PAPER • OPEN ACCESS

Deep null space learning for inverse problems: convergence analysis and rates

To cite this article: Johannes Schwab *et al* 2019 *Inverse Problems* **35** 025008

View the article online for updates and enhancements.

IOP ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

Deep null space learning for inverse problems: convergence analysis and rates

Johannes Schwab, Stephan Antholzer
and Markus Haltmeier[1]

Department of Mathematics, University of Innsbruck Technikerstrasse 13,
6020 Innsbruck, Austria

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

CrossMark

Abstract

Recently, deep learning based methods appeared as a new paradigm for solving inverse problems. These methods empirically show excellent performance but lack of theoretical justification; in particular, no results on the regularization properties are available. In particular, this is the case for two-step deep learning approaches, where a classical reconstruction method is applied to the data in a first step and a trained deep neural network is applied to improve results in a second step. In this paper, we close the gap between practice and theory for a particular network structure in a two-step approach. For that purpose, we propose using so-called null space networks and introduce the concept of Φ-regularization. Combined with a standard regularization method as reconstruction layer, the proposed deep null space learning approach is shown to be a Φ-regularization method; convergence rates are also derived. The proposed null space network structure naturally preserves data consistency which is considered as key property of neural networks for solving inverse problems.

Keywords: convergence analysis, deep learning, convergences rates, null space networks, Phi-regularization

[1] Author to whom any correspondence should be addressed.

1. Introduction

We study the solution of inverse problems of the form

$$\text{Estimate } x \in X \text{ from data} \quad y^\delta = \mathbf{A}x + \xi. \tag{1.1}$$

Here $\mathbf{A} \colon X \to Y$ is a linear operator between Hilbert spaces X and Y, and $\xi \in Y$ models the unknown data error (noise), which is assumed to satisfy the estimate $\|\xi\| \leqslant \delta$ for some noise level $\delta \geqslant 0$. We consider a possibly infinite-dimensional function space setting, but the approach and results apply to a finite dimensional setting as well.

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 [7, 22]). For its stable solution one has to employ regularization methods, which are based on approximating (1.1) by neighboring well-posed problems, which enforce stability, accuracy, and uniqueness.

1.1. Regularization methods

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 $\mathcal{M}$ or that it has small value of some regularizing functional. The most basic regularization method is probably Tikhonov regularization, where the solution is defined as a minimizer of the quadratic Tikhonov functional

$$\mathcal{T}_{\alpha;y_\delta}(x) := \frac{1}{2}\left\|\mathbf{A}(x) - y^\delta\right\|^2 + \frac{\alpha}{2}\|x\|^2. \tag{1.2}$$

Other classical regularization methods for solving linear inverse problems are filter based methods [7], which include Tikhonov regularization as special case.

In the last couple of years variational regularization methods including TV regularization or ℓ^q regularization became popular [22]. They also include classical Tikhonov regularization as special case. In the general version, the regularizer $\frac{1}{2}\|\cdot\|^2$ is replaced by general convex and lower semi-continuous functionals.

In this paper, we develop a new regularization concept that we name $\mathbf{\Phi}$-regularization method. Roughly spoken, a $\mathbf{\Phi}$-regularization method is a tuple $((\mathbf{R}_\alpha)_{\alpha>0}, \alpha^\star)$ where (for a precise definition see definition 2.3)

- $\mathcal{M} \subseteq X$ is the set of admissible elements, defined by the function $\mathbf{\Phi}$;
- $\mathbf{R}_\alpha \colon Y \to X$ are continuous mappings;
- $\alpha^\star = \alpha^\star(\delta, y^\delta)$ is a suitable parameter choice;
- For any $x \in \mathcal{M}$ we have $\mathbf{R}_{\alpha^\star(\delta,y^\delta)}(y^\delta) \to x$ as $\delta \to 0$.

Note that for some cases is might be reasonable to take $\mathbf{R}_\alpha$ multivalued. For the sake of simplicity here we only consider the single-valued case. Classical regularization methods are special cases of $\mathbf{\Phi}$-regularization methods in Hilbert spaces where $\mathcal{M} = \ker(\mathbf{A})^\perp$. A typical regularization method is in this case given by Tikhonov regularization, where $\mathbf{R}_\alpha = (\mathbf{A}^*\mathbf{A} + \alpha\mathrm{Id}_X)^{-1}\mathbf{A}^*$. Here and in the following Id_X denotes the identity on X.

1.2. Solving inverse problems by neural networks

Very recently, deep learning approaches appeared as alternative, very successful methods for solving inverse problems (see, for example [2–6, 8, 10, 12, 15, 25–28]). In most of these approaches, a reconstruction network $\mathbf{R}\colon Y \to X$ is trained to map measured data to the desired output image.

Various reconstruction networks have been introduced in the literature. In the two-step approach, the reconstruction networks take the form $\mathbf{R} = \mathbf{L} \circ \mathbf{B}$ where $\mathbf{B}\colon Y \to X$ maps the data to the reconstruction space (reconstruction layer or backprojection; no free parameters) and $\mathbf{L}\colon X \to X$ is a neural network (NN) whose free parameters are adjusted to the training data. In particular, so called residual networks $\mathbf{L} = \mathrm{Id}_X + \mathbf{N}$ where only the residual part $\mathbf{N}$ is trained [11] show very accurate results for solving inverse problems [3, 6, 12, 13, 16, 18, 21, 26]. Another class of reconstruction networks learns free parameters in iterative schemes. In such approaches, a sequence of reconstruction networks $\mathbf{R} = \mathbf{R}^{(k)}$ is defined by some iterative process $\mathbf{R}^{(k)}(y) = \mathbf{N}_k(\Phi_k(x_{k-1}, \ldots, x_0, y))$ where x_0 is some the initial guess, $\mathbf{N}_k\colon X \to X$ are networks that can be adjusted to available training data, and Φ_k are updates based on the data and the previous iterates [2, 14, 15, 23].

Further existing deep learning approaches for solving inverse problems are based on trained projection operators [5, 8], or use neural networks as trained regularization term [17].

While the above deep learning based reconstruction networks empirically yield good performance, none of them is known to be a convergent regularization method. In this paper, we use a new network structure (null space network) that, when combined with a classical regularization of the Moore Penrose inverse is shown to provide a convergent Φ-regularization method with rates. One of the reviewers of this manuscript kindly brought to our attention that the null space network structure actually has been introduced already by Mardani and collaborators in [19, 20] in a finite dimensional setting. We extend the use of the null space network to operators with non-closed range and analyze its stable approximation in the context of regularization methods.

1.3. Proposed null space networks and main results

As often argued in the recent literature, deep learning based reconstruction approaches (especially using two-stage networks) lack data consistency, in the sense that outputs of existing reconstruction networks fail to accurately predict the given data. In order to overcome this issue, in this paper, we introduce a new network, that we name null space network. The proposed null space network takes the form

$$\mathbf{L} = \mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N} \quad \text{for Lipschitz continuous } \mathbf{N}\colon X \to X. \tag{1.3}$$

The function $\mathbf{N}\colon X \to X$, for example, can be defined by a neural network according to definition 3.2. Note that $\mathrm{Id}_X - \mathbf{A}^+\mathbf{A} = \mathbf{P}_{\mathrm{ker}(\mathbf{A})}$ equals the projector onto the null space $\mathrm{ker}(\mathbf{A})$ of $\mathbf{A}$. Consequently, the null space network $\mathbf{L}$ satisfies the property $\mathbf{A}\mathbf{L}x = \mathbf{A}x$ for all $x \in X$. This yields data consistency, which means that the equation $\mathbf{A}x = y$ is invariant among application of a null space network (compare figure 1).

Suppose $x_1, \ldots, x_N$ are some desired output images and let $\mathbf{L}$ be a trained null space network that approximately maps $\mathbf{A}^+\mathbf{A}x_n$ to x_n. (See the appendix for a possible training strategy.) In this paper, we show that if $(\mathbf{B}_\alpha)_{\alpha > 0}$ is any classical $\mathrm{ker}(\mathbf{A})^\perp$-regularization, then the two-stage reconstruction network

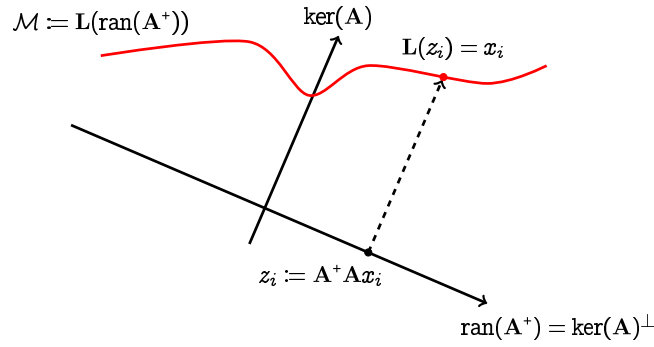$$\mathbf{R}_\alpha := \mathbf{L} \circ \mathbf{B}_\alpha \quad \text{for } \alpha > 0 \tag{1.4}$$

Figure 1. Sketch of the action of a null space network $\mathbf{L}$ that maps points $z_i \in \mathrm{ran}(\mathbf{A}^+)$ to more desirable elements in $z_i + \mathrm{ker}(\mathbf{A})$ along the null space of $\mathbf{A}$. The component $\mathrm{ker}(\mathbf{A})$ is invisible in the data $\mathbf{A}x$, whereas the part $\mathrm{ker}(\mathbf{A})^\perp$ can be found by applying the pseudoinverse $\mathbf{A}^+$ to the data $\mathbf{A}x$.

yields a $\mathbf{\Phi}$-regularization with $\mathcal{M} := \mathbf{L}(\mathrm{ran}(\mathbf{A}^+))$. To the best of our knowledge, these are first results for regularization by neural networks. Additionally, we will derive convergence rates for $(\mathbf{R}_\alpha)_{\alpha>0}$ on suitable function classes.

The intuition behind using the null space network in the two-stage approach (1.4) is that only invisible information in $\mathrm{ker}(\mathbf{A})$ should be learned by the network, whereas the visible part in $\mathrm{ker}(\mathbf{A})^\perp$ should be kept (compare figure 1). Moreover, in the case that $\mathbf{A}$ has non-closed range then the visible part $\mathrm{ker}(\mathbf{A})^\perp$ will be be sensitive with respect to data perturbation. These instabilities with respect to noise can exactly be addressed by a regularizing family $(\mathbf{B}_\alpha)_{\alpha>0}$ of continuous operators that converge pointwise to $\mathbf{A}^+$ in the limit $\alpha \to 0$.

1.4. Outline

This paper is organized as follows. In section 2 we develop a general theory of $\mathbf{\Phi}$-regularization and introduce the notion of $\mathbf{\Phi}$-generalized inverse (definition 2.1) and $\mathbf{\Phi}$-regularization methods (definition 2.3) generalizing the classical Moore–Penrose generalized inverse and regularization concept. We show convergence (see theorem 2.4) and derive convergence rates (theorem 2.8) that include regularization via the null space networks as a special case. In section 3 we introduce the null-space networks and extend the convergence results in the special case of the null space network (theorems 3.4 and 3.5). Possible strategies for training a nullspace network are described in the appendix. The paper concludes with an outlook presented in section 4.

2. A theory of $\mathbf{\Phi}$-regularization

In this section, we introduce the novel concepts of $\mathbf{\Phi}$-generalized inverse and $\mathbf{\Phi}$-regularization. We derive a general class of $\mathbf{\Phi}$-regularization for which we show convergence and derive convergence rates.

Throughout this section, let $\mathbf{A}\colon X \to Y$ be a linear bounded operator and $\mathbf{\Phi}\colon X \to \mathrm{ker}(\mathbf{A}) \subseteq X$ be Lipschitz continuous and define

$$\mathcal{M} := (\mathrm{Id}_X + \mathbf{\Phi})\mathrm{ran}(\mathbf{A}^+). \tag{2.1}$$

The prime example is $\mathbf{\Phi} = \mathbf{P}_{\ker(\mathbf{A})} \circ \mathbf{N}$ being a null space network with a neural network function $\mathbf{N} \colon X \to X$. This case will be studied in the following section. The results presented in this section apply to general Lipschitz continuous functions $\mathbf{\Phi}$ whose image is contained in $\ker(\mathbf{A})$.

2.1. Φ-regularization methods

In the following we denote by $\mathbf{A}^+ \colon \mathrm{dom}(\mathbf{A}^+) \subseteq Y \to X$ the Moore–Penrose generalized inverse of $\mathbf{A}$, defined by $\mathrm{dom}(\mathbf{A}^+) := \mathrm{ran}(\mathbf{A}) \oplus \mathrm{ran}(\mathbf{A})^\perp$ and

$$\mathbf{A}^+(y) := \arg\min\{\|x\|^2 \mid x \in X \wedge \mathbf{A}^*\mathbf{A}x = \mathbf{A}^*y\}. \tag{2.2}$$

It is well known [7] that $\{x \in X \mid \mathbf{A}^*\mathbf{A}x = \mathbf{A}^*y\} \neq \emptyset$ if and only if $y \in \mathrm{ran}(\mathbf{A}) \oplus \mathrm{ran}(\mathbf{A})^\perp$. In particular, $\mathbf{A}^+y$ is well defined, and can be found as the unique minimal norm solution of the normal equation $\mathbf{A}^*\mathbf{A}x = \mathbf{A}^*y$.

Classical regularization methods aim for approximating $\mathbf{A}^+y$. In contrast, the null space network will recover different solutions of the normal equation. For that purpose, we introduce the following concept.

Definition 2.1 (Φ-generalized inverse).　We call $\mathbf{A}^{\mathbf{\Phi}} \colon \mathrm{dom}(\mathbf{A}^+) \subseteq Y \to X$ the $\mathbf{\Phi}$-generalized inverse of $\mathbf{A}$ if

$$\forall y \in \mathrm{dom}(\mathbf{A}^+) \colon \quad \mathbf{A}^{\mathbf{\Phi}}y = (\mathrm{Id}_X + \mathbf{\Phi})(\mathbf{A}^+y). \tag{2.3}$$

Recall that for any $y \in \mathrm{dom}(\mathbf{A}^+)$, the solution set of the normal equation $\mathbf{A}^*\mathbf{A}x = \mathbf{A}^*y$ is given by $\mathbf{A}^+y + \ker(\mathbf{A})$. Hence $\mathbf{A}^{\mathbf{\Phi}}y$ gives a particular solution of the normal equation, that can be adapted to a training set. The $\mathbf{\Phi}$-generalized inverse coincides with the Moore–Penrose generalized inverse if and only if $\mathbf{\Phi}(x) = 0$ for all $x \in X$ in which case $\mathcal{M} = \ker(\mathbf{A})^\perp = \mathrm{ran}(\mathbf{A}^+)$.

Lemma 2.2.　*The $\mathbf{\Phi}$-generalized inverse is continuous if and only if $\mathrm{ran}(\mathbf{A})$ is closed.*

Proof.　If $\mathrm{ran}(\mathbf{A})$ is closed, then classical results show that $\mathbf{A}^+$ is bounded (see for example [7]). Consequently, $(\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{A}^+$ is bounded too. Conversely, if $\mathbf{A}^{\mathbf{\Phi}}$ is continuous, then the identity $\mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)}\mathbf{A}^{\mathbf{\Phi}} = \mathbf{A}^+$ implies that the Moore–Penrose generalized inverse $\mathbf{A}^+$ is bounded and therefore that $\mathrm{ran}(\mathbf{A})$ is closed.　□

Lemma 2.2 shows that as in the case of the classical Moore–Penrose generalized inverse, the $\mathbf{\Phi}$-generalized inverse is discontinuous in the case that $\mathrm{ran}(\mathbf{A})$ is not closed. In order to stably solve the equation $\mathbf{A}x = y$ we therefore require bounded approximations of the $\mathbf{\Phi}$-generalized inverse. For that purpose, we introduce the following concept of regularization methods adapted to $\mathbf{A}^+$.

Definition 2.3 (Φ-regularization method).　Let $(\mathbf{R}_\alpha)_{\alpha > 0}$ be a family of continuous (not necessarily linear) mappings $\mathbf{R}_\alpha \colon Y \to X$ and let $\alpha^\star \colon (0, \infty) \times Y \to (0, \infty)$. We call the pair $((\mathbf{R}_\alpha)_{\alpha > 0}, \alpha^\star)$ a $\mathbf{\Phi}$-regularization method for the equation $\mathbf{A}x = y$ if the following hold:

- $\forall y \in \mathrm{dom}(A^+) \colon \ \lim_{\delta \to 0} \sup\{\alpha^\star(\delta, y^\delta) \mid y^\delta \in \mathrm{dom}(A^+) \wedge \|y^\delta - y\| \leqslant \delta\} = 0.$
- $\forall y \in \mathrm{dom}(A^+) \colon \ \lim_{\delta \to 0} \sup\{\|\mathbf{A}^{\mathbf{\Phi}}y - \mathbf{R}_{\alpha^\star(\delta, y^\delta)}y^\delta\| \mid y^\delta \in \mathrm{dom}(A^+) \wedge \|y^\delta - y\| \leqslant \delta\} = 0.$

In the case that $((\mathbf{R}_\alpha)_{\alpha > 0}, \alpha^\star)$ is a $\mathbf{\Phi}$-regularization method for $\mathbf{A}x = y$, then we call the family $(\mathbf{R}_\alpha)_{\alpha > 0}$ a regularization of $\mathbf{A}^{\mathbf{\Phi}}$ and $\alpha^\star$ an admissible parameter choice.

In our generalized notation, a classical regularization method for the equation $\mathbf{A}x = y$ corresponds to a **0**-regularization method for $\mathbf{A}x = y$. Further note that the null space assumption is required for $(\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{A}^+ y$ with $y \in \mathrm{ran}(\mathbf{A}^+)$ being a solution of $\mathbf{A}x = y$ instead of being any element. We consider this data consistency property central for solving inverse problems with neural network based reconstruction methods.

2.2. Convergence analysis

The following theorem shows that the combination of a null space network and a regularization method of $\mathbf{A}^+$ yields a regularization of $\mathbf{A}^{\mathbf{\Phi}}$.

Theorem 2.4. *Let* $\mathbf{A} \colon X \to Y$ *be a linear operator and* $\mathbf{\Phi} \colon X \to \ker(\mathbf{A}) \subseteq X$ *be Lipschitz continuous. Moreover, suppose* $((\mathbf{B}_\alpha)_{\alpha>0}, \alpha^\star)$ *is any classical regularization method for* $\mathbf{A}x = y$. *Then, the pair* $((\mathbf{R}_\alpha)_{\alpha>0}, \alpha^\star)$ *with* $\mathbf{R}_\alpha := (\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{B}_\alpha$ *is a* $\mathbf{\Phi}$*-regularization method for* $\mathbf{A}x = y$. *In particular, the family* $(\mathbf{R}_\alpha)_{\alpha>0}$ *is a regularization of* $\mathbf{A}^{\mathbf{\Phi}}$.

Proof. Because $((\mathbf{B}_\alpha)_{\alpha>0}, \alpha^\star)$ is a **0**-regularization method, $\lim_{\delta \to 0} \sup\{\alpha^\star(\delta, y^\delta) \,|\, y^\delta \in Y \text{ und } \|y^\delta - y\| \leqslant \delta\} = 0$. Let L be a Lipschitz constant of $\mathrm{Id}_X + \mathbf{\Phi}$. For any y^δ we have

$$\left\| \mathbf{A}^{\mathbf{\Phi}} y - \mathbf{R}_{\alpha^\star(\delta, y^\delta)} y^\delta \right\| = \left\| (\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{A}^+ y - (\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{B}_{\alpha^\star(\delta, y^\delta)} y^\delta \right\|$$
$$\leqslant L \left\| \mathbf{A}^+ y - \mathbf{B}_{\alpha^\star(\delta, y^\delta)} y^\delta \right\|.$$

Consequently

$$\sup \left\{ \| \mathbf{A}^{\mathbf{\Phi}} y - (\mathrm{Id}_X + \mathbf{\Phi}) \mathbf{B}_{\alpha^\star(\delta, y^\delta)} y^\delta \| \,\middle|\, y^\delta \in Y \wedge \|y^\delta - y\| \leqslant \delta \right\}$$
$$\leqslant L \sup \left\{ \| \mathbf{A}^+ y - \mathbf{B}_{\alpha^\star(\delta, y^\delta)} y^\delta \| \,\middle|\, y^\delta \in Y \wedge \|y^\delta - y\| \leqslant \delta \right\} \to 0.$$

In particular, $((\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{B}_\alpha)_{\alpha>0}$ is a regularization of $\mathbf{A}^{\mathbf{\Phi}}$. $\qquad\square$

A wide class of $\mathbf{\Phi}$-regularization methods can be defined by a regularizing filter.

Definition 2.5. A family $(g_\alpha)_{\alpha>0}$ of functions $g_\alpha \colon [0, \|\mathbf{A}^*\mathbf{A}\|] \to \mathbb{R}$ is called a regularizing filter if it satisfies

- For all $\alpha > 0$, g_α is piecewise continuous;
- $\exists C > 0 \colon \sup\{|\lambda g_\alpha(\lambda)| \,|\, \alpha > 0 \wedge \lambda \in [0, \|\mathbf{A}^*\mathbf{A}\|]\} \leqslant C$;
- $\forall \lambda \in (0, \|\mathbf{A}^*\mathbf{A}\|] \colon \lim_{\alpha \to 0} g_\alpha(\lambda) = 1/\lambda$.

Corollary 2.6. *Let* $(g_\alpha)_{\alpha>0}$ *be a regularizing filter and define* $\mathbf{B}_\alpha := g_\alpha(\mathbf{A}^*\mathbf{A})\,\mathbf{A}^*$. *Then* $((\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{B}_\alpha)_{\alpha>0}$ *is a regularization of* $\mathbf{A}^{\mathbf{\Phi}}$.

Proof. The family $(\mathbf{B}_\alpha)_\alpha$ is a regularization of $\mathbf{A}^+$; see [7]. Therefore, according to theorem 2.4, $((\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{B}_\alpha)_{\alpha>0}$ is a regularization of $\mathbf{A}^{\mathbf{\Phi}}$. $\qquad\square$

Basic examples of filter based regularization methods are Tikhonov regularization, where $g_\alpha(\lambda) = 1/(\alpha + \lambda)$, and truncated singular value decomposition where

$$g_\alpha(\lambda) := \begin{cases} 0 & \text{if } \lambda < \alpha \\ 1/\lambda & \text{if } \lambda \geqslant \alpha. \end{cases}$$

Classical regularization methods are based on approximating the Moore–Penrose inverse. The following result shows that $\mathbf{\Phi}$-regularization methods are essentially continuous approximations of $\mathbf{A}^{\mathbf{\Phi}}$.

Proposition 2.7. *Let $(\mathbf{R}_\alpha)_{\alpha>0}$ be a family of continuous mappings $\mathbf{R}_\alpha \colon Y \to X$.*

(a) *If $\mathbf{R}_\alpha|_{\mathrm{dom}(\mathbf{A}^+)} \to \mathbf{A}^{\mathbf{\Phi}}$ pointwise as $\alpha \to 0$, then the family $(\mathbf{R}_\alpha)_{\alpha>0}$ is a regularization of $\mathbf{A}^{\mathbf{\Phi}}$.*
(b) *Suppose that $(\mathbf{R}_\alpha)_{\alpha>0}$ is a regularization of $\mathbf{A}^{\mathbf{\Phi}}$ and that there exists a parameter choice $\alpha^\star$ that is continuous in the first argument. Then $\mathbf{R}_\alpha|_{\mathrm{dom}(\mathbf{A}^+)} \to \mathbf{A}^{\mathbf{\Phi}}$ pointwise as $\alpha \to 0$.*

Proof.

(a) If $\mathbf{R}_\alpha|_{\mathrm{dom}(\mathbf{A}^+)} \to \mathbf{A}^{\mathbf{\Phi}}$ pointwise, then $\mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)} \circ \mathbf{R}_\alpha|_{\mathrm{dom}(\mathbf{A}^+)} \to \mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)} \circ \mathbf{A}^{\mathbf{\Phi}} = \mathbf{A}^+$ pointwise. Hence, classical regularization theory implies that $\mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)} \circ \mathbf{R}_\alpha$ is a regularization of $\mathbf{A}^+$. We have $\mathbf{R}_\alpha = (\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)} \circ \mathbf{R}_\alpha$ and, according to theorem 2.4, the family $(\mathbf{R}_\alpha)_{\alpha>0}$ is a regularization of $\mathbf{A}^{\mathbf{\Phi}}$.
(b) We have

$$\sup\left\{ \|\mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)}(\mathbf{A}^{\mathbf{\Phi}}y - \mathbf{R}_{\alpha^\star(\delta,y^\delta)}y^\delta)\| \mid y^\delta \in Y \wedge \|y^\delta - y\| \leqslant \delta \right\}$$
$$\leqslant \sup\left\{ \|\mathbf{A}^{\mathbf{\Phi}}y - \mathbf{R}_{\alpha^\star(\delta,y^\delta)}y^\delta\| \mid y^\delta \in Y \wedge \|y^\delta - y\| \leqslant \delta \right\} \to 0,$$

which shows that $(\mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)} \circ \mathbf{R}_\alpha)_{\alpha>0}$ is a regularization of $\mathbf{A}^+ = \mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)} \circ \mathbf{A}^{\mathbf{\Phi}}$. Together with standard regularization theory this shows that $\mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)} \circ \mathbf{R}_\alpha|_{\mathrm{dom}(\mathbf{A}^+)} \to \mathbf{A}^+$ pointwise as $\alpha \to 0$. Consequently, $\mathbf{R}_\alpha|_{\mathrm{dom}(\mathbf{A}^+)} = (\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)} \circ \mathbf{R}_\alpha|_{\mathrm{dom}(\mathbf{A}^+)}$ converges pointwise to $\mathbf{A}^{\mathbf{\Phi}} = (\mathrm{Id}_X + \mathbf{\Phi}) \circ \mathbf{A}^+$. $\qquad\square$

2.3. Convergence rates

Next we derive quantitative error estimates. For that purpose, we assume in the following that $\mathbf{B}_\alpha = g_\alpha(\mathbf{A}^*\mathbf{A})\,\mathbf{A}^*$ is defined by the regularizing filter $(g_\alpha)_{\alpha>0}$. We use the notation $\alpha^\star \asymp (\delta/\rho)^a$ as $\delta \to 0$ where $\alpha^\star \colon Y \times (0,\infty) \to (0,\infty)$ and $a,\rho > 0$ to indicate there are positive constants d_1, d_2 such that $d_1(\delta/\rho)^a \leqslant \alpha^\star(\delta) \leqslant d_2(\delta/\rho)^a$.

Theorem 2.8. *Suppose $\mu, \rho > 0$ and let $(g_\alpha)_{\alpha>0}$ be a regularizing filter such that there exist constants $\alpha_0, c_1, c_2 > 0$ with*

- $\forall \alpha > 0 \; \forall \lambda \in [0, \|\mathbf{A}^*\mathbf{A}\|] \colon \lambda^\mu\,|1 - \lambda g_\alpha(\lambda)| \leqslant c_1 \alpha^\mu;$
- $\forall \alpha \in (0, \alpha_0) \colon \|g_\alpha\|_\infty \leqslant c_2/\alpha.$

Consider the $\mathbf{\Phi}$-regularization method $\mathbf{R}_\alpha := (\mathrm{Id}_X + \mathbf{\Phi}) \circ g_\alpha(\mathbf{A}^*\mathbf{A})\mathbf{A}^*$ and set

$$\mathcal{M}_{\mu,\rho,\mathbf{\Phi}} := (\mathrm{Id}_X + \mathbf{\Phi})\,(\mathbf{A}^*\mathbf{A})^\mu\left(\overline{B_\rho(0)}\right). \tag{2.4}$$

Moreover, let $\alpha^\star \colon (0,\infty) \times Y \to (0,\infty)$ be a parameter choice (possible depending on the source set $\mathcal{M}_{\mu,\rho,\mathbf{\Phi}}$) that satisfies $\alpha^\star \asymp (\delta/\rho)^{\frac{2}{2\mu+1}}$ as $\delta \to 0$. Then there exists a constant $c > 0$ such that

$$\sup \left\{ \|\mathbf{R}_{\alpha^\star(\delta,y^\delta)}(y^\delta) - x\| \mid x \in \mathcal{M}_{\mu,\rho,\mathbf{\Phi}} \wedge y^\delta \in Y \wedge \|\mathbf{A}x - y^\delta\| \leqslant \delta \right\}$$
$$\leqslant c\delta^{\frac{2\mu}{2\mu+1}} \rho^{\frac{1}{2\mu+1}}. \qquad (2.5)$$

In particular, for any $x \in \mathrm{ran}((\mathrm{Id}_X + \mathbf{\Phi}) \circ (\mathbf{A}^*\mathbf{A})^\mu)$ *we have the convergence rate result* $\|\mathbf{R}_{\alpha^\star(\delta,y^\delta)}(y^\delta) - x\| = \mathcal{O}(\delta^{\frac{2\mu}{2\mu+1}})$.

Proof. We have $\mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)}\mathcal{M}_{\mu,\rho,\mathbf{\Phi}} = (\mathbf{A}^*\mathbf{A})^\mu \left(\overline{B_\rho(0)}\right)$ and $\mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)}\mathbf{R}_\alpha = g_\alpha(\mathbf{A}^*\mathbf{A})\mathbf{A}^*$. Suppose $x \in \mathcal{M}_{\mu,\rho,\mathbf{\Phi}}$ and $y^\delta \in Y$ with $\|\mathbf{A}x - y^\delta\| \leqslant \delta$. Under the given assumptions, $g_\alpha(\mathbf{A}^*\mathbf{A})\mathbf{A}^*$ is an order optimal regularization method on $(\mathbf{A}^*\mathbf{A})^\mu \left(\overline{B_\rho(0)}\right)$, which implies (see [7])

$$\left\| g_{\alpha^\star(\delta,y^\delta)}(\mathbf{A}^*\mathbf{A})\mathbf{A}^*(y^\delta) - \mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)}x \right\| \leqslant C\delta^{\frac{2\mu}{2\mu+1}} \rho^{\frac{1}{2\mu+1}}$$

for some constant $C > 0$ independent of x, y^δ. Consequently, we have

$$
\begin{aligned}
\|\mathbf{R}_{\alpha^\star(\delta,y^\delta)}(y^\delta) - x\| &= \|(\mathrm{Id}_X + \mathbf{\Phi}) \circ g_{\alpha^\star(\delta,y^\delta)}(\mathbf{A}^*\mathbf{A})\mathbf{A}^*(y^\delta) - (\mathrm{Id}_X + \mathbf{\Phi})\mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)}x\| \\
&\leqslant L \|g_{\alpha^\star(\delta,y^\delta)}(\mathbf{A}^*\mathbf{A})\mathbf{A}^*(y^\delta) - \mathbf{P}_{\mathrm{ran}(\mathbf{A}^+)}x\| \\
&\leqslant LC\delta^{\frac{2\mu}{2\mu+1}} \rho^{\frac{1}{2\mu+1}},
\end{aligned}
$$

where L is the Lipschitz constant of $\mathrm{Id}_X + \mathbf{\Phi}$. Taking the supremum over all $x \in \mathcal{M}_{\mu,\rho,\mathbf{\Phi}}$ and $y^\delta \in Y$ with $\|\mathbf{A}x - y^\delta\| \leqslant \delta$ yields (2.5). $\square$

Note that the filters $(g_\alpha)_{\alpha>0}$ of the truncated SVD and the Landweber iteration satisfy the assumptions of theorem 2.8. In the case of Tikhonov regularization, the assumptions are satisfied for $\mu \leqslant 1$. In particular, under the assumption (resembling the classical source condition)

$$x \in (\mathrm{Id}_X + \mathbf{\Phi})(\mathrm{ran}(\mathbf{A}^+))$$

we obtain the convergence rate $\|\mathbf{R}_{\alpha^\star(\delta,y^\delta)}(y^\delta) - x\| = \mathcal{O}(\delta^{1/2})$.

3. Deep null space learning

Throughout this section let $\mathbf{A}\colon X \to Y$ be a linear bounded operator. In this case, we define $\mathbf{\Phi}$-regularizations by null-space networks. We describe a possible training strategy and derive regularization properties and rates. For the following recall that the projector onto the kernel of $\mathbf{A}$ is given by $\mathbf{P}_{\mathrm{ker}(\mathbf{A})} = \mathrm{Id}_X - \mathbf{A}^+\mathbf{A}$.

3.1. Null space networks

We work with layered feed forward networks in infinite dimensional spaces, although more complicated networks can be applied as long as their Lipschitz constant is finite. As the error estimates depend on the Lipschitz constant it is also desirable that the Lipschitz constant is not too large. Neural network functions in infinite dimensional spaces can be found in [1, 9, 17] and the references therein. Nevertheless, while the notion of neural networks is standard in a finite-dimensional setting, no established definition seems available for general Hilbert spaces. Here we use the following notation in Hilbert space notion.

Definition 3.1 (Layered feed forward network). Let X and Z be Hilbert spaces. We call a function $\mathbf{N}\colon X \to Z$ defined by

$$\mathbf{N} := \sigma_L \circ \mathbf{W}_L \circ \sigma_{L-1} \circ \mathbf{W}_{L-1} \circ \cdots \circ \sigma_1 \circ \mathbf{W}_1, \tag{3.1}$$

a layered feed forward neural network function of depth $L \in \mathbb{N}$ with activations $\sigma_1, \ldots, \sigma_L$ if

(N1) X_ℓ are Hilbert spaces with $X_0 = X$ and $X_L = Z$;
(N2) $\mathbf{W}_\ell \colon X_{\ell-1} \to X_\ell$ are affine, continuous;
(N3) $\sigma_\ell \colon X_\ell \to X_\ell$ are continuous.

Usually the nonlinearities σ_ℓ are fixed and the affine mappings $\mathbf{W}_\ell$ are trained. In the case that X_ℓ is a function space, then a standard operation for σ_ℓ is the ReLU (the rectified linear unit), $\mathrm{ReLU}(x) := \max\{x, 0\}$, that is applied component-wise, or ReLU in combination with max pooling which takes the maximum value $\max\{|x(i)| : i \in I_k\}$ within clusters of transform coefficients. Note that in the typical case of L^2-spaces the elements are equivalence classes of functions. In this case, any representative has to be selected for the application of ReLU, which is clearly independent of the chosen representative. The network in definition 3.1 may in particular be a convolutional neural network (CNN); see [17] for a definition of CNNs in Banach spaces. In a similar manner, one could define more general feed forward networks in Hilbert spaces, for example following the notion of [24] in the finite dimensional case.

We are now able to formally define the concept of a null space network.

Definition 3.2. A function $\mathbf{L} \colon X \to X$ is called a null space network if it has the form $\mathbf{L} = \mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$ where $\mathbf{N} \colon X \to X$ is any Lipschitz continuous neural network function.

An example for the null space network with $\mathbf{N} \colon X \to X$ as defined in (3.1). We however again point out that $\mathbf{N}$ could be a more general Lipschitz continuous neural network function for which the results below equally hold. For the sake of clarity of presentation, we use the simple definition 3.1 of layered neural networks.

An example of a standard residual network $\mathrm{Id}_X + \mathbf{N}$ and a layered null space network $\mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$ both with depth $L = 2$ (i.e. two weight layers) are shown in figure 2.

Remark 3.3. Throughout the following we assume that $\mathbf{L} = \mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$ is a given null-space network. Following the deep learning philosophy, the network would be selected from a parameterized family $(\mathbf{L}_\theta)_{\theta \in \theta}$ based on given training data. A possible training strategy is presented in appendix. The results below hold for any null space network where the Lipschitz constant is finite. It is widely accepted that the Lipschitz constant in typically trained networks is reasonably small. As the error constant depends on the Lipschitz constants of the network it is desirable to keep the Lipschitz constant small. The proposed training strategy also accounts for this issue in the layered neural networks according to definition 3.1.

Another simple way of constructing a null space network is to add a data consistency layer to an existing network. To be specific, let $\mathbf{L}_0$ be any trained network. Then one obtains a null space network by considering

$$\mathbf{L} = \mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})(\mathbf{L}_0 - \mathrm{Id}_X). \tag{3.2}$$

Moreover, one can easily show that $\|x - \mathbf{L}\mathbf{A}^+\mathbf{A}x\| \leqslant \|x - \mathbf{L}_0\mathbf{A}^+\mathbf{A}x\|$ for every $x \in X$. Hence the null space network is always better in terms of reconstruction error than the original network for recovering x from data $y = \mathbf{A}x$.
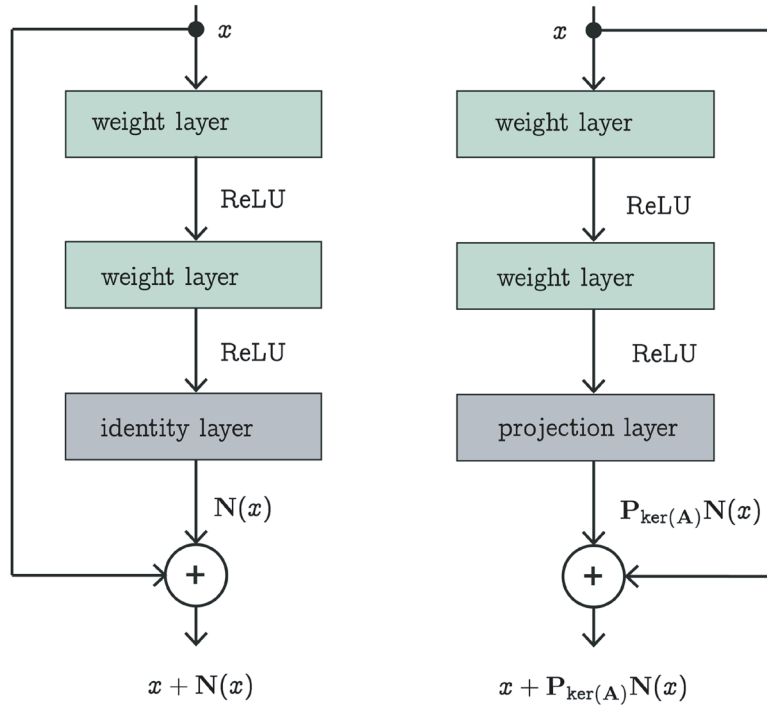
Figure 2. Standard residual network (left) versus null space network (right). The only difference is the projection layer on $\mathbf{P}_{\ker(\mathbf{A})}$ after the last weight layer.

3.2. Convergence and convergence rates

Let $\mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$ be a null-space network, possibly trained as described in appendix by approximately minimizing (A.1). Any such network belongs to the class of functions $\mathrm{Id}_X + \boldsymbol{\Phi}$ by taking $\boldsymbol{\Phi} = (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$. Consequently, the convergence theory of section 2 applies. In particular, theorem 2.4 shows that a regularization $(\mathbf{B}_\alpha)_{\alpha>0}$ of the Moore–Penrose generalized inverse defines a $\boldsymbol{\Phi}$-regularization method via $\mathbf{R}_\alpha := (\mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N})\mathbf{B}_\alpha$. Additionally, theorem 2.8 yields convergence rates for the regularization $(\mathbf{R}_\alpha)_{\alpha>0}$ of $\mathbf{A}^{\boldsymbol{\Phi}}$.

In some cases, the projection $\mathbf{P}_{\ker(\mathbf{A})} = \mathrm{Id}_X - \mathbf{A}^+\mathbf{A}$ might be costly to be computed exactly. For that purpose, in this section we derive more general regularization methods that include approximate evaluations of $\mathbf{A}^+\mathbf{A}$.

Theorem 3.4. *Let* $\mathbf{L} = \mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$ *be a null space network and set* $\mathcal{M} := \mathrm{ran}(\mathbf{L})$. *Suppose* $((\mathbf{B}_\alpha)_{\alpha>0}, \alpha^\star)$ *is a regularization method for* $\mathbf{A}x = y$. *Moreover, let* $(\mathbf{Q}_\alpha)_{\alpha>0}$ *be a family of bounded operators on* X *with* $\|\mathbf{Q}_\alpha - \mathbf{P}_{\ker(\mathbf{A})}\| \to 0$ *as* $\alpha \to 0$. *Then, the pair* $((\mathbf{R}_\alpha)_{\alpha>0}, \alpha^\star)$ *with*

$$\mathbf{R}_\alpha := (\mathrm{Id}_X + \mathbf{Q}_\alpha \mathbf{N}) \circ \mathbf{B}_\alpha \tag{3.3}$$

is a $\boldsymbol{\Phi}$*-regularization method for* $\mathbf{A}x = y$. *In particular, the family* $(\mathbf{R}_\alpha)_{\alpha>0}$ *is a regularization of* $\mathbf{A}^{\boldsymbol{\Phi}}$.

Proof. We have

$$\begin{aligned}
&\left\|(\mathrm{Id}_X + \mathbf{Q}_\alpha\mathbf{N}) \circ \mathbf{B}_\alpha(y^\delta) - \mathbf{A}^{\boldsymbol{\Phi}}y\right\| \\
&\leqslant \left\|(\mathrm{Id}_X + \mathbf{P}_{\ker(\mathbf{A})}\mathbf{N}) \circ \mathbf{B}_\alpha(y^\delta) - \mathbf{A}^{\boldsymbol{\Phi}}y\right\| + \left\|\mathbf{Q}_\alpha - \mathbf{P}_{\ker(\mathbf{A})}\right\| \left\|\mathbf{N}\mathbf{B}_\alpha(y^\delta)\right\|.
\end{aligned}$$

(3.4)

The claim follows from theorem 2.4. □

Theorem 3.5. *Let* $\mathbf{L} = \mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$ *be a null space network and set* $\mathcal{M} := \mathrm{ran}(\mathbf{L})$. *Let* $\mu > 0$, *suppose* $(g_\alpha)_{\alpha>0}$ *satisfies the assumptions of theorem 2.8, and let* $(\mathbf{Q}_\alpha)_{\alpha>0}$ *be a family of bounded operators on* X *with* $\|\mathbf{Q}_\alpha - \mathbf{P}_{\ker(\mathbf{A})}\| = \mathcal{O}(\alpha^\mu)$. *Consider the regularization* $(\mathbf{R}_\alpha)_{\alpha>0}$ *with*

$$\mathbf{R}_\alpha := (\mathrm{Id}_X + \mathbf{Q}_\alpha\mathbf{N}) \circ g_\alpha(\mathbf{A}^*\mathbf{A})\mathbf{A}^*.$$

(3.5)

Then, the parameter choice $\alpha^\star \asymp (\delta/\rho)^{\frac{2}{2\mu+1}}$ *yields the convergence rate results* $\|\mathbf{R}_{\alpha^\star(\delta,y^\delta)}(y^\delta) - x\| = \mathcal{O}(\delta^{\frac{2\mu}{2\mu+1}})$ *for any* $x \in \mathrm{ran}(\mathbf{L}(\mathbf{A}^*\mathbf{A})^\mu)$.

Proof. Follows from the estimate (3.4) with $\mathbf{B}_\alpha = g_\alpha(\mathbf{A}^*\mathbf{A})\mathbf{A}^*$ and theorem 2.8. □

One might use $\mathbf{Q}_\alpha = \mathbf{B}_{\phi(\alpha)}\mathbf{A}$ as a possible approximation to $\mathbf{P}_{\ker(\mathbf{A})^\perp} = \mathbf{A}^+\mathbf{A}$ for some function $\phi\colon [0,\infty) \to [0,\infty)$. In such a situation, one can use existing software packages (for example, for the filtered backprojection algorithm and the discrete Radon transform in case of computed tomography) for evaluating $\mathbf{B}_{\phi(\alpha)}$ and $\mathbf{A}$.

4. Conclusion

In this paper, we introduced the concept of null space networks that have the form $\mathbf{L} = \mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$, where $\mathbf{N}$ is any neural network function (for example a deep convolutional neural network) and $\mathrm{Id}_X - \mathbf{A}^+\mathbf{A} = \mathbf{P}_{\ker(\mathbf{A})}$ is the projector onto the kernel of the forward operator $\mathbf{A}\colon X \to Y$ of the inverse problem to be solved. The null space network shares similarity with a residual network that takes the general form $\mathrm{Id}_X + \mathbf{N}$. However, the introduced projector $\mathrm{Id}_X - \mathbf{A}^+\mathbf{A}$ guarantees data consistency which is an important issue when solving inverse problems.

The null space networks are special members of the class of functions $\mathrm{Id}_X + \boldsymbol{\Phi}$ that satisfy $\mathrm{ran}(\boldsymbol{\Phi}) \subseteq \ker(\mathbf{A})$. For this class, we introduced the concept of $\boldsymbol{\Phi}$-generalized inverse $\mathbf{A}^{\boldsymbol{\Phi}}$ and $\boldsymbol{\Phi}$-regularization as point-wise approximations of $\mathbf{A}^{\boldsymbol{\Phi}}$ on $\mathrm{dom}(\mathbf{A}^+)$. We showed that any classical regularization $(\mathbf{B}_\alpha)_{\alpha>0}$ of the Moore–Penrose generalized inverse defines a $\boldsymbol{\Phi}$-regularization method via $(\mathrm{Id}_X + \boldsymbol{\Phi})\mathbf{B}_\alpha$. In the case of null space networks where $\boldsymbol{\Phi} = (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$, we additionally derived convergence results using only approximation of the projection operator $\mathbf{P}_{\ker(\mathbf{A})}$. Additionally, we derived convergence rates using either exact or approximate projections.

To the best of our knowledge, the obtained convergence and convergence rates are the first regularization results for solving inverse problems with neural networks. Future work has to be done to numerically test the null space networks for typical inverse problems such as limited data problems in CT or deconvolution and compare the performance with standard residual networks, iterative networks or variational networks.

Acknowledgment

Appendix. Possible network training

We may train the null space network $\mathbf{L} = \mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$ to (approximately) map elements to the desired class of training phantoms. For that purpose, fix the following:

- $(x_1, \ldots, x_N)$ is a class of training phantoms;
- For all ℓ fix the nonlinearity $\sigma_\ell \colon X_\ell \to X_\ell$;
- $\mathcal{W}_\ell$ are finite-dimensional spaces of affine continuous mappings;
- $\mathcal{N}$ is the set of all NN functions of the form (3.1) with $\mathbf{W}_\ell \in \mathcal{W}_\ell$.

We then consider null space network $\mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N}$ where $\mathbf{N} \in \mathcal{N}$. To train the null space networks we propose to minimize the regularized error functional $E \colon \mathcal{N} \to \mathbb{R}$ defined by

$$E(\mathbf{N}) := \frac{1}{2} \sum_{n=1}^{N} \left\| x_n - (\mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{A}^+\mathbf{A})\mathbf{N})(\mathbf{A}^+\mathbf{A}x_n) \right\|^2 + \mu \sum_{\ell=1}^{L} \|\mathbf{L}_\ell\| \quad (A.1)$$

where $\mathbf{N}$ is of the form (3.1) and $\mathbf{L}_\ell$ is the linear part of $\mathbf{W}_\ell$ and μ is a regularization parameter.

Network training aims at making $E(\mathbf{N})$ small, for example, by gradient descent. Clearly $\prod_{\ell=1}^{L} \|\mathbf{L}_\ell\|$ is an upper bound on the Lipschitz constant of $\mathbf{N}$. By the inequality of arithmetic and geometric means, the sum $\sum_{\ell=1}^{L} \|\mathbf{L}_\ell\|$ essentially bounds the product $\prod_{\ell=1}^{L} \|\mathbf{L}_\ell\|$. Therefore, the Lipschitz constant of the finally trained network will stay reasonably small. Alternatively, one can directly use the product of the Lipschitz constants $\prod_{\ell=1}^{L} \|\mathbf{L}_\ell\|$ as regularization term in (A.1), but using the sum of the Lipschitz constants seems more in line with standard practice.

Note that it is not required that (A.1) is exactly minimized. Any trained network where $\frac{1}{2} \sum_{\ell=1}^{L} \|x_n - (\mathrm{Id}_X + \mathbf{A}^+\mathbf{A}\mathbf{N})(\mathbf{A}^+\mathbf{A}x_n)\|^2$ is small yields a null space network $\mathrm{Id}_X + \mathbf{A}^+\mathbf{A}\mathbf{N}$ that does, at least on the training set, a better job in estimating x_n from $\mathbf{A}^+\mathbf{A}x_n$ than the identity.

Alternatively, we may train a regularized null space network $\mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{B}_\alpha\mathbf{A})\mathbf{N}$ to map the regularized data $\mathbf{B}_\alpha\mathbf{A}x_n$ (instead of $\mathbf{A}^+\mathbf{A}x_n$) to the outputs x_n. This yields the modified error functional

$$E_\alpha(\mathbf{N}) := \frac{1}{2} \sum_{n=1}^{N} \left\| x_n - (\mathrm{Id}_X + (\mathrm{Id}_X - \mathbf{B}_\alpha\mathbf{A})\mathbf{N})(\mathbf{B}_\alpha x_n) \right\|^2 + \mu \sum_{\ell=1}^{L} \|\mathbf{L}_\ell\|. \quad (A.2)$$

Trying to minimize E_α may be beneficial in the case that many singular values are small but do not vanish exactly. The regularized version $\mathbf{B}_\alpha$ might be defined by truncated SVD or Tikhonov regularization.

ORCID iDs

Stephan Antholzer ◉ https://orcid.org/0000-0002-4923-8146
Markus Haltmeier ◉ https://orcid.org/0000-0001-5715-0331

References

[1] Adler J and Lunz S 2018 Banach Wasserstein GAN (arXiv:1806.06621)
[2] Adler J and Öktem O 2017 Solving ill-posed inverse problems using iterative deep neural networks *Inverse Problems* **33** 124007
[3] Antholzer S, Haltmeier M and Schwab J 2018 Deep learning for photoacoustic tomography from sparse data *Inverse Problems Sci. Eng.* in press
[4] Benning M and Burger M 2018 Modern regularization methods for inverse problems *Acta Numer.* **27** 1–111
[5] Chang J H R, Li C, Poczos B, Kumar V and Sankaranarayanan A C 2017 One network to solve them all—solving linear inverse problems using deep projection models *ICCV* 5889–98
[6] 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
[7] Engl H W, Hanke M and Neubauer A 1996 *Regularization of Inverse Problems* vol 375 (Dordrecht: Kluwer)
[8] Gupta H, Jin K H, Nguyen H Q, McCann M T and Unser M 2018 Cnn-based projected gradient descent for consistent ct image reconstruction *IEEE Trans. Med. Imaging* **37** 1440–53
[9] Guss W H 2016 Deep function machines: generalized neural networks for topological layer expression (arXiv:1612.04799)
[10] 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)
[11] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*
[12] 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
[13] Kang E, Min J and Ye J C 2017 A deep convolutional neural network using directional wavelets for low-dose x-ray CT reconstruction *Med. Phys.* **44** e360–75
[14] Kelly B, Matthews T P and Anastasio M A 2017 Deep learning-guided image reconstruction from incomplete data (arXiv:1709.00584)
[15] Kobler E, Klatzer T, Hammernik K and Pock T 2017 Variational networks: connecting variational methods and deep learning *German Conf. on Pattern Recognition* (Springer)
[16] Lee D, Yoo J and Ye J C 2017 Deep residual learning for compressed sensing mri *IEEE 14th Int. Symp. on Biomedical Imaging* (IEEE)
[17] Li H, Schwab J, Antholzer S and Haltmeier M 2018 NETT: solving inverse problems with deep neural networks (arXiv:1803.00092)
[18] Majumdar A 2015 Real-time dynamic MRI reconstruction using stacked denoising autoencoder (arXiv:1503.06383)
[19] Mardani M *et al* 2017 Deep generative adversarial networks for compressed sensing automates MRI (arXiv:1706.00051)
[20] Mardani M, Monajemi H, Papyan V, Vasanawala S, Donoho D and Pauly J 2017 Recurrent generative adversarial networks for proximal learning and automated compressive image recovery (arXiv:1711.10046)
[21] Rivenson Y, Göröcs Z, Günaydin H, Zhang Y, Wang H and Ozcan A 2017 Deep learning microscopy *Optica* **4** 1437–43
[22] Scherzer O, Grasmair M, Grossauer H, Haltmeier M and Lenzen F 2009 *Variational Methods in Imaging* (*Applied Mathematical Sciences* vol 167) (New York: Springer)
[23] Schlemper J, Caballero J, Hajnal J V, Price A N and Rueckert D 2018 A deep cascade of convolutional neural networks for dynamic mr image reconstruction *IEEE Trans. Med. Imaging* **37** 491–503
[24] Shalev-Shwartz S and Ben-David S 2014 *Understanding Machine Learning: from Theory to Algorithms* (Cambridge: Cambridge University Press)
[25] Wang G 2016 A perspective on deep imaging *IEEE Access* **4** 8914–24
[26] 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*
[27] 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* (Springer)
[28] 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)