Generalization bounds for deep thresholding networks

Arash Behboodi * Holger Rauhut † Ekkehard Schnoor ‡

Abstract

We consider compressive sensing in the scenario where the sparsity basis (dictionary) is not known in advance, but needs to be learned from examples. Motivated by the well-known iterative soft thresholding algorithm for the reconstruction, we define deep networks parametrized by the dictionary, which we call deep thresholding networks. Based on training samples, we aim at learning the optimal sparsifying dictionary and thereby the optimal network that reconstructs signals from their low-dimensional linear measurements. The dictionary learning is performed via minimizing the empirical risk. We derive generalization bounds by analyzing the Rademacher complexity of hypothesis classes consisting of such deep networks. We obtain estimates of the sample complexity that depend only linearly on the dimensions and on the depth.

1 Introduction

Learning representations of or extracting features from data is an important aspect of deep neural networks. In the past decade, this approach has led to impressive results and achieved state-of-the-art performances, e.g., for various classification tasks. However, due to the black-box nature of the end-to-end learning of neural networks, such features are usually abstract and difficult to interpret. On the other hand, it has turned out that algorithms such as iterative soft-thresholding (ISTA) can be regarded as neural networks. Thus, with the help of modern deep learning software libraries, they can easily be implemented and optimized, such that the trained parameters can adapt to data sets of interest. When such algorithms are well understood, it can be possible to transfer results shown for the classical variant to their neural network variant and in this way increase our understanding of deep neural networks. One variant of this is the DISTA, which we propose in the present paper, where we consider a joint reconstruction and dictionary learning problem. Here, the learned representation (a dictionary) is a very well-understood model in image and signal processing, which can be easily interpreted and visualized. As a practical application, one may think of reconstructing images from measurements taken by a medical imaging device. Instead of only trying to reconstruct the image, we would like to implicitly learn also a meaningful representation system which is adapted to the image class of interest, and leads to good generalization (e.g., when taking measurements of new patients). More generally, this is the approach of solving inverse problems in a data-driven way, e.g. by training neural networks [3, 11].

One of the mysteries of deep neural networks is why in practice they generalize so well, despite often being overparameterized, i.e., the number of trainable weights being larger than the sample size [28, 30]. Various techniques have been tried to answer this question, such as implicit bias [31], a compression approach [2], and a PAC-Bayesian approach [29]. However, the possible explanations for the generalization of deep neural networks remain unsatisfactory [27], and the search for good generalization measures is still ongoing [16].

*Institute for Theoretical Information Technology, RWTH Aachen University, Germany
†Chair for Mathematics of Information Processing, RWTH Aachen University, Germany
‡Chair for Mathematics of Information Processing, RWTH Aachen University, Germany

While so far generalization of neural networks has been studied mostly in the context of classification using feed-forward neural networks, the case studied here has received less attention so far from the perspective of generalization. Concretely, our reconstruction problem is a regression problem, and the network is in fact a recurrent neural network, which is known to be difficult to train [32]. Due to the weight sharing, this is a non-overparameterized network; however, it is straightforward to decouple the layers and thus obtain a network which is more similar to standard feed-forward neural networks. Furthermore, we impose an orthogonality constraint on the dictionary, which consists of the learned parameters of the network. We derive generalization bounds for such thresholding networks with orthogonal dictionaries by estimating Dudley's integral (and in particular the covering numbers involved) to upper bound the Rademacher complexity of hypothesis classes consisting of such deep networks. Since the problem is essentially a regression problem, we use a generalization of Talagrands contraction principle [22] for vector-valued functions, which is typically not needed when considering real-valued hypothesis classes, e.g. with the ramp loss (applied to the margin) in a multiclass classification problem [4].

Furthermore, we discuss how imposing structure (such as sparsity) might improve sample complexity bounds. We believe that the techniques presented are of independent interest far beyond the particular problem studied here, e.g., for a theoretical investigation of related iterative schemes, general regression problems using neural networks, and in particular autoencoders and recurrent neural networks.

The paper is structured as follows. In section 2 we precisely introduce DISTA, and formulate it as a machine learning problem. Section 3 is the main section of this paper, where we derive a bound for the generalization error. The proof of the main result is sketched in this section, while detailed proofs of all necessary results are given later in the appendix. Furthermore, we discuss consequences and directions for possible future work. Finally, in section 4 we present the results of our numerical experiments and compare it with our theoretical findings in the section before.

2 Joint Learning of Dictionary and Decoder

2.1 Main definitions and formulation of the problem

We consider the class of signals $\mathbf{x} \in \mathbb{R}^N$ which are sparsely representable with respect to a dictionary $\mathbf{\Psi} \in \mathbb{R}^{N\times N}$. In other words, for each $\mathbf{x}$ there is a sparse vector $\mathbf{z} \in \mathbb{R}^N$ such that $\mathbf{x} = \mathbf{\Psi}\mathbf{z}$. The dictionary $\mathbf{\Psi}$ is assumed to be unknown. We are given a linear observation $\mathbf{y} = \mathbf{A}\mathbf{x} \in \mathbb{R}^n$ where $\mathbf{A} \in \mathbb{R}^{n\times N}$ is a known measurement matrix. We would like to learn a dictionary suitable for decoding purpose from a training sequence $\mathcal{S} := ((\mathbf{x}_i, \mathbf{y}_i))_{i=1,\dots,m}$ with i.i.d. samples drawn from a distribution $\mathcal{D}$. (Formally, this is a distribution over the $\mathbf{x}_i$, and then the corresponding measurements $\mathbf{y}_i$ are given by $\mathbf{y}_i = \mathbf{A}\mathbf{x}_i$, with $\mathbf{A}$ being deterministic.) The decoder is based on the unfolded version of the iterative soft thresholding algorithm (ISTA) with L iterations as follows. The first layer is defined by $f_1(\mathbf{y}) := S_{\tau\lambda}(\tau(\mathbf{A}\mathbf{\Phi})^\top \mathbf{y})$, where $S_\lambda : \mathbb{R} \to \mathbb{R}$ (applied entry-wise) is the shrinkage operator defined as

$$S_\lambda(x) = \begin{cases} 0 & \text{if } |x| \le \lambda, \\ x - \lambda \operatorname{sign}(x) & \text{if } |x| > \lambda. \end{cases} \tag{2.1}$$

which can also be expressed in closed form as $S_\lambda(x) = \operatorname{sign}(x) \cdot \max(0, |x| - \lambda)$. For $l > 1$, the output is given by

$$f_l(\mathbf{z}) := S_{\tau\lambda}\left[\mathbf{z} + \tau(\mathbf{A}\mathbf{\Phi})^\top(\mathbf{y} - (\mathbf{A}\mathbf{\Phi})\mathbf{z})\right] = S_{\tau\lambda}\left[\left(\mathbf{I} - \tau\mathbf{\Phi}^\top\mathbf{A}^\top\mathbf{A}\mathbf{\Phi}\right)\mathbf{z} + \tau(\mathbf{A}\mathbf{\Phi})^\top\mathbf{y}\right], \tag{2.2}$$

which can be interpreted as a layer of a neural network with weight matrix $\mathbf{I} - \tau\mathbf{\Phi}^\top\mathbf{A}^\top\mathbf{A}\mathbf{\Phi}$, bias $\tau(\mathbf{A}\mathbf{\Phi})^\top\mathbf{y}$ and activation function $S_{\tau\lambda}$. Then the decoder is a neural network with shared

weights $\mathbf{\Phi}$ in every layer and given by

$$f_{\mathbf{\Phi}}^L(\mathbf{y}) = \sigma(\mathbf{\Phi} f_L \circ f_{L-1} \cdots \circ f_1(\mathbf{y})), \tag{2.3}$$

where the last activation function σ is a 1-Lipschitz thresholding function such that $\|\sigma(\mathbf{z})\|_2$ is bounded.[1] The hypothesis set consists of of all functions that can be expressed as a L-step soft thresholding, where the dictionary matrix $\mathbf{\Phi}$ parameterizes the hypothesis class

$$\mathcal{H}^L := \{f_{\mathbf{\Phi}}^L : \mathbb{R}^n \to \mathbb{R}^N : f_{\mathbf{\Phi}}^L(x) = \sigma(\mathbf{\Phi} f_L \circ f_{L-1} \cdots \circ f_1(\mathbf{y})), \mathbf{\Phi} \in O(N)\}.$$

Here, we assume that $\mathbf{\Phi}$ ranges over the orthogonal group $O(N)$, which is a typical assumption in dictionary learning; however, different choices of (bounded) parameter sets can be considered as well. The parameters $\tau, \lambda > 0$ will be fixed in the following. Based on the training sequence $\mathcal{S}$ and given the hypothesis space $\mathcal{H}^L$, a learning algorithm yields a function $h_{\mathcal{S}} \in \mathcal{H}^L$ that aims at reconstructing $\mathbf{x}$ from the measurements $\mathbf{Ax}$. The empirical loss is the reconstruction error on the training sequence, i.e., the difference between $\mathbf{x}_i$ and $\hat{\mathbf{x}}_i = h_{\mathcal{S}}(\mathbf{y}_i)$, i.e.

$$\hat{\mathcal{L}}(h) = \frac{1}{m} \sum_{j=1}^{m} \ell(h, \mathbf{x}_j, \mathbf{y}_j).$$

The loss can be chosen, for example, as $\|h(\mathbf{y}_j) - \mathbf{x}_j\|_2$, i.e. measuring the reconstruction error with respect to the ℓ_2-norm. The true loss, i.e., the risk of a hypothesis h is accordingly defined as

$$\mathcal{L}(h) := \mathbb{E}_{\mathbf{x},\mathbf{y}\sim\mathcal{D}} \left(\ell(h, \mathbf{x}, \mathbf{y})\right).$$

The generalization error is defined as the difference between the empirical loss and the true loss,

$$\mathrm{GE}(h_{\mathcal{S}}) := \left|\hat{\mathcal{L}}(h_{\mathcal{S}}) - \mathcal{L}(h_{\mathcal{S}})\right|.$$

(Note that some reference refer to the true loss $\mathcal{L}(h_{\mathcal{S}})$ as the generalization error. However, the above definition is more convenient for our purposes.) We use a Rademacher complexity analysis to bound the generalization error in the next section.

2.2 Related Work

Compressive sensing using dictionaries has been studied before, but, in contrast to the scenario discussed here, typically using a fixed (and possibly even redundant) dictionary and a random measurement matrix [33]. The idea of interpreting gradient-steps of iterative algorithms such as ISTA [7] for sparse recovery as layers of neural networks is well-known since [12] and has since then been an active research topic, e.g., [5, 19, 24, 26, 39, 40]. Thresholding networks fall into the larger class of proximal neural networks studied in [15]. The central problem of sparse coding is to learn weight matrices for an unfolded version of ISTA. Different works focus on different parametrization of the network for faster convergence and better reconstructions. Learning the dictionary can also be implicit in these works. In the present paper, we consider algorithms that try to find a dictionary suitable for reconstruction. Some of the examples of these algorithms are recently suggested Ada-LISTA [1], convolutional sparse coding [37] learning efficient sparse and low-rank models [36]. Like many other related papers, such as ISTA-Net [41], these methods are mainly motivated by applications like impainting [1]. Instead of novel algorithmic aspects, our contribution is to conduct a generalization analysis for these algorithms, which to the best of our knowledge has not been addressed in the literature before in this particular setting. In this way, we connect this line of research with recent developments

[1] One may think of a function that shrinks vectors if their norm is beyond a certain thresholding, but the exact nature of this function is not important for the rest. It is introduced for technical reasons that will be apparent during our proofs.

[4, 10] in the study of generalization of deep neural networks. Particularly, we use a similar framework to [4] by bounding the Rademacher complexity using Dudley's integral. However, the approach of [4] applies only to classes of real-valued neural networks, as typically met in classification problems. The extension to our problem, which is a regression problem with vector-valued functions, involves additional technicalities requiring a generalized contraction principle for hypothesis classes of vector-valued functions. Besides, we show linear dimension dependence, using techniques that are different from the ones in [10]. It is not straightforward to extend the result of [10] to our case because of the weight sharing between different layers of thresholding networks. As already pointed out above, the deep thresholding network we analyse is, due to the weight sharing, a recurrent neural network. The authors of [6] derive VC-dimension of recurrent networks for recurrent perceptrons with binary outputs. The VC-dimension of recurrent neural networks for different classes of activation functions has been studied in [20]. However, their results do not immediately apply to our setup, since they focus on one-dimensional inputs and outputs, which of course does not suit our vector-valued regression problem, and moreover, would correspond to taking just one single measurement. Even in the scenario which is closest to ours, namely fixed piecewise polynomial activation functions with $n = 1$, their bound scales between $\mathcal{O}(Lw)$ and $\mathcal{O}(Lw^2)$, where L is the number of layers and w is the number of trainable parameters in the network. In our case, the number of trainable parameters are equal to the dimension of the orthogonal group $O(N)$, which is $N(N-1)/2$. Therefore, their bounds scale between $\mathcal{O}(LN^2)$ and $\mathcal{O}(LN^4)$. In contrast, if $n = 1$, our bound scales only like $\mathcal{O}(LN)$. Besides, we only make use of Lipschitzness of the activation function. Sample complexity of dictionary learning has been studied before in the literature [9, 13, 14, 34, 38]. The authors in [38] also use a Rademacher complexity analysis for dictionary learning, but they aim at sparse representation of signals rather than reconstruction from compressed measurements and moreover, they do not use neural network structures. Fundamental limits of dictionary learning from an information-theoretic perspective has been studied in [17, 18]. Unique about our perspective and different to the cited papers is our approach for determining the sample complexity based on learning a dictionary implicitly by training a neural network.

2.3 Notation

Before we continue with the main part of the paper, let us fix some notation. Vectors $\mathbf{v} \in \mathbb{R}^n$ and matrices $\mathbf{A} \in \mathbb{R}^{m\times N}$ are denoted with bold letters, unlike scalars $\lambda \in \mathbb{R}$. We will denote the spectral norm by $\|\mathbf{A}\|_{2\to 2}$ and the Frobenius norm by $\|\mathbf{A}\|_F$. The $N \times m$ matrix $\boldsymbol{X}$ contains the data points, $\mathbf{x}_1, \dots, \mathbf{x}_m \in \mathbb{R}^N$, as its columns. As a short notation for indices we use $[m] := \{1, \dots, m\}$, e.g. $(\mathbf{x}_i)_{i\in[m]} = (\mathbf{x}_1, \dots, \mathbf{x}_m)$. Analogously $\boldsymbol{Y} \in \mathbb{R}^{n\times N}$ denotes the matrix collecting the measurements $\mathbf{y}_1, \dots, \mathbf{y}_m \in \mathbb{R}^n$. To make the notation more compact, with a slight abuse of notation, for $f^L_{\boldsymbol{\Phi}} \in \mathcal{H}^L$, we denote by $f^L_{\boldsymbol{\Phi}}(\boldsymbol{Y})$ the matrix whose i-th column is $f^L_{\boldsymbol{\Phi}}(\mathbf{y}_i)$. The unit ball of an n-dimensional normed space $\mathcal{V}$ is denoted by $B^n_{\|\cdot\|} := \{\mathbf{x} \in \mathcal{V} : \|\mathbf{x}\| \le 1\}$. Covering numbers of a metric space $(\mathcal{M}, d)$ at level ε will be denoted by $\mathcal{N}(\mathcal{M}, d, \varepsilon)$. When we consider subsets of normed spaces where the metric is induced by the norm, we write $\mathcal{N}(\mathcal{M}, \|\cdot\|, \varepsilon)$. Furthermore, we have already introduced the hypothesis space $\mathcal{H}^L$ in (2.4). Instead, we write $\mathcal{H}$ if we refer to a general hypothesis space, e.g. when quoting general results from the machine learning literature.

3 Rademacher Complexity Bounds for Deep Thresholding Networks

In order to bound the generalization error we use the Rademacher complexity. For a class $\mathcal{G}$ of functions $g : Z \to \mathbb{R}$ and a sample $\mathcal{S} = (z_1, \dots, z_m)$ the empirical Rademacher complexity is

defined as

$$\mathcal{R}_{\mathcal{S}}(\mathcal{G}) := \mathbb{E}_{\varepsilon} \sup_{g\in\mathcal{G}} \frac{1}{m} \sum_{i=1}^{m} \varepsilon_i g(z_i), \tag{3.1}$$

where ε is a Rademacher vector, i.e., a vector with independent Rademacher variables ε_i, $i = 1, \ldots, m$, taking the value ± 1 with equal probability. The Rademacher complexity is then given as $\mathcal{R}_m(\mathcal{G}) = \mathbb{E}_{\mathcal{S}\sim\mathcal{D}^m}\mathcal{R}_{\mathcal{S}}(\mathcal{G})$, but note that we will exclusively work with the empirical Rademacher complexity. Given a loss function ℓ and a hypothesis class $\mathcal{H}$, one usually considers the Rademacher complexity of the class $\mathcal{G} = \ell \circ \mathcal{H} = \{g((\mathbf{x}, \mathbf{y})) = \ell(h, \mathbf{x}, \mathbf{y}) : h \in \mathcal{H}\}$. We rely on the following theorem which bounds the generalization error in terms of the empirical Rademacher complexity.

Theorem 3.1 ([35, Theorem 26.5]). *Let $\mathcal{H}$ be a family of functions, and let $\mathcal{S}$ be training set $\mathcal{S}$ drawn from $\mathcal{D}^m$. Let ℓ be areal-valued loss function satisfying $|\ell| \leq c$. Then, for $\delta \in (0, 1)$, with probability at least $1 - \delta$ we have, for all $h \in \mathcal{H}$,*

$$\mathcal{L}(h) \leq \hat{\mathcal{L}}(h) + 2\mathcal{R}_{\mathcal{S}}(\ell \circ \mathcal{H}) + 4c\sqrt{\frac{2\log(4/\delta)}{m}}. \tag{3.2}$$

To use the above theorem, the loss function needs to be bounded. We make two main assumptions. Firstly, we assume that the input is bounded in the ℓ_2-norm by $\|\mathbf{x}\|_2 \leq B_{\text{in}}$. Secondly, for the output, we assume that

$$\left\| f_{\mathbf{\Phi}}^{L}(\mathbf{y}) \right\|_2 \leq B_{\text{out}}.$$

Boundedness of the last activation function σ ensures existence of such a constant B_{out}. Moreover, we will show in Lemma A.1 that in the case of bounded inputs $\mathbf{x}$ bounded outputs can even be guaranteed without the boundedness assumption on σ, although the bound we give in Lemma A.1 may be improvable in concrete situations. Under the above assumptions, the loss function $\ell(\cdot)$, chosen as the ℓ_2-distance between the input and the reconstruction, is bounded bounded as

$$\ell(h, \mathbf{y}, \mathbf{x}) = \|h(\mathbf{y}) - \mathbf{x}\|_2 \leq \|\mathbf{x}\|_2 + \|h(\mathbf{y})\|_2 \leq B_{\text{in}} + B_{\text{out}}.$$

The main challenge and focus of the rest of this section is to bound Rademacher complexity of $\ell \circ \mathcal{H}^L$,

$$\mathcal{R}_m(\ell \circ \mathcal{H}^L) = \mathbb{E} \sup_{h\in\mathcal{H}^L} \frac{1}{m} \sum_{i=1}^{m} \varepsilon_i \left\| \mathbf{x}_i - h(\mathbf{y}_i) \right\|_2 .$$

Often, e.g., in multiclass classification problems using the margin loss [4], the function $h(\cdot)$ is real-valued. We can use the classical contraction principle by Talagrand [22] to directly bound the Rademacher complexity of the hypothesis space. However, in our case the function $h(\cdot)$ is vector-valued, and the contraction lemma cease to hold. However, since the norm is 1-Lipschitz, we can use the following generalization of contraction principle for Rademacher complexities of vector-valued hypothesis classes.

Lemma 3.2 ([25, Corollary 4]). *Let $\mathcal{S} = (\mathbf{x}_i)_{i\in[m]}$ be the training sequence. Suppose that the function $h \in \mathcal{H}$ maps $\mathcal{X}$ to $\mathbb{R}^N$, and the function f is K-Lipschitz from $\mathbb{R}^N$ to $\mathbb{R}$. Then*

$$\mathbb{E} \sup_{h\in\mathcal{H}} \sum_{i=1}^{m} \varepsilon_i f \circ h(\mathbf{x}_i) \leq \sqrt{2}K\mathbb{E} \sup_{h\in\mathcal{H}} \sum_{i=1}^{m} \sum_{k=1}^{N} \varepsilon_{ik} h_k(\mathbf{x}_i). \tag{3.3}$$

The ℓ_2-norm appearing in the loss function is 1-Lipschitz. Therefore according to Lemma 3.2, it is enough to bound the following doubly indexed Rademacher complexity

$$\mathcal{R}_{\mathcal{S}}(\ell \circ \mathcal{H}^L) \leq \sqrt{2}\mathcal{R}_{\mathcal{S}}^{(2)}(\mathcal{H}^L) := \sqrt{2}\mathbb{E} \sup_{h\in\mathcal{H}^L} \frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{N} \varepsilon_{ik} h_k(\mathbf{x}_i).$$

3.1 Dudley's Inequality

We use the following version of Dudley's inequality [8, Theorem 8.23], with slightly better constants than [4]. To state the theorem, we require additional definitions. Consider a stochastic process $(X_t)_{t\in\mathcal{T}}$ with the index set $\mathcal{T}$ in a space with pseudometric d given by

$$d(s,t) := \left(\mathbb{E}\left|X_s - X_t\right|^2\right)^{1/2}.$$

A zero-mean process X_t for $t \in \mathcal{T}$ is subgaussian if

$$\mathbb{E}\exp(\theta(X_s - X_t)) \leq \exp\left(\theta^2 d(s,t)^2/2\right) \quad \forall\, s,t \in \mathcal{T},\, \theta > 0.$$

Finally, define the radius of $\mathcal{T}$ as $\Delta(\mathcal{T}) := \sup_{t\in\mathcal{T}} \sqrt{\mathbb{E}|X_t|^2}$. Dudley's inequality is stated as follows.

Theorem 3.3 (Dudley's inequality)**.** *Let $(X_t)_{t\in\mathcal{T}}$ be a centered (i.e. $\mathbb{E}X_t = 0$ for every $t \in \mathcal{T}$) subgaussian process with radius $\Delta(\mathcal{T})$. Then*

$$\mathbb{E}\sup_{t\in\mathcal{T}} X_t \leq 4\sqrt{2}\int_0^{\Delta(\mathcal{T})/2} \sqrt{\log \mathcal{N}(\mathcal{T}, d, u)}\mathrm{d}u. \tag{3.4}$$

We use this inequality to bound the Rademacher complexity term.

3.2 Bounding the Rademacher Complexity

For fixed number of layers $L \in \mathbb{N}$, define the set $\mathcal{M} \subset \mathbb{R}^{N\times m}$ as

$$\mathcal{M} := \left\{(h(\mathbf{x}_1)|\ldots|h(\mathbf{x}_m)) \in \mathbb{R}^{N\times m} : h \in \mathcal{H}^L\right\} = \left\{f_{\mathbf{\Phi}}^L(\boldsymbol{Y}) \in \mathbb{R}^{N\times m} : f_{\mathbf{\Phi}}^L \in \mathcal{H}^L\right\}. \tag{3.5}$$

Note that $\mathcal{M}$ is parameterized by $\mathbf{\Phi} \in \mathbb{R}^{N\times N}$ (as $\mathcal{H}^L$ is), such that we can rewrite (3.4) as

$$\mathcal{R}_{\mathcal{S}}^{(2)}(\mathcal{H}^L) = \mathbb{E}\sup_{\mathbf{M}\in\mathcal{M}} \frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{N} \varepsilon_{ik}M_{ik}. \tag{3.6}$$

We use Dudley's inequality and a covering number argument to bound the Rademacher complexity term The Rademacher process defined in (3.6) is a subgaussian process, and therefore, we can apply Dudley's inequality. For the set of matrices $\mathcal{M}$ defined in (3.5), the radius can be estimated as

$$\begin{aligned}
\Delta(\mathcal{M}) &= \sup_{h\in\mathcal{H}^L} \sqrt{\mathbb{E}\left(\sum_{i=1}^{m}\sum_{k=1}^{N} \varepsilon_{ik}h_k(\mathbf{x}_i)\right)^2} \\
&\leq \sup_{h\in\mathcal{H}^L} \sqrt{\sum_{i=1}^{m}\sum_{k=1}^{N} h_k(\mathbf{x}_i)^2} \\
&\leq \sup_{h\in\mathcal{H}^L} \sqrt{\sum_{i=1}^{m} \|h(\mathbf{x}_i)\|_2^2} \\
&\leq \sqrt{m}B_{\text{out}}.
\end{aligned}$$

Plugging this bound in Dudley's inequality, we obtain (3.7).

$$\mathcal{R}_{\mathcal{S}}^{(2)}(\mathcal{H}^L) \leq \frac{4\sqrt{2}}{m}\int_0^{\sqrt{m}B_{\text{out}}/2} \sqrt{\log \mathcal{N}(\mathcal{M}, \|\cdot\|_F, \varepsilon)}\mathrm{d}\varepsilon.$$

To derive the generalization bound, it suffices to bound the covering number of $\mathcal{M}$. We relate the covering number of $\mathcal{M}$ to the covering number of the parameter space $\mathbf{\Phi}$. The following theorem establishes this connection via a perturbation analysis argument.

Theorem 3.4. *Consider the thresholding networks $f^L_{\mathbf{\Phi}}$ defined as in (2.3) with $L \geq 2$ and dictionary $\mathbf{\Phi}$ in $O(N)$. Then, for any $\mathbf{\Phi}_1, \mathbf{\Phi}_2 \in O(N)$ we have*

$$\|f^L_{\mathbf{\Phi}_1}(\boldsymbol{Y}) - f^L_{\mathbf{\Phi}_2}(\boldsymbol{Y})\|_F \leq K_L \|\mathbf{A}\mathbf{\Phi}_1 - \mathbf{A}\mathbf{\Phi}_2\|_{2\to 2} \tag{3.7}$$

where K_L is given by

$$K_L = \tau\|\boldsymbol{Y}\|_F \left[1 + \sum_{l=2}^{L} (1 + \tau\|\mathbf{A}\|^2_{2\to 2})^{L-l} + \left(1 + 2\tau\|\mathbf{A}\|^2_{2\to 2} \sum_{k=0}^{l-2} \|\mathbf{I} - \tau\mathbf{A}^T\mathbf{A}\|^k_{2\to 2}\right)\right]. \tag{3.8}$$

If $\tau\|\mathbf{A}\|^2_{2\to 2} \leq 1$, it has the simplified upper bound

$$K_L \leq \tau\|\boldsymbol{Y}\|_F(1 + 6 \cdot 2^L) \leq 7\tau\|\boldsymbol{Y}\|_F 2^L. \tag{3.9}$$

The proof is provided in the supplementary material in Appendix A. In particular, the proof of (3.9) is based on the following observation. Because it will be useful sometimes in the sequel, we already state it at this point.

Remark 3.5. The $N \times N$ - matrix $\mathbf{A}^\top\mathbf{A}$ is rank deficient in the compressive sensing setup ($n < N$). If in addition $\tau \|\mathbf{A}\|^2_{2\to 2} \leq 1$ then $\left\|\mathbf{I} - \tau\mathbf{A}^\top\mathbf{A}\right\|_{2\to 2} = 1$.

Before we state the main result, we need the following covering number estimate, which, together with Theorem 3.4, will give us an estimate of the covering number $\mathcal{N}(\mathcal{M}, \|\cdot\|_F, \varepsilon)$ appearing in Dudley's integral (3.7).

Lemma 3.6 (Covering numbers of $\mathbf{A}$ applied to the orthogonal group)**.** *For a fixed matrix $\mathbf{A} \in \mathbb{R}^{n\times N}$ consider*

$$\mathcal{W} := \{\mathbf{A}\mathbf{\Phi} : \mathbf{\Phi} \in O(N)\} \subset \mathbb{R}^{n\times N},$$

i.e., $\mathbf{A}$ applied to the orthogonal group. The covering number estimate is given by

$$\mathcal{N}(\mathcal{W}, \|\cdot\|_{2\to 2}, \varepsilon) \leq \left(1 + \frac{2\|\mathbf{A}\|_{2\to 2}}{\varepsilon}\right)^{nN}.$$

Proof. The following lemma is a standard result for covering number estimates which can be found in various sources; as a reference, see [8, Proposition C.3].

Lemma 3.7. *Let $\varepsilon > 0$ and let $\|\cdot\|$ be a norm on a n-dimensional vector space V. Then, for any subset $U \subseteq B_{\|\cdot\|} := \{x \in V : \|x\| \leq 1\}$ it holds*

$$\mathcal{N}(U, \|\cdot\|, \varepsilon) \leq \left(1 + \frac{2}{\varepsilon}\right)^n.$$

Consequently, we obtain the following covering number estimate for the orthogonal group $O(N)$.

Corollary 3.8 (Covering numbers of the orthogonal group)**.** For the covering numbers of the orthogonal group $(O(N), \|\cdot\|_{2\to 2})$ equipped with the spectral norm we have

$$\mathcal{N}(O(N), \|\cdot\|_{2\to 2}, \varepsilon) \leq \left(1 + \frac{2}{\varepsilon}\right)^{N^2}.$$

Proof. The orthogonal group $O(N)$ is contained in $B^{N\times N}_{\|\cdot\|_{2\to 2}}$ and therefore Lemma 3.7 applies. □

Finally, we can prove the covering number estimate from Lemma 3.6, which we use to prove our main result. Note that considering $\mathbf{A}$ applied to the orthogonal group $O(N)$ instead of $O(N)$ itself leads to a much better better dimension dependence with nN instead of N^2 in the exponent (recall, that $n < N$ or even $n \ll N$ in the compressive sensing setting).

Using that for any $\mathbf{\Phi} \in O(N)$ the matrix $\mathbf{A\Phi}/\|\mathbf{A}\|_{2\to 2}$ is contained in the unit ball of $\mathbb{R}^{n\times N}$ with respect to the spectral norm, Lemma 3.7 gives

$$\begin{aligned}\mathcal{N}(\mathcal{W}, \|\cdot\|_{2\to 2}, \varepsilon) =& \mathcal{N}(\{\mathbf{A\Phi} : \mathbf{\Phi} \in O(N)\}, \|\cdot\|_{2\to 2}, \varepsilon) \\ =& \mathcal{N}(\{\mathbf{A\Phi}/\|\mathbf{A}\|_{2\to 2} : \mathbf{\Phi} \in O(N)\}, \|\cdot\|_{2\to 2}, \varepsilon/\|\mathbf{A}\|_{2\to 2}) \\ \leq& \left(1 + \frac{2\|\mathbf{A}\|_{2\to 2}}{\varepsilon}\right)^{nN},\end{aligned}$$

which proves the lemma. □

Note that considering the set $\mathcal{W}$ instead of the orthogonal group $O(N)$ itself results in a better dimension dependence and takes n, the number of measurements, into account. Now we are ready to state and prove our main result.

Theorem 3.9. *Consider the hypothesis space $\mathcal{H}^L$ defined in 2.4. With probability at least $1-\delta$, the generalization error for any $f^L_{\mathbf{\Phi}}$ is bounded as*

$$\mathcal{L}(f^L_{\mathbf{\Phi}}) \leq \hat{\mathcal{L}}(f^L_{\mathbf{\Phi}}) + 8B_{out}\sqrt{\frac{Nn}{m}\log e\left(1 + \frac{4K_L\|\mathbf{A}\|_{2\to 2}}{\sqrt{m}B_{out}}\right)} + 4(B_{in} + B_{out})\sqrt{\frac{2\log(4/\delta)}{m}},$$

where K_L is the perturbation bound in (3.8).

Proof. Using Lemma 3.6, the covering numbers of $\mathcal{M}$ are bounded by

$$\begin{aligned}\mathcal{N}(\mathcal{M}, \|\cdot\|_F, \varepsilon) \leq& \mathcal{N}(K_L\{\mathbf{A\Phi} : \mathbf{\Phi} \in O(N)\}, \|\cdot\|_{2\to 2}, \varepsilon) \\ =& \mathcal{N}(\{\mathbf{A\Phi} : \mathbf{\Phi} \in O(N)\}, \|\cdot\|_{2\to 2}, \varepsilon/K_L) \\ \leq& \left(1 + \frac{2\|\mathbf{A}\|_{2\to 2}K_L}{\varepsilon}\right)^{nN}.\end{aligned}$$

If we plug this into Dudley's integral, we obtain

$$\begin{aligned}\mathbb{E}\sup_{\mathbf{M}\in\mathcal{M}}\frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{N}\varepsilon_{ik}M_{ik} \leq& \frac{4\sqrt{2}}{m}\int_0^{\sqrt{m}B_{\text{out}}/2}\sqrt{\log\mathcal{N}(\mathcal{M}, \|\cdot\|_F, \varepsilon)}\mathrm{d}\varepsilon \\ \leq& \frac{4\sqrt{2}}{m}\int_0^{\sqrt{m}B_{\text{out}}/2}\sqrt{nN}\sqrt{\log\left(1 + \frac{2\|\mathbf{A}\|_{2\to 2}K_L}{\varepsilon}\right)}\mathrm{d}\varepsilon \\ \leq& \frac{4\sqrt{2nN}}{m}\frac{\sqrt{m}B_{\text{out}}}{2}\sqrt{\log e\left(1 + \frac{2K_L\|\mathbf{A}\|_{2\to 2}}{\sqrt{m}B_{\text{out}}/2}\right)} \\ =& 2\sqrt{2}B_{\text{out}}\sqrt{\frac{Nn}{m}\log e\left(1 + \frac{2K_L\|\mathbf{A}\|_{2\to 2}}{\sqrt{m}B_{\text{out}}/2}\right)},\end{aligned}$$

where we have used the following inequality for the last step [8][Lemma C.9]

$$\int_0^{\alpha}\sqrt{\log\left(1 + \frac{\beta}{t}\right)}\mathrm{d}t \leq \alpha\sqrt{\log e(1 + \beta/\alpha)}. \tag{3.10}$$

The theorem is obtained using Theorem 3.1, Lemma 3.3 and 3.4. □

Theorem 3.9 holds for general τ and $\mathbf{A}$. However, the convergence analysis of ISTA [7] shows that the algorithm may not converge if these parameters are not properly chosen. We assume that τ and $\mathbf{A}$ are such that $\tau\|\mathbf{A}\|^2_{2\to 2} \leq 1$ to ensure convergence. The above theorem is then simplified to the following corollary using 3.9.

Corollary 3.10. With the hypothesis space $\mathcal{H}^L$ defined as 2.4, assume that $\tau \|\mathbf{A}\|_{2\to 2}^2 \leq 1$. Then, with probability at least $1-\delta$, the generalization error for any $f_{\mathbf{\Phi}}^L$ is bounded as

$$\mathcal{L}(f_{\mathbf{\Phi}}^L) \leq \hat{\mathcal{L}}(f_{\mathbf{\Phi}}^L) + 8B_{\text{out}}\sqrt{\frac{NnL}{m}\left(1+\log\left(2+\frac{14B_{\text{in}}}{B_{\text{out}}}\right)\right)} \quad + 4(B_{\text{in}}+B_{\text{out}})\sqrt{\frac{2\log(4/\delta)}{m}}. \quad (3.11)$$

3.3 Further remarks and outlook

Some remarks are in order. The above generalization bound holds for general data distributions. However, the corollary is particularly interesting in the compressive sensing setup where the number of measurements n is smaller than N. Suppose that the input data $\mathbf{x}$ is s-sparse in a basis $\mathbf{\Psi}$. According to compressive sensing theory, $n = \Omega(s\log(N/s))$ random subgaussian measurements is sufficient for reconstruction of input using many algorithms including ISTA. In other words, the hypothesis class includes a hypothesis with recovery guarantees. Ignoring logarithmic factors, this means that $\tilde{O}(NsL)$ samples are required for controlling both the generalization error and the reconstruction error. Under certain conditions, ISTA with a known dictionary can perfectly recover the input. One example is if $\mathbf{A}$ satisfies the so-called (ε, s) - restricted isometry property (RIP), that is

$$(1-\varepsilon)\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1+\varepsilon)\|\mathbf{x}\|_2^2$$

for all s-sparse vectors $\mathbf{x} \in \mathbb{R}^N$. In such a case, empirical risk minimization (ERM) not only generalizes well, but also yields a small reconstruction error. Let us summarize this in the following remark.

Remark 3.11. With the hypothesis space $\mathcal{H}^L$ defined as 2.4, assume that $\tau \|\mathbf{A}\|_{2\to 2}^2 \leq 1$. Suppose that a dictionary has been found such that ISTA can perfectly recover all the inputs from the distribution $\mathcal{D}$. Then, with probability at least $1-\delta$, we have

$$\mathcal{L}(\text{ERM}_{\mathcal{S}}) \leq 8B_{\text{out}}\sqrt{\frac{Nn}{m}\log e\left(1+\frac{7\tau\|\boldsymbol{Y}\|_F 2^L}{B_{\text{out}}\sqrt{m}}\right)} + 4(B_{\text{in}}+B_{\text{out}})\sqrt{\frac{2\log(4/\delta)}{m}}, \quad (3.12)$$

where $\text{ERM}_{\mathcal{S}}$ is the empirical risk minimizer hypothesis.

We expect that our theoretical results can be extended to far more general scenarios. The method used for the generalization error can be applied as well for cases where a different dictionary or a non-orthogonal dictionary is used at each layer. One might expect to obtain better results when the structure of data is taken into account. All these derivations are expected to follow from a similar framework presented in this paper.

4 Thresholding Networks for Sparse Recovery

We obtained a worst-case bound on the sample complexity that holds uniformly over the hypothesis space and for any arbitrary data distribution. Although the bound is quite simple and general, it is interesting to see if it can be improved for data from (realistic) low complexity distributions, or whether the generalization error behaves similarly when it is applied e.g. to sparse recovery tasks. Since ISTA is used mainly in sparse coding and recovery, this scenario suggests itself.

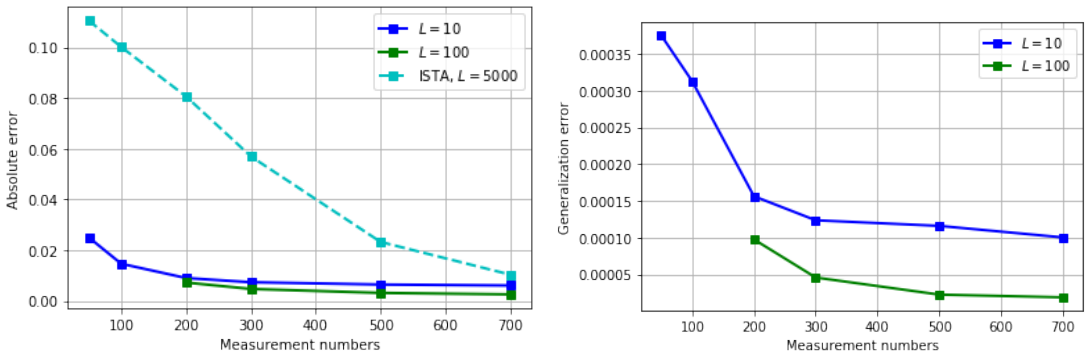
We consider a synthetic dataset as well as the MNIST data set [21]. For both cases, the measurement matrix is a random Gaussian matrix properly normalized to guarantee convergence of soft-thresholding algorithms. The synthetic data is generated for different input and output dimensions and sparsity level. The original dictionary is a random orthogonal matrix. The

default parameters are $N = 100$, $n = 80$ and sparsity equal to 20. Sparse vectors in the dictionary basis are generated by choosing their support uniformly randomly and non-zero values according to the standard normal distribution. The experiments for the synthetic data are repeated 10 times, and the results are averaged over the repetitions. For both the MNIST and the synthetic dataset, we sweep over L, N and n to see how the generalization error behaves.

There are different ways to implement the orthogonality constraint for weight matrices. One way [23] is based on the fact that the matrix exponential mapping provides a bijective mapping from the skew-symmetric matrices onto the special orthogonal group $SO(N)$. However, we use an alternative method of adding a regularization term $\|\mathbf{I} - \mathbf{\Phi}^\top \mathbf{\Phi}\|_F$ (or another matrix norm) to the loss function, which means to penalize if $\mathbf{\Phi}$ is far from being orthogonal.

We choose different number of measurements and layers for both datasets. For each one, the network is trained for a few epochs. Mostly not more than 10 epochs are required to get first promising results, and often times, the loss goes down very slowly after 10 epochs.

All experiments (see Figure 1a) show that it is possible to recover the original vectors $\mathbf{x}$ with as few as 10 layers, which is less than typical when using ISTA (see supplementary materials for some visuals). Note that the error in the MNIST experiments is the pixel-based error normalized by the image dimension. We have chosen Iterative soft thersholding algorithm (ISTA) with a similar structure and 5000 iterations. The result warrants the applicability of dictionary learning for sparse reconstruction.



(a) Absolute reconstruction error for different measurements of MNIST

(b) Generalization error for different measurements of MNIST

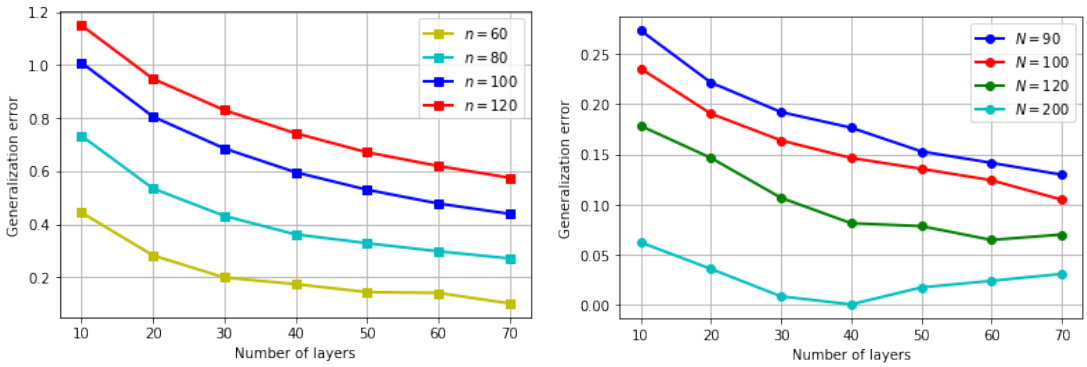
Figure 1: MNIST dataset

Figure 2a confirms the dependence of the generalization error on the number of measurements n. Increasing the number of measurements increases the generalization error for a fixed number of layers (compare plots for various values of n). However, for the MNIST dataset, it seems that increasing the number of measurements decreases the generalization error. The generalization error also decreases by increasing the number of layers for both synthetic and MNIST dataset. See Figure 1b, 2a and 2b. Besides, Figure 2b shows that increasing N decreases the generalization error.

This is, however, not unexpected. As we mentioned above, the sample complexity is supposed to apply to all possible input distributions, If we restrict ourselves to distributions over low complexity sets, then various worst-case bounds in our analysis might be improved. The experiments seem to confirm this intuition. Namely, for the MNIST dataset there is a clear improvement with increasing measurement numbers and the number of layers. This is intuitive from a compressive sensing standpoint, as more number of layers in ISTA leads to better results and more measurements provide more information about the input.

On the other hand, the synthetic dataset shows that the generalization error increases with the input dimension and the number of layers. Note that the bound of this paper is obtained for a very general setting where nothing is assumed on the data structure. Additional assumptions

on the structure of the problem can be used the improve the current bound. Nonetheless, the linear dimension dependency of the current bound makes it a very good baseline for future comparisons.



(a) Generalization error for different measurements of synthetic data

(b) Generalization error for different input dimensions of synthetic data

Figure 2: Synthetic dataset

There are many ideas for improving the performance of this method experimentally. Firstly, it has been noted in many works that training RNN architectures are difficult in general. Many works on LISTA, however, use a different dictionary at each layer, which eases the training procedure. We expect that the proposed method can be improved and tested on various benchmarks with ideas borrowed from research on LISTA.

5 Conclusion and Outlook

In this paper, we have derived a generalization bound for deep thresholding networks like LISTA. To the best of our knowledge, this is the first result of its kind, where most works so far focused on applications. Our proof utilizes a Rademacher complexity analysis and obtains generalization bounds with only linear dependence on the dimension. Particularly, we have applied the contraction principle Lemma 3.3 for vector-valued functions in the context of deep neural networks. With this tool, it is possible to analyze considerably more general situations than just hypothesis classes consisting of real-valued functions. In this way, it is also possible to study general regression problems, whereas so far, research has strongly focused on classification using feed-forward neural networks. Regression problems of particular interest (that are similar to the present scenario) are in general all autoencoders, which may be analysed using the same approach. In particular, analogously to the number of measurements n appearing in our generalization bound, we expect that the number of latent variables plays a similar role for general autoencoders. The comparison of our theoretical results and the numerical results suggests that we might be able to obtain tighter generalization bounds of neural networks for structured input data. Future works consist of also considering more intricate structures with more flexible weight sharing between the layers and also learning the parameters simultaneously.

Acknowledgement

The third author acknowledges funding from the Deutsche Forschungsgemeinschaft (DFG) through the project *Structured Compressive Sensing via Neural Network Learning* (SCoSNeL, MA 1184/36-1) within the SPP 1798 *Compressed Sensing in Information Processing* (CoSIP).

References

[1] Aviad Aberdam, Alona Golts, and Michael Elad. Ada-lista: Learned solvers adaptive to varying models. *Preprint arXiv:2001.08456*, 2020.

[2] Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. *Preprint arXiv:1802.05296*, 2018.

[3] Simon Arridge, Peter Maass, Ozan Öktem, and Carola-Bibiane Schönlieb. Solving inverse problems using data-driven models. *Acta Numerica*, 28:1–174, 2019.

[4] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6240–6249. Curran Associates, Inc., 2017.

[5] Xiaohan Chen, Jialin Liu, Zhangyang Wang, and Wotao Yin. Theoretical linear convergence of unfolded ista and its practical weights and thresholds. In *Advances in Neural Information Processing Systems*, pages 9061–9071, 2018.

[6] B. DasGupta and E.D. Sontag. Sample complexity for learning recurrent perceptron mappings. *IEEE Transactions on Information Theory*, 42(5):1479–1487, September 1996. Conference Name: IEEE Transactions on Information Theory.

[7] Ingrid Daubechies, Michel Defrise, and Christine De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 57(11):1413–1457, 2004.

[8] Simon Foucart and Holger Rauhut. *A Mathematical Introduction to Compressive Sensing*. Applied and Numerical Harmonic Analysis. Springer New York, New York, NY, 2013.

[9] Alexandros Georgogiannis. The generalization error of dictionary learning with moreau envelopes. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1617–1625, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

[10] Noah Golowich, Alexander Rakhlin, and Ohad Shamir. Size-Independent Sample Complexity of Neural Networks. In *Conference On Learning Theory*, pages 297–299, July 2018.

[11] Nina M Gottschling, Vegard Antun, Ben Adcock, and Anders C Hansen. The troublesome kernel: why deep learning for inverse problems is typically unstable. *Preprint arXiv:2001.01258*, 2020.

[12] Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 399–406, 2010.

[13] Rémi Gribonval, Rodolphe Jenatton, Francis Bach, Martin Kleinsteuber, and Matthias Seibert. Sample complexity of dictionary learning and other matrix factorizations. *IEEE Transactions on Information Theory*, 61(6):3469–3486, 2015.

[14] Rémi Gribonval and Karin Schnass. Dictionary identification - sparse matrix-factorisation via ℓ_1-minimisation. *IEEE Transactions on Information Theory*, 56(7):3523–3539, 2010.

[15] Marzieh Hasannasab, Johannes Hertrich, Sebastian Neumayer, Gerlind Plonka, Simon Setzer, and Gabriele Steidl. Parseval proximal neural networks. 2019.

[16] Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. *Preprint arXiv:1912.02178*, 2019.

[17] A. Jung, Y. C. Eldar, and N. Görtz. Performance limits of dictionary learning for sparse coding. In *2014 22nd European Signal Processing Conference (EUSIPCO)*, pages 765–769, 2014.

[18] A. Jung, Y. C. Eldar, and N. Görtz. On the minimax risk of dictionary learning. *IEEE Transactions on Information Theory*, 62(3):1501–1515, 2016.

[19] Ulugbek S Kamilov and Hassan Mansour. Learning optimal nonlinearities for iterative thresholding algorithms. *IEEE Signal Processing Letters*, 23(5):747–751, 2016.

[20] Pascal Koiran and Eduardo D. Sontag. Vapnik-Chervonenkis dimension of recurrent neural networks. *Discrete Applied Mathematics*, 86(1):63–79, August 1998.

[21] Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*.

[22] Michel Ledoux and Michel Talagrand. *Probability in Banach spaces: isoperimetry and processes*. Classics in mathematics. Springer, Berlin ; London, 2011. OCLC: ocn751525992.

[23] Mario Lezcano-Casado and David Martínez-Rubio. Cheap orthogonal constraints in neural networks: A simple parametrization of the orthogonal and unitary group. *Preprint arXiv:1901.08428*, 2019.

[24] Jialin Liu, Xiaohan Chen, Zhangyang Wang, and Wotao Yin. Alista: Analytic weights are as good as learned weights in lista. 2018.

[25] Andreas Maurer. A vector-contraction inequality for Rademacher complexities. *Preprint arXiv:1605.00251*, May 2016. arXiv: 1605.00251.

[26] Ali Mousavi, Ankit B Patel, and Richard G Baraniuk. A deep learning approach to structured signal recovery. In *2015 53rd annual allerton conference on communication, control, and computing (Allerton)*, pages 1336–1343. IEEE, 2015.

[27] Vaishnavh Nagarajan and J Zico Kolter. Uniform convergence may be unable to explain generalization in deep learning. In *Advances in Neural Information Processing Systems*, pages 11611–11622, 2019.

[28] Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nati Srebro. Exploring generalization in deep learning. In *Advances in Neural Information Processing Systems*, pages 5947–5956, 2017.

[29] Behnam Neyshabur, Srinadh Bhojanapalli, and Nathan Srebro. A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *Preprint arXiv:1707.09564*, 2017.

[30] Behnam Neyshabur, Zhiyuan Li, Srinadh Bhojanapalli, Yann LeCun, and Nathan Srebro. Towards understanding the role of over-parametrization in generalization of neural networks. *Preprint arXiv:1805.12076*, 2018.

[31] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. *Preprint arXiv:1412.6614*, 2014.

[32] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318, 2013.

[33] Holger Rauhut, Karin Schnass, and Pierre Vandergheynst. Compressed sensing and redundant dictionaries. *IEEE Transactions on Information Theory*, 54(5):2210–2219, 2008.

[34] Karin Schnass. On the identifiability of overcomplete dictionaries via the minimisation principle underlying K-SVD. *Applied and Computational Harmonic Analysis*, (3):37, 2014.

[35] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: from theory to algorithms*. Cambridge University Press, New York, NY, USA, 2014.

[36] Pablo Sprechmann, Alexander M Bronstein, and Guillermo Sapiro. Learning efficient sparse and low rank models. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1821–1833, 2015.

[37] Hillel Sreter and Raja Giryes. Learned convolutional sparse coding. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2191–2195. IEEE, 2018.

[38] Daniel Vainsencher, Shie Mannor, and Alfred M Bruckstein. The sample complexity of dictionary learning. *Journal of Machine Learning Research*, 12(Nov):3259–3281, 2011.

[39] Kailun Wu, Yiwen Guo, Ziang Li, and Changshui Zhang. Sparse coding with gated learned ista. In *International Conference on Learning Representations*, 2020.

[40] Bo Xin, Yizhou Wang, Wen Gao, David Wipf, and Baoyuan Wang. Maximal sparsity with deep networks? In *Advances in Neural Information Processing Systems*, pages 4340–4348, 2016.

[41] Jian Zhang and Bernard Ghanem. Ista-net: Interpretable optimization-inspired deep network for image compressive sensing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1828–1837, 2018.

Appendix

A Proof of Theorem 3.4

In this section we prove Theorem 3.4. It is a key ingredient for the proof of the main result, Theorem 3.9, but also interesting in its own right. As a preparation, we need the following Lemma.

Lemma A.1. *For any $\mathbf{\Phi} \in O(N)$, $l \in \mathbb{N}$, and arbitrary $\tau, \lambda > 0$ in $S_{\tau\lambda}$ in the definition (2.2) of $f^l_{\mathbf{\Phi}}$, we have*

$$\left\| f^l_{\mathbf{\Phi}}(\boldsymbol{Y}) \right\|_F \leq \left\| \tau(\mathbf{A\Phi})^\top \boldsymbol{Y} \right\|_F \sum_{k=0}^{l-1} \left\| \mathbf{I} - \tau \mathbf{\Phi}^\top \mathbf{A}^\top \mathbf{A} \mathbf{\Phi} \right\|_{2\to 2}^k \tag{A.1}$$

$$\leq \tau \|\mathbf{A}\|_{2\to 2} \|\boldsymbol{Y}\|_F \sum_{k=0}^{l-1} \left\| \mathbf{I} - \tau \mathbf{A}^\top \mathbf{A} \right\|_{2\to 2}^k . \tag{A.2}$$

Proof. We only need to prove (A.1). The second inequality (A.2) then follows immediately using the orthogonality of $\mathbf{\Phi}$. Thus, we turn to the prove of (A.1) via induction. Clearly, for $l = 1$, we have $\left\| f^1_{\mathbf{\Phi}}(\boldsymbol{Y}) \right\|_F = \left\| \tau(\mathbf{A\Phi})^\top \boldsymbol{Y} \right\|_F$. Assuming the statement is true for l, we obtain it for $l+1$ by the following chain of inequalities, using in particular the contractivity $S_{\tau\lambda}$ with respect to the Frobenius norm,

$$\begin{aligned}
\left\| f^{l+1}_{\mathbf{\Phi}}(\boldsymbol{Y}) \right\|_F &= \left\| S_{\tau\lambda} \left[\left(\mathbf{I} - \tau \mathbf{\Phi}^\top \mathbf{A}^\top \mathbf{A} \mathbf{\Phi} \right) f^l_{\mathbf{\Phi}}(\boldsymbol{Y}) + \tau(\mathbf{A\Phi})^\top \boldsymbol{Y} \right] \right\|_F \\
&\leq \left\| \left(\mathbf{I} - \tau \mathbf{\Phi}^\top \mathbf{A}^\top \mathbf{A} \mathbf{\Phi} \right) f^l_{\mathbf{\Phi}}(\boldsymbol{Y}) \|_F + \|\tau(\mathbf{A\Phi})^\top \boldsymbol{Y} \right\|_F \\
&\leq \left\| \mathbf{I} - \tau \mathbf{\Phi}^\top \mathbf{A}^\top \mathbf{A} \mathbf{\Phi} \right\|_{2\to 2} \left\| f^l_{\mathbf{\Phi}}(\boldsymbol{Y}) \right\|_F + \left\| \tau(\mathbf{A\Phi})^\top \boldsymbol{Y} \right\|_F \\
&\leq \left\| \tau(\mathbf{A\Phi})^\top \boldsymbol{Y} \right\|_F \sum_{k=0}^{l-1} \left\| \mathbf{I} - \tau \mathbf{\Phi}^\top \mathbf{A}^\top \mathbf{A} \mathbf{\Phi} \right\|_{2\to 2}^{k+1} + \left\| \tau(\mathbf{A\Phi})^\top \boldsymbol{Y} \right\|_F \\
&= \left\| \tau(\mathbf{A\Phi})^\top \boldsymbol{Y} \right\|_F \sum_{k=0}^{l} \left\| \mathbf{I} - \tau \mathbf{\Phi}^\top \mathbf{A}^\top \mathbf{A} \mathbf{\Phi} \right\|_{2\to 2}^{k} .
\end{aligned}$$

□

Now we turn to the actual proof of Theorem 3.4.

Proof of Theorem 3.4. We formally set $f^0_{\mathbf{\Phi}_1}(\boldsymbol{Y}) = f^0_{\mathbf{\Phi}_2}(\boldsymbol{Y}) = \boldsymbol{Y}$ for a unified treatment of all layers $l \geq 1$. Using the fact that $S_{\tau\lambda}$ is 1-Lipschitz we obtain

$$\begin{aligned}
&\left\| f^l_{\mathbf{\Phi}_1}(\boldsymbol{Y}) - f^l_{\mathbf{\Phi}_2}(\boldsymbol{Y}) \right\|_F \\
\leq &\left\| \left(\mathbf{I} - \tau(\mathbf{A\Phi}_1)^\top \mathbf{A\Phi}_1 \right) f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y}) + \tau(\mathbf{A\Phi}_1)^\top \boldsymbol{Y} \right. \\
&\qquad \left. - \left(\mathbf{I} - \tau(\mathbf{A\Phi}_2)^\top \mathbf{A\Phi}_2 \right) f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y}) - \tau(\mathbf{A\Phi}_2)^\top \boldsymbol{Y} \right\|_F \qquad (A.3) \\
\leq &\left\| f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y}) - f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y}) \right\|_F + \tau \|\mathbf{A\Phi}_1 - \mathbf{A\Phi}_2\|_{2\to 2} \|\boldsymbol{Y}\|_F \\
&\qquad + \tau \left\| (\mathbf{A\Phi}_2)^\top \mathbf{A\Phi}_2 f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y}) - (\mathbf{A\Phi}_1)^\top \mathbf{A\Phi}_1 f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y}) \right\|_F . \qquad (A.4)
\end{aligned}$$

We further estimate the term in (A.4) by introducing mixed terms and applying the triangle

inequality,

$$
\begin{aligned}
&\left\|(\mathbf{A}\mathbf{\Phi}_2)^\top \mathbf{A}\mathbf{\Phi}_2 f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y}) - (\mathbf{A}\mathbf{\Phi}_1)^\top \mathbf{A}\mathbf{\Phi}_1 f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y})\right\|_F \\
\leq &\left\|(\mathbf{A}\mathbf{\Phi}_2)^\top \mathbf{A}\mathbf{\Phi}_2 f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y}) - (\mathbf{A}\mathbf{\Phi}_1)^\top \mathbf{A}\mathbf{\Phi}_2 f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\| \\
&\quad + \left\|(\mathbf{A}\mathbf{\Phi}_1)^\top \mathbf{A}\mathbf{\Phi}_2 f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y}) - (\mathbf{A}\mathbf{\Phi}_1)^\top \mathbf{A}\mathbf{\Phi}_1 f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F \\
&\quad\quad + \left\|(\mathbf{A}\mathbf{\Phi}_1)^\top \mathbf{A}\mathbf{\Phi}_1 f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y}) - (\mathbf{A}\mathbf{\Phi}_1)^\top \mathbf{A}\mathbf{\Phi}_1 f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y})\right\|_F \\
\leq &\left\|(\mathbf{A}\mathbf{\Phi}_2 - \mathbf{A}\mathbf{\Phi}_1)^\top\right\|_{2\to 2} \|\mathbf{A}\mathbf{\Phi}_2\|_{2\to 2} \left\|f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F \\
&\quad + \left\|(\mathbf{A}\mathbf{\Phi}_1)^\top\right\|_{2\to 2} \|\mathbf{A}\mathbf{\Phi}_2 - \mathbf{A}\mathbf{\Phi}_1\|_{2\to 2} \left\|f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F \\
&\quad\quad + \left\|(\mathbf{A}\mathbf{\Phi}_1)^\top \mathbf{A}\mathbf{\Phi}_1\right\|_{2\to 2} \left\|f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y}) - f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y})\right\|_F \\
\leq &\|\mathbf{A}\mathbf{\Phi}_2 - \mathbf{A}\mathbf{\Phi}_1\|_{2\to 2} \|\mathbf{A}\|_{2\to 2} \left\|f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F \\
&\quad + \|\mathbf{A}\|_{2\to 2} \|\mathbf{A}\mathbf{\Phi}_2 - \mathbf{A}\mathbf{\Phi}_1\|_{2\to 2} \left\|f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F \\
&\quad\quad + \left\|\mathbf{A}^\top \mathbf{A}\right\|_{2\to 2} \left\|f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y}) - f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y})\right\|_F \\
\leq &2\|\mathbf{A}\mathbf{\Phi}_2 - \mathbf{A}\mathbf{\Phi}_1\|_{2\to 2} \|\mathbf{A}\|_{2\to 2} \left\|f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F + \left\|\mathbf{A}^\top \mathbf{A}\right\|_{2\to 2} \left\|f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y}) - f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y})\right\|_F .
\end{aligned}
$$

Plugging this estimate into (A.4) and and applying Lemma A.1 in the second inequality below yields

$$
\begin{aligned}
&\left\|f^{l}_{\mathbf{\Phi}_1}(\boldsymbol{Y}) - f^{l}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F \qquad\qquad (A.5)\\
\leq &\left(1 + \tau \left\|\mathbf{A}^\top \mathbf{A}\right\|_{2\to 2}\right) \left\|f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y}) - f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F \\
&\quad + \tau \|\mathbf{A}\mathbf{\Phi}_2 - \mathbf{A}\mathbf{\Phi}_1\|_{2\to 2} \left(\|\boldsymbol{Y}\|_F + 2\|\mathbf{A}\|_{2\to 2} \left\|f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F\right) \\
\leq &\left(1 + \tau\|\mathbf{A}\|^2_{2\to 2}\right) \left\|f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y}) - f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F \\
&\quad + \tau\|\boldsymbol{Y}\|_F \|\mathbf{A}\mathbf{\Phi}_2 - \mathbf{A}\mathbf{\Phi}_1\|_{2\to 2} \left(1 + 2\tau \|\mathbf{A}\|^2_{2\to 2} Z_{l-1}\right) \qquad (A.6)\\
\leq &A \left\|f^{l-1}_{\mathbf{\Phi}_1}(\boldsymbol{Y}) - f^{l-1}_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F + B_l \|\mathbf{A}\mathbf{\Phi}_2 - \mathbf{A}\mathbf{\Phi}_1\|_{2\to 2}, \qquad (A.7)
\end{aligned}
$$

where Z_l (with $l \geq 0$) in (A.6) and A, B_l (with $l \geq 1$) in (A.7) are defined as

$$
\begin{aligned}
A &:= \left(1 + \tau\|\mathbf{A}\|^2_{2\to 2}\right), \\
Z_0 &:= 0, \qquad Z_l := \sum_{k=0}^{l-1} \left\|\mathbf{I} - \tau \mathbf{A}^\top \mathbf{A}\right\|^k_{2\to 2}, \quad l \geq 1, \\
B_l &:= \tau\|\boldsymbol{Y}\|_F \left(1 + 2\tau \|\mathbf{A}\|^2_{2\to 2} Z_{l-1}\right), \qquad l \geq 1.
\end{aligned}
$$

Using these abbreviations, the general formula for K_L in (3.8) has the compact form

$$
K_L = \sum_{l=1}^{L} A^{L-l} B_l, \qquad L \geq 1. \qquad (A.8)
$$

Based on (A.7) we prove via induction that (3.7) holds for any number of layers $L \in \mathbb{N}$ with K_L given by (A.8). For $L = 1$, we can directly calculate the constant K_1 via

$$
\begin{aligned}
\left\|f^1_{\mathbf{\Phi}_1}(\boldsymbol{Y}) - f^1_{\mathbf{\Phi}_2}(\boldsymbol{Y})\right\|_F &= \left\|S_{\tau\lambda}(\tau(\mathbf{A}\mathbf{\Phi}_1)^\top \boldsymbol{Y}) - S_{\tau\lambda}(\tau(\mathbf{A}\mathbf{\Phi}_2)^\top \boldsymbol{Y})\right\|_F \\
&\leq \tau\|\boldsymbol{Y}\|_F \|\mathbf{A}\mathbf{\Phi}_1 - \mathbf{A}\mathbf{\Phi}_2\|_{2\to 2},
\end{aligned}
$$

so that $K_1 = \tau\|\boldsymbol{Y}\|_F = B_1$, as claimed in (A.8). Plugging this into the estimate for the second layer $L = 2$ and using (A.7), we obtain

$$\begin{aligned}\left\|f^2_{\boldsymbol{\Phi}_1}(\boldsymbol{Y}) - f^2_{\boldsymbol{\Phi}_2}(\boldsymbol{Y})\right\|_F &\leq A\left\|f^1_{\boldsymbol{\Phi}_1}(\boldsymbol{Y}) - f^{l-1}_{\boldsymbol{\Phi}_2}(\boldsymbol{Y})\right\|_F + B_2\|\mathbf{A}\boldsymbol{\Phi}_2 - \mathbf{A}\boldsymbol{\Phi}_1\|_{2\to 2}\\ &\leq AK_1\|\mathbf{A}\boldsymbol{\Phi}_2 - \mathbf{A}\boldsymbol{\Phi}_1\|_{2\to 2} + B_2\|\mathbf{A}\boldsymbol{\Phi}_2 - \mathbf{A}\boldsymbol{\Phi}_1\|_{2\to 2}\\ &\leq (AB_1 + B_2)\|\mathbf{A}\boldsymbol{\Phi}_2 - \mathbf{A}\boldsymbol{\Phi}_1\|_{2\to 2}.\end{aligned}$$

Hence, $K_2 = AB_1 + B_2 = \sum_{l=1}^{2} A^{2-l}B_l$, which is of the claimed form (A.8) with $L = 2$. Now we proceed with the induction step, assuming formula (A.8) to hold for some $L \in \mathbb{N}$. Applying the estimate after (A.5) for the output after layer $L + 1$, we obtain

$$\begin{aligned}\left\|f^{L+1}_{\boldsymbol{\Phi}_1}(\boldsymbol{Y}) - f^{L+1}_{\boldsymbol{\Phi}_2}(\boldsymbol{Y})\right\|_F \leq& A\left\|f^{L}_{\boldsymbol{\Phi}_1}(\boldsymbol{Y}) - f^{L}_{\boldsymbol{\Phi}_2}(\boldsymbol{Y})\right\|_F + B_{L+1}\|\mathbf{A}\boldsymbol{\Phi}_2 - \mathbf{A}\boldsymbol{\Phi}_1\|_{2\to 2}\\ \leq& AK_L\|\mathbf{A}\boldsymbol{\Phi}_2 - \mathbf{A}\boldsymbol{\Phi}_1\|_{2\to 2} + B_{L+1}\|\mathbf{A}\boldsymbol{\Phi}_2 - \mathbf{A}\boldsymbol{\Phi}_1\|_{2\to 2}\\ \leq& (AK_L + B_{L+1})\|\mathbf{A}\boldsymbol{\Phi}_2 - \mathbf{A}\boldsymbol{\Phi}_1\|_{2\to 2},\end{aligned}$$

and therefore,

$$K_{L+1} = AK_L + B_{L+1} = A\sum_{l=1}^{L} A^{L-l}B_l + B_{L+1} = \sum_{l=1}^{L} A^{L-l+1}B_l + B_{L+1} = \sum_{l=1}^{L+1} A^{(L+1)-l}B_l.$$

This is the desired expression for K_{L+1} and finishes the proof of (3.7). It remains to prove the upper bound (3.9). Plugging in the assumption $\tau\,\|\mathbf{A}\|^2_{2\to 2} \leq 1$ as well as the observation $\left\|\mathbf{I} - \tau\mathbf{A}^\top\mathbf{A}\right\|_{2\to 2} = 1$ from Remark 3.5 into (3.8), we obtain

$$K_L \leq \tau\|\boldsymbol{Y}\|_F\left[1 + \sum_{l=2}^{L} 2^{L-l}\,(1 + 2(l-2))\right] = \tau\|\boldsymbol{Y}\|_F\left[1 + 2^{L-2}\sum_{k=0}^{L-2} 2^{-k}\,(1+2k)\right].$$

The sum can be further bounded by passing to the infinite series, which can be directly calculated (similar to the case of a geometric series).

$$\sum_{l=0}^{L-2} 2^{-l}\,(1+2l) \leq \sum_{l=0}^{\infty} 2^{-l}\,(1+2l) = 6.$$

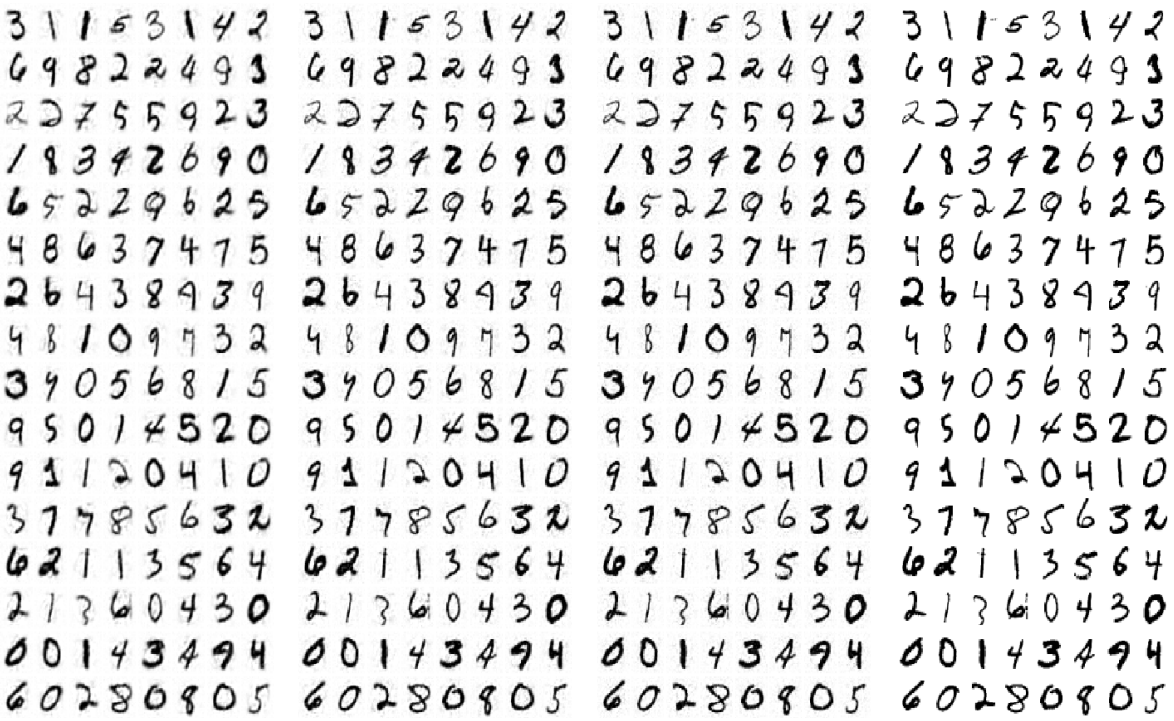
Together, this gives us the desired result. □

B Reconstruction Comparisons

In Figure 3 and 4, examples of reconstructed images are shown for different number of layers, using 200 (Figure 3) and 500 (Figure 4) measurements. (Recall that the MNIST images are of pixel size $28 \times 28 = 784$.)

(a) $L = 10$ (b) $L = 100$ (c) $L = 1000$ (d) Original

Figure 3: MNIST examples: reconstruction from 200 measurements with different number of layers L.



(a) $L = 10$ (b) $L = 100$ (c) $L = 1000$ (d) Original

Figure 4: MNIST examples: reconstruction from 500 measurements with different number of layers L.