NETT: solving inverse problems with deep neural networks

OPEN ACCESS
IOP Publishing
Inverse Problems
Inverse Problems **36** (2020) 065005 (23pp)
https://doi.org/10.1088/1361-6420/ab6d57

NETT: solving inverse problems with deep neural networks

Housen Li[1], **Johannes Schwab**[2], **Stephan Antholzer**[2] **and Markus Haltmeier**[2,3]

[1] Institute for Mathematical Stochastics, University of Göttingen, Goldschmidtstrasse 7, 37077 Göttingen, Germany
[2] Department of Mathematics, University of Innsbruck, Technikerstrasse 13, 6020 Innsbruck, Austria

E-mail: markus.haltmeier@uibk.ac.at

Received 13 July 2019, revised 9 January 2020
Accepted for publication 16 January 2020
Published 28 May 2020

Abstract

Recovering a function or high-dimensional parameter vector from indirect measurements is a central task in various scientific areas. Several methods for solving such inverse problems are well developed and well understood. Recently, novel algorithms using deep learning and neural networks for inverse problems appeared. While still in their infancy, these techniques show astonishing performance for applications like low-dose CT or various sparse data problems. However, there are few theoretical results for deep learning in inverse problems. In this paper, we establish a complete convergence analysis for the proposed NETT (network Tikhonov) approach to inverse problems. NETT considers nearly data-consistent solutions having small value of a regularizer defined by a trained neural network. We derive well-posedness results and quantitative error estimates, and propose a possible strategy for training the regularizer. Our theoretical results and framework are different from any previous work using neural networks for solving inverse problems. A possible data driven regularizer is proposed. Numerical results are presented for a tomographic sparse data problem, which demonstrate good performance of NETT even for unknowns of different type from the training data. To derive the convergence and convergence rates results we introduce a new framework based on the absolute Bregman distance generalizing the standard Bregman distance from the convex to the non-convex case.

[3] The author to whom any correspondence should be addressed.

Keywords: convergence analysis, convolutional neural networks, deep learning, total nonlinearity, inverse problems, absolute Bregman distance, image reconstruction

(Some figures may appear in colour only in the online journal)

1. Introduction

We study the stable solution of inverse problems of the form

$$\text{Estimate } x \in D \text{ from data } \quad y_\delta = \mathbf{F}(x) + \xi_\delta. \tag{1.1}$$

Here $\mathbf{F} : D \subseteq X \to Y$ is a possibly non-linear operator between reflexive Banach spaces $(X, \|\cdot\|)$ and $(Y, \|\cdot\|)$ with domain D. We allow a possibly infinite-dimensional function space setting, but clearly the approach and results apply to a finite dimensional setting as well. The element $\xi_\delta \in Y$ models the unknown data error (noise) which is assumed to satisfy the estimate $\|\xi_\delta\| \leq \delta$ for some noise level $\delta \geq 0$. We focus on the ill-posed (or ill-conditioned) case where without additional information, the solution of (1.1) is either highly unstable, highly underdetermined, or both. Many inverse problems in biomedical imaging, geophysics, engineering sciences, or elsewhere can be written in such a form (see, for example, [18, 38, 45]).

The stable solution of inverse problems requires regularization methods, which are based on approximating (1.1) by neighboring well-posed problems that enforce stability and uniqueness. On the other hand, recently, several deep learning approaches for inverse problems have been developed (see[4] for example, [1, 5, 13, 28, 30–32, 46, 48, 51, 53, 54]). In these approaches, a trained network $\mathbf{\Phi}_{\text{rec}}(\mathbb{V}, \cdot) : Y \to X$ maps measured data to the desired output image. Two-step reconstruction networks take the form $\mathbf{\Phi}_{\text{rec}}(\mathbb{V}, \cdot) = \mathbf{\Phi}_{\text{CNN}}(\mathbb{V}, \cdot) \circ \mathbf{B}$, where $\mathbf{B} : Y \to X$ maps the data to the reconstruction space (backprojection; no free parameters) and $\mathbf{\Phi}_{\text{CNN}}(\mathbb{V}, \cdot) : X \to X$ is a neural network, for example a convolutional neural network (CNN), whose free parameters are adjusted to the training data. This basic form allows the use of well established CNNs for image reconstruction [20] and already demonstrates impressing results. Network cascades [32, 46] and trained iterative schemes [1, 2, 12, 27] learn free parameters in iterative schemes. In such approaches, the reconstruction network can be written in the form $\mathbf{\Phi}_{\text{rec}}(\mathbb{V}, y) = (\mathbf{\Phi}_N(\mathbb{V}_N, \cdot) \circ \mathbf{B}_N(y, \cdot) \circ \cdots \circ \mathbf{\Phi}_1(\mathbb{V}_1, \cdot) \circ \mathbf{B}_1(y, \cdot))(x_0)$, where x_0 is the initial guess, $\mathbf{\Phi}_k(\mathbb{V}_k, \cdot) : X \to X$ are CNNs that can be trained, and $\mathbf{B}_k(y, \cdot) : X \to X$ are iterative updates based on the forward operator and the data. The iterative updates may be defined by a gradient step with respect to the given inverse problem. The free parameters are again adjusted to available training data.

In this paper we introduce the NETT as a deep learning framework that uses neural networks as regularization terms and that is shown to provide a convergent regularization method. We note that an early related work [43] uses denoisers as regularizers which also includes certain CNNs. In [2], the norm of a residual network is used as a regularizer. Another related work [12] uses a learned proximal operator instead of a regularization term. After the present paper was initially submitted, other works explored the idea of neural networks as regularizers. In

[4] We initially submitted our paper to a recognized journal in February 28, 2018. On June 18, 2019 (first decision date), we have been informed that the paper is rejected. Since so much work has been done in the emerging field of deep learning on inverse problems, for the present version, we have not updated the reference list with all the interesting papers, but only with closely related work.

particular, in [35] a regularizer has been proposed that distinguishes the distributions of desired images and noisy images. In [39] a related synthesis approach has been proposed. We note, however, that neither convergence nor convergence rates results have been derived by any work using neural networks as regularizer.

1.1. NETT framework

Any method for the stable solution of (1.1) uses, either implicitly or explicitly, *a priori* information about the unknowns to be recovered. Such information can be that x belongs to a certain set of admissible elements or that x has small value of a regularizer (or regularization functional) $\mathcal{R}: X \to [0, \infty]$. In this paper we focus on the latter situation and assume that the regularizer takes the form

$$\forall x \in X: \quad \mathcal{R}(x) = \mathcal{R}(\mathbb{V}, x) := \psi(\mathbf{\Phi}(\mathbb{V}, x)). \tag{1.2}$$

Here $\psi: \mathbb{X}_L \to [0, \infty]$ is a scalar functional and $\mathbf{\Phi}(\mathbb{V}, \cdot): X \to \mathbb{X}_L$ a neural network of depth L where $\mathbb{V} \in \mathcal{V}$, for some vector space $\mathcal{V}$, contains free parameters that can be adjusted to available training data (see section 2.1 for a precise formulation). With the regularizer (1.2), we approach (1.1) via

$$\mathcal{T}_{\alpha; y_\delta}(x) := \mathcal{D}(\mathbf{F}(x), y_\delta) + \alpha \mathcal{R}(\mathbb{V}, x) \to \min_{x \in D}, \tag{1.3}$$

where $\mathcal{D}: Y \times Y \to [0, \infty]$ is an appropriate similarity measure in the data space enforcing data consistency. One may take $\mathcal{D}(\mathbf{F}(x), y_\delta) = \|\mathbf{F}(x) - y_\delta\|^2$ but also other similarity measures such as the Kullback–Leibler divergence (which, among others, is used in emission tomography) are reasonable choices. Optimization problem (1.3) can be seen as a particular instance of generalized Tikhonov regularization for solving (1.1) with a neural network as regularizer. We therefore name (1.3) network Tikhonov (NETT) approach for inverse problems.

The network regularizer $\mathcal{R}(\mathbb{V}, \cdot)$ can either be user-specified, or a trained network, where free parameters are adjusted on appropriate training data. Some examples are as follows.

(a) Non-linear ℓ^q-regularizer: a simple user-specified instance of the regularizer (1.2) is the ℓ^q-regularizer $\mathcal{R}(\mathbb{V}, x) = \sum_{\lambda \in \Lambda} v_\lambda |\langle x, \varphi_\lambda \rangle|^q$. Here $(\varphi_\lambda)_{\lambda \in \Lambda}$ is a prescribed basis or frame and $(v_\lambda)_{\lambda \in \Lambda}$ are weights. In this case, the neural network is simply given by the analysis operator $\mathbf{\Phi}(\mathbb{V}, \cdot): X \to \ell^2(\Lambda): x \mapsto \big(\langle x, \varphi_\lambda \rangle\big)_{\lambda \in \Lambda}$ and NETT regularization reduces to sparse ℓ^q-regularization [17, 22, 23, 34, 41]. This form of the regularizer can also be combined with a training procedure by adjusting the weights $(v_\lambda)_{\lambda \in \Lambda}$ to a class of training data.

In this paper we study a non-linear extension of ℓ^q-regularization, where the (in general) non-convex network regularizer takes the form

$$\mathcal{R}(\mathbb{V}, x) = \sum_{\lambda \in \Lambda} v_\lambda |\mathbf{\Phi}_\lambda(\mathbb{V}, x)|^q, \tag{1.4}$$

with $q \geq 1$ and $\mathbf{\Phi}(\mathbb{V}, \cdot) = (\mathbf{\Phi}_\lambda(\mathbb{V}, \cdot))_{\lambda \in \Lambda}$ being a possible non-linear neural network with multiple layers. In section 3.4 we present convergence results for this non-linear generalization of ℓ^q-regularization. By selecting non-negative weights, one can easily construct networks that are convex with respect to the inputs [4]. In this work, however, we consider the general situation of arbitrary weights, in which the network regularizer (1.4) can be non-convex.

(b) CNN regularizer: the network regularizer $\mathcal{R}(\mathbb{V}, \cdot)$ in (1.2) may also be defined by a convolutional neural network (CNN) $\mathbf{\Phi}(\mathbb{V}, \cdot)$, containing free parameters that can be

adjusted on appropriate training data. The CNN can be trained in such a way, that the regularizer has small value for elements x in a set of training phantoms and larger value on a class of un-desirable phantoms. The class of un-desirable phantoms can be elements containing undersampling artifacts, noise, or both. In section 5, we present a possible regularizer design together with a strategy for training the CNN to remove undersampling artifacts. We present numerical results demonstrating that our approach performs well in practice for a sparse tomographic data problem.

1.2. Main contributions

In this paper, we show that under reasonable assumptions, the NETT approach (1.3) is stably solvable. As $\delta \to 0$, the regularized solutions $x_{\alpha,\delta} \in \arg\min_x \mathcal{T}_{\alpha;y_\delta}(x)$ are shown to converge to $\mathcal{R}(\mathbb{V},\cdot)$-minimizing solutions of $\mathbf{F}(x) = y_0$. Here and below $\mathcal{R}(\mathbb{V},\cdot)$-minimizing solutions of $\mathbf{F}(x) = y_0$ are defined as any element

$$x_+ \in \arg\min \{\mathcal{R}(\mathbb{V},x) \mid x \in D \wedge \mathbf{F}(x) = y_0\}. \tag{1.5}$$

Additionally, we derive convergence rates (quantitative error estimates) between $\mathcal{R}(\mathbb{V},\cdot)$-minimizing solutions x_+ and regularized solutions $x_{\alpha,\delta}$. As a consequence, (1.3) provides a stable solution scheme for (1.1) using near data-consistency and encoding *a priori* knowledge via neural networks. For proving norm convergence and convergence rates, we introduce the absolute Bregman distance as a new generalization of the standard Bregman distance for non-convex regularization.

The results in this paper are a main step for the regularization of inversion problems with neural networks. For the first time, we present a complete convergence analysis and derive convergence rate under reasonable assumptions. Many additional issues can be addressed in future work. This includes the design of appropriate CNN regularizers, the development of efficient algorithms for minimizing (1.3), and the consideration of other regularization strategies for (1.5). The focus of the present paper is on the theoretical analysis of NETT and demonstrating the feasibility of our approach; detailed comparison with other methods in terms of reconstruction quality, computational performance and applicability to real-world data is beyond our scope here and will be addressed in future work.

In contrast to network cascades, iterative schemes and two-step reconstruction networks, the proposed NETT approach bounds the data consistency term $\mathcal{D}(\mathbf{F}(x_{\alpha,\delta}), y)$ also for data outside the training set. We expect the combination of the forward problem and a neural network via (1.3) (or, for the noiseless case, (1.5)) to increase reconstruction quality, especially in the case of limited access to a large amount of appropriate training data. We point out that the formulation of NETT (1.3) separates the noise characteristic and the *a priori* information of unknowns. This allows us to incorporate the knowledge of data generating mechanism, e.g. Poisson noise or Gaussian noise, by choosing the corresponding log-likelihood as the data consistency term, and also simplifies the training process of $\mathcal{R}(\mathbb{V},\cdot)$, as it to some extend avoids the impact of noise. Meanwhile, this enhances the interpretability of the resulting approach: we on the one hand require its fidelity to the data, and on the other penalize unfavorable features (e.g. artifacts in tomography).

1.3. Outline

The rest of this paper is organized as follows. In section 2, we describe the proposed NETT framework for solving inverse problems. We show its stability and derive convergence in the weak topology (see theorem 2.6). To obtain the strong convergence of NETT, we introduce a new notion of total non-linearity of non-convex functionals. For totally non-linear regularizers,

we show norm convergence of NETT (see theorem 2.11). Convergence rates (quantitative error estimates) for NETT are derived in section 3. Among others, we derive a convergence rate result in terms of the absolute Bregman distance (see proposition 3.3). A framework for learning the data driven regularizer is proposed in section 4, and applied to a sparse data problem in photoacoustic tomography in section 5. The paper concludes with a short summary and outlook presented in section 6.

2. NETT regularization

In the section we introduce the proposed NETT and analyze its well-posedness (existence, stability and weak convergence). We introduce the novel concepts of absolute Bregman distance and total non-linearity, which are applied to establish convergence of NETT with respect to the norm.

2.1. The NETT framework

Our goal is to solve (1.1) with $\|\xi_\delta\| \leq \delta$ and $\delta > 0$. For that purpose we consider minimizing the NETT functional (1.3), where the regularizer $\mathcal{R}(\mathbb{V}, \cdot) : X \to [0, \infty]$ in (1.2) is defined by a neural network of the form

$$\mathbf{\Phi}(\mathbb{V}, x) := (\sigma_L \circ \mathbb{V}_L \circ \sigma_{L-1} \circ \mathbb{V}_{L-1} \circ \cdots \circ \sigma_1 \circ \mathbb{V}_1)(x). \tag{2.1}$$

Here L is the depth of the network (the number of layers after the input layer) and $\mathbb{V}_\ell(x) = \mathbb{A}_\ell(x) + b_\ell$ are affine linear operators between Banach spaces $\mathbb{X}_{\ell-1}$ and $\mathbb{X}_{\ell-1/2}$; we take $\mathbb{X}_0 := X$. The operators $\mathbb{A}_\ell : \mathbb{X}_{\ell-1} \to \mathbb{X}_{\ell-1/2}$ are the linear parts and $b_\ell \in \mathbb{X}_{\ell-1/2}$ the so-called bias terms. The operators $\sigma_\ell : \mathbb{X}_{\ell-1/2} \to \mathbb{X}_\ell$ are possibly non-linear and the functionals $\psi : \mathbb{X}_L \to [0, \infty]$ are possibly non-convex. Note that we use two different spaces $\mathbb{X}_{\ell-1}$ and $\mathbb{X}_{\ell-1/2}$ in each layer because common operations in networks like max-pooling, downsampling or upsampling change the domain space.

As common in machine learning, the affine mappings $\mathbb{V}_\ell$ depend on free parameters that can be adjusted in the training phase, whereas the non-linearities σ_ℓ are fixed. Therefore $\mathbb{V}_\ell$ and σ_ℓ are treated separately and only the affine part $\mathbb{V} = (\mathbb{V}_\ell)_{\ell=1}^L$ is indicated in the notion of the neural network regularizer $\mathcal{R}(\mathbb{V}, \cdot)$. Throughout our theoretical analysis we assume $\mathcal{R}(\mathbb{V}, \cdot)$ to be given and all free parameters to be trained before the minimization of (1.3). In section 4, we present a possible framework for training a neural network regularizer.

Remark 2.1 (CNNs in Banach space setting). A typical instance for the neural network in NETT (1.2), is a deep convolutional neural network (CNN). In a possible infinite dimensional setting, such CNNs can be written in the form (2.1), where the involved spaces satisfy $\mathbb{X}_\ell := \ell^p(\Lambda_\ell, X_\ell)$ and $\mathbb{X}_{\ell-1/2} := \ell^p(\Lambda_\ell, X_{\ell-1/2})$ with $p \geq 1$, X_ℓ and $X_{\ell-1/2}$ being function spaces, and Λ_ℓ being an at most countable set that specifies the number of different filters (depth) of the ℓth layer. The linear operators $\mathbb{A}_\ell : \ell^p(\Lambda_{\ell-1}, X_{\ell-1}) \to \ell^p(\Lambda_\ell, X_{\ell-1/2})$ are taken as

$$\forall x \in \ell^p(\Lambda_{\ell-1}, X_{\ell-1})\, \forall \ell \in \Lambda_\ell : \quad \mathbb{A}_\ell(x) = \left(\sum_{\mu \in \Lambda_{\ell-1}} K_{\lambda,\mu}^{(\ell)}(x_\mu) \right)_{\lambda \in \Lambda_\ell}, \tag{2.2}$$

where $K_{\lambda,\mu}^{(\ell)} : X_{\ell-1} \to X_{\ell-1/2}$ are convolution operators.

We point out, that in the existing machine learning literature, only finite dimensional settings have been considered so far, where $\mathbb{X}_\ell$ and $\mathbb{X}_{\ell-1/2}$ are finite dimensional spaces. In such a finite dimensional setting, we can take $X_\ell = \mathbb{R}^{N_1^\ell \times N_2^\ell}$, and Λ_ℓ as a set with N_c^ℓ elements. One then can identify $\mathbb{X}_\ell = \ell^p(\Lambda_\ell, X_\ell) \simeq \mathbb{R}^{N_1^\ell \times N_2^\ell \times N_c^\ell}$ and interpreted its elements as stack of discrete images (the same holds for $\mathbb{X}_{\ell-1/2}$). In typical CNNs, either the dimensions $N_1^\ell \times N_2^\ell$ of the base space X_ℓ are progressively reduced and number of channels N_c^ℓ increased, or vice versa. While we are not aware of any infinite dimensional general formulation of CNNs, our proposed formulation (2.1), (2.2) is the natural infinite-dimensional Banach space version of CNNs, which reduces to standard CNNs [20] in the finite dimensional setting.

Basic convex regularizers are sparse ℓ^q-penalties $\mathcal{R}(\mathbb{V}, x) = \sum_{\lambda\in\Lambda} \nu_\lambda |\langle \varphi_\lambda, x\rangle|^q$. In this case one may take (2.1) as a single-layer neural network with $\mathbb{X}_1 = \ell^2(\Lambda_1, \mathbb{R})$, $\sigma = Id$ and $\mathbf{\Phi}(\mathbb{V}, x) = \mathbb{V}(x) = (\langle \varphi_\lambda, x\rangle)_\lambda$ being the analysis operator to some frame $(\varphi_\lambda)_{\lambda\in\Lambda}$. The functional $\psi(x) = \sum_{\lambda\in\Lambda} \nu_\lambda |x_\lambda|^q$ is a weighted ℓ^q-norm. The frame $(\varphi_\lambda)_{\lambda\in\Lambda}$ may be a prescribed wavelet or curvelet basis [11, 16, 37] or a trained dictionary [3, 25]. In section 3.4, we analyze a non-linear version of ℓ^q-regularization, where $\langle \phi_\lambda, \cdot\rangle$ are replaced by non-linear functionals. In this case the resulting regularizer will in general be non-convex even if $q \geq 1$.

2.2. Well-posedness and weak convergence

For the convergence analysis of NETT regularization, we use the following assumptions on the regularizer and the data consistency term in (1.3).

Condition 2.2 (Convergence of NETT regularization).

(a) Network regularizer $\mathcal{R}$:
- 1 The regularizer $\mathcal{R}(\mathbb{V}, \cdot)$ is defined by (1.2) and (2.1)
- 2 $\mathbb{V}_\ell : \mathbb{X}_{\ell-1} \to \mathbb{X}_{\ell-1/2}$ are affine operators of the form $\mathbb{V}_\ell(x) = \mathbb{A}_\ell x + b_\ell$;
- 3 $\mathbb{A}_\ell$ are bounded linear;
- 4 σ_ℓ are weakly continuous;
- 5 The functional ψ is weakly lower semi-continuous.

(b) Data consistency term $\mathcal{D}$:
- 1 For some $\tau \geq 1$ we have $\forall y_0,\, y_1,\, y_2 \in Y : \mathcal{D}(y_0, y_1) \leq \tau\mathcal{D}(y_0, y_2) + \tau\mathcal{D}(y_2, y_1)$;
- 2 $\forall y_0,\, y_1 \in Y : \mathcal{D}(y_0, y_1) = 0 \iff y_0 = y_1$;
- 3 $\forall (y_k)_{k\in\mathbb{N}} \in Y^{\mathbb{N}} : y_k \to y \implies \mathcal{D}(y_k, y) \to 0$;
- 4 The functional $(x, y) \mapsto \mathcal{D}(\mathbf{F}(x), y)$ is sequentially lower semi-continuous.

(c) Coercivity condition:
$\mathcal{R}(\mathbb{V}, \cdot)$ is coercive, that is $\mathcal{R}(\mathbb{V}, x) \to \infty$ as $\|x\| \to \infty$.

The conditions in (a) guarantees the lower semicontinuity of the regularizer. The conditions in (b) for the data consistency term are not very restrictive and, for example, are satisfied for the squared norm distance. The coercivity condition (c) might be the most restrictive condition. Several strategies how it can be obtained are discussed in the following.

Remark 2.3 (Coercivity via skip or residual connections). Coercivity (c) clearly holds for regularizers of the form

$$\mathcal{R}(\mathbb{V}, \cdot) = \mathcal{R}^{(1)}(\mathbb{V}, \cdot) + \psi^{(2)}(x), \tag{2.3}$$

where $\mathcal{R}^{(1)}(\mathbb{V},\cdot)$ is a trained regularizer as in (a) and $\psi^{(2)}$ is coercive and weakly lower semi-continuous. The regularizer (2.3) fits to our general framework and results from a network using a skip connection between the input and the output layer. In this case, the overall network takes the form $\mathbf{\Phi}(\mathbb{V},x) = [\mathbf{\Phi}^{(1)}(\mathbb{V},x),x]$ where $\mathbf{\Phi}^{(1)}(\mathbb{V},x)$ is of the form (2.1).

Another possibility to obtain coercivity is to use a residual connection in the network structure which results in a regularizer of the form

$$\mathcal{R}(\mathbb{V},x) = \psi\left(\mathbf{\Phi}^{(r)}(\mathbb{V},x) - x\right). \tag{2.4}$$

If the last non-linearity σ_ℓ in the network $\mathbf{\Phi}^{(r)}(\mathbb{V},x)$ is a bounded function and the functional ψ is coercive, then the resulting regularizer is coercive. Coercivity also holds if $\mathbf{\Phi}^{(r)}(\mathbb{V},x)$ has Lipschitz constant < 1, which can be achieved by appropriate training [7].

Remark 2.4 (Layer-wise coercivity). A set of specific conditions that implies coercivity of the regularizer is to assume that, for all ℓ, the activation functions σ_ℓ are coercive and there exists $c_\ell \in [0,\infty)$ such that $\forall x \in X : \|x\| \leq c_\ell \|\mathbb{A}_\ell x\|$. The coercivity of $\mathbb{A}_\ell$ can be obtained by including a skip connection, in which case the operator $\mathbb{A}_\ell$ takes the form $\mathbb{A}_\ell(x) = [\mathbb{A}_\ell^{(1)}(x),x]$, where $\mathbb{A}_\ell$ is bounded linear.

In CNNs, the spaces $\mathbb{X}_\ell$ and $\mathbb{X}_{\ell-1/2}$ are function spaces (see remark 2.1) and a standard operation for σ_ℓ is the ReLU (the rectified linear unit), $\mathrm{ReLU}(x) := \max\{x,0\}$, that is applied component-wise. The plain form of the ReLU is not coercive. However, the slight modification $x \mapsto \max\{x,ax\}$ for some $a \in (0,1)$, named leaky ReLU, is coercive, see [29, 36]. Another coercive standard operation for σ_ℓ in CNNs is max pooling which takes the maximum value $\max\{|x(i)| : i \in I_k\}$ within clusters of transform coefficients. We emphasize however that by using one of the strategies described in remark 2.3, one can use any common activation function without worrying about its coercivity.

Remark 2.5 (Generalization of the coercivity condition). The results derived below also hold under the following weaker alternative to the coercivity condition (c) in condition 2.2:

(c′) For all $y \in Y$ and $\alpha > 0$, there exists a $C > 0$ such that

$$\{x \in X \mid \mathcal{D}(\mathbf{F}(x),y) + \alpha\mathcal{R}(\mathbb{V},x) \leq C\} \quad \text{is nonempty and bounded in } X. \tag{2.5}$$

Condition (c′) ensures that the level set in (2.5) is sequentially weakly pre-compact for all $y \in Y$ and $\alpha > 0$. It is indeed weaker than Condition (c). For instance, in case that $\mathbf{F}$ is linear and $\mathcal{D}(\cdot,0)$ is convex, (c′) amounts to require that $\mathcal{R}(\mathbb{V},\cdot)$ is coercive on the null space of $\mathbf{F}$, whereas (c) requires coercivity of $\mathcal{R}(\mathbb{V},\cdot)$ on the whole space X.

Now we are ready to show that NETT provides a convergent regularization method.

Theorem 2.6 *(Well-posedness and convergence of NETT-regularization). Let condition 2.2 be satisfied. Then the following assertions hold true:*

(a) Existence: for all $y \in Y$ and $\alpha > 0$, there exists a minimizer of $\mathcal{T}_{\alpha;y}$.

(b) Stability: if $y_k \to y$ and $x_k \in \operatorname{argmin}\mathcal{T}_{\alpha;y_k}$, then weak accumulation points of $(x_k)_{k\in\mathbb{N}}$ exist and are minimizers of $\mathcal{T}_{\alpha;y}$.

(c) Convergence: let $x \in X$, $y := \mathbf{F}(x)$, $(y_k)_{k\in\mathbb{N}}$ satisfy $\mathcal{D}(y_k,y), \mathcal{D}(y,y_k) \leq \delta_k$ for some sequence $(\delta_k)_{k\in\mathbb{N}} \in (0,\infty)^{\mathbb{N}}$ with $\delta_k \to 0$, suppose $x_k \in \arg\min_x \mathcal{T}_{\alpha(\delta_k),\,y_k}(x)$ and let the parameter choice $\alpha : (0,\infty) \to (0,\infty)$ satisfy

$$\lim_{\delta\to 0}\alpha(\delta)=\lim_{\delta\to 0}\frac{\delta}{\alpha(\delta)}=0. \tag{2.6}$$

Then the following holds:

1 *Weak accumulation points of* $(x_k)_{k\in\mathbb{N}}$ *are* $\mathcal{R}(\mathbb{V},\cdot)$*-minimizing solutions of* $\mathbf{F}(x)=y$.
2 $(x_k)_{k\in\mathbb{N}}$ *has at least one weak accumulation point* x_+.
3 *Any subsequence* $(x_{k(n)})_{n\in\mathbb{N}}$ *weakly converging to* x_+ *satisfies* $\mathcal{R}(\mathbb{V},x_{k(n)})\to\mathcal{R}(\mathbb{V},x_+)$.
4 *If the* $\mathcal{R}(\mathbb{V},\cdot)$*-minimizing solution of* $\boldsymbol{F}(x)=y$ *is unique, then* $x_k\rightharpoonup x_+$.

Proof. According to [21, 45] it is sufficient to show that the functional $\mathcal{R}(\mathbb{V},\cdot)$ is weakly sequentially lower semi-continuous and the set $\{x\mid\mathcal{T}_{\alpha;y}(x)\leq t\}$ is sequentially weakly pre-compact for all $t>0$ and $y\in Y$ and $\alpha>0$. By the Banach–Alaoglu theorem, the latter condition is satisfied if $\mathcal{R}(\mathbb{V},\cdot)$ is coercive. The coercivity of $\mathcal{R}(\mathbb{V},\cdot)$ however is assumed in condition 2.2 (for sufficient coercivity conditions see remarks 2.3 and 2.4). Also from condition 2.2 it follows that $\mathcal{R}(\mathbb{V},\cdot)$ is sequentially lower semi-continuous. □

Note that the convergence and stability results of theorem 2.6 are valid for any data independent of the training data used for optimizing the network regularizer. Clearly, if the considered inverse problem is under-determined, a $\mathcal{R}(\mathbb{V},\cdot)$-minimizing solutions is not necessarily the one corresponding to the desired signal class for test data very different from this class.

2.3. Absolute Bregman distance and total non-linearity

For convex regularizers, the notion of Bregman distance is a powerful concept [8, 45]. For non-convex regularizers, the standard definition of the Bregman distance takes negative values. In this paper, we therefore use the notion of absolute Bregman distance. To the best of our knowledge, the absolute Bregman distance has not been used in regularization theory so far.

Definition 2.7 (Absolute Bregman distance). Let $\mathcal{F}:D\subseteq X\to\mathbb{R}$ be Gâteaux differentiable at $x\in X$. The *absolute Bregman distance* $\mathcal{B}_{\mathcal{F}}(\cdot,x):D\to[0,\infty]$ with respect to $\mathcal{F}$ at x is defined by

$$\forall\tilde{x}\in X:\quad \mathcal{B}_{\mathcal{F}}(\tilde{x},x):=\left|\mathcal{F}(\tilde{x})-\mathcal{F}(x)-\mathcal{F}'(x)(\tilde{x}-x)\right|. \tag{2.7}$$

Here $\mathcal{F}'(x)$ denotes the Gâteaux derivative of $\mathcal{F}$ at x.

From theorem 2.6 we can conclude convergence of $x_{\alpha,\delta}$ to the exact solution in the absolute Bregman distance. Below we show that this implies strong convergence under some additional assumption on the regularization functional. For this purpose we introduce the new total non-linearity, which has not been studied before.

Definition 2.8 (Total non-linearity). Let $\mathcal{F}:D\subseteq X\to\mathbb{R}$ be Gâteaux differentiable at $x\in D$. We define the *modulus of total non-linearity* of $\mathcal{F}$ at x as $\nu_{\mathcal{F}}(x,\cdot):[0,\infty)\to[0,\infty]$:

$$\forall t>0:\quad \nu_{\mathcal{F}}(x,t):=\inf\left\{\mathcal{B}_{\mathcal{F}}(\tilde{x},x)\mid\tilde{x}\in D\wedge\|\tilde{x}-x\|=t\right\}. \tag{2.8}$$

The function $\mathcal{F}$ is called *totally non-linear* at x if $\nu_{\mathcal{F}}(x,t)>0$ for all $t\in(0,\infty)$.

The notion of total non-linearity is similar to total convexity [10] for convex functionals. Opposed to total convexity we do not assume convexity of $\mathcal{F}$, and use the absolute Bregman

distance instead of the standard Bregman distance. For convex functions, the total non-linearity reduces to total convexity, as the Bregman distance is always non-negative for convex functionals. For a Gâteaux differentiable convex function, the total non-linearity essentially requires that its second derivative at x is bounded away from zero. The functional $\mathcal{F}(x) := \sum_{\lambda\in\Lambda} v_\lambda |x_\lambda|^q$ with $v_\lambda > 0$ is totally non-linear at every $x = (x_\lambda)_{\lambda\in\Lambda} \in \ell^\infty(\Lambda)$ if $q > 1$.

We have the following result, which generalizes [42, proposition 2.2] (see also [45, theorem 3.49]) from the convex to the non-convex case.

Proposition 2.9 *(Characterization of total non-linearity). For $\mathcal{F} : D \subseteq X \to \mathbb{R}$ and any $x \in D$ the following assertions are equivalent:*

(a) The function $\mathcal{F}$ is totally non-linear at x;

(b) $\forall (x_n)_{n\in\mathbb{N}} \subseteq D^{\mathbb{N}} : (\lim_{n\to\infty} \mathcal{B}_{\mathcal{F}}(x_n, x) = 0 \wedge (x_n)_n$ bounded$) \Rightarrow \lim_{n\to\infty} \|x_n - x\| = 0$.

Proof. The proof of the implication (b) $\Rightarrow$ (a) is the same as [42, proposition 2.2]. For the implication (a) $\Rightarrow$ (b), let (a) hold, let $(x_n)_{n\in\mathbb{N}} \subseteq D^{\mathbb{N}}$ satisfy $\mathcal{B}_{\mathcal{F}}(x_n, x) \to 0$, and suppose $\lim_{n\to\infty} \|x_n - x\| = \delta > 0$ for the moment. For any $\varepsilon > 0$, by the continuity of $\mathcal{B}_{\mathcal{F}}(\cdot, x)$, there exist $\tilde{x}_n$ with $\|\tilde{x}_n - x\| = \delta$ such that for sufficiently large n

$$\varepsilon \geq \mathcal{B}_{\mathcal{F}}(x_n, x) + \frac{\varepsilon}{2} \geq \mathcal{B}_{\mathcal{F}}(\tilde{x}_n, x) \geq \nu_{\mathcal{F}}(x, \delta).$$

This leads to $\nu_{\mathcal{F}}(x, \delta) = 0$, which contradicts with the total non-linearity of $\mathcal{F}$ at x. Then, the assertion follows by considering subsequences of $(x_n)_{n\in\mathbb{N}}$. □

Remark 2.10. We point out that proposition 2.9 remains true, if we replace the absolute value $|\cdot| : \mathbb{R} \to [0, \infty]$ in (2.7) by the ReLU function, the leaky ReLU function, or any other nonnegative continuous function $\phi : \mathbb{R} \to [0, \infty]$ that satisfies $\phi(0) = 0$.

2.4. Strong convergence of NETT regularization

For totally non-linear regularizers $\mathcal{R}(\mathbb{V}, \cdot)$ we can prove convergence of NETT with respect to the norm topology.

Theorem 2.11 *(Strong convergence of NETT). Let condition 2.2 hold and assume additionally that $\mathbf{F}(x) = y$ has a solution, $\mathcal{R}(\mathbb{V}, \cdot)$ is totally non-linear at $\mathcal{R}(\mathbb{V}, \cdot)$-minimizing solutions, and α satisfies (2.6). Then for every sequence $(y_k)_{k\in\mathbb{N}}$ with $\mathcal{D}(y_k, y), \mathcal{D}(y, y_k) \leq \delta_k$ where $\delta_k \to 0$ and every sequence $x_k \in \arg\min_x \mathcal{T}_{\alpha(\delta_k), y_k}(x)$ there exist a subsequence $(x_{k(n)})_{n\in\mathbb{N}}$ and an $\mathcal{R}(\mathbb{V}, \cdot)$-minimizing solution x_+ with $\|x_{k(n)} - x_+\| \to 0$. If the $\mathcal{R}(\mathbb{V}, \cdot)$-minimizing solution is unique, then $x_k \to x_+$ with respect to the norm topology.*

Proof. It follows from theorem 2.6 that there exists a subsequence $(x_{k(n)})_{n\in\mathbb{N}}$ weakly converging to some $\mathcal{R}(\mathbb{V}, \cdot)$-minimizing solution x_+ such that $\mathcal{R}(\mathbb{V}, x_{k(n)}) \to \mathcal{R}(\mathbb{V}, x_+)$. From the weak convergence of $(x_{k(n)})_{n\in\mathbb{N}}$ and the convergence of $(\mathcal{R}(\mathbb{V}, x_{k(n)}))_{n\in\mathbb{N}}$ it follows that $\mathcal{B}_{\mathcal{R}(\mathbb{V},\cdot)}(x_{k(n)}, x_+) \to 0$. Thus it follows from proposition 2.9, that $\|x_{k(n)} - x_+\| \to 0$. If x_+ is the unique $\mathcal{R}$-minimizing solution, the strong convergence to x_+ again follows from theorem 2.6 and proposition 2.9. □

3. Convergence rates

In this section, we derive convergence rates for NETT in terms of general error measures under certain variational inequalities. Throughout we denote by $\delta > 0$ the noise level and $\alpha > 0$

the regularization parameter. We discuss instances where the variational inequality is satisfied for the absolute Bregman distance. Additionally, we consider a non-linear generalization of ℓ^q-regularization.

3.1. General convergence rates result

We study convergence rates in terms of a general functional $\mathcal{E}: X \times X \to [0, \infty]$ measuring closeness in the space X. For convex $\Psi: [0, \infty) \to [0, \infty)$, let $\Psi^*: \mathbb{R} \to \mathbb{R}$ denote the Fenchel conjugate of Ψ defined by $\Psi^*(\tau) := \sup\{\tau t - \Psi(t) \mid t \geq 0\}$.

Theorem 3.1 *(Convergence rates for NETT). Suppose $\mathcal{E}: X \times X \to [0, \infty]$, let $x_+ \in D$ and assume that there exist a concave, continuous and strictly increasing function $\Phi: [0, \infty) \to [0, \infty)$ with $\Phi(0) = 0$ and a constant $\beta > 0$ such that for all $\varepsilon > 0$ and $x \in D$ with $|\mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+)| < \varepsilon$ we have*

$$\beta \mathcal{E}(x, x_+) \leq \mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+) + \Phi(\mathcal{D}(\mathbf{F}(x), \mathbf{F}(x_+))). \tag{3.1}$$

Additionally, let condition 2.2 hold, let $y_\delta \in Y$ and $\delta > 0$ satisfy $\mathcal{D}(y, y_\delta), \mathcal{D}(y_\delta, y) \leq \delta$ and write Φ^{-*} for the Fenchel conjugate of the inverse function Φ^{-1}. Then the following assertions hold true:

(a) For sufficiently small α and δ, we have

$$\forall x_{\alpha,\delta} \in \arg\min \mathcal{T}_{\alpha;y_\delta}: \quad \beta \mathcal{E}(x_{\alpha,\delta}, x_+) \leq \frac{\delta}{\alpha} + \Phi(\tau\delta) + \frac{\Phi^{-*}(\tau\alpha)}{\tau\alpha}. \tag{3.2}$$

(b) If $\alpha \sim \delta/\Phi(\tau\delta)$, then $\mathcal{E}(x_{\alpha,\delta}, x_+) = \mathcal{O}(\Phi(\tau\delta))$ as $\delta \to 0$.

Proof.

(a) By theorem 2.6(c), we can assume that $|\mathcal{R}(\mathbb{V}, \cdot)(x_{\alpha,\delta}) - \mathcal{R}(\mathbb{V}, x_+)| \leq \varepsilon$. From the definition of $x_{\alpha,\delta}$ it follows that $\mathcal{D}(\mathbf{F}(x_{\alpha,\delta}), y_\delta) + \alpha\mathcal{R}(\mathbb{V}, \cdot)(x_{\alpha,\delta}) \leq \mathcal{D}(\mathbf{F}(x_+), y_\delta) + \alpha\mathcal{R}(\mathbb{V}, x_+)$. Then from (3.1) we obtain

$$\begin{aligned}
\alpha\beta\mathcal{E}(x_{\alpha,\delta}, x_+) &\leq \delta - \mathcal{D}(\mathbf{F}(x_{\alpha,\delta}), y_\delta) + \alpha\Phi(\mathcal{D}(\mathbf{F}(x_{\alpha,\delta}), \mathbf{F}(x_+))) \\
&\leq \delta - \mathcal{D}(\mathbf{F}(x_{\alpha,\delta}), y_\delta) + \alpha\Phi(\tau\delta + \tau\mathcal{D}(\mathbf{F}(x_{\alpha,\delta}), y_\delta)) \\
&\leq \delta - \mathcal{D}(\mathbf{F}(x_{\alpha,\delta}), y_\delta) + \alpha\Phi(\tau\delta) + \alpha\Phi(\tau\mathcal{D}(\mathbf{F}(x_{\alpha,\delta}), y_\delta)) \\
&\leq \delta + \alpha\Phi(\tau\delta) + \tau^{-1}\Phi^{-*}(\tau\alpha).
\end{aligned}$$

The last estimate is an application of Young's inequality $\alpha\Phi(\tau t) \leq t + \tau^{-1}\Phi^{-*}(\tau\alpha)$.

(b) Elementary computations show $\limsup_{\delta\to 0}\Phi^{-*}(\tau\delta/\Phi(\tau\delta))/\delta < \infty$, such that the right hand side of (3.2) is bounded by $\Phi(\tau\delta)$ up to a constant if $\alpha \sim \delta/\Phi(\tau\delta)$. □

Remark 3.2. If $\Phi(t) \leq Ct$ for sufficiently small t, then from (3.2) it follows that $\beta\mathcal{E}(x_{\alpha,\delta}, x) \leq \delta/\alpha + C\tau\delta = \mathcal{O}(\delta)$ if $\alpha \leq 1/C$. This says that the regularization parameter α needs not to vanish for $\delta \to 0$, which is often referred to as exact penalization (for the convex case, see the discussions in [8, 45]).

3.2. Rates in the absolute Bregman distance

We next derive conditions under which a variational inequality in form of (3.1) is possible for the absolute Bregman distance as error measure, $\mathcal{E}(x, x_+) := \mathcal{B}_{\mathcal{R}(\mathbb{V},\cdot)}(x, x_+)$.

Proposition 3.3 *(Rates in the absolute Bregman distance). Let X and Y be Hilbert spaces and let* $\mathbf{F}: X \to Y$ *be a bounded linear operator. Assume that* $\mathcal{R}(\mathbb{V}, \cdot)$ *is Gâteaux differentiable, that* $\mathcal{R}'(\mathbb{V}, x_+) \in \mathrm{Ran}(\mathbf{F}^*)$*, and that there exist positive constants* γ*,* ε *with*

$$\mathcal{R}(\mathbb{V}, x_+) - \mathcal{R}(\mathbb{V}, x) \leq \gamma \|\mathbf{F}(x) - \mathbf{F}(x_+)\| \tag{3.3}$$

for all x satisfying $|\mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+)| < \varepsilon$*. Then,*

$$\mathcal{B}_{\mathcal{R}(\mathbb{V},\cdot)}(x, x_+) \leq \mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+) + C\|\mathbf{F}(x) - \mathbf{F}(x_+)\| \quad \text{for some constant } C.$$

In particular, for the similarity measure $\mathcal{D}(z, y) = \|z - y\|^2$ *and under condition 2.2, items a and b of theorem 3.1 hold true.*

Proof. Let ξ satisfy that $\mathcal{R}'(\mathbb{V}, x_+) = \mathbf{F}^*\xi$. Then $|\langle \mathcal{R}'(\mathbb{V}, x_+), x - x_+\rangle| \leq \|\xi\|\|\mathbf{F}(x) - \mathbf{F}(x_+)\|$. Note that $|\mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+)| = \mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+)$ if $\mathcal{R}(\mathbb{V}, x) \geq \mathcal{R}(\mathbb{V}, x_+)$ and $|\mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+)| = \mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+) + 2(\mathcal{R}(\mathbb{V}, x_+) - \mathcal{R}(\mathbb{V}, x)) \leq \mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+) + 2\gamma\|\mathbf{F}(x) - \mathbf{F}(x_+)\|$ otherwise. This yields

$$\begin{aligned}\mathcal{B}_{\mathcal{R}(\mathbb{V},\cdot)}(x, x_+) &\leq |\mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+)| + |\langle \mathcal{R}'(\mathbb{V}, x_+), x - x_+\rangle| \\ &\leq \mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+) + C\|\mathbf{F}(x) - \mathbf{F}(x_+)\|\end{aligned}$$

with the constant $C := \|\xi\| + 2\gamma$, and concludes the proof. □

Remark 3.4. Proposition 3.3 shows that a variational inequality of the form (3.1) with $\beta = 1$ and $\Phi(t) = C\sqrt{t}$ follows from a classical source condition $\mathcal{R}'(x_+) \in \mathrm{Ran}(\mathbf{F}^*)$. By theorem 3.1, it further implies that $\mathcal{B}_{\mathcal{R}(\mathbb{V},\cdot)}(x_{\alpha,\delta}, x_+) = \mathcal{O}(\delta)$ if $\|y - y_\delta\| \leq \delta$. Moreover, we point out that the additional assumption (3.3) is rather weak, and follows from the classical source condition $\mathcal{R}'(x_+) \in \mathrm{Ran}(\mathbf{F}^*)$ if $\mathcal{R}$ is convex, see [22]. It is clear that a sufficient condition to (3.3) is

$$|\mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+) - \langle \mathcal{R}'(\mathbb{V}, x_+), x - x_+\rangle| \leq c|\mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+)| \quad \text{for some } c < 1,$$

which resembles a tangential-cone condition. Choosing the squared Hilbert space norm for the similarity measure, $\mathcal{D}(z, y) = \|z - y\|^2$, the error estimate (3.2) takes the form

$$\forall x_{\alpha,\delta} \in \arg\min \mathcal{T}_{\alpha;y_\delta}: \quad \mathcal{B}_{\mathcal{R}(\mathbb{V},\cdot)}(x_{\alpha,\delta}, x_+) \leq \frac{\delta}{\alpha} + C\sqrt{\delta} + \frac{C^4}{4}\alpha. \tag{3.4}$$

In particular, choosing $\alpha \sim \sqrt{\delta}$ yields the convergence rate $\mathcal{B}_{\mathcal{R}(\mathbb{V},\cdot)}(x_{\alpha,\delta}, x_+) = \mathcal{O}(\sqrt{\delta})$.

3.3. General regularizers

So far we derived well-posedness, convergence and convergence rates for regularizers of the form (1.2). These results can be generalized to Tikhonov regularization

$$\mathcal{T}_{\alpha;y_\delta}(x) := \mathcal{D}(\mathbf{F}(x), y_\delta) + \alpha\mathcal{R}(x) \to \min_x, \tag{3.5}$$

where the regularization term is not necessarily defined by a neural network. These results are derived by replacing condition 2.2 with the following one.

Condition 3.5 (Convergence for general regularizers).

(a) The functional $\mathcal{R}$ is sequentially lower semi-continuous.
(b) The set $\{x \mid \mathcal{T}_{\alpha;y}(x) \leq t\}$ is sequentially pre-compact for all t, y and $\alpha > 0$.
(c) The data consistency term satisfies (b).

Then we have the following:

Theorem 3.6 *(Results for general Tikhonov regularization). Under condition 3.5, general Tikhonov regularization (3.5) satisfies the following:*

(a) The conclusions from theorem 2.6 (well-posedness and convergence) hold true.
(b) If $\mathcal{R}$ is totally non-linear at $\mathcal{R}$-minimizing solutions, the strong convergence from theorem 2.11 holds.
(c) The convergence rates result from theorem 3.1 holds.
(d) If $\mathbf{F}$ is bounded linear, the assertions of proposition 3.3 hold for $\mathcal{R}$.

Proof. All assertions are shown as in the special case $\mathcal{R} = \mathcal{R}(\mathbb{V}, \cdot)$. □

Note that item (a) in the above theorem is contained in [21]. Items (b–d) have not been obtained previously for non-convex regularizers.

3.4. Non-linear ℓ^q-regularization

We now analyze a special instance of NETT regularization (1.2), generalizing classical ℓ^q-regularization by including non-linear transformations. More precisely, we consider the following ℓ^q-Tikhonov functional

$$\mathcal{T}_{\alpha,y_\delta}(x) = \|\mathbf{F}(x) - y_\delta\|^2 + \alpha\sum_{\lambda\in\Lambda} v_\lambda|\phi_\lambda(x)|^q \qquad \text{with} \quad q > 1. \tag{3.6}$$

Here Λ is a countable set and $\phi_\lambda : X \to \mathbb{R}$ are possibly non-linear functionals. Theorem 2.6 assures existence and convergence of minimizers of (3.6) provided that $(\phi_\lambda)_{\lambda\in\Lambda}$ is coercive and weakly continuous. If $(\phi_\lambda)_{\lambda\in\Lambda}$ is non-linear, minimizers are not necessarily unique.

The regularizer $\mathcal{R}(x) := \sum_{\lambda\in\Lambda} v_\lambda|\phi_\lambda(x)|^q$ is a particular instance of NETT (1.2), if we take $\sigma_L \circ \mathbb{V}_L \circ \cdots \circ \sigma_1 \circ \mathbb{V}_1 = (\phi_\lambda)_{\lambda\in\Lambda}$, and ψ as a weighted ℓ^q-norm. However, in (3.6) also more general choices for ϕ_λ are allowed (see condition 3.7).

We assume the following:

Condition 3.7.

(a) $\mathbf{F} : X \to Y$ is a bounded linear operator between Hilbert spaces X and Y.
(b) $\phi_\lambda : X \to \mathbb{R}$ is Gâteaux differentiable for every $\lambda \in \Lambda$.
(c) There is a $\mathcal{R}(\mathbb{V}, \cdot)$-minimizing solution x_+ with $\mathcal{R}'(\mathbb{V}, x_+) \in \mathrm{Ran}(\mathbf{F}^*)$.
(d) There exist constants $C, \varepsilon > 0$ such that for all x with $|\mathcal{R}(\mathbb{V}, x) - \mathcal{R}(\mathbb{V}, x_+)| \leq \varepsilon$, it holds that

$$\forall \lambda \in \Lambda: \ \operatorname{sign}(\phi_\lambda(x_+))(\phi_\lambda(x_+) - \phi_\lambda(x)) \le C \ \operatorname{sign}(\phi_\lambda(x_+))\phi'_\lambda(x_+)(x_+ - x).$$

Here $\operatorname{sign}(t) = 1$ for $t > 0$, $\operatorname{sign}(t) = 0$ for $t = 0$, and $\operatorname{sign}(t) = -1$ otherwise.

Proposition 3.8. *Let condition 3.7 hold, suppose that $y_\delta \in Y$ is such that $\|y - y_\delta\| \le \delta$, and let $x_{\alpha,\delta} \in \arg\min \mathcal{T}_{\alpha,y_\delta}$. If choosing $\alpha \sim \delta$, then $\mathcal{B}_{\mathcal{R}}(x_{\alpha,\delta}, x_+) = \mathcal{O}(\delta)$.*

Proof. The convexity of $t \mapsto |t|^q$ implies that

$$\mathcal{R}(x_+) - \mathcal{R}(x) \le \sum_{\lambda\in\Lambda} v_\lambda |\phi_\lambda(x_+)|^{q-1} \operatorname{sign}(\phi_\lambda(x_+))(\phi_\lambda(x_+) - \phi_\lambda(x)).$$

By (d), we obtain $\mathcal{R}(x_+) - \mathcal{R}(x) \le C\langle \mathcal{R}'(x_+), x_+ - x\rangle$. This together with (c) implies (3.3). Thus, the assertion follows from proposition 3.3. □

Remark 3.9. Consider the case that $\phi_\lambda(x) := \langle \varphi_\lambda, x\rangle$ for an orthonormal basis $(\varphi_\lambda)_{\lambda\in\Lambda}$ of X. It is known that $\|x - x_+\|^2 = \mathcal{O}\left(\mathcal{B}_{\mathcal{R}(\mathbb{V},\cdot)}(x, x_+)\right)$, see e.g. [45]. Then, proposition 3.8 gives $\|x_{\alpha,\delta} - x_+\| = \mathcal{O}(\delta^{1/2})$, which reproduces the result of [45, theorem 3.54]. This rate can be improved to $\mathcal{O}(\delta^{1/q})$ if we further assume sparsity of x_+ and restricted injectivity of $\mathbf{F}$. It can be shown by theorem 3.1 because in such a situation (3.1) holds with $\Phi(t) \sim \sqrt{t}$ and $\mathcal{E}(x, x_+) = \|x - x_+\|^q$, see [22] for details.

3.5. Comparison to the W-Bregman distance

A different framework for deriving convergence rates for non-convex regularization functionals is based on the W-Bregman distance introduced in [21, 24]. In this subsection we compare our absolute Bregman distance with the W-Bregman for some specific examples.

Definition 3.10 (W-Bregman distance). Let W be a set of functions $w: X \to \mathbb{R}$, $\mathcal{R}: X \to [0, \infty]$ be a functional and $x_+ \in X$.

(a) $\mathcal{R}$ is called W-convex at x_+ if $\mathcal{R}(x_+) = \sup_{w\in W} \inf_{x\in X}(\mathcal{R}(x) - w(x) + w(x_+))$.
(b) Let $\mathcal{R}$ be W-convex at x_+. The W-subdifferential of $\mathcal{R}$ at x_+ is defined by $\partial_W \mathcal{R}(x_+) := \{w \in W \mid \forall x \in X: \mathcal{R}(x) \ge \mathcal{R}(x_+) + w(x) - w(x_+)\}$. Moreover, the W-Bregman distance with respect to $w \in \partial_W \mathcal{R}(x_+)$ between x_+ and $x \in X$ is defined by

$$\mathcal{B}^w_{\mathcal{R},W}(x, x_+) := \mathcal{R}(x) - \mathcal{R}(x_+) - w(x) + w(x_+).$$

The notion of W-Bregman distance reduces to its classical counterpart if we take W as the set of all bounded linear functionals on X. Allowing more general function sets W provides an extension of the Bregman distance to non-convex functionals that can be used as an alterative approach to convergence rates. It, however, requires finding a suitable function set such that $\mathcal{R}$ is W-convex at x_+.

Relations between the absolute Bregman distance and the W-Bregman distance depends on the particular choice of W. We illustrate this by an example.

Example 3.11. Consider the functional $\mathcal{R}(x) := \left(2\mathrm{ReLU}(x - x_+) - 1\right)|x - x_+|^q$ for $x \in \mathbb{R}$ with $q > 1$. The absolute Bregman distance according to definition 2.7 is given by $\mathcal{B}_{\mathcal{R}}(x, x_+) = |x - x_+|^q$. For the W-Bregman distance consider the family $W := \{w_{\alpha,\beta}: \mathbb{R} \to \mathbb{R} \mid \alpha, \beta \in \mathbb{R}\}$

where $w_{\alpha,\beta}(x) := \alpha(x - x_+) - \beta|x - x_+|^q$. Then $\mathcal{R}$ is locally convex at x_+ with respect to W. Moreover, $w_{\alpha,\beta} \in \partial_W \mathcal{R}(x_+)$ if $\alpha = 0$ and $\beta \geq 1$. For $w_{0,\beta} \in \partial_W \mathcal{R}(x_+)$, it follows that

$$\mathcal{B}^{w_{0,\beta}}_{\mathcal{R},W}(x, x_+) = \begin{cases} (\beta + 1)|x - x_+|^q & \text{if } x \geq x_+ \\ (\beta - 1)|x - x_+|^q & \text{otherwise.} \end{cases}$$

In case of $q = 2$, the W-subdifferential is closely related to the notion of proximal subdifferentiability used in [14].

If $\beta > 1$, then $\mathcal{B}^{w_{0,\beta}}_{\mathcal{R},W}(\cdot, x_+)$ and $\mathcal{B}_{\mathcal{R}}(\cdot, x_+)$ only differ in terms of the front constants. As a consequence, the rates with respect to both Bregman distances will be of the same order. However, if $\beta = 1$, then $\mathcal{B}^{w_{0,1}}_{\mathcal{R},W}(\cdot, x_+)$ equals 0 when $x \leq x_+$. In contrast, $\mathcal{B}_{\mathcal{R}}(\cdot, x_+)$ always treats both $x \geq x_+$ and $x \leq x_+$ equally.

We conclude that the relation between the absolute Bregman distance and the W-Bregman distance depends on the particular situation and the choice of the family W. Using the W-Bregman distance for the analysis of NETT and studying relations between the two generalized Bregman distances are interesting lines of research that we aim to address in future work.

4. A data driven regularizer for NETT

In this section we present a framework for constructing a trained neural network regularizer $\mathcal{R}(\mathbb{V}, \cdot)$ of the form (2.1). Additionally, we develop a strategy for network training and minimizing the NETT functional.

4.1. A trained regularizer

For the regularizer we propose $\mathcal{R}(\mathbb{V}, \cdot) = \sum_{\lambda \in \Lambda_L} \|\mathbf{\Phi}_\lambda(\mathbb{V}, x)\|_q^q$ with a network $\mathbf{\Phi}(\mathbb{V}, \cdot) = (\mathbf{\Phi}_\lambda(\mathbb{V}, \cdot))_{\lambda \in \Lambda}$ of the form (2.1), that itself is part of encoder-decoder type network:

$$\mathbf{\Psi}(\mathbb{W}, \cdot) \circ \mathbf{\Phi}(\mathbb{V}, \cdot) : X \to X. \tag{4.1}$$

Here $\mathbf{\Phi}(\mathbb{V}, \cdot) : X \to \mathbb{X}_L$ can be interpreted as encoding network and $\mathbf{\Psi} : \mathbb{X}_L \to X$ as decoding network. We note, however, that any network with at least one hidden layer can be written in the form (4.1). Moreover we also allow $\mathbb{X}_L$ to be of large dimension in which case the encoder $\mathbf{\Phi}(\mathbb{V}, \cdot)$ does not perform any form of dimensionality reduction or compression.

Training of the network is performed such that $\mathcal{R}(\mathbb{V}, \cdot)$ is small for artifact free images and large for images with artifacts. The proposed training strategy is presented below.

For suitable network training of the encoder–decoder scheme (4.1), we propose the following strategy (compare figure 1). We choose a set of training phantoms $z_n \in X$ for $n = 1, \ldots, N_1 + N_2$ from which we construct back-projection images $x_n := \mathbf{F}^+(\mathbf{F}z_n)$ for the first N_1 training examples, and set $x_n = z_n$ for the last N_2 training images. Here $\mathbf{F}^+$ denotes any (approximate) right inverse of $\mathbf{F}$, such as the pseudo-inverse in the linear case. From this we define the training data $\{(x_n, r_n)\}_{n=1}^{N_1+N_2}$, where

$$\forall n = 1, \ldots, N_1 : \quad r_n = z_n - x_n = z_n - \mathbf{F}^+(\mathbf{F}z_n) \tag{4.2}$$

$$\forall n = N_1 + 1, \ldots, N_1 + N_2 : \quad r_n = z_n - x_n = 0. \tag{4.3}$$

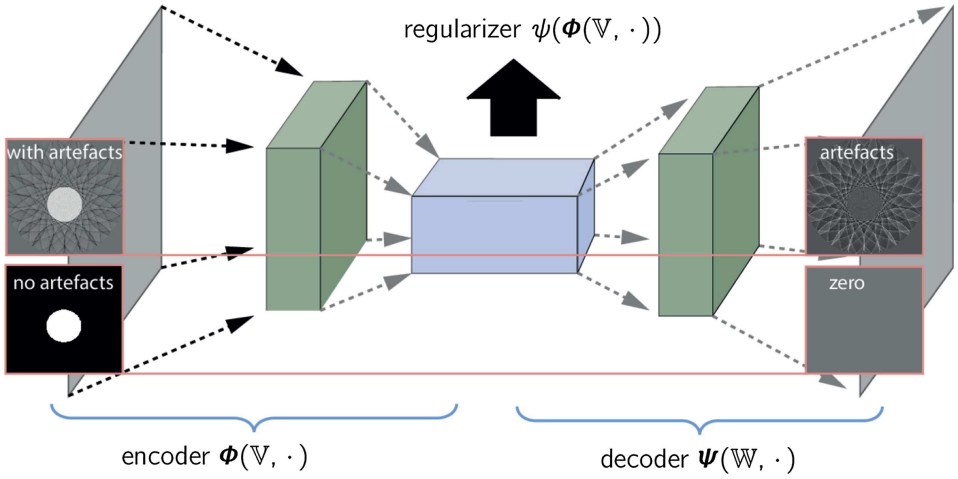

Figure 1. Encoder-decoder scheme and proposed training strategy. The network consists of the encoder part $\boldsymbol{\Phi}(\mathbb{V}, \cdot)$ and decoder part $\boldsymbol{\Psi}(\mathbb{W}, \cdot)$, and is trained to map any potential solution x to the corresponding artifact part. The norm of $\boldsymbol{\Phi}(\mathbb{V}, x)$ is used as trained regularizer.

The free parameters in (4.1) are adjusted in such a way, that $\boldsymbol{\Psi}(\mathbb{W}, \boldsymbol{\Phi}(\mathbb{V}, x_n)) \simeq r_n$ for any training pair (x_n, r_n). This is achieved by minimizing the error function

$$E_N(\mathbb{V}, \mathbb{W}) := \sum_{n=1}^{N_1+N_2} \ell(\boldsymbol{\Psi}(\mathbb{W}, \boldsymbol{\Phi}(\mathbb{V}, x_n)), r_n), \tag{4.4}$$

where ℓ is a suitable distance measure (or loss function) that quantifies the error made by the network function on the nth training sample. Typical choices for d are mean absolute error or mean squared error.

Given an arbitrary unknown $x \in X$, the trained network estimates the artifact part. As a consequence, $\mathcal{R}(\mathbb{V}, x)$ is expected to be large, if x contains severe artifacts and small if it is almost artifact free. If x is similar to elements in the training set, this should produce almost artifact free results with NETT regularization. Even if the true unknown is of different type from the training data, artifacts as well as noise will have large value of the regularizer. Thus our approach is applicable for a wider range of images apart from training ones. This claim is confirmed by our numerical results in section 5. Note that we did not explicitly account for the coercivity condition during the training phase. Several possibilities for ensuring coercivity are discussed in section 2.2. Moreover, note that the class of methods we have in mind for the above training strategy are underdetermined problems such as undersampled CT, MRI or PAT. We expect that similar training strategies can be designed for problems that have many small but not vanishing singular values. Investigating such issues in more detail (theoretically and numerically) is an interesting line of future research.

Remark 4.1 (Alternative trained regularizers). Another natural choice would be to simply take $\mathcal{R}(\mathbb{W}, \mathbb{V}, x) := \|\boldsymbol{\Psi}(\mathbb{W}, \boldsymbol{\Phi}(\mathbb{V}, x))\|^2$ for the regularizer. Such a regularizer has been used in the proceedings [6] combined with quite simple network architectures for $\boldsymbol{\Psi}$ and $\boldsymbol{\Phi}$. The main

Algorithm 1. Incremental gradient descent for minimizing NETT.

Choose family of step-sizes $(s_i) > 0$
Choose initial iterate x_0
for $i = 1$ to maxiter **do**
  $\bar{x}_i \leftarrow x_{i-1} - s_i \mathbf{F}'(x_{i-1})^*(\mathbf{F}(x_{i-1}) - y^\delta)$ {gradient step for $\frac{1}{2}\|\mathbf{F}(x) - y_\delta\|^2$}
  $x_i \leftarrow \bar{x}_i - s_i \alpha \nabla_x \mathcal{R}(\mathbb{V}, \cdot)(\bar{x}_i)$ {gradient step for $\mathcal{R}(\mathbb{V}, \cdot)$}
end for

emphasis of this paper is the convergence analysis of NETT, so the investigation of the effects of different trained regularizers is beyond its scope. We nevertheless point out that including training data corresponding to (4.3) makes the trained network $\mathbf{\Psi}(\mathbb{W}, \mathbf{\Phi}(\mathbb{V}, x))$ different to standard artifact removal network [30], which only use training data corresponding to (4.2) to remove artifacts. Detailed investigation of benefits of each approach is an interesting aspect of future research.

4.2. Minimizing the NETT functional

Using the encoder–decoder scheme, regularized solutions are defined as minimizers of the NETT functional

$$\mathcal{T}_{\alpha;y}(x) = \frac{1}{2}\|\mathbf{F}(x) - y_\delta\|^2 + \alpha \sum_{\lambda \in \Lambda_L} \|\mathbf{\Phi}_\lambda(\mathbb{V}, x)\|_q^q, \tag{4.5}$$

where $\mathbf{\Phi}$ is trained as above. The optimization problem (4.5) is non-convex (due to the presence of the non-linear network) and non-smooth. Note that the gradient of the regularization term $\mathcal{R}(\mathbb{V}, x) = \sum_{\lambda \in \Lambda_L} \|\mathbf{\Phi}(\mathbb{V}, x)_\lambda\|_q^q$ can be evaluated by standard software for network training with the backpropagation algorithm. We therefore propose to use an incremental gradient method for minimizing the Tikhonov functional (4.5), which alternates between a gradient descent step for $\frac{1}{2}\|\mathbf{F}(x) - y_\delta\|^2$ and a gradient descent step for the regularizer $\mathcal{R}(\mathbb{V}, x)$.

The resulting minimization procedure is summarized in algorithm 1.

In practice, we found that algorithm 1 gives favorable performance, and is stable with respect to tuning parameters. Also other algorithms such as proximal gradient methods [15] or Newton type methods might be used for the minimization of (4.5). A detailed comparison with other algorithms is beyond the scope of this article.

Note that the regularizer may be taken $\mathcal{R}(\mathbb{V}, x) = \|\mathbf{\Phi}(x)\|_L$ with an arbitrary norm $\|\cdot\|_L$ on $\mathbb{X}_L$. The concrete training procedure is described below. In the form (4.5), NETT constitutes a non-linear generalization of ℓ^q-regularization.

5. Application to sparse data tomography

As a demonstration, we use NETT regularization with the encoder–decoder scheme presented in section 4 to the sparse data problem in photoacoustic tomography (PAT). PAT is an emerging hybrid imaging method based on the conversion of light in sound, and beneficially combines the high contrast of optical imaging with the good resolution of ultrasound tomography (see, for example, [33, 40, 49, 50]).

5.1. Sparse sampling problem in PAT

The aim of PAT is to recover the initial pressure $p_0 : \mathbb{R}^d \to \mathbb{R}$ in the wave equation

$$\begin{cases} \partial_t^2 p(x,t) - \Delta_x p(x,t) = 0, & \text{for } (x,t) \in \mathbb{R}^d \times (0,\infty), \\ p(x,0) = p_0(x), & \text{for } x \in \mathbb{R}^d, \\ \partial_t p(x,0) = 0, & \text{for } x \in \mathbb{R}^d. \end{cases} \tag{5.1}$$

form measurements of p made on an observation surface S outside the support of p_0. Here d is the spatial dimension, Δ_x the spatial Laplacian, and ∂_t the derivative with respect to the time variable t. Both cases $d = 2, 3$ for the spatial dimension are relevant in PAT: the case $d = 3$ corresponds to classical point-wise measurements; the case $d = 2$ to integrating line detectors [9, 40]. In this paper we consider the case of $d = 3$ and assume the initial pressure $p_0 : \mathbb{R}^2 \to \mathbb{R}$ vanishes outside the unit disc D_1, the ball of radius 1, and that acoustic data are collected at the boundary sphere $\mathbb{S}^1 = \partial D_1$. In particular, we are interested in the sparse sampling case, where data are only given for a small number of sensor locations on $\mathbb{S}^1$. This is the case that one often faces in practical applications.

In the full sampling case, the discrete PAT forward operator is written as $\mathcal{F} : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^{m_{\text{full}} \times m_2}$ where m_{full} corresponds to the number of complete spatial sampling points and M_2 to the number of temporal sampling points. Sufficient sampling conditions for PAT in the circular geometry have been derived in [26]. We discretize the exact inversion formula of [19] to obtain an approximation $\mathcal{F}^\sharp : \mathbb{R}^{m_{\text{full}} \times m_2} \to \mathbb{R}^{n_1 \times n_2}$ to the inverse of $\mathbf{F}$. In the full data case, application of $\mathcal{F}^\sharp$ to data $\mathcal{F}x \in \mathbb{R}^{m_{\text{full}} \times m_2}$ gives an almost artifact free reconstruction $x \in \mathbb{R}^{n_1 \times n_2}$, see [26]. Note that $\mathcal{F}^\sharp$ is the discretization of the continuous adjoint of $\mathcal{F}$ with respect to a weighted L^2-inner product (see [19]).

In the sparse sampling case, the PAT forward operator is given by

$$\mathbf{F} = \mathbf{S} \circ \mathcal{F} : X = \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^{m_1 \times m_2}. \tag{5.2}$$

Here $\mathbf{S} : \mathbb{R}^{m_{\text{full}} \times m_2} \to \mathbb{R}^{m_1 \times m_2}$ is the subsampling operator, which restricts the full data in to a small number of spatial sampling points. In the case of spatial under-sampling, the filtered backprojection (FBP) reconstruction $\mathbf{F}^\sharp := \mathcal{F}^\sharp \circ \mathbf{S}^\mathsf{T}$ yields typical streak-like under-sampling artifacts (see, for example, the examples in figure 2).

5.2. Implementation details

Consider NETT where the regularizer is defined by the encoder–decoder framework described in section 4. The network $\boldsymbol{\Psi}(\mathbb{W}, \cdot) \circ \boldsymbol{\Phi}(\mathbb{V}, \cdot)$ is taken as the Unet, where the $\boldsymbol{\Phi}(\mathbb{V}, x)$ corresponds to the output of the bottom layer with smallest image size and largest depth. The Unet has been proposed in [44] for image segmentation and successfully applied to PAT in [5, 47]. However, we point out, that any network that has the encoder–decoder of the form $\boldsymbol{\Psi}(\mathbb{W}, \cdot) \circ \boldsymbol{\Phi}(\mathbb{V}, \cdot)$ can be used in an analogous manner.

The network was trained on a set of training pairs $\{(x_n, r_n)\}_{n=1}^{N_1+N_2}$, with $N_1 = N_2 = 975$, where exactly half of them contained under-sampling artifacts. For generating such training data we used (4.2), (4.3) where z_n are taken as randomly generated piecewise constant Shepp–Logan type phantoms. The Shepp–Logan type phantoms have position, angle, shape and intensity of every ellipse chosen uniformly at random under the side constraints that the support of every ellipse lies inside the unit disc and the intensity of the phantom is in the range

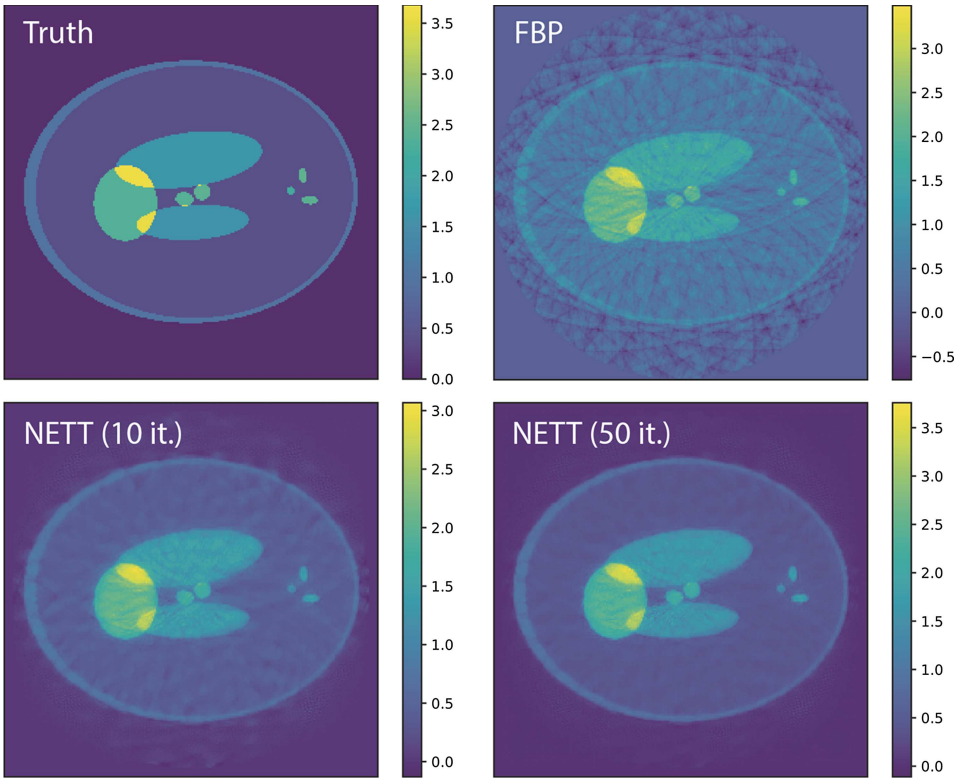

Figure 2. Reconstruction results for a phantom of Shepp–Logan type. Top: Phantom x (left) and corresponding FBP reconstruction (right); bottom: iterates x_{10} (left) and x_{50} (right) with the proposed algorithm for minimizing the NETT functional.

$[0, 6]$. During training we had no problems with overfitting or instability and thus we did not use dropout or batch normalization. However, such techniques might be needed for different networks or training sets.

In this discrete sparse sampling case, we take the forward operator $\mathbf{F}$ as in (5.2) with $n_1 = n_2 = 256$ and $m_1 = 30$ spatial samples distributed equidistantly on the boundary circle. We used $m_2 = 2000$ times sampled evenly in the interval $[0, 2.5]$. The under-sampling problem in PAT is solved by FBP, and NETT regularization using $\alpha = 1/4$. We minimize (4.5) using algorithm 1, where we chose a constant step size of $s_i = 0.4$ and take the zero image $x_0 = 0$ for the initial guess. These parameters have been selected by hand using similar phantoms as reference.

5.3. Results and discussion

The top left image in figure 2 shows a Shepp–Logan type phantom $x \in \mathbb{R}^{256\times256}$ corresponding to a function on the domain $[-1, 1]^2$. It is of the same type as the training data, but is not contained in the training data. The NETT reconstruction x_{10} and x_{50} with algorithm 1 after 10 and 50 iterations for the Shepp–Logan type phantom are shown in the bottom row of figure 2. The top right image figure 2 shows the reconstruction $x_{\text{FBP}} = \mathbf{F}^\sharp\mathbf{F}x$ with the FBP algorithm of [19]. The relative L^2-errors $E(z) := \|x - z\|_2/\|x\|_2$ of the iterates for the Shepp–Logan

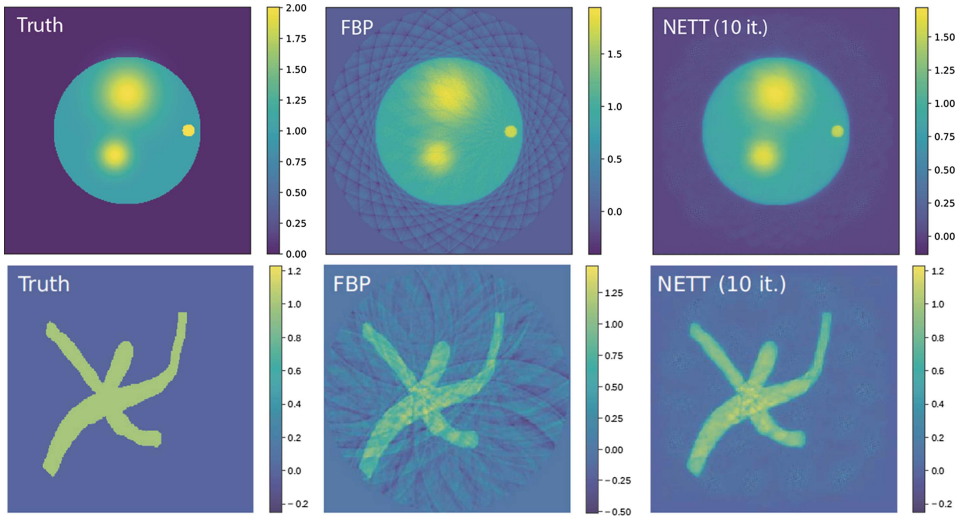

Figure 3. Reconstruction results for phantoms of different type from training data. Top: phantom with smooth blobs (left), corresponding FBP reconstruction (middle) and NETT reconstruction with the proposed algorithm for minimizing the NETT functional. Bottom: same for another phantom.

type phantom are $E(x_{10}) = 0.262$ and $E(x_{50}) = 0.192$, whereas the relative error of the FBP reconstruction is $E(x_{\mathrm{FBP}}) = 0.338$. From figure 2 one recognizes that NETT is able to well remove under-sampling artifacts while preserving high resolution information.

We also consider a phantom image (blobs phantom) that additionally includes smooth parts and is of different type from the phantoms used for training. The blobs phantom as well as the FBP reconstruction x_{FBP} and NETT reconstructions x_{10} and x_{50} are shown in the top row of figure 3. Again, for this phantom different from the training set, NETT removes under-sampling artifacts and at the same time preserves high resolution. Results for another phantom of different type from training data are shown in the bottom row of figure 3. Finally, figure 4 shows reconstruction results with NETT from noisy data where we added 5% additive Gaussian noise to the data. We performed 15 iterations with algorithm 1. Parameters have been taken as in the noiseless data case. We also calculated the average relative L^2-error and average structured similarity index (SSIM) of [52] on a test set of 50 phantoms, which were similar to the training set. The errors for both the noiseless and noisy case can be seen in table 1. We used the same parameters as above for both the noiseless and the noisy case.

In all cases, the reconstructions are free from under-sampling artifacts and contain high frequency information, which demonstrates the applicability of NETT for noisy data as well. The above results demonstrate the proposed NETT regularization using the encoder–decoder framework and with algorithm 1 for minimization is able to remove under-sampling artifacts. It also gives consistent results even on images with smooth structures not contained in the training data. This shows that in the NETT framework, learning the regularization functional on one class of training data, can lead to good results even for images beyond that class.

We finally note that the NETT reconstructions contain negative values even if the network is trained on non-negative phantoms only. This comes from the fact that the network regularizer in

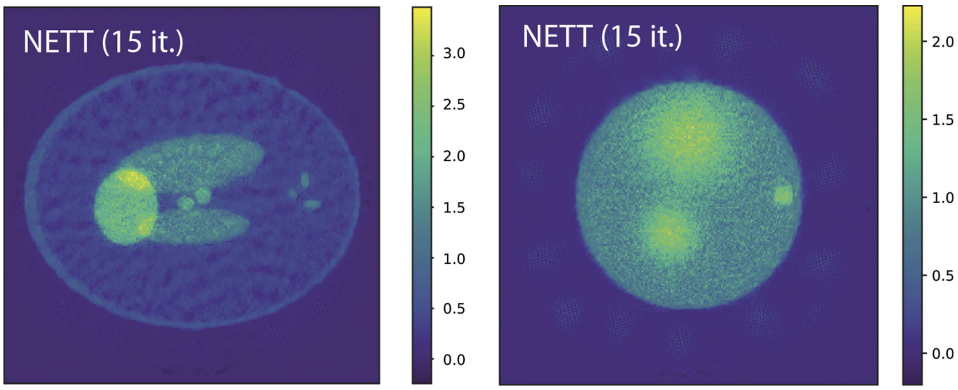

Figure 4. Reconstruction results from noisy data using NETT with 15 iterations. Left: Shepp–Logan phantom; right: blobs phantom.

Table 1. Average error measures for reconstruction from noisy data from on the test set using FBP and NETT using 20 iterations.

	FBP	NETT
Relative L^2	0.834	0.198
SSIM	0.261	0.502

(4.5) does not impose any hard constraint but enforces similarity to the training data instead. In order to obtain nonnegative reconstruction results, one can add positivity as an additional constraint to the NETT functional. Formally, this is obtained by using $\sum_{\lambda\in\Lambda_L}\|\Phi_\lambda(\mathbb{V},x)\|_q^q + \boldsymbol{i}_+(x)$ as regularizer in (4.5). Here the additional term $\boldsymbol{i}_+$ in the regularizer denotes the indicator function of the cone of nonnegative elements, defined by $\boldsymbol{i}_+(x) = 0$ if all entries of x are nonnegative and $\boldsymbol{i}_+(x) = \infty$ otherwise. Clearly one can add further terms to the regularizers, such as a TV or a smoothness penalty. Numerical investigations of such procedures are beyond the scope of this paper.

6. Conclusion and outlook

In this paper we developed a new framework for the solution of inverse problems via NETT (1.3). We presented a complete convergence analysis and derived well-posedness and weak convergence (theorem 2.6), norm-convergence (theorem 2.11), as well as various convergence rates results (see section 3). For these results we introduced the absolute Bregman distance as a new generalization of the standard Bregman distance from the convex to the non-convex setting. NETT combines deep neural networks with a Tikhonov regularization strategy. The regularizer is defined by a network that might be a user-specified function (generalizing frame based regularization), or might be a CNN trained on an appropriate training data set. We have developed a possible strategy for learning a deep CNN (using an encoder–decoder framework, see section 4). Initial numerical results for a sparse data problem in PAT (see section 5) demonstrated that NETT with the trained regularizer works well and also yields good results for

phantoms different from the class of training data. This may be a result of the fact, that opposed to other deep learning approaches for image reconstruction, the NETT includes a data consistency term as well as the trained network that focuses on identifying artifacts. Detailed comparison with other deep learning methods for inverse problems as well as variational regularization methods (including TV-minimization) is subject of future studies.

Many possible lines of future research arise from the proposed NETT regularization and the corresponding network-minimizing solution concept (1.5). For example, instead of the Tikhonov variant (1.3) one can employ and analyze the residual method (or Ivanov regularization) for approximating (1.5), see [24]. Instead of the simple incremental gradient descent algorithm (cf. algorithm 1) for minimizing NETT one could investigate different algorithms such as proximal gradient or semi-smooth Newton methods. Studying network designs and training strategies different from the encoder–decoder scheme is a promising aspect of future studies. Finally, application of NETT to other inverse problems is another interesting research direction.

Acknowledgment

S A and M H acknowledge support of the Austrian Science Fund (FWF), project P 30747. The work of H L has been support through the DFG Cluster of Excellence Multiscale Bioimaging EXC 2067.

ORCID iDs

Markus Haltmeier https://orcid.org/0000-0001-5715-0331

References

[1] Adler J and Öktem O 2017 Solving ill-posed inverse problems using iterative deep neural networks *Inverse Problems* **33** 124007
[2] Aggarwal H K, Mani M P and Jacob M 2018 MODL: model-based deep learning architecture for inverse problems *IEEE Trans. Med. Imaging* **38** 394–405
[3] Aharon M, Elad M and Bruckstein A 2006 K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation *IEEE Trans. Signal Process.* **54** 4311–22
[4] Amos B, Xu L and Kolter J Z 2017 Input convex neural networks *Proc. 34th Int. Conf. on Machine Learning* vol 70 pp 146–55
[5] Antholzer S, Haltmeier M and Schwab J 2019 Deep learning for photoacoustic tomography from sparse data *Inverse Problems Sci. Eng.* **27** 987–1005
[6] Antholzer S, Schwab J, Bauer-Marschallinger J, Burgholzer P and Haltmeier M 2019 NETT regularization for compressed sensing photoacoustic tomography *Proc. Photons Plus Ultrasound: Imaging and Sensing (3–6 February 2019, San Francisco, California)* vol 10878 p 108783B
[7] Behrmann J, Grathwohl W, Chen R, Duvenaud D and Jacobsen J 2019 Invertible residual networks *Proc. 36th Int. Conf. on Machine Learning* ed K Chaudhuri and R Salakhutdinov vol 97 pp 573–82
[8] Burger M and Osher S 2004 Convergence rates of convex variational regularization *Inverse Problems* **20** 1411–21
[9] Burgholzer P, Bauer-Marschallinger J, Grün H, Haltmeier M and Paltauf G 2007 Temporal back-projection algorithms for photoacoustic tomography with integrating line detectors *Inverse Problems* **23** S65–80
[10] Butnariu D and Iusem A N 2000 *Totally Convex Functions for Fixed Points Computation and Infinite Dimensional Optimization* (Applied Optimization vol 40) (Dordrecht: Kluwer Academic)

[11] Candès E J and Donoho D L 2004 New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities *Commun. Pure Appl. Math.* **57** 219–66
[12] Chang R, Li C, Poczos B, Kumar V and Sankaranarayanan A 2017 One network to solve them all–solving linear inverse problems using deep projection models *Proc. IEEE Int. Conf. on Computer Vision* pp 5888–97
[13] Chen H, Zhang Y, Zhang W, Liao P, Li K, Zhou J and Wang G 2017 Low-dose CT via convolutional neural network *Biomed. Opt. Express* **8** 679–94
[14] Clarke F H, Ledyaev Y S, Stern R J and Wolenski P R 1997 *Nonsmooth Analysis and Control Theory* (Berlin: Springer) vol 178
[15] Combettes P L and Pesquet J-C 2011 Proximal splitting methods in signal processing *Fixed-Point Algorithms for Inverse Problems in Science and Engineering* (Berlin: Springer) pp 185–212
[16] Daubechies I 1988 Orthonormal bases of compactly supported wavelets *Commun. Pure Appl. Math.* **41** 909–96
[17] Daubechies I, Defrise M and De Mol C 2004 An iterative thresholding algorithm for linear inverse problems with a sparsity constraint *Commun. Pure Appl. Math.* **57** 1413–57
[18] Engl H W, Hanke M and Neubauer A 1996 *Regularization of Inverse Problems* (Dordrecht: Kluwer Academic) vol 375
[19] Finch D, Haltmeier M and Rakesh 2007 Inversion of spherical means and the wave equation in even dimensions *SIAM J. Appl. Math.* **68** 392–412
[20] Goodfellow I, Bengio Y and Courville A 2016 *Deep Learning* (Cambridge, MA: MIT Press) http://www.deeplearningbook.org
[21] Grasmair M 2010 Generalized Bregman distances and convergence rates for non-convex regularization methods *Inverse Problems* **26** 115014
[22] Grasmair M, Haltmeier M and Scherzer O 2008 Sparse regularization with l^q penalty term *Inverse Problems* **24** 055020
[23] Grasmair M, Haltmeier M and Scherzer O 2011 Necessary and sufficient conditions for linear convergence of ℓ^1-regularization *Commun. Pure Appl. Math.* **64** 161–82
[24] Grasmair M, Haltmeier M and Scherzer O 2011 The residual method for regularizing ill-posed problems *Appl. Math. Comput.* **218** 2693–710
[25] Gribonval R and Schnass K 2010 Dictionary identification — sparse matrix-factorization via ℓ_1-minimization *IEEE Trans. Inf. Theory* **56** 3523–39
[26] Haltmeier M 2016 Sampling conditions for the circular Radon transform *IEEE Trans. Image Process.* **25** 2910–19
[27] Hammernik K, Klatzer T, Kobler E, Recht M P, Sodickson D K, Pock T and Knoll F 2018 Learning a variational network for reconstruction of accelerated mri data *Magn. Reson. Med.* **79** 3055–71
[28] Han Y, Yoo J J and Ye J C 2016 Deep residual learning for compressed sensing CT reconstruction via persistent homology analysis arXiv:1611.06391
[29] He K, Zhang X, Ren S and Sun J 2015 Delving deep into rectifiers: surpassing human-level performance on imagenet classification *Proc. ICCV* pp 1026–34
[30] Jin K H, McCann M T, Froustey E and Unser M 2017 Deep convolutional neural network for inverse problems in imaging *IEEE Trans. Image Process.* **26** 4509–22
[31] Kelly B, Matthews T P and Anastasio M A 2017 Deep learning-guided image reconstruction from incomplete data arXiv:1709.00584
[32] Kofler A, Haltmeier M, Kolbitsch C, Kachelrieß M and Dewey M 2018 A u-nets cascade for sparse view computed tomography *International Workshop on Machine Learning for Medical Image Reconstruction* (Berlin: Springer) pp 91–9
[33] Kuchment P and Kunyansky L A 2008 Mathematics of thermoacoustic and photoacoustic tomography *Eur. J. Appl. Math.* **19** 191–224
[34] Lorenz D 2008 Convergence rates and source conditions for Tikhonov regularization with sparsity constraints *J. Inverse Ill-Posed Problems* **16** 463–78
[35] Du Simon S, Hu Wei and Lee Jason D 2018 Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced *Advances in Neural Information Processing Systems 31* ed S Bengio, H Wallach, H Larochelle, N Cesa-Bianchi and R Garnett (Curran Associates, Inc.) pp 384–395 URL: http://papers.nips.cc/paper/7321-algorithmic-regularization-in-learning-deep-homogeneous-models-layers-are-automatically-balanced.pdf
[36] Maas A L, Hannun A Y and Ng A Y 2013 Rectifier nonlinearities improve neural network acoustic models *Proc. ICML* vol 30

[37] Mallat S 2009 *A Wavelet Tour of Signal Processing: The Sparse Way* 3rd edn (Amsterdam: Elsevier/Academic)
[38] Natterer F and Wübbeling F 2001 *Mathematical Methods in Image Reconstruction* (Monographs on Mathematical Modeling and Computation vol 5) (Philadelphia, PA: SIAM)
[39] Obmann D, Schwab J and Haltmeier M 2019 Sparse synthesis regularization with deep neural networks *2017 Int. Conf. on Sampling Theory and Applications* (SampTA) https://arxiv.org/abs/1902.00390
[40] Paltauf G, Nuster R, Haltmeier M and Burgholzer P 2007 Photoacoustic tomography using a Mach-Zehnder interferometer as an acoustic line detector *Appl. Opt.* **46** 3352–58
[41] Ramlau R and Teschke G 2006 A Tikhonov-based projection iteration for nonlinear ill-posed problems with sparsity constraints *Numer. Math.* **104** 177–203
[42] Resmerita E 2004 On total convexity, Bregman projections and stability in Banach spaces *J. Convex Anal.* **11** 1–16
[43] Romano Y, Elad M and Milanfar P 2017 The little engine that could: regularization by denoising (red) *SIAM J. Imag. Sci.* **10** 1804–44
[44] Ronneberge O, Fischer P and Brox T 2015 U-net: Convolutional networks for biomedical image segmentation *Int. Conf. on Medical Image Computing and Computer-Assisted Intervention* pp 234–41
[45] Scherzer O, Grasmair M, Grossauer H, Haltmeier M and Lenzen F 2009 *Variational Methods in Imaging* (Applied Mathematical Sciences vol 167) (New York: Springer)
[46] Schlemper J, Caballero J, Hajnal J V, Price A N and Rueckert D 2017 A deep cascade of convolutional neural networks for dynamic mr image reconstruction *IEEE Trans. Med. Imaging* **37** 491–503
[47] Schwab J, Antholzer S, Nuster R and Haltmeier M 2018 Real-time photoacoustic projection imaging using deep learning arXiv:1801.06693
[48] Wang G 2016 A perspective on deep imaging *IEEE Access* **4** 8914–24
[49] Wang K and Anastasio M A 2011 Photoacoustic and thermoacoustic tomography: image formation principles *Handbook of Mathematical Methods in Imaging* (Berlin: Springer) pp 781–815
[50] Wang L V 2009 Multiscale photoacoustic microscopy and computed tomography *Nat. Photon.* **3** 503–9
[51] Wang S, Su Z, Ying L, Peng X, Zhu S, Liang F, Feng D and Liang D 2016 Accelerating magnetic resonance imaging via deep learning *IEEE 13th Int. Symp. on Biomedical Imaging (ISBI)* pp 514–7
[52] Wang Z, Bovik A C and Sheikh H R *et al* 2004 Image quality assessment: from error visibility to structural similarity *IEEE Trans. Image Process.* **13** 600–12
[53] Würfl T, Ghesu F C, Christlein V and Maier A 2016 Deep learning computed tomography Int. Conf. on Medical Image Computing and Computer-Assisted Intervention (Berlin: Springer) pp 432–40
[54] Zhang H, Li L, Qiao K, Wang L, Yan B, Li L and Hu G 2016 Image prediction for limited-angle tomography via deep learning with convolutional neural network arXiv:1607.08707