasibility. Strong empirical evidence suggests that using problem-specific deep neural networks that include knowledge-driven models outperform purely knowledge- or data-driven approaches. In contrast, there is little rigorous mathematical theory supporting these empirical observations, but the results clearly show that it is worth the effort to develop the necessary theory.

Below are some further key observations we believe are worth pointing out regarding the role of deep learning in solving inverse problems.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

The functional analytic and Bayesian viewpoints. The way deep learning is used for solving an inverse problem depends on whether one adopts the functional analytic or the Bayesian viewpoint.

Within the functional analytic viewpoint, a deep neural network is simply a parametrized family of operators, and learning amounts to calibrating the parameters against example data by minimizing some appropriate loss function.

In Bayesian inversion, a deep neural network corresponds to a statistical decision rule, so methods from deep learning constitute a computational framework for statistical decision making in high dimensions. For example, many of the estimators that have previously been computationally unfeasible are now computable: for example, the conditional mean seems to be well approximated by learned iterative schemes (Section 5.1.4). Likewise, a trained generative network can be used to sample from the posterior (Section 5.2) in a computationally feasible manner, as shown in Section 5.2.

Computational feasibility. Essentially all methods from Bayesian inversion and many from functional analytic regularization are computationally very demanding. Those techniques based on unrolling an iterative scheme (Section 4.9.1) are for designing a deep neural network that approximates a computationally demanding operator, such as one that solves a large-scale optimization problem.

The training of such a deep neural network may take quite some time, but once it is trained, it is fast to evaluate. In some sense, this is a way to redistribute the computational burden from the execution to the training.

Handling lack of training data. Inverse problems in the sciences and engineering often have little training data compared to the dimensionality of the model parameter. Furthermore, it is impractical to have a method that requires retraining as soon as the measurement protocol changes. This becomes an issue is medical imaging where data in multi-centre studies is typically acquired using different CT or MRI scanners.

For these reasons, black-box machine learning algorithms (Section 7.1) are not suitable for solving such inverse problems. On the other hand, in these inverse problems there is often a knowledge-driven model for how data are generated and it is important to integrate this information into the data-driven method. Learned iterative schemes (Section 5.1.4) employ a deep neural network that embeds this model for data into its architecture.

Encoding a priori *information.* In functional analytic regularization, much of the theoretical research (Section 2) has focused on finding optimal convergence rates as the noise level tends to zero. Likewise, theoretical research in Bayesian inversion (Section 3) focuses on contraction rates for certain classes of priors and in deriving an asymptotic closed-form characterization

of the posterior distribution as the noise level tends to zero. Here, the regularization functional (in functional analytic regularization) and the prior distribution (in Bayesian inversion) primarily act as a regularizers.

The above viewpoint does not acknowledge the potential that lies in encoding knowledge about the true model parameter into the regularization functional or prior. Furthermore, in applications data are fixed with some given noise level, and there is little, if any, guidance from the above theory on which regularizer or prior to select in such a setting. Empirical evidence suggests that instead of hand-crafting a regularization functional or a prior, one can learn it from example data. This allows us to pick up information related to the inverse problem that is difficult, if not impossible, to account for otherwise.

Unrolling. A key technique in many approaches for using deep learning to solve inverse problems is to construct problem-specific deep neural network architectures by unrolling an iterative scheme, as outlined in Section 4.9.1.

This technique allows us to use compressed sensing theory to derive properties for certain classes of deep neural networks, for example those resulting from multi-layer convolutional sparse coding (ML-CSC) (Section 4.4.2). Next, as shown in Section 4.9, the same technique is also useful in accelerating the evaluation of computationally demanding operators, such as those that solve large-scale optimization problems. Finally, unrolling is also used to embed a knowledge-driven forward operator and the adjoint of its derivative into a deep neural network that seeks to approximate an estimator (Section 5.1).

The above principle of unrolling can be used in a much wider context than solving an optimization problem or an inverse problem. It can be seen as constructing a deep neural network architecture for approximating an operator that is given implicitly through an iterative scheme. Hence, as pointed out in Section 4.9.4, unrolling establishes a link between numerical analysis and deep learning.

8.2. Outlook

We identify several interesting directions for future research in the context of inverse problems and machine learning.

8.2.1. Approximation and convergence properties of deep inversion

We believe a key component of future research is to analyse the mathematical–statistical approximation and convergence properties of inversion strategies that use deep neural networks, for example statistical recovery guarantees and generalization limits, as well as bounds on the number of training samples necessary for reaching prescribed accuracies, and estimates

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

for uncertainty in terms of stability properties and statistical confidence statements for the algorithms used.

Convergence and stability properties of denoising by deep neural networks can be analysed using techniques from sparse signal processing, as in Papyan *et al.* (2017). Likewise, techniques from applied harmonic analysis can be used to analyse approximation properties of feed-forward deep neural networks, as in Bölcskei *et al.* (2019). This paper establishes a connection between the complexity of a function class in $L^2(\mathbb{R}^d)$ and the complexity (model capacity) of feed-forward deep neural networks approximating functions from this class to within a prescribed accuracy. A specific focus is on function classes in $L^2(\mathbb{R}^d)$ that are optimally approximated by general affine systems, which include a wide range of representation systems from applied harmonic analysis such as wavelets, ridgelets, curvelets, shearlets, α-shearlets and, more generally, α-molecules. The central result in Bölcskei *et al.* (2019) is that feed-forward deep neural networks achieve the optimum approximation properties of all affine systems combined with minimal connectivity and memory requirements.

None of these papers, however, consider deep neural networks in the context of inverse problems.

8.2.2. Robustness against adversarial attacks

Sensitivity towards adversarial attacks is a known issue that has mostly been studied in the context of classification (Szegedy *et al.* 2014); see also the surveys by Chakraborty *et al.* (2018) and Akhtar and Mian (2018). However, little work has been done regarding adversarial stability of reconstruction methods for solving inverse problems that are based on deep neural networks.

One recent work along these lines is that of Antun *et al.* (2019), who extend the approach in Szegedy *et al.* (2014) to the case of regression. The idea is to perturb the model parameter in a way that is hard to distinguish, yet the data from the perturbed model parameter have a large influence on the reconstruction. The perturbation (adversarial example) is computed by solving an optimization and, in contrast to classification, different optimization problems can be constructed to test for different types of instabilities. The adversarial stability test in Antun *et al.* (2019) is demonstrated on MRI image reconstruction. The paper tests two fully learned approaches given by Zhu *et al.* (2018) and Schlemper *et al.* (2018), two learned post-processing approaches given by Jin *et al.* (2017) and Yang *et al.* (2018*b*), and finally the learned iterative approach of Hammernik *et al.* (2018). All approaches show adversarial instability, which could come as a surprise for the learned iterative approach that seeks to approximate a conditional mean that is known to be stable (Section 3.2.2). One reason for this could be that the learned iterative approach of Hammernik *et al.* (2018), which is based on variational

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

networks, has a model capacity too limited to properly approximate the conditional mean.

On a final note, a key element in Antun *et al.* (2019) is that data from the perturbation model parameter is noise-free. It is known that adding white noise to the input helps against adversarial attacks for classifiers (Cohen, Rosenfeld and Kolter 2019). Moreover, in inverse problems one always has noisy data, so it remains unclear whether the computed perturbation in Antun *et al.* (2019) actually acts as an adversarial example when noise is added.

Clearly, theory for robustness against adversarial attacks in the context of inverse problems is very much an emerging field.

8.2.3. Theory for learned iterative reconstruction

More specifically, the theory of statistical regularization applied to learned iterative methods in Section 5.1.4 is fairly incomplete, especially when interested in theoretical guarantees in the presence of empirical distributions for the data and model parameter. This will require studying estimates of the posterior in a non-asymptotic setting. Another key element is to estimate the generalization gap for learned iterative methods. Here one could consider theory for empirical Bayes methods, but current results focus on analysing Bayesian inversion methods where hyper-parameters defining a hierarchical prior are selected from data (Knapik, Szabó, van der Vaart and van Zanten 2016, Szabó, van der Vaart and van Zanten 2013).

8.2.4. 'Convergence' of training data

For all approaches where supervised training data are used, there is a discrepancy between theoretical error estimates in the infinite-dimensional setting and the practical case of finite-dimensional training data used in presented data-driven inversion approaches. For instance, error estimates are needed between solutions of a neural network trained on finitely many samples (that describe a particular empirical distribution) and solutions trained with infinitely many samples from a joint distribution.

8.2.5. Bespoke neural network architectures for inverse problems

In many inverse problems the model parameter space X and the data space Y are not the same, and in particular the data g often live in non-Euclidean spaces. On the other hand, existing data-driven inversion models, as discussed in this survey, which make use of neural networks as data-driven parametrizations, usually employ off-the-shelf network architectures such as U-Net, for instance. For future research it would be interesting to investigate neural network architectures that are specifically designed as mappings between non-Euclidean spaces. Some developments along these lines can be found in the paper by Bronstein *et al.* (2017), who investigate how the

notion of a CNN can be generalized to non-Euclidean structures such as graphs and manifolds.

The above is also closely related to work in developing neural network architectures that are equivariant and/or invariant to certain transformations given as group action. This is highly relevant when one seeks to solve inverse problems whose solutions enjoy such equivariance and/or invariance. Examples of work in this direction are those of Esteves, Allen-Blanchette, Makadia and Daniilidis (2017), Zhao *et al.* (2018), Weiler *et al.* (2018) and Veeling *et al.* (2018), but none of this is pursued in the context of inverse problems.

Another feature of inverse problems is that one can often prescribe how singularities in data are related to those in the model parameter (Section 6.3). Hence it is natural to seek network architectures that encode not only the forward operator but also such a relation. This is likely to further improve the robustness and generalization properties.

8.2.6. Continuous notion of neural network architectures

Some of the recent attempts to build a continuous framework for neural networks have been touched upon in Section 6.2.1. Continuous formulations to neural networks make them amenable to the rich toolkit of functional analysis and theoretical results as outlined in Section 2. Moreover, starting with a continuous model such as a partial differential equation, for instance, may give rise to new discretizations and new neural network architectures – a development that has clearly happened before in mathematical imaging (*e.g.* Perona and Malik 1990).

8.2.7. Theoretical guarantees for learning-to-optimize approaches

If a neural network is used to approximate and consequently computationally speed up a knowledge-driven approach (*e.g.* the learning-to-optimize methods in Section 4.9), it is important to understand the error committed by such an approximation. What is the correct notion of such an approximation error? How does it depend on the training set and the network architecture?

Acknowledgements

This article builds on lengthy discussions and long-standing collaborations with a large number of people. These include Jonas Adler, Sebastian Banert, Martin Benning, Marta Betcke, Luca Calatroni, Juan Carlos De Los Reyes, Andreas Hauptmann, Lior Horesh, Bangti Jin, Iasonas Kokkinos, Felix Lucka, Sebastian Lunz, Thomas Pock, Tuomo Valkonen and Olivier Verdier. The authors are moreover grateful to the following people for proofreading the manuscript and providing valuable feedback on its content

and structure: Andrea Aspri, Martin Benning, Matthias Ehrhardt, Barbara Kaltenbacher, Yury Korolev, Mike McCann, Erkki Somersalo and Michael Unser.

SA acknowledges support from EPSRC grant EP/M020533/1. OÖ acknowledges support from the Swedish Foundation of Strategic Research grant AM13-004 and the Alan Turing Institute. CBS acknowledges support from the Leverhulme Trust project 'Breaking the non-convexity barrier', EPSRC grant EP/M00483X/1, EPSRC grant EP/N014588/1, the RISE projects CHiPS and NoMADS, the Cantab Capital Institute for the Mathematics of Information and the Alan Turing Institute.

Acronyms

ADMM alternating direction method of multipliers
AutoMap automated transform by manifold approximation
BFGS Broyden–Fletcher–Goldfarb–Shanno
CG conjugate gradient
CNN convolutional neural network
CSC convolutional sparse coding
CT computed tomography
DGD deep gradient descent
DIP deep inverse prior
FBP filtered backprojection
FFP field-free point
FoE Field of Experts
GAN generative adversarial network
ICA independent component analysis
ICTV infimal-convolution total variation
ISTA Iterative Soft-Thresholding Algorithm
KL Kullback–Leibler
LISTA Learned Iterative Soft-Thresholding Algorithm
MAP maximum *a posteriori*
MCMC Markov chain Monte Carlo
ML-CSC multi-layer convolutional sparse coding
ML-EM maximum likelihood expectationmaximization
MPI magnetic particle imaging
MRF Markov random field
MRI magnetic resonance imaging
NETT neural network Tikhonov
P^3 Plug-and-Play Prior
PAT photoacoustic tomography
PCA principal component analysis
PDE partial differential equation

PDHG primal–dual hybrid gradient
PET positron emission tomography
PG proximal gradient
PoE Product of Experts
PSNR peak signal-to-noise ratio
RED regularization by denoising
RIP restricted isometry property
R-TLS regularized total least-squares
SGD stochastic gradient descent
SPECT single photon emission computed tomography
SSIM structural similarity index
SVD singular value decomposition
TGV total generalized variation
TLS total least-squares
TV total variation

Appendices

A. Optimization of convex non-smooth functionals

Suppose in general that we want to optimize a problem defined as the sum of two parts,

$$\min_{f\in X}[\mathcal{J}(f) := \Phi(f) + \mathcal{S}(f)], \tag{A.1}$$

where $\Phi\colon X \to \mathbb{R}$ is a continuously differentiable convex function, and $\mathcal{S}\colon X \to \mathbb{R}$ is convex but possibly *non-differentiable*. We say that the combined function is *convex non-smooth*.

A.1. Proximal methods

First we define a *proximal operator* for a functional $\mathcal{S}\colon X \to \mathbb{R}$:

$$\text{prox}_{\mathcal{S}}(h) := \arg\min_{f\in X}\left[\frac{1}{2}\|f-h\|^2 + \mathcal{S}(h)\right]. \tag{A.2}$$

Clearly, if $\mathcal{S}$ is differentiable, then $p = \text{prox}_{\mathcal{S}}(h)$ satisfies

$$h - p = \nabla\mathcal{S}\,|_{f=p}. \tag{A.3}$$

When $\mathcal{S}$ is non-differentiable, we instead have

$$h - p \in \partial\mathcal{S}\,|_{f=p}, \tag{A.4}$$

where $\partial\mathcal{S}(f)$ is the *subdifferential* of $\mathcal{S}$. This allows us to write a formal expression for (A.2) as

$$\text{prox}_{\mathcal{S}}(h) := (\text{id} + \partial\mathcal{S})^{-1}(h). \tag{A.5}$$

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

A.2. Proximal gradient method for inverse problems

For inverse problems, $\Phi(f)$ corresponds to the data discrepancy $\mathcal{L}(\mathcal{A}(f), g)$ and $\mathcal{S}(f)$ to the regularization functional.

Now consider the minimization of $f \mapsto \mathcal{J}_\lambda(f)$ in (2.10). Defining $[\partial\mathcal{A}(f)]$ to be the Fréchet derivative of $\mathcal{A}$ at f, then, exploiting the first-order necessary conditions for such minima, we have

$$0 \in [\partial\mathcal{A}(f)]^*(\mathcal{A}(f) - g) + \lambda\partial\mathcal{S}(f), \tag{A.6}$$

which after multiplying both sides with τ, adding f on both sides and reordering terms yields the fixed-point condition for a minimizer:

$$f = \operatorname{prox}_{\tau\lambda\mathcal{S}}(f - \tau[\partial\mathcal{A}(f)]^*(\mathcal{A}(f) - g)). \tag{A.7}$$

The step length is given by $0 < \tau < 2/L$, where L is the Lipschitz constant of $\nabla\Phi$ (Combettes and Wajs 2005). For linear inverse problems, L can be approximated by the largest eigenvalue of $\mathcal{A}^*\mathcal{A}$, *i.e.* the square of the largest singular value of the forward operator $\mathcal{A}$. For non-linear problems $L(f)$ is the square of the largest singular value of $[\partial\mathcal{A}(f)]$ and thus changes over iteration. We have the following examples.

- Multivariate Gaussian noise $e \sim \mathcal{N}(0, \Gamma_e)$: the likelihood is

  $$\mathcal{L}(\mathcal{A}(f), g) = \|g - \mathcal{A}f\|^2_{\Gamma_e},$$

  and

  $$f^{(n+1)} \leftarrow f^{(n)} + \tau\lambda\,\mathcal{A}^*\,\Gamma_e^{-1}(g - \mathcal{A}f^{(n)}).$$

- Poisson noise $g = \text{Poisson}(\mathcal{A}f)$: the likelihood is

  $$\mathcal{L}(\mathcal{A}(f), g) = \int_\Omega g \ln \mathcal{A}f - \mathcal{A}f + g - \ln g,$$

  and

  $$f^{(n+1)} \leftarrow f^{(n)} + \tau\,\mathcal{A}^*\left(\mathbf{1} - \frac{g}{\mathcal{A}f^{(n)}}\right).$$

The related iteration scheme to (A.7), which can also be derived by minimizing surrogate functionals (Daubechies *et al.* 2004) or by the method of quadratic relaxation, yields the following algorithm.

Algorithm A.1 (generalized gradient projection method).

Choose f^0 and iterate for $k > 0$.

(1) Choose τ_k, *e.g.* $\tau_k = \tau$ constant for all k.
(2) Determine $v^k = f^k - \tau_k[\partial\mathcal{A}(f)]^*(\mathcal{A}(f) - g)$.
(3) Determine $f^{k+1} = \operatorname{prox}_{\tau_k\lambda\mathcal{S}}(v^k)$.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

This version, along with several accelerated variants incorporating a step size selection or primal–dual iterations, has been studied intensively (Bredies, Lorenz and Maass 2009, Nesterov 2007, Figueiredo *et al.* 2007).

The convergence proofs of such methods are typically based on rephrasing the first-order necessary condition.

Theorem A.1. Assume that $\mathcal{A} : X \to Y$ is Fréchet-differentiable and assume that $\mathcal{S} : X \to \mathbb{R}$ is proper and convex. Then, a (first-order) necessary condition for a minimizer $\hat{f}$ of $f \mapsto \mathcal{J}_\lambda(f)$ in (2.10) is given by

$$\langle [\partial \mathcal{A}(f)]^*(\mathcal{A}(f) - g), h - f\rangle_X \geq \mathcal{S}(f) - \mathcal{S}(h) \quad \text{for all } h \in X,$$

which is equivalent to

$$\langle [\partial \mathcal{A}(f)]^*(\mathcal{A}(f) - g), f\rangle + \mathcal{S}(f) = \min_{h \in X} \langle [\partial \mathcal{A}(f)]^*(\mathcal{A}(f) - g), h\rangle + \mathcal{S}(h).$$

This characterization motivates the definition of an auxiliary functional

$$D_\lambda(f^k) := \lambda(\mathcal{S}(f^k) - \mathcal{S}(f^{k+1})) + \langle [\partial \mathcal{A}(f^k)]^*(\mathcal{A}(f^k) - g), f^k - f^{k+1}\rangle, \quad \text{(A.8)}$$

which is decreased during the iteration and whose minimizing f allows an estimate in terms of the Bregman distance related to $\mathcal{S}$.

A.3. Iterative Soft-Thresholding Algorithm (ISTA)

The success of proximal methods usually depends on finding a fast 'trick' for performing the projection for a given functional $\mathcal{S}(f)$. One notable such method is for the L^1-functional $\mathcal{S}(f) = \lambda\|f\|_1$ whereby the proximal operator is the soft-thresholding (or shrinkage operator), denoted by

$$S_\alpha(z) := \begin{cases} z - \alpha & \text{if } z \geq \alpha, \\ 0 & \text{if } |z| \leq \alpha, \\ z + \alpha & \text{if } z \leq -\alpha. \end{cases} \quad \text{(A.9)}$$

We arrive at the following split method, known as the Iterative Soft-Thresholding Algorithm (ISTA) (Daubechies *et al.* 2004, Figueiredo *et al.* 2007):

$$\begin{aligned} &\text{gradient descent} && f^{(n+1/2)} \leftarrow f^{(n)} - \tau \nabla \Phi(f), \\ &\text{thresholding} && f^{(n+1)} \leftarrow S_{\tau\lambda}(f^{(n+1/2)}). \end{aligned}$$

(Note that the threshold is the product of τ and λ.)

Now consider applying this principle to the problem of minimizing Tikhonov functionals of type $f \mapsto \mathcal{J}_\lambda(f)$ defined in (2.10). In this case $\Phi(f) := \frac{1}{2}\|\mathcal{A}(f) - g\|^2$ and the necessary first-order condition for a minimizer is given by

$$0 \in \mathcal{A}^*(\mathcal{A} f - g) + \lambda \partial \|f\|_1.$$

Multiplying with an arbitrary real positive real number τ and adding f plus

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

rearranging yields

$$f - \tau\,\mathcal{A}^*(\mathcal{A}\,f - g) \in f + \tau\lambda\partial\|f\|_1.$$

Using (A.9) to invert the term on the right-hand side yields

$$S_{\tau\lambda}(f - \lambda\,\mathcal{A}^*(\mathcal{A}\,f - g)) = f.$$

Hence this is a fixed-point condition, which is a necessary condition for all minimizers of $f \mapsto \mathcal{J}_\lambda(f)$. Turning the fixed-point condition into an iteration scheme yields

$$\begin{aligned} f^{k+1} &= S_{\tau\lambda}(f^k - \tau\,\mathcal{A}^*(\mathcal{A}\,f^k - g)) \\ &= S_{\tau\lambda}((\mathrm{id} - \tau\,\mathcal{A}^*\,\mathcal{A})f^k + \tau\,\mathcal{A}^*\,g). \end{aligned} \tag{A.10}$$

A.4. Alternating direction method of multipliers (ADMM)
Consider solving (A.1) as a constrained problem,

$$\hat{f} = \arg\min_{f,v}[\Phi(f) + \mathcal{S}(v)] \quad \text{such that } f = v,$$

making use of the augmented Lagrangian with dual (adjoint) variable u,

$$\begin{aligned} \mathcal{J}(f, v, u) &= \Phi(f) + \mathcal{S}(v) + \langle u, f - v\rangle + \frac{\beta}{2}\|f - v\|_2^2 \\ &= \Phi(f) + \mathcal{S}(v) + \frac{\beta}{2}\|f - v + \frac{1}{\beta}u\|_2^2 - \frac{1}{2\beta}\|u\|_2^2, \end{aligned} \tag{A.11}$$

which results in the sequential update sequence

$$f^{(n+1)} \leftarrow \mathrm{prox}_{(1/\beta)\Phi}\left[v^{(n)} - \frac{1}{\beta}u^{(n)}\right], \tag{A.12}$$

$$v^{(n+1)} \leftarrow \mathrm{prox}_{(1/\beta)\,\mathcal{S}}\left[f^{(n+1)} + \frac{1}{\beta}u^{(n)}\right], \tag{A.13}$$

$$u^{(n+1)} \leftarrow u^{(n)} + \beta(f^{(n+1)} - v^{(n+1)}). \tag{A.14}$$

B. The Wasserstein 1-distance

Let X be a measurable separable Banach space and $\mathscr{P}_X$ the space of probability measures on X. The Wasserstein 1-distance $\mathcal{W}$: $\mathscr{P}_X \times \mathscr{P}_X \to \mathbb{R}$ is a metric on $\mathscr{P}_X$ that can be defined as (Villani 2009, Definition 6.1)

$$\mathcal{W}(p, q) := \inf_{\mu\in\Pi(p,q)} \mathbb{E}_{(\mathbb{f},\mathbb{h})\sim\mu}[\|\mathbb{f} - \mathbb{h}\|_X] \quad \text{for } p, q \in \mathscr{P}_X. \tag{B.1}$$

In the above, $\Pi(p, q) \subset \mathscr{P}_{X\times X}$ denotes the family of joint probability measures on $X \times X$ that has p and q as marginals. Note also that we assume

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

$\mathscr{P}_X$ only contains measures where the Wasserstein distance takes finite values (Wasserstein space): see Villani (2009, Definition 6.4) for the formal definition.

The Wasserstein 1-distance in (B.1) can be rewritten using the Kantorovich–Rubinstein dual characterization (Villani 2009, Remark 6.5, p. 95), resulting in

$$\mathcal{W}(p,q) = \sup_{\substack{\mathrm{D}\colon X\to\mathbb{R} \\ \mathrm{D}\in\mathrm{Lip}(X)}} \{\mathbb{E}_{\mathbb{f}\sim q}[\mathrm{D}(\mathbb{f})] - \mathbb{E}_{\mathbb{h}\sim p}[\mathrm{D}(\mathbb{h})]\} \quad \text{for } p,q \in \mathscr{P}_X. \tag{B.2}$$

Here, $\mathrm{Lip}(X)$ denotes real-valued 1-Lipschitz maps on X, that is,

$$\mathrm{D} \in \mathrm{Lip}(X) \iff |\,\mathrm{D}(f_1) - \mathrm{D}(f_2)| \leq \|f_1 - f_2\|_X \quad \text{for all } f_1, f_2 \in X.$$

The above constraint can be hard to enforce in (B.2) as is, so following Gulrajani *et al.* (2017) and Adler and Lunz (2018) we prefer the *gradient characterization*

$$\mathrm{D} \in \mathrm{Lip}(X) \iff \|\partial \mathrm{D}(f)\|_{X^*} \leq 1 \quad \text{for all } f \in X,$$

where ∂ indicates the Fréchet derivative and X^* is the dual space of X. In our setting, X is an L_2-space, which is a Hilbert space so $X^* = X$, and the Fréchet derivative becomes the (Hilbert space) gradient of D.

REFERENCES[10]

M. Abadi *et al.* (2015), TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from https://www.tensorflow.org.

B. Adcock and A. C. Hansen (2016), 'Generalized sampling and infinite-dimensional compressed sensing', *Found. Comput. Math.* **16**, 1263–1323.

J. Adler and S. Lunz (2018), Banach Wasserstein GAN. In *Advances in Neural Information Processing Systems 31 (NIPS 2018)* (S. Bengio *et al.*, eds), Curran Associates, pp. 6754–6763.

J. Adler and O. Öktem (2017), 'Solving ill-posed inverse problems using iterative deep neural networks', *Inverse Problems* **33**, 124007.

J. Adler and O. Öktem (2018*a*), Deep Bayesian inversion: Computational uncertainty quantification for large scale inverse problems. arXiv:1811.05910

J. Adler and O. Öktem (2018*b*), 'Learned primal–dual reconstruction', *IEEE Trans. Medical Imaging* **37**, 1322–1332.

J. Adler, S. Lunz, O. Verdier, C.-B. Schönlieb and O. Öktem (2018), Task adapted reconstruction for inverse problems. arXiv:1809.00948

L. Affara, B. Ghanem and P. Wonka (2018), Supervised convolutional sparse coding. arXiv:1804.02678

[10] The URLs cited in this work were correct at the time of going to press, but the publisher and the authors make no undertaking that the citations remain live or are accurate or appropriate.

https://doi.org/10.1017/S0962492919000059 Published online by Cambridge University Press

S. Agapiou, S. Larsson and A. M. Stuart (2013), 'Posterior contraction rates for the Bayesian approach to linear ill-posed inverse problems', *Stoch. Process. Appl.* **123**, 3828–3860.

S. Agapiou, A. M. Stuart and Y. X. Zhang (2014), 'Bayesian posterior contraction rates for linear severely ill-posed inverse problems', *J. Inverse Ill-Posed Problems* **22**, 297–321.

H. K. Aggarwal, M. P. Mani and M. Jacob (2019), 'MoDL: Model-based deep learning architecture for inverse problems', *IEEE Trans. Medical Imaging* **38**, 394–405.

M. Aharon, M. Elad and A. M. Bruckstein (2006), 'K-SVD: An algorithm for designing of over-complete dictionaries for sparse representation', *IEEE Trans. Signal Process.* **54**, 4311–4322.

A. Ahmed, M. Aly, J. Gonzalez, S. Narayanamurthy and A. Smola (2012), Scalable inference in latent variable models. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining (WSDM '12)*, pp. 123–132.

N. Akhtar and A. Mian (2018), Threat of adversarial attacks on deep learning in computer vision: A survey. arXiv:1801.00553

W. K. Allard, G. Chen and M. Maggioni (2012), 'Multi-scale geometric methods for data sets, II: Geometric multi-resolution analysis', *Appl. Comput. Harmon. Anal.* **32**, 435–462.

D. Allman, A. Reiter and M. A. L. Bell (2018), 'Photoacoustic source detection and reflection artifact removal enabled by deep learning', *IEEE Trans. Medical Imaging* **37**, 1464–1477.

L. Ambrosio, N. Fusco and D. Pallara (2000), *Functions of Bounded Variation and Free Discontinuity Problems*, Oxford University Press.

H. Andrade-Loarca, G. Kutyniok, O. Öktem and P. Petersen (2019), Extraction of digital wavefront sets using applied harmonic analysis and deep neural networks. arXiv:1901.01388

M. Andrychowicz, M. Denil, S. Gomez, M. Hoffman, D. Pfau, T. Schaul and N. de Freitas (2016), Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems 29 (NIPS 2016)* (D. D. Lee *et al.*, eds), Curran Associates, pp. 3981–3989.

V. Antun, F. Renna, C. Poon, B. Adcock and A. C. Hansen (2019), On instabilities of deep learning in image reconstruction: Does AI come at a cost? arXiv:1902.05300v1

L. Ardizzone, J. Kruse, S. Wirkert, D. Rahner, E. W. Pellegrini, R. S. Klessen, L. Maier-Hein, C. Rother and U. Köthe (2018), Analyzing inverse problems with invertible neural networks. arXiv:1808.04730

M. Argyrou, D. Maintas, C. Tsoumpas and E. Stiliaris (2012), Tomographic image reconstruction based on artificial neural network (ANN) techniques. In *2012 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, pp. 3324–3327.

M. Arjovsky, S. Chintala and L. Bottou (2017), Wasserstein generative adversarial networks. In *34th International Conference on Machine Learning (ICML '17)*, pp. 214–223.

S. Armato, G. McLennan, L. Bidaut, M. McNitt-Gray, C. Meyer, A. Reeves, B. Zhao, D. Aberle, C. Henschke, E. Hoffman *et al.* (2011), 'The lung image database consortium (LIDC) and image database resource initiative (IDRI): A completed reference database of lung nodules on CT scans', *Med. Phys.* **38**, 915–931.

S. R. Arridge and O. Scherzer (2012), 'Imaging from coupled physics', *Inverse Problems* **28**, 080201.

S. R. Arridge and J. C. Schotland (2009), 'Optical tomography: Forward and inverse problems', *Inverse Problems* **25**, 123010.

A. Aspri, S. Banert, O. Öktem and O. Scherzer (2018), A data-driven iteratively regularized Landweber iteration. arXiv:1812.00272

P. Auer, M. Herbster and M. K. Warmuth (1996), Exponentially many local minima for single neurons. In *8th International Conference on Neural Information Processing Systems (NIPS)*, MIT Press, pp. 316–322.

T. Bai, H. Yan, X. Jia, S. Jiang, G. Wang and X. Mou (2017), Volumetric computed tomography reconstruction with dictionary learning. In *14th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine (Fully3D 2017)*.

A. B. Bakushinskii (1984), 'Remarks on choosing a regularization parameter using the quasi-optimality and ratio criterion', *USSR Comput. Math. Math. Phys.* **24**, 181–182.

G. Bal, F. Chung and J. Schotland (2016), 'Ultrasound modulated bioluminescence tomography and controllability of the radiative transport equation', *SIAM J. Math. Anal.* **48**, 1332–1347.

G. Balakrishnan, A. Zhao, M. R. Sabuncu, J. Guttag and A. V. Dalca (2019), 'VoxelMorph: A learning framework for deformable medical image registration', *IEEE Trans. Imaging*, to appear. arXiv:1809.05231

L. Baldassarre, Y.-H. Li, J. Scarlett, B. Gözcü, I. Bogunovic and V. Cevher (2016), 'Learning-based compressive subsampling', *IEEE J. Selected Topics Signal Process.* **10**, 809–822.

A. Banerjee, X. Guo and H. Wang (2005), 'On the optimality of conditional expectation as a Bregman predictor', *IEEE Trans. Inform. Theory* **51**, 2664–2669.

S. Banert, A. Ringh, J. Adler, J. Karlsson and O. Öktem (2018), Data-driven nonsmooth optimization. arXiv:1808.00946

A. R. Barron (1994), 'Approximation and estimation bounds for artificial neural networks', *Machine Learning* **14**, 115–133.

F. Baus, M. Nikolova and G. Steidl (2014), 'Fully smoothed L1-TV models: Bounds for the minimizers and parameter choice', *J. Math. Imaging Vision* **48**, 295–307.

H. H. Bauschke and P. L. Combettes (2017), *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, second edition, CMS Books in Mathematics, Springer.

P. Beard (2011), 'Biomedical photoacoustic imaging', *Interface Focus* **1**, 602–631.

A. Beck and M. Teboulle (2009), 'A fast iterative shrinkage–thresholding algorithm for linear inverse problems', *SIAM J. Imaging Sci.* **2**, 183–202.

A. Beck, S. Sabach and M. Teboulle (2016), 'An alternating semiproximal method for nonconvex regularized structured total least squares problems', *SIAM J. Matrix Anal. Appl.* **37**, 1129–1150.
S. Becker, Y. Zhang and A. A. Lee (2018), Geometry of energy landscapes and the optimizability of deep neural networks. arXiv:1805.11572
Y. Bengio, P. Simard and P. Frasconi (1994), 'Learning long-term dependencies with gradient descent is difficult', *IEEE Trans. Neural Networks* **5**, 157–166.
M. Benning and M. Burger (2018), Modern regularization methods for inverse problems. In *Acta Numerica*, Vol. 27, Cambridge University Press, pp. 1–111.
M. Benning, C. Brune, M. Burger and J. Müller (2013), 'Higher-order TV methods: Enhancement via Bregman iteration', *J. Sci. Comput.* **54**, 269–310.
F. Benvenuto, A. L. Camera, C. Theys, A. Ferrari, H. Lantéri and M. Bertero (2008), 'The study of an iterative method for the reconstruction of images corrupted by Poisson and Gaussian noise', *Inverse Problems* **24**, 035016.
J. O. Berger (1985), *Statistical Decision Theory and Bayesian Analysis*, second edition, Springer.
R. Berinde, A. C. Gilbert, P. Indyk, H. Karloff and M. J. Strauss (2008), Combining geometry and combinatorics: A unified approach to sparse signal recovery. In *46th Annual Allerton Conference on Communication, Control, and Computing*, pp. 798–805.
M. Bertero and P. Boccacci (1998), *Introduction to Inverse Problems in Imaging*, Institute of Physics Publishing.
M. Bertero, H. Lantéri and L. Zanni (2008), Iterative image reconstruction: A point of view. In *Interdisciplinary Workshop on Mathematical Methods in Biomedical Imaging and Intensity-Modulated Radiation (IMRT)* (Y. Censor *et al.*, eds), pp. 37–63.
D. Bertsekas (1999), *Nonlinear Programming*, second edition, Athena Scientific.
J. Besag (1974), 'Spatial interaction and the statistical analysis of lattice systems', *J. Royal Statist. Soc.* B **36**, 192–236.
J. Besag and P. J. Green (1993), 'Spatial statistics and Bayesian computation', *J. Royal Statist. Soc.* B **55**, 25–37.
M. Betancourt (2017), A conceptual introduction to Hamiltonian Monte Carlo. arXiv:1701.02434
L. Biegler, G. Biros, O. Ghattas, M. Heinkenschloss, D. Keyes, B. Mallick, L. Tenorio, B. van Bloemen Waanders, K. Willcox and Y. Marzouk (2011), *Large-Scale Inverse Problems and Quantification of Uncertainty*, Vol. 712 of Wiley Series in Computational Statistics, Wiley.
N. Bissantz, T. Hohage, A. Munk and F. Ruymgaart (2007), 'Convergence rates of general regularization methods for statistical inverse problems and applications', *SIAM J. Numer. Anal.* **45**, 2610–2636.
A. Blake and A. Zisserman (1987), *Visual Reconstruction*, MIT Press.
D. M. Blei, A. Küçükelbir and J. D. McAuliffe (2017), 'Variational inference: A review for statisticians', *J. Amer. Statist. Assoc.* **112** (518), 859–877.
I. R. Bleyer and R. Ramlau (2013), 'A double regularization approach for inverse problems with noisy data and inexact operator', *Inverse Problems* **29**, 025004.
T. Blumensath (2013), 'Compressed sensing with nonlinear observations and related nonlinear optimization problems', *IEEE Trans. Inform. Theory* **59**, 3466–3474.

T. Blumensath and M. E. Davies (2008), 'Iterative thresholding for sparse approximations', *J. Fourier Anal. Appl.* **14**, 629–654.

N. Bochkina (2013), 'Consistency of the posterior distribution in generalized linear inverse problems', *Inverse Problems* **29**, 095010.

Y. E. Boink, S. A. van Gils, S. Manohar and C. Brune (2018), 'Sensitivity of a partially learned model-based reconstruction algorithm', *Proc. Appl. Math. Mech.* **18**, e201800222.

H. Bölcskei, P. Grohs, G. Kutyniok and P. Petersen (2019), 'Optimal approximation with sparsely connected deep neural networks', *SIAM J. Math. Data Sci.* **1**, 8–45.

J. F. Bonnans and D. Tiba (1991), 'Pontryagin's principle in the control of semilinear elliptic variational inequalities', *Appl. Math. Optim.* **23**, 299–312.

E. Bostan, U. S. Kamilov and L. Waller (2018), 'Learning-based image reconstruction via parallel proximal algorithm', *IEEE Signal Process. Lett.* **25**, 989–993.

R. Boţ and E. Csetnek (2015), 'On the convergence rate of a forward–backward type primal–dual splitting algorithm for convex optimization problems', *Optimization* **64**, 5–23.

R. Boţ and C. Hendrich (2013), 'A Douglas–Rachford type primal–dual method for solving inclusions with mixtures of composite and parallel-sum type monotone operators', *SIAM J. Optim.* **23**, 2541–2565.

L. Bottou, F. E. Curtis and J. Nocedal (2018), 'Optimization methods for large-scale machine learning', *SIAM Review* **60**, 223–311.

S. Boyd, N. Parikh, E. Chu, B. Peleato and J. Eckstein (2011), 'Distributed optimization and statistical learning via the alternating direction method of multipliers', *Found. Trends Mach. Learn.* **3**, 1–122.

K. Bredies and T. Valkonen (2011), Inverse problems with second-order total generalized variation constraints. In *9th International Conference on Sampling Theory and Applications (SampTA 2011)*.

K. Bredies, K. Kunisch and T. Pock (2011), 'Total generalized variation', *SIAM J. Imaging Sci.* **3**, 492–526.

K. Bredies, K. Kunisch and T. Valkonen (2013), 'Properties of l^1-TGV^2: The one-dimensional case', *J. Math. Anal. Appl.* **398**, 438–454.

K. Bredies, D. A. Lorenz and P. Maass (2009), 'A generalized conditional gradient method and its connection to an iterative shrinkage method', *Comput. Optim. Appl.* **42**, 173–193.

L. Breiman, L. Le Cam and L. Schwartz (1965), 'Consistent estimates and zero-one sets', *Ann. Math. Statist.* **35**, 157–161.

H. Bristow, A. Eriksson and S. Lucey (2013), Fast convolutional sparse coding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2013)*, pp. 391–398.

M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam and P. Vandergheynst (2017), 'Geometric deep learning: Going beyond Euclidean data', *IEEE Signal Process. Mag.* **34**, 18–42.

A. M. Bruckstein, D. L. Donoho and M. Elad (2009), 'From sparse solutions of systems of equations to sparse modeling of signals and images', *SIAM Review* **51**, 34–18.

J. Bruna and S. Mallat (2013), 'Invariant scattering convolution networks', *IEEE Trans. Pattern Anal. Mach. Intel.* **35**, 1872–1886.

A. Buades, B. Coll and J.-M. Morel (2005), A non-local algorithm for image denoising. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005)*, Vol. 2, pp. 60–65.

T. A. Bubba, G. Kutyniok, M. Lassas, M. März, W. Samek, S. Siltanen and V. Srinivasan (2018), Learning the invisible: A hybrid deep learning–shearlet framework for limited angle computed tomography. arXiv:1811.04602

S. Bubeck (2015), 'Convex optimization: Algorithms and complexity', *Found. Trends Mach. Learn.* **8**, 231–357.

A. Buccini, M. Donatelli and R. Ramlau (2018), 'A semiblind regularization algorithm for inverse problems with application to image deblurring', *SIAM J. Sci. Comput.* **40**, A452–A483.

T. Bui-Thanh, K. Willcox and O. Ghattas (2008), 'Model reduction for large-scale systems with high-dimensional parametric input space', *SIAM J. Sci. Comput.* **30**, 3270–3288.

M. Burger and F. Lucka (2014), 'Maximum *a posteriori* estimates in linear inverse problems with log-concave priors are proper Bayes estimators', *Inverse Problems* **30**, 114004.

R. H. Byrd, G. M. Chin, J. Nocedal and Y. Wu (2012), 'Sample size selection in optimization methods for machine learning', *Math. Program.* **134**, 127–155.

C. L. Byrne (2008), *Applied Iterative Methods*, Peters/CRC Press.

L. Calatroni (2015), New PDE models for imaging problems and applications. PhD thesis, University of Cambridge.

L. Calatroni, C. Cao, J. C. De los Reyes, C.-B. Schönlieb and T. Valkonen (2016), 'Bilevel approaches for learning of variational imaging models', *Variational Methods* **18**, 252–290.

L. Calatroni, J. C. De los Reyes and C.-B. Schönlieb (2014), Dynamic sampling schemes for optimal noise learning under multiple nonsmooth constraints. In *26th IFIP Conference on System Modeling and Optimization (CSMO 2013)*, Springer, pp. 85–95.

L. Calatroni, J. C. De los Reyes and C.-B. Schönlieb (2017), 'Infimal convolution of data discrepancies for mixed noise removal', *SIAM J. Imaging Sci.* **10**, 1196–1233.

A. P. Calderón (1958), 'Uniqueness in the Cauchy problem for partial differential equations', *Amer. J. Math.* **80**, 16–36.

A. P. Calderón and A. Zygmund (1952), 'On the existence of certain singular integrals', *Acta Math.* **88**, 85.

A. P. Calderón and A. Zygmund (1956), 'On singular integrals', *Amer. J. Math.* **78**, 289–309.

D. Calvetti and E. Somersalo (2008), 'Hypermodels in the Bayesian imaging framework', *Inverse Problems* **24**, 034013.

D. Calvetti and E. Somersalo (2017), 'Inverse problems: From regularization to Bayesian inference', *WIREs Comput. Statist.* **10**, e1427.

D. Calvetti, B. Lewis and L. Reichel (2002), 'On the regularizing properties of the GMRES method', *Numer. Math.* **91**, 605–625.

D. Calvetti, E. Somersalo and A. Strang (2019), 'Hierarchical Bayesian models and sparsity: ℓ_2-magic', *Inverse Problems* **35**, 035003.
E. J. Candès and D. L. Donoho (2005), 'Continuous curvelet transform, I: resolution of the wavefront set', *Appl. Comput. Harmon. Anal.* **19**, 162–197.
E. J. Candès, L. Demanet and L. Ying (2007), 'Fast computation of Fourier integral operators', *SIAM J. Sci. Comput.* **29**, 2464–2493.
E. J. Candès, J. K. Romberg and T. Tao (2006), 'Robust uncertainty principles: Exact signal reconstruction from highly incomplete Fourier information', *IEEE Trans. Inform. Theory* **52**, 489–509.
M. Carriero, A. Leaci and F. Tomarelli (1996), A second order model in image segmentation: Blake & Zisserman functional. In *Variational Methods for Discontinuous Structures* (R. Serapioni and F. Tomarelli *et al.*), Springer, pp. 57–72.
I. Castillo and R. Nickl (2013), 'Nonparametric Bernstein–von Mises theorems in Gaussian white noise', *Ann. Statist.* **41**, 1999–2028.
I. Castillo and R. Nickl (2014), 'On the Bernstein–von Mises phenomenon for nonparametric Bayes procedures', *Ann. Statist.* **42**, 1941–1969.
A. Chakraborty, M. Alam, V. Dey, A. Chattopadhyay and D. Mukhopadhyay (2018), Adversarial attacks and defences: A survey. arXiv:1810.00069
A. Chambolle and P.-L. Lions (1997), 'Image recovery via total variation minimization and related problems', *Numer. Math.* **76**, 167–188.
A. Chambolle and T. Pock (2011), 'A first-order primal–dual algorithm for convex problems with applications to imaging', *J. Math. Imaging Vision* **40**, 120–145.
A. Chambolle and T. Pock (2016), An introduction to continuous optimization for imaging. In *Acta Numerica*, Vol. 25, Cambridge University Press, pp. 161–319.
A. Chambolle, M. Holler and T. Pock (2018), A convex variational model for learning convolutional image atoms from incomplete data. arXiv:1812.03077v1
T. F. Chan and J. Shen (2006), 'Image processing and analysis: Variational, PDE, wavelet, and stochastic methods', *BioMed. Engng OnLine* **5**, 38.
J. H. R. Chang, C.-L. Li, B. Póczos, B. V. K. V. Kumar and A. C. Sankaranarayanan (2017), One network to solve them all: Solving linear inverse problems using deep projection models. arXiv:1703.09912v1
B. Chen, K. Xiang, Z. Gong, J. Wang, and S. Tan (2018), 'Statistical iterative CBCT reconstruction based on neural network', *IEEE Trans. Medical Imaging* **37**, 1511–1521.
G. Chen and D. Needell (2016), 'Compressed sensing and dictionary learning', *Proc. Sympos. Appl. Math.* **73**, 201–241.
H. Chen, Y. Zhang, Y. Chen, J. Zhang, W. Zhang, H. Sun, Y. Lv, P. Liao, J. Zhou and G. Wang (2019), 'LEARN: Learned experts' assessment-based reconstruction network for sparse-data CT', *IEEE Trans. Medical Imaging* **37**, 1333–1347.
H. Chen, Y. Zhang, M. K. Kalra, F. Lin, Y. Chen, P. Liao, J. Zhou and G. Wang (2017*a*), 'Low-dose CT with a residual encoder–decoder convolutional neural network', *IEEE Trans. Medical Imaging* **36**, 2524–2535.
H. Chen, Y. Zhang, W. Zhang, P. Liao, K. Li, J. Zhou and G. Wang (2017*b*), Low-dose CT denoising with convolutional neural network. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pp. 143–146.

Y. Chen, T. Pock and H. Bischof (2012), Learning ℓ_1-based analysis and synthesis sparsity priors using bi-level optimization. In *Workshop on Analysis Operator Learning vs. Dictionary Learning (NIPS 2012)*.

Y. Chen, T. Pock, R. Ranftl and H. Bischof (2013), Revisiting loss-specific training of filter-based MRFs for image restoration. In *German Conference on Pattern Recognition (GCPR 2013)*, Vol. 8142 of Lecture Notes in Computer Science, Springer, pp. 271–281.

Y. Chen, R. Ranftl and T. Pock (2014), 'Insights into analysis operator learning: From patch-based sparse models to higher order MRFs', *IEEE Trans. Image Process.* **23**, 1060–1072.

Y. Chen, W. Yu and T. Pock (2015), On learning optimized reaction diffusion processes for effective image restoration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015)*, pp. 5261–5269.

F. Chollet *et al.* (2015), Keras: The Python Deep Learning library. https://keras.io

A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous and Y. LeCun (2015), The loss surfaces of multilayer networks. In *18th International Conference on Artificial Intelligence and Statistics (AISTATS 2015)*, pp. 192–204.

I. Y. Chun, X. Zheng, Y. Long and J. A. Fessler (2017), Sparse-view X-ray CT reconstruction using ℓ_1 regularization with learned sparsifying transform. In *14th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine (Fully3D 2017)*.

J. Chung and M. I. Espanol (2017), 'Learning regularization parameters for general-form Tikhonov', *Inverse Problems* **33**, 074004.

C. Clason, T. Helin, R. Kretschmann and P. Piiroinen (2018), Generalized modes in Bayesian inverse problems. arXiv:1806.00519

A. Cohen, W. Dahmen and R. DeVore (2009), 'Compressed sensing and best k-term approximation', *J. Amer. Math. Soc.* **22**, 211–231.

J. Cohen, E. Rosenfeld and J. Z. Kolter (2019), Certified adversarial robustness via randomized smoothing. arXiv:1902.02918v1

P. L. Combettes and J.-C. Pesquet (2011), Proximal splitting methods in signal processing. In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering* (H. H. Bauschke *et al.*, eds), Vol. 49 of Springer Optimization and its Applications, Springer, pp. 185–212.

P. L. Combettes and J.-C. Pesquet (2012), 'Primal–dual splitting algorithm for solving inclusions with mixtures of composite, Lipschitzian, and parallel-sum type monotone operators', *Set-Valued Var. Anal.* **20**, 307–330.

P. L. Combettes and V. R. Wajs (2005), 'Signal recovery by proximal forward–backward splitting', *Multiscale Model. Simul.* **4**, 1168–1200.

R. Costantini and S. Susstrunk (2004), Virtual sensor design. In *Electronic Imaging 2004*, International Society for Optics and Photonics, pp. 408–419.

A. Courville, I. Goodfellow and Y. Bengio (2017), *Deep Learning*, MIT Press.

G. R. Cross and A. K. Jain (1983), 'Markov random field texture models', *IEEE Trans. Pattern Anal. Mach. Intel.* **5**, 25–39.

G. Cybenko (1989), 'Approximation by superpositions of a sigmoidal function', *Math. Control Signals Syst.* **2**, 303–314.

C. O. da Luis and A. J. Reader (2017), Deep learning for suppression of resolution-recovery artefacts in MLEM PET image reconstruction. In *2017 IEEE*

Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), pp. 1–3.
K. Dabov, A. Foi, V. Katkovnik and K. Egiazarian (2007), 'Image denoising by sparse 3-D transform-domain collaborative filtering', *IEEE Trans. Image Process.* **16**, 2080–2095.
A. V. Dalca, G. Balakrishnan, J. Guttag and M. R. Sabuncu (2018), Unsupervised learning for fast probabilistic diffeomorphic registration. In *21st International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2018)* (A. F. Frangi *et al.*, eds), Vol. 11070 of Lecture Notes in Computer Science, Springer, pp. 729–738.
M. Dashti and A. M. Stuart (2017), The Bayesian approach to inverse problems. In *Handbook of Uncertainty Quantification* (R. Ghanem *et al.*, eds), Springer, chapter 10.
M. Dashti, K. J. H. Law, A. M. Stuart and J. Voss (2013), 'MAP estimators and their consistency in Bayesian nonparametric inverse problems', *Inverse Problems* **29**, 095017.
I. Daubechies, M. Defrise and C. De Mol (2004), 'An iterative thresholding algorithm for linear inverse problems with a sparsity constraint', *Commun. Pure Appl. Math.* **57**, 1413–1457.
I. Daubechies *et al.* (1991), *Ten Lectures on Wavelets*, Vol. 61 of CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM.
M. E. Davison (1983), 'The ill-conditioned nature of the limited angle tomography problem', *SIAM J. Appl. Math.* **43**, 428–448.
E. Davoli and P. Liu (2018), 'One dimensional fractional order TGV: Gamma-convergence and bi-level training scheme', *Commun. Math. Sci.* **16**, 213–237.
A. Dax (1993), 'On row relaxation methods for large constrained least squares problems', *SIAM J. Sci. Comput.* **14**, 570–584.
J. C. De los Reyes (2011), 'Optimal control of a class of variational inequalities of the second kind', *SIAM J. Control Optim.* **49**, 1629–1658.
J. C. De los Reyes (2015), *Numerical PDE-Constrained Optimization*, Springer.
J. C. De los Reyes and C.-B. Schönlieb (2013), 'Image denoising: Learning the noise model via nonsmooth PDE-constrained optimization', *Inverse Problems* **7**, 1183–1214.
J. C. De los Reyes, C.-B. Schönlieb and T. Valkonen (2016), 'The structure of optimal parameters for image restoration problems', *J. Math. Anal. Appl.* **434**, 464–500.
J. C. De los Reyes, C.-B. Schönlieb and T. Valkonen (2017), 'Bilevel parameter learning for higher-order total variation regularisation models', *J. Math. Imaging Vision* **57**, 1–25.
A. P. Dempster, N. M. Laird, D. B. Rubin *et al.* (1977), 'Maximum likelihood from incomplete data via the EM algorithm', *J. Royal Statist. Soc.* B **39**, 1–38.
E. Y. Derevtsov, A. V. Efimov, A. K. Louis and T. Schuster (2011), 'Singular value decomposition and its application to numerical inversion for ray transforms in 2D vector tomography', *J. Inverse Ill-Posed Problems* **19**, 689–715.
A. Diaspro, M. Schneider, P. Bianchini, V. Caorsi, D. Mazza, M. Pesce, I. Testa, G. Vicidomini and C. Usai (2007), Two-photon excitation fluorescence microscopy. In *Science of Microscopy* (P. W. Hawkes and J. C. H. Spence, eds), Vol. 2, Springer, Chapter 11, pp. 751–789.

S. Dittmer, T. Kluth, P. Maass and D. O. Baguer (2018), Regularization by architecture: A deep prior approach for inverse problems. arXiv:1812.03889

I. Dokmanić, J. Bruna, S. Mallat and M. de Hoop (2016), Inverse problems with invariant multiscale statistics. arXiv:1609.05502

K. Doksum (1974), 'Tail-free and neutral random probabilities and their posterior distributions', *Ann. Probab.* **2**, 183–201.

W. Dong, G. Shi, Y. Ma and X. Li (2015), 'Image restoration via simultaneous sparse coding: Where structured sparsity meets Gaussian scale mixture', *Internat. J. Comput. Vision* **114**, 217–232.

W. Dong, L. Zhang, G. Shi and X. Li (2013), 'Nonlocally centralized sparse representation for image restoration', *IEEE Trans. Image Process.* **22**, 1620–1630.

W. Dong, L. Zhang, G. Shi and X. Wu (2011), 'Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization', *IEEE Trans. Image Process.* **20**, 1838–1857.

D. L. Donoho, Y. Tsaig, I. Drori and J.-L. Starck (2012), 'Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit', *IEEE Trans. Inform. Theory* **58**, 1094–1121.

J. L. Doob (1948), Application of the theory of martingales. In *International Colloquium du CNRS: Probability Theory and its Application*, pp. 22–28.

F. Draxler, K. Veschgini, M. Salmhofer and F. Hamprecht (2018), 'Essentially no barriers in neural network energy landscape', *Proc. Mach. Learning Res.* **80**, 1309–1318.

Y. Drori and M. Teboulle (2014), 'Performance of first-order methods for smooth convex minimization: A novel approach', *Math. Program.* **145**, 451–482.

A. Durmus, E. Moulines and M. Pereyra (2018), 'Efficient Bayesian computation by proximal Markov chain Monte Carlo: When Langevin meets Moreau', *SIAM J. Imaging Sci.* **11**, 473–506.

W. E, J. Han and A. Jentzen (2017), Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. arXiv:1706.04702

J. Eckstein and D. Bertsekas (1992), 'On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators', *Math. Program.* **55**, 293–318.

M. Elad (2010), *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, Springer.

M. Elad and M. Aharon (2006), 'Image denoising via sparse and redundant representations over learned dictionaries', *IEEE Trans. Image Process.* **15**, 3736–3745.

Y. C. Eldar and G. Kutyniok (2012), *Compressed Sensing: Theory and Applications*, Cambridge University Press.

H. W. Engl, M. Hanke and A. Neubauer (2000), *Regularization of Inverse Problems*, Vol. 375 of Mathematics and its Applications, Springer.

H. W. Engl, K. Kunisch and A. Neubauer (1989), 'Convergence rates for Tikhonov regularisation of non-linear ill-posed problems', *Inverse Problems* **5**, 523.

C. Esteves, C. Allen-Blanchette, A. Makadia and K. Daniilidis (2017), Learning SO(3) equivariant representations with spherical CNNs. arXiv:1711.06721

S. N. Evans and P. B. Stark (2002), 'Inverse problems as statistics', *Inverse Problems* **18**, R1–R55.
V. Faber, A. I. Katsevich and A. G. Ramm (1995), 'Inversion of cone-beam data and helical tomography', *J. Inverse Ill-Posed Problems* **3**, 429–446.
C. Farabet, C. Couprie, L. Najman and Y. LeCun (2013), 'Learning hierarchical features for scene labeling', *IEEE Trans. Pattern Anal. Mach. Intel.* **35**, 1915–1929.
M. A. T. Figueiredo, R. D. Nowak and S. J. Wright (2007), 'Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems', *IEEE J. Selected Topics Signal Process.* **1**, 586–598.
C. E. Floyd (1991), 'An artificial neural network for SPECT image reconstruction', *IEEE Trans. Medical Imaging* **10**, 485–487.
M. Fornasier and H. Rauhut (2008), 'Iterative thresholding algorithms', *Appl. Comput. Harmon. Anal.* **25**, 187–208.
S. Foucart (2016), 'Dictionary-sparse recovery via thresholding-based algorithms', *J. Funct. Anal. Appl.* **22**, 6–19.
S. Foucart and H. Rauhut (2013), *A Mathematical Introduction to Compressive Sensing*, Applied and Numerical Harmonic Analysis, Birkhäuser.
C. Fox and S. Roberts (2012), 'A tutorial on variational Bayes', *Artif. Intel. Rev.* **38**, 85–95.
J. Franke, R. Lacroix, H. Lehr, M. Heidenreich, U. Heinen and V. Schulz (2017), 'MPI flow analysis toolbox exploiting pulsed tracer information: An aneurysm phantom proof', *Internat. J. Magnetic Particle Imaging* **3**, 1703020.
D. Freedman (1963), 'On the asymptotic behavior of Bayes estimates in the discrete case, I', *Ann. Math. Statist.* **34**, 1386–1403.
D. Freedman (1965), 'On the asymptotic behavior of Bayes estimates in the discrete case, II', *Ann. Math. Statist.* **36**, 454–456.
A. Frommer and P. Maass (1999), 'Fast CG-based methods for Tikhonov–Phillips regularization', *SIAM J. Sci. Comput.* **20**, 1831–1850.
L. Fu, T.-C. Lee, S. M. Kim, A. M. Alessio, P. E. Kinahan, Z. Chang, K. Sauer, M. K. Kalra and B. D. Man (2017), 'Comparison between pre-log and post-log statistical models in ultra-low-dose CT reconstruction', *IEEE Trans. Medical Imaging* **36**, 707–720.
C. Garcia-Cardona and B. Wohlberg (2017), Convolutional dictionary learning. arXiv:1709.02893
M. U. Ghani and W. C. Karl (2018), Deep learning-based sinogram completion for low-dose CT. In *2018 IEEE 13th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*.
S. Ghosal and N. Ray (2017), 'Deep deformable registration: Enhancing accuracy by fully convolutional neural net', *Pattern Recog. Lett.* **94**, 81–86.
S. Ghosal and A. W. van der Vaart (2017), *Fundamentals of Nonparametric Bayesian Inference*, Cambridge Series in Statistical and Probabilistic Mathematics, Cambridge University Press.
S. Ghosal, J. K. Ghosh and R. V. Ramamoorthi (1999), 'Posterior consistency of dirichlet mixtures in density estimation', *Ann. Statist.* **27**, 143–158.
S. Ghosal, J. K. Ghosh and A. W. van der Vaart (2000), 'Convergence rates of posterior distributions', *Ann. Statist.* **28**, 500–531.

G. Gilboa and S. Osher (2008), 'Nonlocal operators with applications to image processing', *Multiscale Model. Simul.* **7**, 1005–1028.

D. Gilton, G. Ongie and R. Willett (2019), Neumann networks for inverse problems in imaging. arXiv:1901.03707

M. Giordano and H. Kekkonen (2018), Bernstein–von Mises theorems and uncertainty quantification for linear inverse problems. arXiv:1811.04058v1

M. Girolami and B. Calderhead (2011), 'Riemann manifold Langevin and Hamiltonian Monte Carlo methods', *J. Royal Statist. Soc.* B **73**, 123–214.

R. Giryes, Y. C. Eldar, A. M. Bronstein and G. Sapiro (2017), Tradeoffs between convergence speed and reconstruction accuracy in inverse problems. arXiv:1605.09232v2

B. Gleich and J. Weizenecker (2005), 'Tomographic imaging using the nonlinear response of magnetic particles', *Nature* **435** (7046), 1214–1217.

G. H. Golub and C. F. Van Loan (1980), 'An analysis of the total least squares problem', *SIAM J. Numer. Anal.* **17**, 883–893.

G. H. Golub, P. C. Hansen and D. P. O'Leary (1999), 'Tikhonov regularization and total least squares', *SIAM J. Matrix Anal. Appl.* **21**, 185–194.

G. H. Golub, M. Heat and G. Wahba (1979), 'Generalized cross validation as a method for choosing a good ridge parameter', *Technometrics* **21**, 215–223.

A. N. Gomez, M. Ren, R. Urtasun and R. B. Grosse (2017), The reversible residual network: Backpropagation without storing activations. arXiv:1707.04585v1

D. Gong, Z. Zhang, Q. Shi, A. van den Hengel, C. Shen and Y. Zhang (2018), Learning an optimizer for image deconvolution. arXiv:1804.03368v1

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio (2014), Generative adversarial nets. In *Advances in Neural Information Processing Systems 27 (NIPS 2014)* (Z. Ghahramani *et al.*, eds), Curran Associates, pp. 2672–2680.

B. Gözcü, R. K. Mahabadi, Y.-H. Li, E. Ilcak, T. Çukur, J. Scarlett and V. Cevher (2018), 'Learning-based compressive MRI', *IEEE Trans. Medical Imaging* **37**, 1394–1406.

M. Grasmair, M. Haltmeier and O. Scherzer (2008), 'Sparse regularization with ℓ_q penalty term', *Inverse Problems* **24**, 055020.

P. J. Green, K. Łatuszyśński, M. Pereyra and C. P. Robert (2015), 'Bayesian computation: A summary of the current state, and samples backwards and forwards', *Statist. Comput.* **25**, 835–862.

A. Greenleaf, Y. Kurylev, M. Lassas and G. Uhlmann (2007), 'Full-wave invisibility of active devices at all frequencies', *Comm. Math. Phys.* **275**, 749–789.

K. Gregor and Y. LeCun (2010), Learning fast approximations of sparse coding. In *27th International Conference on Machine Learning (ICML '10)*, pp. 399–406.

R. Gribonval and M. Nikolova (2018), On Bayesian estimation and proximity operators. arXiv:1807.04021

S. Gu, L. Zhang, W. Zuo and X. Feng (2014), Weighted nuclear norm minimization with application to image denoising. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2014)*, pp. 2862–2869.

S. Gu, W. Zuo, Q. Xie, D. Meng, X. Feng and L. Zhang (2015), Convolutional sparse coding for image super-resolution. In *IEEE International Conference on Computer Vision (ICCV 2015)*, pp. 1823–1831.

S. Gugushvili, A. van der Vaart and D. Yan (2018), Bayesian linear inverse problems in regularity scales. arXiv:1802.08992v1

I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin and A. C. Courville (2017), Improved training of Wasserstein GANs. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)* (I. Guyon *et al.*, eds), Curran Associates, pp. 5767–5777.

Y. Guo, Y. Liu, T. Georgiou and M. S. Lew (2018), 'A review of semantic segmentation using deep neural networks', *Internat. J. Multimedia Information Retrieval* **7**, 87–93.

H. Gupta, K. H. Jin, H. Q. Nguyen, M. T. McCann and M. Unser (2018), 'CNN-based projected gradient descent for consistent CT image reconstruction', *IEEE Trans. Medical Imaging* **37**, 1440–1453.

S. Gutta, M. Bhatt, S. K. Kalva, M. Pramanik and P. K. Yalavarthy (2019), 'Modeling errors compensation with total least squares for limited data photoacoustic tomography', *IEEE J. Selected Topics Quantum Electron.* **25**, 1–14.

E. Haber and L. Ruthotto (2017), 'Stable architectures for deep neural networks', *Inverse Problems* **34**, 014004.

E. Haber and L. Tenorio (2003), 'Learning regularization functionals', *Inverse Problems* **19**, 611–626.

E. Haber, L. Horesh and L. Tenorio (2010), 'Numerical methods for the design of large-scale nonlinear discrete ill-posed inverse problems', *Inverse Problems* **26**, 025002.

J. Hadamard (1902), 'Sur les problèmes aux dérivées partielles et leur signification physique', *Princeton University Bulletin*, pp. 49–52.

J. Hadamard (1923), *Lectures on Cauchy's Problem in Linear Partial Differential Equations*, Yale University Press.

J. Haegele, J. Rahmer, B. Gleich, J. Borgert, H. Wojtczyk, N. Panagiotopoulos, T. Buzug, J. Barkhausen and F. Vogt (2012), 'Magnetic particle imaging: Visualization of instruments for cardiovascular intervention', *Radiology* **265**, 933–938.

B. N. Hahn (2015), 'Dynamic linear inverse problems with moderate movements of the object: Ill-posedness and regularization', *Inverse Probl. Imaging* **9**, 395–413.

U. Hämarik, B. Kaltenbacher, U. Kangro and E. Resmerita (2016), 'Regularization by discretization in Banach spaces', *Inverse Problems* **32**, 035004.

K. Hammernik, T. Klatzer, E. Kobler, M. P. Recht, D. K. Sodickson, T. Pock and F. Knoll (2018), 'Learning a variational network for reconstruction of accelerated MRI data', *Magnetic Reson. Med.* **79**, 3055–3071.

K. Hammernik, F. Knoll, D. Sodickson and T. Pock (2016), Learning a variational model for compressed sensing MRI reconstruction. In *Proceedings of the International Society of Magnetic Resonance in Medicine (ISMRM)*.

Y. Han and J. C. Ye (2018), 'Framing U-Net via deep convolutional framelets: Application to sparse-view CT', *IEEE Trans. Medical Imaging* **37**, 1418–1429.

M. Hanke-Bourgeois (1995), *Conjugate Gradient Type Methods for Ill-Posed Problems*, Vol. 327 of Pitman Research Notes in Mathematics, Longman.

M. Hanke and P. C. Hansen (1993), 'Regularization methods for large-scale problems', *Surveys Math. Indust.* **3**, 253–315.

P. C. Hansen (1992), 'Analysis of discrete ill-posed problems by means of the L-curve', *SIAM Review* **34**, 561–580.

A. Hauptmann, F. Lucka, M. Betcke, N. Huynh, J. Adler, B. Cox, P. Beard, S. Ourselin and S. Arridge (2018), 'Model-based learning for accelerated limited-view 3-D photoacoustic tomography', *IEEE Trans. Medical Imaging* **37**, 1382–1393.

B. He and X. Yuan (2012), 'Convergence analysis of primal–dual algorithms for a saddle-point problem: From contraction perspective', *SIAM J. Imaging Sci.* **5**, 119–149.

J. He, Y. Yang, Y. Wang, D. Zeng, Z. Bian, H. Zhang, J. Sun, Z. Xu and J. Ma (2019), 'Optimizing a parameterized plug-and-play ADMM for iterative low-dose CT reconstruction', *IEEE Trans. Medical Imaging* **38**, 371–382.

K. He, X. Zhang, S. Ren and J. Sun (2016), Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, pp. 770–778.

T. Helin and M. Burger (2015), 'Maximum *a posteriori* probability estimates in infinite-dimensional Bayesian inverse problems', *Inverse Problems* **31**, 085009.

S. W. Hell, A. Schönle and A. Van den Bos (2007), Nanoscale resolution in far-field fluorescence microscopy. In *Science of Microscopy* (P. W. Hawkes and J. C. H. Spence, eds), Vol. 2, Springer, pp. 790–834.

C. F. Higham and D. J. Higham (2018), Deep learning: An introduction for applied mathematicians. arXiv:1801.05894

M. Hintermüller and C. N. Rautenberg (2017), 'Optimal selection of the regularization function in a weighted total variation model, I: Modelling and theory', *J. Math. Imaging Vision* **59**, 498–514.

M. Hintermüller and T. Wu (2015), 'Bilevel optimization for calibrating point spread functions in blind deconvolution', *Inverse Probl. Imaging* **9**, 1139–1169.

M. Hintermüller, A. Laurain, C. Löbhard, C. N. Rautenberg and T. M. Surowiec (2014), Elliptic mathematical programs with equilibrium constraints in function space: Optimality conditions and numerical realization. In *Trends in PDE Constrained Optimization* (G. Leugering *et al.*, eds), Springer, pp. 133–153.

M. Hintermüller, C. N. Rautenberg, T. Wu and A. Langer (2017), 'Optimal selection of the regularization function in a weighted total variation model, II: Algorithm, its analysis and numerical tests', *J. Math. Imaging Vision* **59**, 515–533.

K. Hirakawa and T. W. Parks (2006), 'Image denoising using total least squares', *IEEE Trans. Image Process.* **15**, 2730–2742.

B. Hofmann (1994), 'On the degree of ill-posedness for nonlinear problems', *J. Inverse Ill-Posed Problems* **2**, 61–76.

B. Hofmann, B. Kaltenbacher, C. Pöschl and O. Scherzer (2007), 'A convergence rates result for Tikhonov regularization in Banach spaces with non-smooth operators', *Inverse Problems* **23**, 987.

B. Hofmann, S. Kindermann *et al.* (2010), 'On the degree of ill-posedness for linear problems with noncompact operators', *Methods Appl. Anal.* **17**, 445–462.

T. Hohage and F. Weidling (2016), 'Characterizations of variational source conditions, converse results, and maxisets of spectral regularization methods', *SIAM J. Numer. Anal.* **55**, 598–620.

T. Hohage and F. Werner (2016), 'Inverse problems with Poisson data: Statistical regularization theory, applications and algorithms', *Inverse Problems* **32**, 093001.

X. Hong, Y. Zan, F. Weng, W. Tao, Q. Peng and Q. Huang (2018), 'Enhancing the image quality via transferred deep residual learning of coarse PET sinograms', *IEEE Trans. Medical Imaging* **37**, 2322–2332.

L. Hörmander (1971), 'Fourier integral operators, I', *Acta Math.* **127**, 79–183.

H. Hornik (1991), 'Approximation capabilities of multilayer feedforward networks', *Neural Networks* **4**, 251–257.

K. Hornik, M. Stinchcombe and H. White (1989), 'Multilayer feedforward networks are universal approximators', *Neural Networks* **2**, 359–366.

J.-T. Hsieh, S. Zhao, S. Eismann, L. Mirabella and S. Ermon (2019), Learning neural PDE solvers with convergence guarantees. In *Seventh International Conference on Learning Representations (ICLR 2019)*, to appear.

J. Huang and D. Mumford (1999), Statistics of natural images and models. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 1999)*, Vol. 1, pp. 541–547.

Y. Huizhuo, J. Jinzhu and Z. Zhanxing (2018), SIPID: A deep learning framework for sinogram interpolation and image denoising in low-dose CT reconstruction. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 1521–1524.

C. M. Hyun, H. P. Kim, S. M. Lee, S. Lee and J. K. Seo (2018), 'Deep learning for undersampled MRI reconstruction', *Phys. Med. Biol.* **63**, 135007.

M. Igami (2017), Artificial intelligence as structural estimation: Economic interpretations of Deep Blue, Bonanza, and AlphaGo. arXiv:1710.10967

N. Ikeda and S. Watanabe (1989), *Stochastic Differential Equations and Diffusion Processes*, second edition, North-Holland.

J. Ingraham, A. Riesselman, C. Sander and D. Marks (2019), Learning protein structure with a differentiable simulator. In *International Conference on Learning Representations (ICLR 2019)*.

E. Janssens, J. D. Beenhouwer, M. V. Dael, T. D. Schryver, L. V. Hoorebeke, P. Verboven, B. Nicolai and J. Sijbers (2018), 'Neural network Hilbert transform based filtered backprojection for fast inline X-ray inspection', *Measurement Sci. Tech.* **29**, 034012.

B. Jin and P. Maass (2012*a*), 'An analysis of electrical impedance tomography with applications to Tikhonov regularization', *ESAIM Control Optim. Calc. Var.* **18**, 1027–1048.

B. Jin and P. Maass (2012*b*), 'Sparsity regularization for parameter identification problems', *Inverse Problems* **28**, 123001.

K. H. Jin, M. T. McCann, E. Froustey and M. Unser (2017), 'Deep convolutional neural network for inverse problems in imaging', *IEEE Trans. Image Process.* **26**, 4509–4522.

F. John (1955*a*), 'A note on "improper" problems in partial differential equations', *Commun. Pure Appl. Math.* **8**, 591–594.

F. John (1955*b*), 'Numerical solution of the equation of heat conduction for preceding times', *Ann. Mat. Pura Appl.* (4) **40**, 129–142.

F. John (1959), Numerical solution of problems which are not well posed in the sense of Hadamard. In *Proc. Rome Symp. Prov. Int. Comp. Center*, pp. 103–116.

F. John (1960), 'Continuous dependence on data for solutions of partial differential equations with a prescribed bound', *Commun. Pure Appl. Math.* **13**, 551–585.

D. J. Kadrmas (2004), 'LOR-OSEM: Statistical PET reconstruction from raw line-of-response histograms', *Phys. Med. Biol.* **49**, 4731–4744.

J. P. Kaipio and E. Somersalo (2005), *Statistical and Computational Inverse Problems*, Vol. 160 of Applied Mathematical Sciences, Springer.

J. P. Kaipio and E. Somersalo (2007), 'Statistical inverse problems: Discretization, model reduction and inverse crimes', *J. Comput. Appl. Math.* **198**, 493–504.

O. Kallenberg (2002), *Foundations of Modern Probability*, second edition, Springer.

B. Kaltenbacher, A. Kirchner and B. Vexler (2011), 'Adaptive discretizations for the choice of a Tikhonov regularization parameter in nonlinear inverse problems', *Inverse Problems* **27**, 125008.

B. Kaltenbacher, A. Neubauer and O. Scherzer (2008), *Iterative Regularization Methods for Nonlinear Ill-posed Problems*, Vol. 6 of Radon Series on Computational and Applied Mathematics, De Gruyter.

E. Kang and J. C. Ye (2018), Framelet denoising for low-dose CT using deep learning. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 311–314.

E. Kang, W. Chang, J. Yoo and J. C. Ye (2018), 'Deep convolutional framelet denoising for low-dose CT via wavelet residual network', *IEEE Trans. Medical Imaging* **37**, 1358–1369.

E. Kang, J. Min and J. C. Ye (2017), 'A deep convolutional neural network using directional wavelets for low-dose X-ray CT reconstruction', *Med. Phys.* **44**, 360–375.

A. Karpathy and L. Fei-Fei (2017), 'Deep visual-semantic alignments for generating image descriptions', *IEEE Trans. Pattern Anal. Mach. Intel.* **39**, 664–676.

K. Kekkonen, M. Lassas and S. Siltanen (2016), 'Posterior consistency and convergence rates for Bayesian inversion with hypoelliptic operators', *Inverse Problems* **32**, 085005.

A. Khandhar, P. Keselman, S. Kemp, R. Ferguson, P. Goodwill, S. Conolly and K. Krishnan (2017), 'Evaluation of peg-coated iron oxide nanoparticles as blood pool tracers for preclinical magnetic particle imaging', *Nanoscale* **9**, 1299–1306.

Y. Khoo and L. Ying (2018), SwitchNet: A neural network model for forward and inverse scattering problems. arXiv:1810.09675v1

Y. Khoo, J. Lu and L. Ying (2017), Solving parametric PDE problems with artificial neural networks. arXiv:1707.03351v2

D. Kim and J. A. Fessler (2016), 'Optimized first-order methods for smooth convex minimization', *Math. Program.* **159**, 81–107.

K. Kim, G. E. Fakhri and Q. Li (2017), 'Low-dose CT reconstruction using spatially encoded nonlocal penalty', *Med. Phys.* **44**, 376–390.

K. Kim, D. Wu, K. Gong, J. Dutta, J. H. Kim, Y. D. Son, H. K. Kim, G. E. Fakhri and Q. Li (2018), 'Penalized PET reconstruction using deep learning prior and local linear fitting', *IEEE Trans. Medical Imaging* **37**, 1478–1487.

S.-J. Kim, K. Koh, M. Lustig, S. Boyd and D. Gorinevsky (2007), 'An interior-point method for large-scale ℓ_1-regularized least squares', *IEEE J. Selected Topics Signal Process.* **1**, 606–617.

S. Kindermann (2011), 'Convergence analysis of minimization-based noise level-free parameter choice rules for linear ill-posed problems', *Electron. Trans. Numer. Anal.* **38**, 233–257.

A. Kirsch (2011), *An Introduction to the Mathematical Theory of Inverse Problems*, second edition, Vol. 120 of Applied Mathematical Sciences, Springer.

T. Klatzer and T. Pock (2015), Continuous hyper-parameter learning for support vector machines. In *20th Computer Vision Winter Workshop (CVWW)*.

B. J. K. Kleijn and Y. Y. Zhao (2018), Criteria for posterior consistency. arXiv:1308.1263v5

T. Kluth (2018), 'Mathematical models for magnetic particle imaging', *Inverse Problems* **34**, 083001.

T. Kluth and P. Maass (2017), 'Model uncertainty in magnetic particle imaging: Nonlinear problem formulation and model-based sparse reconstruction', *Internat. J. Magnetic Particle Imaging* **3**, 1707004.

B. T. Knapik, B. T. Szabó, A. W. van der Vaart and J. H. van Zanten (2016), 'Bayes procedures for adaptive inference in inverse problems for the white noise model', *Probab. Theory Related Fields* **164**, 771–813.

B. T. Knapik, A. W. van der Vaart and J. H. van Zanten (2011), 'Bayesian inverse problems with Gaussian priors', *Ann. Statist.* **39**, 2626–2657.

B. T. Knapik, A. W. van der Vaart and J. H. van Zanten (2013), 'Bayesian recovery of the initial condition for the heat equation', *Commun. Statist. Theory Methods* **42**, 1294–1313.

T. Knopp, N. Gdaniec and M. Möddel (2017), 'Magnetic particle imaging: From proof of principle to preclinical applications', *Phys. Med. Biol.* **62**, R124.

T. Knopp, T. Viereck, G. Bringout, M. Ahlborg, J. Rahmer and M. Hofmann (2016), MDF: Magnetic particle imaging data format. arXiv:1602.06072

S. Ko, D. Yu and J.-H. Won (2017), On a class of first-order primal–dual algorithms for composite convex minimization problems. arXiv:1702.06234

E. Kobler, T. Klatzer, K. Hammernik and T. Pock (2017), Variational networks: connecting variational methods and deep learning. In *German Conference on Pattern Recognition (GCPR 2017)*, Vol. 10496 of Lecture Notes in Computer Science, Springer, pp. 281–293.

F. Kokkinos and S. Lefkimmiatis (2018), Deep image demosaicking using a cascade of convolutional residual denoising networks. arXiv:1803.05215

V. Kolehmainen, M. Lassas, K. Niinimäki and S. Siltanen (2012), 'Sparsity-promoting Bayesian inversion', *Inverse Problems* **28**, 025005.

V. P. Krishnan and E. T. Quinto (2015), *Microlocal Analysis in Tomography*, Handbook of Mathematical Methods in Imaging, Springer.
A. Küçükelbir, R. Ranganath, A. Gelman and D. Blei (2017), 'Automatic variational inference', *J. Mach. Learn. Res.* **18**, 430–474.
J. Kukačka, V. Golkov and D. Cremers (2017), Regularization for deep learning: A taxonomy. arXiv:1710.10686
K. Kunisch and T. Pock (2013), 'A bilevel optimization approach for parameter learning in variational models', *SIAM J. Imaging Sci.* **6**, 938–983.
H. Kushner and G. Yin (1997), *Stochastic Approximation Algorithms and Applications*, Springer.
G. Kutyniok and D. Labate (2009), 'Resolution of the wavefront set using continuous shearlets', *Trans. Amer. Math. Soc.* **361**, 2719–2754.
G. Kutyniok and D. Labate (2012), *Shearlets: Multiscale Analysis for Multivariate Data*, Springer.
R. R. Lam, L. Horesh, H. Avron and K. E. Willcox (2017), Should you derive, or let the data drive? An optimization framework for hybrid first-principles data-driven modeling. arXiv:1711.04374
F. Lanusse, J.-L. Starck, A. Woiselle and J. M. Fadili (2014), '3-D sparse representations', *Adv. Imaging Electron Phys.* **183**, 99–204.
A. Lanza, S. Morigi, F. Sgallari and Y.-W. Wen (2014), 'Image restoration with Poisson–Gaussian mixed noise', *Comput. Methods Biomech. Biomed. Engng Imaging Vis.* **2**, 12–24.
M. Lassas, E. Saksman and S. Siltanen (2009), 'Discretization invariant Bayesian inversion and Besov space priors', *Inverse Probl. Imaging* **3**, 87–122.
P. Latafat and P. Patrinos (2017), 'Asymmetric forward–backward–adjoint splitting for solving monotone inclusions involving three operators', *Comput. Optim. Appl.* **68**, 57–93.
R. Lattès and J.-L. Lions (1969), *The Method of Quasi-Reversibility: Applications to Partial Differential Equations*, Vol. 18 of Modern Analytic and Computational Methods in Science and Mathematics, American Elsevier.
L. Le Cam (1986), *Asymptotic Methods in Statistical Decision Theory*, Springer Series in Statistics, Springer.
Y. LeCun, Y. Bengio and G. Hinton (2015), 'Deep learning', *Nature* **521** (7553), 436–444.
D. Lee, J. Yoo and J. C. Ye (2017), Deep residual learning for compressed sensing MRI. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pp. 15–18.
H. Lee, J. Lee, H. Kim, B. Cho and S. Cho (2019), 'Deep-neural-network based sinogram synthesis for sparse-view CT image reconstruction', *IEEE Trans. Radiation Plasma Med. Sci.* **3**, 109–119.
S. Lefkimmiatis (2017), Non-local color image denoising with convolutional neural networks. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2017)*, pp. 3587–3596.
M. S. Lehtinen, L. Päivärinta and E. Somersalo (1989), 'Linear inverse problems for generalised random variables', *Inverse Problems* **5**, 599–612.
C. Y. Li, X. Liang, Z. Hu and E. P. Xing (2018*a*), Hybrid retrieval-generation reinforced agent for medical image report generation. arXiv:1805.08298

H. Li, J. Schwab, S. Antholzer and M. Haltmeier (2018*b*), NETT: Solving inverse problems with deep neural networks. arXiv:1803.00092

H. Li, Z. Xu, G. Taylor and T. Goldstein (2018*c*), Visualizing the loss landscape of neural nets. In *6th International Conference on Learning Representations (ICLR 2018)*.

F. Liese and K.-J. Miescke (2008), *Statistical Decision Theory: Estimation, Testing, and Selection*, Springer Series in Statistics, Springer.

R. Liu, Z. Lin, W. Zhang and Z. Su (2010), Learning PDEs for image restoration via optimal control. In *European Conference on Computer Vision (ECCV 2010)*, Vol. 6311 of Lecture Notes in Computer Science, Springer, pp. 115–128.

Z. Long, Y. Lu, X. Ma and B. Dong (2018), PDE-Net: Learning PDEs from data. arXiv:1710.09668v2

A. K. Louis (1989), *Inverse und schlecht gestellte Probleme*, Vieweg/Teubner.

A. K. Louis (1996), 'Approximate inverse for linear and some nonlinear problems', *Inverse Problems* **12**, 175–190.

A. K. Louis (2011), 'Feature reconstruction in inverse problems', *Inverse Problems* **27**, 065010.

A. K. Louis and P. Maass (1990), 'A mollifier method for linear operator equations of the first kind', *Inverse Problems* **6**, 427.

S. Lu, S. V. Pereverzev and U. Tautenhahn (2009), 'Regularized total least squares: computational aspects and error bounds', *SIAM J. Matrix Anal. Appl.* **31**, 918–941.

A. Lucas, M. Iliadis, R. Molina and A. K. Katsaggelos (2018), 'Using deep neural networks for inverse problems in imaging: Beyond analytical methods', *IEEE Signal Process. Mag.* **35**, 20–36.

S. Lunz, O. Öktem and C.-B. Schönlieb (2018), Adversarial regularizers in inverse problems. In *Advances in Neural Information Processing Systems 31 (NIPS 2018)* (S. Bengio *et al.*, eds), Curran Associates, pp. 8507–8516.

Z.-Q. Luo, J.-S. Pang and D. Ralph (1996), *Mathematical programs with equilibrium constraints*, Cambridge University Press.

P. Maass (2019), Deep learning for trivial inverse problems. In *Compressed Sensing and its Applications*, Birkhäuser.

J. Mairal and J. Ponce (2014), Sparse modeling for image and vision processing. arXiv:1411.3230v2

J. Mairal, F. Bach, J. Ponce and G. Sapiro (2010), 'Online learning for matrix factorization and sparse coding', *J. Mach. Learn. Res.* **11**, 19–60.

S. Mallat (2009), *A Wavelet Tour of Signal Processing: The Sparse Way*, third edition, Academic Press.

S. Mallat (2012), 'Group invariant scattering', *Commun. Pure Appl. Math.* **65**, 1331–1398.

S. Mallat (2016), 'Understanding deep convolutional networks', *Philos. Trans. Royal Soc.* A **374**, 20150203.

S. G. Mallat and Z. Zhang (1993), 'Matching pursuits with time-frequency dictionaries', *IEEE Trans. Signal Process.* **41**, 3397–3415.

A. Mandelbaum (1984), 'Linear estimators and measurable linear transformations on a Hilbert space', *Z. Wahrsch. verw. Gebiete* **65**, 385–397.

X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang and S. P. Smolley (2016), Least squares generative adversarial networks. arXiv:1611.04076

M. Mardani, E. Gong, J. Y. Cheng, J. Pauly and L. Xing (2017*a*), Recurrent generative adversarial neural networks for compressive imaging. In *IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2017)*.

M. Mardani, E. Gong, J. Y. Cheng, S. Vasanawala, G. Zaharchuk, M. Alley, N. Thakur, S. Han, W. Dally, J. M. Pauly and L. Xing (2017*b*), Deep generative adversarial networks for compressed sensing (GANCS) automates MRI. arXiv:1706.00051

M. Markkanen, L. Roininen, J. M. J. Huttunen and S. Lasanen (2019), 'Cauchy difference priors for edge-preserving Bayesian inversion', *J. Inverse Ill-Posed Problems* **27**, 225–240.

I. Markovsky and S. Van Huffel (2007), 'Overview of total least-squares methods', *Signal Processing* **87**, 2283–2302.

J. Martens and I. Sutskever (2012), Training deep and recurrent networks with Hessian-free optimization. In *Neural Networks: Tricks of the Trade*, Vol. 7700 of Lecture Notes in Computer Science, Springer, pp. 479–535.

D. Martin, C. Fowlkes, D. Tal and J. Malik (2001), A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *8th International Conference on Computer Vision (ICCV 2001)*, Vol. 2, pp. 416–423.

M. T. McCann and M. Unser (2019), Algorithms for biomedical image reconstruction. arXiv:1901.03565

M. T. McCann, K. H. Jin and M. Unser (2017), 'Convolutional neural networks for inverse problems in imaging: A review', *IEEE Signal Process. Mag.* **34**, 85–95.

T. Meinhardt, M. Moeller, C. Hazirbas and D. Cremers (2017), Learning proximal operators: Using denoising networks for regularizing inverse imaging problems. In *IEEE International Conference on Computer Vision (ICCV 2017)*, pp. 1799–1808.

K. Miller (1970), 'Least squares methods for ill-posed problems with a prescribed bound', *SIAM J. Math. Anal.* **1**, 52–74.

D. D. L. Minh and D. Le Minh (2015), 'Understanding the Hastings algorithm', *Commun. Statist. Simul. Comput.* **44**, 332–349.

T. Minka (2001), Expectation propagation for approximate Bayesian inference. In *17th Conference on Uncertainty in Artificial Intelligence (UAI '01)* (J. S. Breese and D. Koller, eds), Morgan Kaufmann, pp. 362–369.

F. Monard, R. Nickl and G. P. Paternain (2019), Efficient nonparametric Bayesian inference for X-ray transforms. *Ann. Statist.* **47**, 1113–1147.

N. Moriakov, K. Michielsen, J. Adler, R. Mann, I. Sechopoulos and J. Teuwen (2018), Deep learning framework for digital breast tomosynthesis reconstruction. arXiv:1808.04640

V. A. Morozov (1966), 'On the solution of functional equations by the method of regularization', *Soviet Math. Doklady* **7**, 414–417.

A. Mousavi and R. G. Baraniuk (2017), Learning to invert: Signal recovery via deep convolutional networks. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2272–2276.

J. L. Mueller and S. Siltanen (2012), *Linear and Nonlinear Inverse Problems with Practical Applications*, SIAM.

D. Mumford and J. Shah (1989), 'Optimal approximations by piecewise smooth functions and associated variational problems', *Commun. Pure Appl. Math.* **42**, 577–685.

K. Murase, M. Aoki, N. Banura, K. Nishimoto, A. Mimura, T. Kuboyabu and I. Yabata (2015), 'Usefulness of magnetic particle imaging for predicting the therapeutic effect of magnetic hyperthermia', *Open J. Medical Imaging* **5**, 85.

F. Natterer (1977), 'Regularisierung schlecht gestellter Probleme durch Projektionsverfahren', *Numer. Math.* **28**, 329–341.

F. Natterer (2001), *The Mathematics of Computerized Tomography*, Vol. 32 of Classics in Applied Mathematics, SIAM.

F. Natterer and F. Wübbeling (2001), *Mathematical Methods in Image Reconstruction*, SIAM.

R. M. Neal (2003), 'Slice sampling', *Ann. Statist.* **31**, 705–767.

D. Needell and J. A. Tropp (2009), 'CoSaMP: iterative signal recovery from incomplete and inaccurate samples', *Appl. Comput. Harmon. Anal.* **26**, 301–321.

D. Needell and R. Vershynin (2009), 'Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit', *Found. Comput. Math.* **9**, 317–334.

Y. Nesterov (2004), *Introductory Lectures on Convex Optimization: A Basic Course*, Vol. 87 of Applied Optimization, Springer.

Y. Nesterov (2007), Gradient methods for minimizing composite objective function. CORE Discussion Papers no. 2007076, Center for Operations Research and Econometrics (CORE), Université Catholique de Louvain.

A. Neubauer and H. K. Pikkarainen (2008), 'Convergence results for the Bayesian inversion theory', *J. Inverse Ill-Posed Problems* **16**, 601–613.

R. Nickl (2013), *Statistical Theory.* Lecture notes, University of Cambridge. http://www.statslab.cam.ac.uk/~nickl/Site/_files/stat2013.pdf

R. Nickl (2017*a*), 'Bernstein–von Mises theorems for statistical inverse problems, I: Schrödinger equation,' *J. Eur. Math. Soc.*, to appear. arXiv:1707.01764

R. Nickl (2017*b*), 'On Bayesian inference for some statistical inverse problems with partial differential equations', *Bernoulli News* **24**, 5–9.

R. Nickl and J. Söhl (2017), 'Nonparametric Bayesian posterior contraction rates for discretely observed scalar diffusions', *Ann. Statist.* **45**, 1664–1693.

R. Nickl, S. van de Geer and S. Wang (2018), Convergence rates for penalised least squares estimators in PDE-constrained regression problems. arXiv:1809.08818

L. Nie and X. Chen (2014), 'Structural and functional photoacoustic molecular tomography aided by emerging contrast agents', *Chem. Soc. Review* **43**, 7132–70.

J. Nocedal and S. Wright (2006), *Numerical Optimization*, Springer Series in Operations Research and Financial Engineering, Springer.

C. Oh, D. Kim, J.-Y. Chung, Y. Han and H. W. Park (2018), ETER-net: End to end MR image reconstruction using recurrent neural network. In *International*

Workshop on Machine Learning for Medical Image Reconstruction (MLMIR 2018) (F. Knoll *et al.*, eds), Vol. 11074 of Lecture Notes in Computer Science, Springer.

B. A. Olshausen and D. J. Field (1997), 'Sparse coding with an overcomplete basis set: A strategy employed by V1?', *Vision Research* **37**, 3311–3325.

J. V. Outrata (2000), 'A generalized mathematical program with equilibrium constraints', *SIAM J. Control Optim.* **38**, 1623–1638.

S. Oymak and M. Soltanolkotabi (2017), 'Fast and reliable parameter estimation from nonlinear observations', *SIAM J. Optim.* **27**, 2276–2300.

V. Papyan, Y. Romano and M. Elad (2017), 'Convolutional neural networks analysed via convolutional sparse coding', *J. Mach. Learn. Res.* **18**, 1–52.

V. Papyan, J. Sulam and M. Elad (2016*a*), Working locally thinking globally, I: Theoretical guarantees for convolutional sparse coding. arXiv:1607.02005

V. Papyan, J. Sulam and M. Elad (2016*b*), Working locally thinking globally, II: Stability and algorithms for convolutional sparse coding. arXiv:1607.02009

P. Paschalis, N. D. Giokaris, A. Karabarbounis, G. K. Loudos, D. Maintas, C. N. Papanicolas, V. Spanoudaki, C. Tsoumpas and E. Stiliaris (2004), 'Tomographic image reconstruction using artificial neural networks', *Nucl. Instrum. Methods Phys. Res.* A **527**, 211–215.

D. M. Pelt and K. J. Batenburg (2013), 'Fast tomographic reconstruction from limited data using artificial neural networks', *IEEE Trans. Image Process.* **22**, 5238–5251.

D. M. Pelt, K. J. Batenburg and J. A. Sethian (2018), 'Improving tomographic reconstruction from limited data using mixed-scale dense convolutional neural networks', *J. Imaging* **4**, 128.

P. Perona and J. Malik (1990), 'Scale-space and edge detection using anisotropic diffusion', *IEEE Trans. Pattern Anal. Mach. Intel.* **12**, 629–639.

G. Peyré, S. Bougleux and L. D. Cohen (2011), 'Non-local regularization of inverse problems', *Inverse Probl. Imaging* **5**, 511–530.

D. L. Phillips (1962), 'A technique for the numerical solution of certain integral equations of the first kind', *J. Assoc. Comput. Mach.* **9**, 84–97.

L. Plantagie and J. K. Batenburg (2015), 'Algebraic filter approach for fast approximation of nonlinear tomographic reconstruction methods', *J. Electron. Imaging* **24**, 013026.

R. Plato and G. Vainikko (1990), 'On the regularization of projection methods for solving ill-posed problems', *Numer. Math.* **57**, 63–79.

Y. Pu, X. Yuan, A. Stevens, C. Li and L. Carin (2016), A deep generative deconvolutional image model. In *19th International Conference on Artificial Intelligence and Statistics (AISTATS 2016)*, pp. 741–750.

P. Putzky and M. Welling (2017), Recurrent inference machines for solving inverse problems. arXiv:1706.04008

C. Qin, J. Schlemper, J. Caballero, A. N. Price, J. V. Hajnal and D. Rueckert (2019), 'Convolutional recurrent neural networks for dynamic MR image reconstruction', *IEEE Trans. Medical Imaging* **38**, 280–290.

T. M. Quan, S. Member, T. Nguyen-Duc and W.-K. Jeong (2018), 'Compressed sensing MRI reconstruction using a generative adversarial network with a cyclic loss', *IEEE Trans. Medical Imaging* **37**, 1488–1497.

E. T. Quinto (1993), 'Singularities of the X-ray transform and limited data tomography in $\mathbb{R}^2$ and $\mathbb{R}^3$', *SIAM J. Math. Anal.* **24**, 1215–1225.

E. T. Quinto and O. Öktem (2008), 'Local tomography in electron microscopy', *SIAM J. Appl. Math.* **68**, 1282–1303.

J. Radon (1917), 'Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten', *Ber. Verh. Sächs. Akad. Wiss. (Leipzig)* **69**, 262–277.

M. Raissi and G. E. Karniadakis (2017), Hidden physics models: Machine learning of nonlinear partial differential equations. arXiv:1708.00588v2

R. Ranftl and T. Pock (2014), A deep variational model for image segmentation. In *36th German Conference on Pattern Recognition (GCPR 2014)*, Vol. 8753 of Lecture Notes in Computer Science, Springer, pp. 107–118.

K. Ray (2013), 'Bayesian inverse problems with non-conjugate priors', *Electron. J. Statist.* **7**, 2516–2549.

E. T. Reehorst and P. Schniter (2018), Regularization by denoising: Clarifications and new interpretations. arXiv:1806.02296

A. Repetti, M. Pereyra and Y. Wiaux (2019), 'Scalable Bayesian uncertainty quantification in imaging inverse problems via convex optimization', *SIAM J. Imaging Sci.* **12**, 87–118.

W. Ring (2000), 'Structural properties of solutions to total variation regularization problems', *ESAIM Math. Model. Numer. Anal.* **34**, 799–810.

S. Rizzo, F. Botta, S. Raimondi, D. Origgi, C. Fanciullo, A. G. Morganti and M. Bellomi (2018), 'Radiomics: The facts and the challenges of image analysis', *European Radiol. Exp.* **2**, 36.

G. Rizzuti, A. Siahkoohi and F. J. Herrmann (2019), Learned iterative solvers for the Helmholtz equation. Submitted to *81st EAGE Conference and Exhibition 2019*. Available from https://www.slim.eos.ubc.ca/content/learned-iterative-solvers-helmholtz-equation.

H. Robbins and S. Monro (1951), 'A stochastic approximation method', *Ann. Math. Statist.* **22**, 400–407.

C. P. Robert and G. Casella (2004), *Monte Carlo Statistical Methods*, Springer Texts in Statistics, Springer.

R. T. Rockafellar and R. J.-B. Wets (1998), *Variational Analysis*, Springer.

Y. Romano and M. Elad (2015), Patch-disagreement as a way to improve K-SVD denoising. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1280–1284.

Y. Romano, M. Elad and P. Milanfar (2017*a*), 'The little engine that could: Regularization by denoising (RED)', *SIAM J. Imaging Sci.* **10**, 1804–1844.

Y. Romano, J. Isidoro and P. Milanfar (2017*b*), 'RAISR: Rapid and accurate image super resolution', *IEEE Trans. Comput. Imaging* **3**, 110–125.

O. Ronneberger, P. Fischer and T. Brox (2015), U-Net: Convolutional networks for biomedical image segmentation. In *18th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2015)* (N. Navab *et al.*, eds), Vol. 9351 of Lecture Notes in Computer Science, Springer, pp. 234–241.

S. Roth and M. J. Black (2005), Fields of experts: A framework for learning image priors. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005)*, Vol. 2, pp. 860–867.

R. Rubinstein, A. M. Bruckstein and M. Elad. (2010), 'Dictionaries for sparse representation modeling', *Proc. IEEE* **98**, 1045–1057.

S. Ruder (2016), An overview of gradient descent optimization algorithms. arXiv:1609.04747

L. I. Rudin, S. Osher and E. Fatemi (1992), 'Nonlinear total variation based noise removal algorithms', *Phys.* D **60**, 259–268.

D. E. Rumelhart, G. E. Hinton and R. J. Williams (1986), Learning internal representation by error propagation. In *Parallel distributed processing: Explorations in the Microstructures of Cognition*, Vol. 1: *Foundations* (D. E. Rumelhart, J. L. McClelland and the PDP Research Group, eds), MIT Press, pp. 318–362.

L. Ruthotto and E. Haber (2018), Deep neural networks motivated by partial differential equations. arXiv:1804.04272

J. Salamon, M. Hofmann, C. Jung, M. G. Kaul, F. Werner, K. Them, R. Reimer, P. Nielsen, A. vom Scheidt, G. Adam, T. Knopp and H. Ittrich (2016), 'Magnetic particle/magnetic resonance imaging: *In-vitro* MPI-guided real time catheter tracking and 4D angioplasty using a road map and blood pool tracer approach', *PloS ONE* **11**, e0156899.

K. G. Samuel and M. F. Tappen (2009), Learning optimized map estimates in continuously-valued MRF models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pp. 477–484.

M. Sato (1971), Regularity of hyperfunctions solutions of partial differential equations. In *Actes du Congrès International des Mathématiciens*, Vol. 2, Gauthier-Villars, pp. 785–794.

A. Sawatzky, C. Brune, J. Müller and M. Burger (2009), Total variation processing of images with Poisson statistics. In *Computer Analysis of Images and Patterns* (X. Jiang and N. Petkov, eds), Vol. 5702 of Lecture Notes in Computer Science, Springer, pp. 533–540.

M. J. Schervish (1995), *Theory of Statistics*, Springer Series in Statistics, Springer.

O. Scherzer (1998), 'A modified Landweber iteration for solving parameter estimation problems', *Appl. Math. Optim.* **38**, 45–68.

O. Scherzer, M. Grasmair, H. Grossauer, M. Haltmeier and F. Lenzen (2009), *Variational Methods in Imaging*, Vol. 167 of Applied Mathematical Sciences, Springer.

J. Schlemper, J. Caballero, J. V. Hajnal, A. Price and D. Rueckert (2017), A deep cascade of convolutional neural networks for MR image reconstruction. In *25th International Conference on Information Processing in Medical Imaging (IPMI 2017)*, Vol. 10265 of Lecture Notes in Computer Science, Springer, pp. 647–658.

J. Schlemper, J. Caballero, J. V. Hajnal, A. N. Price and D. Rueckert (2018), 'A deep cascade of convolutional neural networks for dynamic MR image reconstruction', *IEEE Trans. Medical Imaging* **37**, 491–503.

U. Schmidt and S. Roth (2014), Shrinkage fields for effective image restoration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2014)*, pp. 2774–2781.

U. Schmitt and A. K. Louis (2002), 'Efficient algorithms for the regularization of dynamic inverse problems, I: Theory', *Inverse Problems* **18**, 645–658.
T. Schuster (2007), *The Method of Approximate Inverse: Theory and Applications*, Vol. 1906 of Lecture Notes in Mathematics, Springer.
T. Schuster, B. Hahn and M. Burger (2018), 'Dynamic inverse problems: Modelling – regularization – numerics [Preface]', *Inverse Problems* **34**, 040301.
T. Schuster, B. Kaltenbacher, B. Hofmann and K. Kazimierski (2012), *Regularization Methods in Banach Spaces*, Radon Series on Computational and Applied Mathematics, De Gruyter.
J. Schwab, S. Antholzer and M. Haltmeier (2018), Deep null space learning for inverse problems: Convergence analysis and rates. arXiv:1806.06137
L. Schwartz (1965), 'On Bayes procedures', *Z. Wahrsch. verw. Gebiete* **4**, 10–26.
T. I. Seidman and C. R. Vogel (1989), 'Well-posedness and convergence of some regularisation methods for nonlinear ill posed problems', *Inverse Problems* **5**, 227–238.
P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus and Y. LeCun (2013), OverFeat: Integrated recognition, localization and detection using convolutional networks. arXiv:1312.6229
S.-S. Shai and B.-D. Shai (2014), *Understanding Machine Learning: From Theory to Algorithms*, Cambridge University Press.
H. Shan, Y. Zhang, Q. Yang, U. Kruger, M. K. Kalra, L. Sun, W. Cong and G. Wang (2018), '3-D convolutional encoder–decoder network for low-dose CT via transfer learning from a 2-D trained network', *IEEE Trans. Medical Imaging* **37**, 1522–1534.
J. Sirignano and K. Spiliopoulos (2017), DGM: A deep learning algorithm for solving partial differential equations. arXiv:1708.07469v1
B. Sprungk (2017), Numerical methods for Bayesian inference in Hilbert spaces. PhD thesis, Technische Universität Chemnitz.
H. Sreter and R. Giryes (2017), Learned convolutional sparse coding. arXiv:1711.00328
R. L. Streit (2010), *Poisson Point Processes: Imaging, Tracking, and Sensing*, Springer.
D. M. Strong, T. F. Chan *et al.* (1996), Exact solutions to total variation regularization problems. In *UCLA CAM Report*, Citeseer.
A. M. Stuart (2010), Inverse problems: A Bayesian perspective. In *Acta Numerica*, Vol. 19, Cambridge University Press, pp. 451–559.
A. M. Stuart and A. L. Teckentrup (2018), 'Posterior consistency for Gaussian process approximations of Bayesian posterior distributions', *Math. Comp.* **87**, 721–753.
J. Sulam and M. Elad (2015), Expected patch log likelihood with a sparse prior. In *Energy Minimization Methods in Computer Vision and Pattern Recognition: Proceedings of the 10th International Conference (EMMCVPR 2015)*, pp. 99–111.
J. Sulam, V. Papyan, Y. Romano and M. Elad (2017), Multi-layer convolutional sparse modeling: Pursuit and dictionary learning. arXiv:1708.08705
N.-S. Syu, Y.-S. Chen and Y.-Y. Chuang (2018), Learning deep convolutional networks for demosaicing. arXiv:1802.03769

B. T. Szabó, A. W. van der Vaart and J. H. van Zanten (2013), 'Empirical Bayes scaling of Gaussian priors in the white noise model', *Electron. J. Statist.* **7**, 991–1018.

C. Szegedy, W. Zaremb, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow and R. Fergus (2014), Intriguing properties of neural networks. arXiv:1312.6199v4

M. F. Tappen (2007), Utilizing variational optimization to learn Markov random fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2007)*, pp. 1–8.

A. Tarantola (2005), *Inverse Problem Theory and Methods for Model Parameter Estimation*, second edition, SIAM.

A. Tarantola and B. Valette (1982), 'Inverse Problems = Quest for Information', *J. Geophys* **50**, 159–170.

S. Tariyal, A. Majumdar, R. Singh and M. Vatsa (2016), 'Deep dictionary learning', *IEEE Access* **4**, 10096–10109.

U. Tautenhahn (2008), 'Regularization of linear ill-posed problems with noisy right hand side and noisy operator', *J. Inverse Ill-Posed Problems* **16**, 507–523.

A. Taylor, J. Hendrickx and F. Glineur (2017), 'Smooth strongly convex interpolation and exact worst-case performance of first-order methods', *Math. Program.* **161**, 307–345.

M. Thoma (2016), A survey of semantic segmentation. arXiv:1602.06541

R. Tibshirani (1996), 'Regression shrinkage and selection via the Lasso', *J. Royal Statist. Soc.* B **58**, 267–288.

A. N. Tikhonov (1943), On the stability of inverse problems. *Dokl. Akad. Nauk SSSR* **39**, 195–198.

A. N. Tikhonov (1963), Solution of incorrectly formulated problems and the regularization method. *Dokl. Akad. Nauk.* **151**, 1035–1038.

A. N. Tikhonov and V. Y. Arsenin (1977), *Solutions of Ill-Posed Problems*, Winston.

J. Tompson, K. Schlachter, P. Sprechmann and K. Perlin (2017), Accelerating Eulerian fluid simulation with convolutional networks. arXiv:1607.03597v6

A. Traverso, L. Wee, A. Dekker and R. Gillies (2018), 'Repeatability and reproducibility of radiomic features: A systematic review', *Imaging Radiation Oncology* **102**, 1143–1158.

J. A. Tropp and A. C. Gilbert (2007), 'Signal recovery from random measurements via orthogonal matching pursuit', *IEEE Trans. Inform. Theory* **53**, 4655–4666.

G. Uhlmann and A. Vasy (2012), 'The inverse problem for the local geodesic X-ray transform', *Inventio. Math.* **205**, 83–120.

D. Ulyanov, A. Vedaldi and V. Lempitsky (2018), Deep image prior. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, pp. 9446–9454.

M. Unser and T. Blu (2000), 'Fractional splines and wavelets', *SIAM Review* **42**, 43–67.

K. Valluru, K. Wilson and J. Willmann (2016), 'Photoacoustic imaging in oncology: Translational preclinical and early clinical experience', *Radiology* **280**, 332–349.

C. Van Chung, J. De los Reyes and C.-B. Schönlieb (2017), 'Learning optimal spatially-dependent regularization parameters in total variation image denoising', *Inverse Problems* **33**, 074005.

D. Van Veen, A. Jalal, E. Price, S. Vishwanath and A. G. Dimakis (2018), Compressed sensing with deep image prior and learned regularization. arXiv:1806.06438

Y. Vardi, L. Shepp and L. Kaufman (1985), 'A statistical model for positron emission tomography', *J. Amer. Statist. Assoc.* **80** (389), 8–20.

B. S. Veeling, J. Linmans, J. Winkens, T. Cohen and M. Welling (2018), Rotation equivariant CNNs for digital pathology. arXiv:1806.03962

S. V. Venkatakrishnan, C. A. Bouman and B. Wohlberg (2013), Plug-and-play priors for model based reconstruction. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP 2013)*, pp. 945–948.

R. Vidal, J. Bruna, R. Giryes and S. Soatto (2017), Mathematics of deep learning. arXiv:1712.04741

C. Villani (2009), *Optimal Transport: Old and New*, Vol. 338 of Grundlehren der mathematischen Wissenschaften, Springer.

C. Viroli and G. J. McLachlan (2017), Deep Gaussian mixture models. arXiv:1711.06929

C. Vogel and T. Pock (2017), A primal dual network for low-level vision problems. In *GCPR 2017: Pattern Recognition* (V. Roth and T. Vetter, eds), Vol. 10496 of Lecture Notes in Computer Science, Springer, pp. 189–202.

G. Wang, J. C. Ye, K. Mueller and J. A. Fessler (2018), 'Image reconstruction is a new frontier of machine learning', *IEEE Trans. Medical Imaging* **37**, 1289–1296.

L. V. Wang (2009), 'Multiscale photoacoustic microscopy and computed tomography', *Nature Photonics* **3**, 503–509.

S. Wang, Z. Su, L. Ying, X. Peng, S. Zhu, F. Liang, D. Feng and D. Liang (2016), Accelerating magnetic resonance imaging via deep learning. In *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pp. 514–517.

Y. Wang and D. M. Blei (2017), Frequentist consistency of variational Bayes. arXiv:1705.03439

Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli (2004), 'Image quality assessment: From error visibility to structural similarity', *IEEE Trans. Image Process.* **13**, 600–612.

J. Weickert (1998), *Anisotropic Diffusion in Image Processing*, ECMI series, Teubner.

M. Weiler, M. Geiger, M. Welling, W. Boomsma and T. Cohen (2018), 3D steerable CNNs: Learning rotationally equivariant features in volumetric data. arXiv:1807.02547

J. Weizenecker, B. Gleich, J. Rahmer, H. Dahnke and J. Borgert (2009), 'Three-dimensional real-time *in vivo* magnetic particle imaging', *Phys. Med. Biol.* **54**, L1–L10.

M. Welling and Y. W. Teh (2011), Bayesian learning via stochastic gradient Langevin dynamics. In *28th International Conference on Machine Learning (ICML '11)*, pp. 681–688.

M. Welling, S. Osindero and G. E. Hinton (2003), Learning sparse topographic representations with products of Student-t distributions. In *15th International Conference on Neural Information Processing Systems (NIPS '02)*, MIT Press, pp. 1383–1390.

B. Wohlberg (2014), Efficient convolutional sparse coding. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7173–7177.

J. M. Wolterink, A. M. Dinkla, M. H. F. Savenije, P. R. Seevinck, C. A. T. van den Berg and I. Išgum (2017), Deep MR to CT synthesis using unpaired data. In *Simulation and Synthesis in Medical Imaging SASHIMI 2017* (S. Tsaftaris *et al.*, eds), Vol. 10557 of Lecture Notes in Computer Science, Springer, pp. 14–23.

Y. Wu, P. Zhang, H. Shen and H. Zhai (2018), Visualizing neural network developing perturbation theory. arXiv:1802.03930v2

D. Wu, K. Kim and Q. Li (2018*a*), Computationally efficient cascaded training for deep unrolled network in CT imaging. arXiv:1810.03999v2

D. Wu, K. Kim, G. E. Fakhri and Q. Li (2017), 'Iterative low-dose CT reconstruction with priors trained by artificial neural network', *IEEE Trans. Medical Imaging* **36**, 2479–2486.

T. Würfl, M. Hoffmann, V. Christlein, K. Breininger, Y. Huang, M. Unberath and A. K. Maier (2018), 'Deep learning computed tomography: Learning projection-domain weights from image domain in limited angle problems', *IEEE Trans. Medical Imaging* **37**, 1454–1463.

J. Xia and L. V. Wang (2014), 'Small-animal whole-body photoacoustic tomography: A review', *IEEE Trans. Biomedical Engng* **61**, 1380–1389.

J. Xie, L. Xu and E. Chen (2012), Image denoising and inpainting with deep neural networks. In *Advances in Neural Information Processing Systems 25 (NIPS 2012)* (F. Pereira *et al.*, eds), Curran Associates, pp. 341–349.

Q. Xu, H. Yu, X. Mou, L. Zhang, J. Hsieh and G. Wang (2012), 'Low-dose X-ray CT reconstruction via dictionary learning', *IEEE Trans. Medical Imaging* **31**, 1682–1697.

B. Yang, L. Ying and J. Tang (2018*a*), 'Artificial neural network enhanced bayesian PET image reconstruction', *IEEE Trans. Medical Imaging* **37**, 1297–1309.

G. Yang, S. Yu, H. Dong, G. Slabaugh, P. L. Dragotti, X. Ye, F. Liu, S. Arridge, J. Keegan, Y. Guo and D. Firmin (2018*b*), 'DAGAN: Deep De-Aliasing Generative Adversarial Networks for fast compressed sensing MRI reconstruction', *IEEE Trans. Medical Imaging* **37**, 1310–1321.

Q. Yang, P. Yan, Y. Zhang, H. Yu, Y. Shi, X. Mou, M. K. Kalra, Y. Zhang, L. Sun and G. Wang (2018*c*), 'Low-dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss', *IEEE Trans. Medical Imaging* **37**, 1348–1357.

X. Yang, R. Kwitt, M. Styner and M. Niethammer (2017), 'Quicksilver: Fast predictive image registration: A deep learning approach', *NeuroImage* **158**, 378–396.

Y. Yang, J. Sun, H. Li and Z. Xu (2016), Deep ADMM-Net for compressive sensing MRI. In *Advances in Neural Information Processing Systems 29 (NIPS 2016)* (D. D. Lee *et al.*, eds), Curran Associates, pp. 10–18.

J. C. Ye, Y. S. Han and E. Cha (2018), 'Deep convolutional framelets: A general deep learning for inverse problems', *SIAM J. Imaging Sci.* **11**, 991–1048.

S. Ye, S. Ravishankar, Y. Long and J. A. Fessler (2018*b*), SPULTRA: Low-dose CT image reconstruction with joint statistical and learned image models. arXiv:1808.08791v2

J. Yoo, S. Sabir, D. Heo, K. H. Kim, A. Wahab, Y. Choi, S.-I. Lee, E. Y. Chae, H. H. Kim, Y. M. Bae, Y.-W. Choi, S. Cho and J. C. Ye (2017), Deep learning can reverse photon migration for diffuse optical tomography. arXiv:1712.00912

C. You, G. Li, Y. Zhang, X. Zhang, H. Shan, S. Ju, Z. Zhao, Z. Zhang, W. Cong, M. W. Vannier, P. K. Saha and G. Wang (2018*a*), CT super-resolution GAN constrained by the identical, residual, and cycle learning ensemble (GAN-CIRCLE). arXiv:1808.04256v3

C. You, Q. Yang, H. Shan, L. Gjesteby, G. Li, S. Ju, Z. Zhang, Z. Zhao, Y. Zhang, W. Cong and G. Wang (2018*b*), 'Structurally-sensitive multi-scale deep neural network for low-dose CT denoising', *IEEE Access* **6**, 41839–41855.

J. You and G. L. Zeng (2007), 'Hilbert transform based FBP algorithm for fan-beam CT full and partial scans', *IEEE Trans. Medical Imaging* **26**, 190–199.

E. Y. Yu, M. Bishop, B. Zheng, R. M. Ferguson, A. P. Khandhar, S. J. Kemp, K. M. Krishnan, P. W. Goodwill and S. M. Conolly (2017), 'Magnetic particle imaging: A novel *in vivo* imaging platform for cancer detection', *Nano Letters* **17**, 1648–1654.

M. D. Zeiler, D. Krishnan, G. W. Taylor and R. Fergus (2010), Deconvolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2010)*, pp. 2528–2535.

C. Zhang, T. Zhang, M. Li, C. Peng, Z. Liu and J. Zheng (2016), 'Low-dose CT reconstruction via L1 dictionary learning regularization using iteratively reweighted least-squares', *BioMed. Engng OnLine* **15**, 66.

Y. Zhang and H. Yu (2018), 'Convolutional neural network based metal artifact reduction in X-ray computed tomography', *IEEE Trans. Medical Imaging* **37**, 1370–1381.

Z. Zhang, X. Liang, X. Dong, Y. Xie and G. Cao (2018), 'A sparse-view CT reconstruction method based on combination of densenet and deconvolution', *IEEE Trans. Medical Imaging* **37**, 1407–1417.

C. Zhao, J. Zhang, R. Wang and W. Gao (2018*a*), 'CREAM: CNN-REgularized ADMM framework for compressive-sensed image reconstruction', *IEEE Access* **6**, 76838–76853.

J. Zhao, Z. Chen, L. Zhang and X. Jin (2016), Few-view CT reconstruction method based on deep learning. In *2016 IEEE Nuclear Science Symposium, Medical Imaging Conference and Room-Temperature Semiconductor Detector Workshop (NSS/MIC/RTSD)*.

R. Zhao, Y. Hu, J. Dotzel, C. D. Sa and Z. Zhang (2018), Building efficient deep neural networks with unitary group convolutions. arXiv:1811.07755

X. Zheng, S. Ravishankar, Y. Long and J. A. Fessler (2018), 'PWLS-ULTRA: An efficient clustering and learning-based approach for low-dose 3D CT image reconstruction', *IEEE Trans. Medical Imaging* **37**, 1498–1510.

Y. Zhou, J. Yao and L. V. Wang (2016), 'Tutorial on photoacoustic tomography', *J. Biomedical Optics* **21**, 061007.

B. Zhu, J. Z. Liu, S. F. Cauley, B. R. Rosen and M. S. Rosen (2018), 'Image reconstruction by domain-transform manifold learning', *Nature* **555**, 487–492.

H. Zhu, G. Leus and G. B. Giannakis (2011), 'Sparsity-cognizant total least-squares for perturbed compressive sampling', *IEEE Trans. Signal Process.* **59**, 2002–2016.

S. C. Zhu, Y. Wu and D. Mumford (1998), 'Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling', *Internat. J. Comput. Vision* **27**, 107–126.

D. Zoran and Y. Weiss (2011), From learning models of natural image patches to whole image restoration. In *IEEE International Conference on Computer Vision (ICCV 2011)*, pp. 479–486.