
Degrees of Freedom Analysis of Unrolled Neural Networks

Morteza Mardani¹, Qingyun Sun², Vardan Papyan², Shreyas Vasanaawala³,
John Pauly¹, and David Donoho²

Depts. of Electrical Engineering¹, Statistics², and Radiology³, Stanford University
{morteza, qysun, papyan, vasanaawala, pauly, donoho}@stanford.edu

Abstract

Unrolled neural networks emerged recently as an effective model for learning inverse maps appearing in image restoration tasks. However, their generalization risk (i.e., test mean-squared-error) and its link to network design and train sample size remains mysterious. Leveraging the Stein’s Unbiased Risk Estimator (SURE), this paper analyzes the generalization risk with its bias and variance components for recurrent unrolled networks. We particularly investigate the degrees-of-freedom (DOF) component of SURE, trace of the end-to-end network Jacobian, to quantify the prediction variance. We prove that DOF is well-approximated by the weighted *path sparsity* of the network under incoherence conditions on the trained weights. Empirically, we examine the SURE components as a function of train sample size for both recurrent and non-recurrent (with many more parameters) unrolled networks. Our key observations indicate that: 1) DOF increases with train sample size and converges to the generalization risk for both recurrent and non-recurrent schemes; 2) recurrent network converges significantly faster (with less train samples) compared with non-recurrent scheme, hence recurrence serves as a regularization for low sample size regimes.

1 Introduction

Training deep neural networks typically demands abundant labeled data to achieve an acceptable generalization. Collecting valid labels, however, is costly if not impossible for certain applications such as medical imaging due to physical constraints and privacy concerns. This paper deals with imaging from compressive and noisy measurements, where labels are high-quality images that in medical imaging drive diagnostic decisions.

Outside the scarce-label setting, recent works adopts *unrolled* neural networks to learn the inversion map for recovering an underlying image from compressive and corrupted measurements; see e.g., [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13] and references therein. The crux of unrolled schemes is to cast recovery as an image-to-image translation task mapping a low-quality image (e.g., found as linear estimate) to a high quality label image. It is essentially a cascade of alternating denoisers and data adjustment units with denoisers modeled via neural networks.

The denoiser networks are either allowed to be shared, leading to RNN architectures (see e.g., [10, 8]), or separately vary across iterations (see e.g., [4, 6, 9]). The latter seems to entail smaller networks that are easier to train, and they have been observed to achieve promising generalization performance for solving inverse problems. A noteworthy example of which includes the neural proximal gradient descent algorithm (NPGD) that models the proximal map using residual networks (ResNets) and after training using a recurrent neural network (RNN) it achieves state-of-the-art performance for compressed sensing tasks such as MR imaging. The recurrent inference machines in [14] also leverages RNNs to learn the prior distribution parameters. For natural image superresolution tasks also cross-layer weight sharing achieves state-of-the-art quality [8].

All in all, the generalization risk of unrolled neural networks for inverse problems has not been studied to date. This paper aims to extensively study this and reveals the influence of network architecture and train sample size.

Contributions. In order to study the prediction error of unrolled neural networks we leverage the Stein’s Unbiased Risk Estimator (SURE) [15, 16] as a proxy for the generalization MSE. In statistical estimation, SURE comprises two terms, residual sum of squares (RSS) plus degrees of freedom (DOF), where RSS typically accounts for the prediction bias, while DOF captures the prediction uncertainty. We particularly focus on DOF, that is the trace of the end-to-end network Jacobian. Adopting a single layer residual unit with skip connection for the denoiser, the achievable DOF is analyzed for the denoising task. Under certain incoherence conditions on the weights of the trained network, DOF is derived in terms of the weighted path sparsity of the network activation as a useful surrogate to assess the generalization risk.

We conducted extensive empirical evaluations for denoising natural images, confirming the theoretically-predicted behavior of SURE. The adopted RNN architecture with weight sharing (WS) is compared with an alternative scheme where the weights are allowed to freely change across iterations, the so termed weight changing (WC). The comparisons reveal an interesting trade off. WS achieves higher DOF but lower RSS than WC. The overall SURE for WC is however smaller. The SURE gap between WS and WC schemes is shown to be significant for low sample sizes, but decreases as the sample size grows; eventually WS and WC agree as label scarcity abates. Further experiments for natural image deblurring show superior PSNR for WS vs. WC. We also compared the filtering behavior of the learned proximals for WS and WC inspired by deep scattering networks [17, 18]. For this purpose we analyzed the frequency spectrum of different iterations that show WS performs bandpass filtering, while WC alternates between low and bandpass filtering to denoise the images.

In summary, these findings rest on several novel contributions:

- Theoretical analysis of generalization risk using SURE for recurrent unrolled networks
- Proved that DOF is well-approximated by the weighted *path sparsity* under proper incoherence conditions
- Extensive empirical evaluation of SURE for recurrent and non-recurrent networks for natural image denoising and deblurring (compressed sensing MRI in supplementary materials).
- Filtering interpretation of the trained weights in recurrent and non-recurrent unrolled networks.

The rest of this paper is organized as follows. Section 2 introduces the preliminaries on neural proximal algorithms and states the problem. The generalization risk is analyzed in Section 3. Empirical evaluations are then reported in Section 4, while Section 5 discusses the conclusions.

Notations. In the paper $(\cdot)^\dagger$, $(\cdot)^H$, $\|\cdot\|_2$, \mathbb{E} , tr , \circ , and I_n refer to the matrix pseudo inverse, Hermitian, ℓ_2 -norm, statistical expectation, trace, composition operator, and $n \times n$ identity matrix. $\mathbb{1}_{\{x\}}$ also denotes the indicator function that is unity for $x > 0$, and zero otherwise.

2 Preliminaries and Problem Statement

Consider the linear system

$$y = \Phi x + v \tag{1}$$

with $\Phi \in \mathbb{C}^{m \times n}$ and $m \leq n$, where the Gaussian noise $v \sim \mathcal{N}(0, \sigma^2)$ captures the noise and unmodeled dynamics. Suppose the unknown image x lies in a low-dimensional manifold. No information is known about the manifold besides the training samples $\mathcal{X} := \{x_i\}_{i=1}^N$ drawn from it, and the corresponding noisy observations $\mathcal{Y} := \{y_i\}_{i=1}^N$. Given a new undersampled observation y , the goal is to quickly recover a plausible image \hat{x} that is close to x .

The stated problem covers a wide range of image recovery tasks. For instance, for image denoising $\Phi = I$ [19, 20, 2], for image deblurring Φ [21, 22] signifies the local convolution operator, for image superresolution [23, 24] Φ is the downsampling operator that averages out nonoverlapping image regions to arrive at a low resolution image, and for compressed sensing MRI [25] Φ refers to the subsampled Fourier operator.

2.1 Neural proximal learning

In order to invert the linear system (1) a variation of the proximal algorithm advocated in [10] is adopted. Given a pre-trained proximal operator \mathcal{P}_ψ [26] modeled via a neural network, the overall iterative procedure evolves according to the state-space equations

$$\begin{aligned} \text{step 1.} \quad & s^{t+1} = g(x^t; y) \\ \text{step 2.} \quad & x^{t+1} = \mathcal{P}_\psi(s^{t+1}) \end{aligned}$$

for a fixed number of iterations, i.e., $t = 1, \dots, T$. The first step invokes a linear operation that assures the state consistency with the measurements y . The second step executes the proximal mapping for denoising the image estimate. The recursion starts with the initial linear estimate $x^0 = \Phi^H y$ as the match filtered input y . For the first step, we can perform a first-order gradient step as in [10], or (preferably) a second-order least-squares step if computationally affordable. They are expressed as follows (for a learnable step size α):

- Gradient step

$$g(x^t; y) := \alpha \Phi^H y + (I - \alpha \Phi^H \Phi) x^t.$$

- Least-squares step

$$g(x^t; y) := (\alpha \Phi \Phi^H + (1 - \alpha)I)^{-1} (\alpha \Phi^H y + (1 - \alpha)x^t).$$

2.2 Proximal modeling with neural networks

A network with K residual units (RU) is adopted to model the proximal map \mathcal{P}_ψ . Adopting the ReLU activation $\sigma(x) = D(x) \cdot x$, where $D(x) = \mathbb{1}_x$, the outer iteration t (mapping x^{t-1} to x^t) can be decomposed as follows,

- $h_0^t = g(x_{t-1}; y)$
- $h_{k+1}^t = h_k^t + W_k^H \sigma(\bar{W}_k h_k^t)$, $k = 1, \dots, K$
- $x^t = h_K^t$.

Neural proximal algorithm is recurrent in nature to mimic the fixed point iteration for traditional proximal algorithm [26]. We thus shared weights $\{W_k\}_{k=1}^K$ for different outer iterations t . When $\bar{W}_k = W_k$ we call the model symmetric residual unit, which can provide further regularization through weight sharing.

However, we could also learn different weights $\{W_k^t\}_{k=1}^K$ for different t , which changes the hidden layers at t -th iteration to

$$h_{k+1}^t = h_k^t + W_k^{t,H} \sigma(\bar{W}_k^t h_k^t), \quad k = 1, \dots, K \quad (2)$$

This scheme known as weight changing is used later in the numerical experiments as the benchmark for performance comparison.

Pseudo linear representation. We adopt a pseudo-linear representation for the activation, where D_k is a diagonal mask matrix with binary values for ReLU. Note, during inference, the mask D_k is dependent on the input data examples, while W_k is fixed. Accordingly, we can write $h_{k+1}^t = M_k^t h_k^t$, where $M_k^t = I + W_k^H D_k^t \bar{W}_k$. The overall proximal map M_t at t -th iteration then admits

$$x^{t+1} = \underbrace{M_t^K \dots M_t^2 M_t^1}_{:=M_t} s^{t+1}. \quad (3)$$

Apparently, the map M_t is input data dependent due to nonlinear activation.

Unrolling the T outer iterations of the proximal algorithm and the K inner iterations of the K RUs, the end-to-end recurrent map with input y_i yields

$$\hat{x}_i := x_i^T := (M_T \circ g) \circ \dots \circ (M_1 \circ g)(\Phi^H y_i). \quad (4)$$

Training. One optimizes the network weights $\mathcal{W} := \{W_k\}_{k=1}^K$ to fit $\hat{\mathcal{X}} := \{\hat{x}_i\}$ to $\mathcal{X} := \{x_i\}$ for the training population using pixel-wise empirical loss

$$\text{minimize}_{\mathcal{W}} \quad \frac{1}{N} \sum_{i=1}^N \|\hat{x}_i - x_i\|_2^2. \quad (5)$$

3 Risk Analysis

In order to ease the analytical exposition for the generalization risk we focus on the denoising task ($\Phi = I$),

$$y = x + v, \quad v \sim \mathcal{N}(0, \sigma^2) \quad (6)$$

with $\|x\|_2 = 1$. The derivation presented here could be generalized to an arbitrary Φ with noise following an exponential distribution using ideas presented in [27]. Let $x^T = h_\Theta(y)$ denote the prediction obtained via neural proximal algorithm for the function $h_\Theta(\cdot)$ in (4) with a test sample y as input argument. Assume h is weakly differentiable. The Stein's unbiased risk estimator (SURE) [15, 16] for $h_\Theta(y)$ is then expressed as

$$\text{SURE}(y) = -n\sigma^2 + \underbrace{\|h_\Theta(y) - y\|_2^2}_{\text{RSS}(y)} + 2\sigma^2 \underbrace{\nabla_y \cdot h_\Theta(y)}_{\text{DOF}(y)}, \quad (7)$$

where ∇ is the divergence operator. Note, SURE has two nice properties. First, it does not depend on the ground truth x . Second, it is an unbiased estimate of the test mean-squared-error (MSE), namely

$$\text{MSE} := \mathbb{E} [\|h_\Theta(y) - x\|_2^2] = \mathbb{E} [\text{SURE}(y)]. \quad (8)$$

SURE in (7) comprises two main terms. The residual sum of squares (RSS) measures the error between the corrupted and denoised input, while the DOF measures the *achievable* degrees of freedom for the denoiser.

Lemma 1 *For the considered neural proximal algorithm with the end-to-end nonlinear map J_T , namely $x^T = J_T y$ suppose that $\|x^T\|_2 = 1$ (after normalization). It then holds that*

$$\begin{aligned} \text{DOF} &= \mathbb{E}[\text{tr}(J_T)] \\ \text{RSS} &= n\sigma^2 + 2(1 - \mathbb{E}[\text{tr}(y^H J_T y)]) \end{aligned}$$

A natural question then pertains to the behavior of DOF and RSS terms as well as the overall SURE. In particular, we are interested in DOF that is known in statistical estimation as a measure of the predictor variability and uncertainty [15]. It has been adopted as a notion of optimism or prediction error in [28]. Parameter tuning using SURE has also been advocated in [29, 16]. SURE has also been used recently for unsupervised training of deep inverse maps [12, 11]. DOF captures the neural network capacity (free parameters), the number of training samples, and the training algorithm. A rigorous analysis of DOF for the trained neural PGD algorithm is the subject of next section.

3.1 Degrees of freedom

To facilitate the SURE analysis, we consider the proximal map with a single RU. We assume that RU is symmetric, meaning that the deconvolution operation is simply the convolution transposed with a negative scaling. The corresponding proximals then simply admit $M_t = I - W^H D_t W$ at t -th iteration (see (3)). In addition, assume that for the gradient step $\alpha = 0$, meaning that $g(x^t; y) = x^t$. One can alternatively encourage data consistency by imposing data fidelity during the training. Accordingly, from (4) it is easy to derive the end-to-end map relating the iteration outputs x^{t+1} to the initial estimate $x^0 = y$ as

$$x_{t+1} = \underbrace{(I - W^H D_{t+1} W) \dots (I - W^H D_1 W)}_{:= J_{t+1}} y \quad (9)$$

One can then expand J_T as a linear combination of $J_{\mathcal{I}}$'s where there are in total 2^T different index choices \mathcal{I} that can be associated with an index array (i_1, \dots, i_j) . Accordingly,

$$J_T = I + \sum_{\mathcal{I}} (-1)^{|\mathcal{I}|} J_{\mathcal{I}}, \quad (10)$$

$$J_{\mathcal{I}} = W^H D_{i_j} W W^H \dots W W^H D_{i_1} W, \quad (11)$$

3.1.1 Linear networks

To ease the exposition we begin with a linear network where the proximal map $\mathcal{P}_\Psi(x) = (I - W^H W)x$ is repeated for infinitely many iterations, namely $T \rightarrow \infty$. As a result of end-to-end

training, the overall mapping is bounded and admits

$$\lim_{T \rightarrow \infty} J_T y = \underbrace{(I - WW^\dagger)}_{:= J_\infty} y \quad (12)$$

where $\mathcal{P}_W := WW^\dagger$ is orthogonal projection onto the range space of W .

The overall network mapping J_∞ resembles singular value thresholding for denoising [30]. The expected DOF then admits a closed-form expression as stated next (proofs are included in the supplementary materials).

Lemma 2. *For the unrolled network with single-layer linear residual units with $T \rightarrow \infty$, let $\{\sigma_i\}_{i=1}^n$ denote the singular values of the sample correlation matrix $C_x := \frac{1}{N} \sum_{i=1}^N x_i x_i^H$, and suppose $\hat{\sigma}^2 \approx \frac{1}{N} \sum_{i=1}^N v_i v_i^H$. Then, DOF admits*

$$\text{DOF} \approx n - \sum_{i=1}^{\min\{\ell, n\}} \mathbb{I}_{\{\sigma_i - \hat{\sigma}^2\}}$$

3.1.2 Non-linear networks

In the nonlinear setting to gain some intuition first we focus on the simpler case where the network entails infinitely many iterations. The end-to-end mapping is then expressed as

$$J_T y = \prod_{\tau=0}^T (I - W^H D_{T-\tau} W) y \quad (13)$$

for the mask sequence $\{D_t\}_{t=1}^\infty$. Assuming that the system returns a unique output, the mask sequence converges to the mask D of the latent code elements for the input y . Let the set \mathcal{S} include the support set for \bar{D} . Using similar arguments as the linear case it can be shown that the end-to-end mapping becomes a projection operator as follows

$$J_\infty y = (I - W_{\mathcal{S}} W_{\mathcal{S}}^\dagger) y \quad (14)$$

Lemma 3. *For an unrolled network with single-layer nonlinear residual units and $T \rightarrow \infty$ if the sequence of activation masks converge to \bar{D} (for each input y), the DOF admits*

$$\text{DOF} = n - \mathbb{E}[\text{tr}(\bar{D})] \quad (15)$$

Apparently, the sparsity level of the last layers determines the DOF.

In practice, we are interested in understanding the behavior of unrolled neural networks with a finite number of iterations. Let $\rho_t = \mathbb{E}[\text{tr}(D_t)]$ be the expected sparsity level at t -th iteration. Define also the incoherence of the matrix W as the largest off-diagonal inner product

$$\mu_W := \sup_{i \neq j} |[WW^H]_{ij}| \quad (16)$$

For the trained network each noisy input activates neurons at certain hidden layers. Accordingly, one can imagine a connectivity graph through which the input pixels would traverse different paths to reach and form the output. As discussed earlier in (11) due to skip connections there are a total of 2^T possible paths. Introduce the diagonal matrix B with the i -th diagonal element $[WW^H]_{ii} = \|W_i\|_2^2$, and the cascade of the activation masks $D_{\mathcal{I}} := D_{i_j} \dots D_{i_1}$. Define then the weighted *path sparsity*

$$p_{\mathcal{I}} = \mathbb{E}[\text{tr}(D_{\mathcal{I}} B^{|\mathcal{I}|})].$$

The following Lemma then bounds the deviation of the individual terms (associated with different paths) in the Jacobian expansion (11) from the neural network with orthogonal weight matrices.

Lemma 4. *For the trained network with the expected sparsity levels $\rho_i := \mathbb{E}[\text{tr}(D_i)]$ at i -th iteration, for the index subset (i_1, \dots, i_j) , it holds that*

$$\left| \mathbb{E}[\text{tr}(W^H D_{i_j} W W^H \dots W W^H D_{i_1} W)] - p_{\mathcal{I}} \right| \leq \prod_{l=1}^j [\sqrt{s_{i_l}} (s_{i_l} - 1) \mu_W]$$

Combining Lemma 4 with (11), the main result is established as follows.

Theorem 1. *For an unrolled network with a cascade of T recurrent single-layer residual units let $\{\rho_t = \mathbb{E}[\text{tr}(D_t)]\}_{t=1}^T$. If the network weights satisfy $\mu_W \leq \epsilon \max_t \rho_t^{-3/2}$ for a constant $\epsilon < 1$, the DOF is bounded as*

$$\left| \mathbb{E}[\text{tr}(J_T)] - n - \sum_{\mathcal{I}} (-1)^{|\mathcal{I}|} p_{\mathcal{I}} \right| \leq (1 + \epsilon)^T - 1 - \epsilon T.$$

Accordingly, one can adopt $n + \sum_{\mathcal{I}} (-1)^{|\mathcal{I}|} p_{\mathcal{I}}$ as a surrogate for DOF of the trained network.

4 Empirical Evaluations

Extensive experiments were performed to assess our findings for natural image denoising and deblurring. In particular, we aimed to address the following important questions:

- Q1. How would the RSS, DOF, and SURE behave empirically for WS and WC schemes?
- Q2. How would MSE/PSNR behave as a function of the train sample size for WS and WC schemes?
- Q3. What is the filtering interpretation of the learned denoisers for WS and WC schemes?

4.1 Network Architecture and Training

To address the above questions, we adopted a ResNet with 2 residual blocks (RBs) where each RB consists of two convolutional layers with 3×3 kernels and a fixed number (128) of feature maps, that were followed by batch normalization (BN) and ReLU activation. ResNet is used in the feature domain, and thus we add a convolutional layer with 3×3 kernels that lift up the image from previous iterations to 128 feature maps. Similarly, ResNet is followed by a convolutional layer with 1×1 kernels that lifts off the feature maps to create the next estimate. We used the Adam optimizer [31] with the momentum parameter 0.9 and initial learning rate varying across the experiments. Training was performed with TensorFlow and PyTorch interface on NVIDIA Titan X Pascal GPUs with 12GB RAM. PSNR (dB) is used as the figure of merit that is simply related to MSE as $PSNR = -10 \log_{10}(MSE)$ since the images are normalized.

4.2 Denoising

This section addresses Q1 and Q2 for natural image denoising task, where $\Phi = I$.

Dataset. 400 natural images of size 481×321 were selected from the Berkeley image segmentation dataset (a.k.a BSD68) [32]. Patches of size 40×40 were extracted as labels, resulting in 230, 400 training samples. 68 full images were chosen for test data.

The ResNet architecture described before was adopted with $K = 2$ RBs and $T = 4$ iterations. It is trained for 50 epochs with minibatch size 256. The initial learning rate was annealed by a factor of 10 at the 40-th epoch. We run experiments independently (with random initialization) for several initial learning rates, namely 0.0075, 0.005, 0.0025, 0.001, 0.00075, 0.0005, 0.00025, 0.0001, 0.000075, 0.00005, and pick the one leading to the best PSNR on test data during the last epoch. The aforementioned experiments were repeated for various noise levels $\sigma \in \{15, 25, 50, 100\}$. Moreover, the experiments were repeated with and without weight sharing.

We assess SURE, DOF, and RSS with sample sizes within the range $[10, 230400]$ (logarithmically spaced). It is first observed that the SURE estimate is in perfect agreement with the test MSE (or PSNR) when having the true labels available for validation purposes. We thus plot the PSNR evolution in Fig. 1 as the train sample size grows (orange line for WS and the blue line for WC). For all noise levels, we observe a consistent benefit for WS in sample sizes less than 1K. Interestingly, after 1K they coincide and no benefit is observed for WC even for very large sample sizes in the order of 10^5 . Note also that the non-smooth behavior of the curve is mainly attributed to Adam optimizer that may not necessarily converge to the globally optimum network weights.

Error bars for the individual SURE components including DOF and RSS are also plotted in Fig. 2, respectively. The upper (res. bottom) rows correspond to WC (res. WS). Fig. 2 depicts the evolution

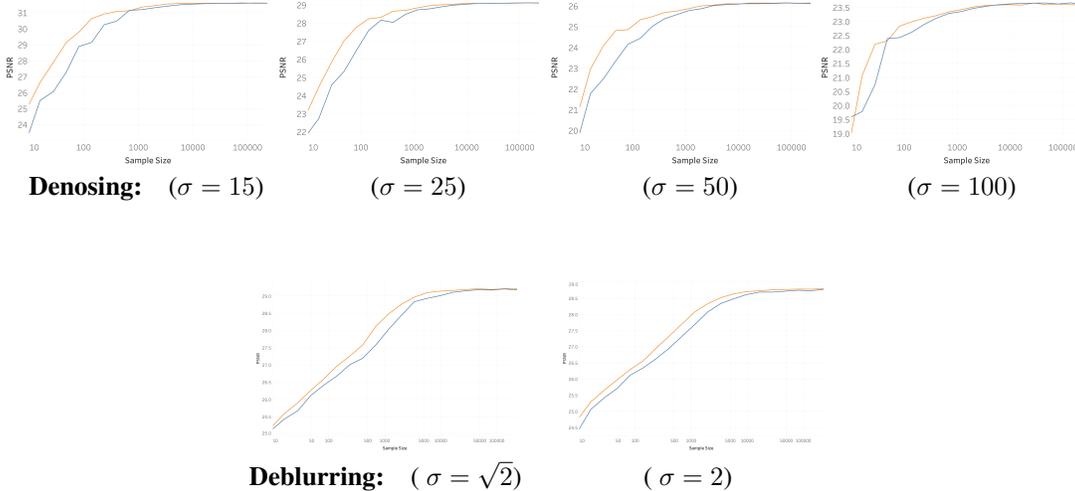


Figure 1: *Effects of weight sharing and sample size on the image denoising and deblurring performance*. The upper row corresponds to denoising and the lower row corresponds to image deblurring. Different columns correspond to different noise levels. Each panel depicts the test PSNR as function of train sample size. The orange line corresponds to WS and the blue line to WC.

of normalized RSS, namely $\frac{1}{\sigma^2} \|y - h_{\Theta}(y)\|_2^2$ over train sample size. Similarly, Fig. 2 plots the DOF $\nabla_y \cdot h_{\Theta}(y)$. The blue dots correspond to 68 test image samples. Box-and-whisker plots also depict RSS percentile. It is first observed that for both WS and WC scenario, DOF (res. RSS) tend to be increasing (res. decreasing) with the train sample size, where it finally saturates at a limiting value that is identical for both WS and WC. Interestingly, the limiting value coincides with the generalization MSE as per (7). The DOF for WS scheme, however, ramps off quickly, suggesting that fewer samples are required to construct the bases and attain the degrees of freedom embedded in the network. On the contrary, RSS would drop quickly for WS, which contributes to small SURE values.

Our second observation compares RSS and DOF values for WS and WC. It appears that, under different noise regimes, WS consistently achieves larger DOF. The achieved RSS however is much smaller which renders the overall SURE (or MSE) smaller for WS in low train sample complexity regimes. In addition, upon using sufficient train samples, RSS converges to unity for all noise levels, which corresponds to $\|y - h_{\Theta}(y; \Phi)\|_2^2 = n\sigma^2$. We explain this by noting that any sensible denoising algorithm should output estimated images within the noise level $n\sigma^2$ of the corrupted image.

4.3 Deblurring

This section addresses Q2 for natural image deblurring. The sensing matrix Φ is a linear operator that convolves an image with a Gaussian kernel of standard deviation 1.6. We prepared data as described for denoising, except that we extracted patches of size 50×50 . The same ResNet as in the denoising case was adopted to model the proximal. However, instead of encouraging data consistency with a gradient step as for denoising (see 2.1), the full least square problem is simply solved after each proximal step. As a result, the state variable update is modified as

$$s^{t+1} = (\Phi\Phi^H + \alpha I)^{-1} (\Phi^H y + \alpha x^t). \quad (17)$$

It is worth commenting that the approach of tackling a general image restoration problems by repeatedly denoising and solving a least-squares problem is a common practice in image processing; see e.g., [33, 21, 3].

We train the architecture in the same way described in the previous subsection. The experiments are repeated for two noise levels $\sigma \in \{\sqrt{2}, 2\}$ at different panels. Each panel depicts the test PSNR of neural proximal algorithm as a function of training sample size. The orange (res. blue) line corresponds to WS (res. WC). We observe a consistent benefit from using weight sharing in sample sizes smaller than 50K.

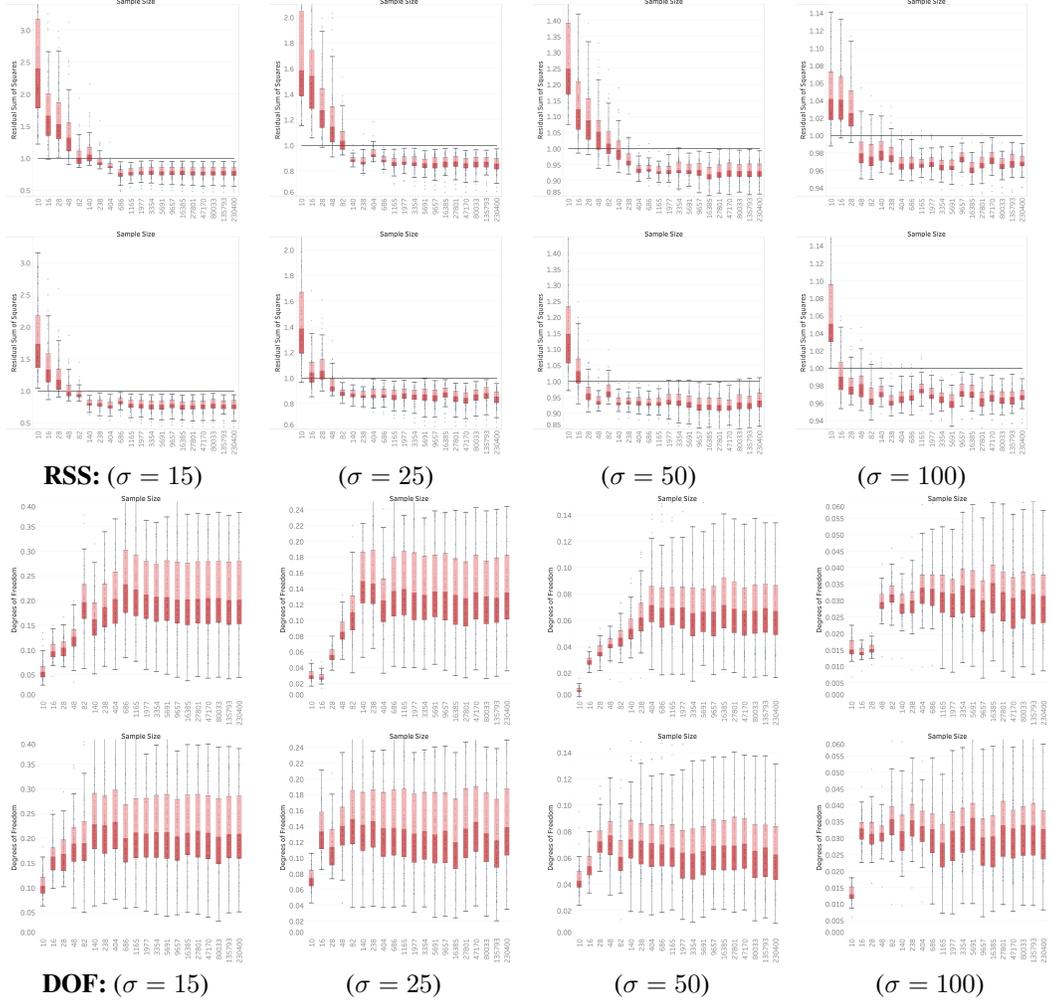


Figure 2: Effects of weight sharing and non-weight sharing and sample size on the RSS and DOF for denoising. Each column corresponds to a different noise level. The upper row corresponds to WC while the bottom to WS. Each upper panel depicts the normalized RSS $\frac{1}{\sigma_x^2} \|y - h_{\Theta}(y)\|_2^2$ as a function of train sample size. Each lower panel depicts the degrees of freedom $\nabla_y \cdot h_{\Theta}(y)$ of as a function of training sample size. The blue dots correspond to 68 test images on which the RSS was computed. The box and whisker depict the percentiles.

4.4 Compressed sensing MRI

To further investigate Q2, we consider also the task of compressed sensing [34] for MRI reconstruction. Looking at the linear model (1), compressed sensing (CS) assumes there are typically much less measurements than the unknown image pixels, i.e., $m \ll n$. A prime example for CS is reconstruction of MR images [25], that is widely adopted in the clinical scanners. In essence, the MR scanner acquires a fraction of Fourier coefficients (k-space data) of the underlying image across various coils. We focused on a single-coil MR acquisition model, where for a patient the acquired k-space data admits

$$y_{i,j} = [\mathcal{F}(x)]_{i,j}, \quad (i, j) \in \Omega \quad (18)$$

Here, \mathcal{F} refers to the 2D Fourier transform, and the set Ω indexes the sampled Fourier coefficients. Just as in conventional CS MRI, we selected Ω based on variable-density sampling with radial view ordering that is more likely to pick low frequency components from the center of k-space [25]. Only 20% of Fourier coefficients were collected.

Dataset. It includes 19 subjects scanned with a 3T GE MR750 whole body MR scanner. Fully sampled sagittal images were acquired with a 3D FSE CUBE sequence with proton density weighting including fat saturation. Other parameters include FOV=160mm, TR=1550 (sagittal) and 2,000

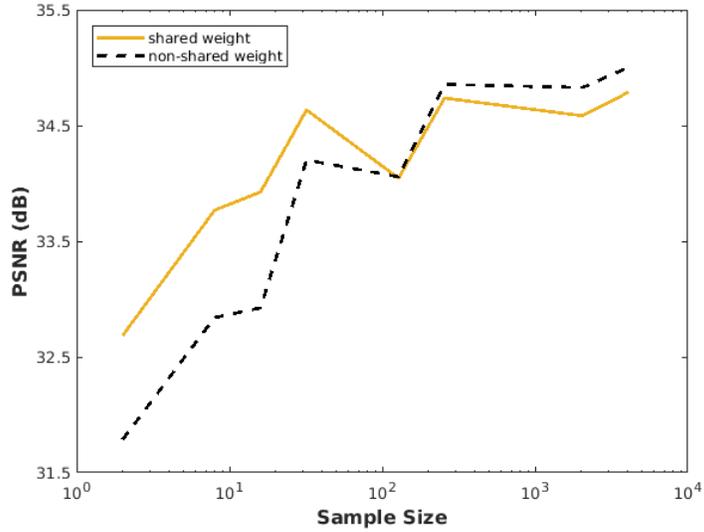


Figure 3: Effects of weight sharing and sample size on the performance for CS-MRI with 5-fold undersampling of the k -space data. The panel depicts the test PSNR as a function of training sample size. The orange line corresponds to the result obtained with WS and the blue line to WC.

(axial), TE=25 (sagittal) and 35 (axial), slice thickness 0.6mm (sagittal) and 2.5mm (axial). The dataset is publicly available at [35]. Each image is a complex valued 3D volume of size $320 \times 320 \times 256$. Axial slices of size 320×256 were considered as the input for train and test. 16 patients are used for training (5, 120 image slices) and 3 patients for test (960 image slices).

Neural proximal algorithm with $T = 5$ iterations was run with minibatch of size 4. For any train sample size, training is performed for various learning rates $3 \times \{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}\}$, choosing the one achieving the highest PSNR. The input and output were complex-valued images of the same size and each included two channels for real and imaginary components. The input image $x_0 = \Phi^H y$ was indeed the inverse 2D FFT of the k -space data where the missing frequencies were filled with zeros. It was thus severely contaminated with aliasing artifacts. The benefits of weight sharing for small sample sizes is evident, when using only a few MR images for training would lead to around 1 dB gain compared with the weight changing scheme. The gap kept decreasing with the train sample size and finally after 10^2 samples the weight changing scheme led to a slight improvement, possibly due to the larger representation capacity. Note also that compared with the denoising and deblurring experiments, the gap disappears for a smaller sample sizes, as train images are of large dimension with 320×256 training pixels.

4.5 Filtering interpretation of proximals

Along with Q3, it is of interest to *explain* how the cascade of learned proximals of WS and WC contribute to recover the input image from errors. To do so, we focus on the natural image denoising described in Section 4.2 with $T = 4$ proximal iterations. Recall the t -th proximal network mapping s_{t+1} to x_{t+1} through a ResNet with 2RBs. We focus on the first convolutional layer with 128 kernels collected as $\{f_{t,i}\}_{i=1}^{128}$ per iteration t . For visualization purposes, we propose to compute the two-dimensional Fourier transform for each of filters $\{f_{t,i}\}_{i=1}^{128}$ and then to sum over all filters the magnitude of the Fourier coefficients. We repeat this process for T iterations in WC case and the single set of filters in WS.

The results are shown in Fig. 4 for the noise level $\sigma = 100$. The first four panel columns represent the weights obtained in four iterations of WC network, while the fifth column represents WS. Each row corresponds to a different sample size. It is observed that for WS at high sample sizes the filters converge to a spectrum associated with a bandpass filter. The pattern observed for WC however is interesting; the odd iterations converge to a lowpass, while the even iterations converge to bandpass

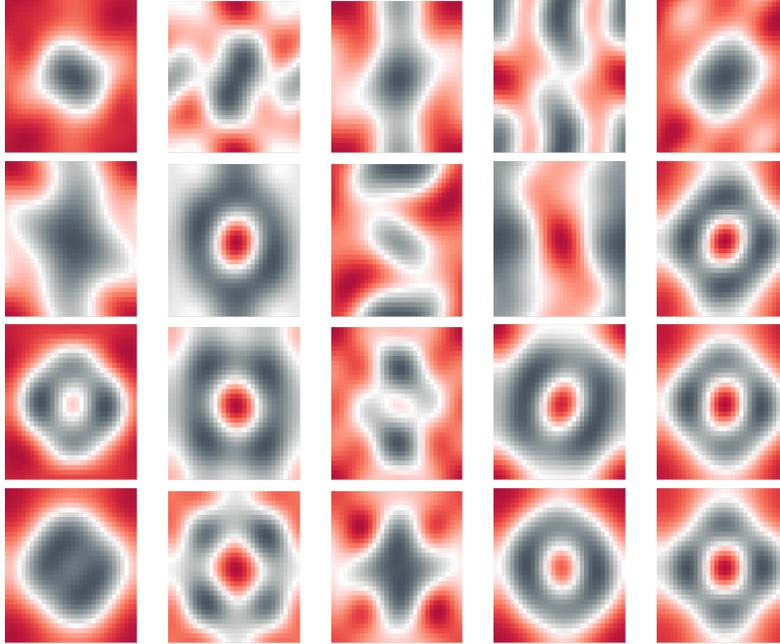


Figure 4: *Frequency visualization of the learned proximals in the shared and non-shared case.* The first four columns correspond to the weights obtained in the 4 iterations of the non-shared weights. The fifth column corresponds to the shared-weights case. Rows from top to bottom are for 10; 686; 5, 691; 135, 793 train samples. Each panel depicts the summation of the magnitudes of the 2D Fourier transforms of the filters. The noise level is fixed to $\sigma = 100$. The color coding associates large magnitudes to black, and small ones to red.

filters. This is reminiscent of the scattering transform [17, 18], where a cascade of lowpass and bandpass operations is applied.

In contrast with scattering networks however, our *learned* proximal network applies lowpass filtering followed by bandpass filtering. The scattering transform applies several lowpass filters, and then a highpass filter at the last layer. We also observe that the shared weights converge to the final spectrum at approximately 686 examples, while the non-shared case requires more than 5,691 samples to converge. Moreover, for WC case the filters in the first two iterations are very similar to their final versions already at 686 examples, meaning that the filters in the first iterations are trained first.

5 Conclusions

This paper investigates the generalization risk of unrolled neural networks appearing in image translation tasks. The Stein’s Unbiased Risk Estimator (SURE) is adopted as an estimate for the generalization risk, and the DOF term quantifying the prediction variance that is the trace of the end-to-end network Jacobian is analyzed. Under certain incoherence conditions on the train weight matrices, DOF is derived in terms of the weight path sparsity of the network. Extensive empirical evaluations are performed with natural images for image denoising and deblurring tasks. While the analysis are performed for the recurrent unrolled networks, non-recurrent networks are empirically tested for comparison. The observations indicate that the DOF increases with train sample size and converges to the generalization risk for both recurrent and non-recurrent schemes. In addition, the recurrent network converges significantly quicker (with less train samples) compared with non-recurrent scheme, hence recurrent scheme serves as a regularization for low sample size regimes. All in all, this is the first attempt to apply SURE for generalization risk analysis of unrolled neural networks.

There are still important avenues to explore that are left for future research. One such avenue pertains to extending the SURE analysis to arbitrary sensing matrices. Another one includes understanding the link between early stopping and weight sharing for training unrolled neural networks.

6 Appendix

6.1 Proof of Lemma 2

Upon defining the sample correlation matrix $C_x^N := \frac{1}{N} \sum_{i=1}^N x_i x_i^H$, the network weights come from the training process that optimizes

$$\begin{aligned} W &= \arg \min_W \frac{1}{N} \sum_{i=1}^N \|x_i - (I - \mathcal{P}_W)y_i\|^2 \\ &\stackrel{(a)}{\approx} \arg \min_W \frac{1}{N} \sum_{i=1}^N \|\mathcal{P}_W x_i\|^2 + \hat{\sigma}^2 \text{tr}((I - \mathcal{P}_W)) \\ &= \arg \min_W \text{tr}(\mathcal{P}_W(C_x - \hat{\sigma}^2 I)) \end{aligned}$$

where the approximation (a) comes from $\frac{1}{N} \sum_{i=1}^N v_i v_i^H \approx \hat{\sigma}^2 I$, and quickly approaches $\sigma^2 I$ for relatively large N .

Apparently, the training objective amounts to learning the bases for principal component analysis (PCA). The training process then tunes the network weights to the singular vectors of the sample correlation matrix. Let us decompose the sample correlation matrix as $C_x^N = U \Sigma U^H$. In essence, the optimal $W \in \mathbb{R}^{\ell \times n}$ is orthogonal with the rows that include the singular vectors $\{u_i\}$ where $\sigma_i^N \leq \hat{\sigma}^2$; if $\sigma_i^N > \hat{\sigma}^2$, we set the corresponding row to zero. For the end-to-end map $J_\infty = I - \mathcal{P}_W$ the DOF is then reduced to $n - \text{tr}(\mathcal{P}_W)$, and the result of Lemma 2 immediately follows.

6.2 Proof of Lemma 4

Let

$$Q = WW^H - \text{diag}(b) \quad (19)$$

then the diagonal of Q are all zeros, therefore

$$\text{tr}(D_i Q) = 0. \quad (20)$$

Let $\lambda_j(D_i Q)$ be eigenvalues of $D_i Q$, we rank the eigenvalues so that $\lambda_1(D_i Q)$ be the eigenvalue with largest absolute value, then $|\lambda_1(D_i Q)|$ is the spectral norm of the matrix as $\|D_i Q\|_2$.

Now we find upper bound on the spectral norm. Using Gershgorin circle theorem, there exist at least one index k such that

$$|\lambda_1(D_i Q)| \leq \sum_{j \neq k} |(D_i Q)_{kj}| \quad (21)$$

Since the diagonal entry of $D_i Q$ is zero, so

$$\mathbb{E} \|D_i Q\|_2 = \mathbb{E} |\lambda_1(Q)| \quad (22)$$

$$\leq \mathbb{E} \left[\sum_{j \neq 1} |[D_i Q]_{1j}| \right] \quad (23)$$

$$\leq (s_i - 1) \mu_W \quad (24)$$

Now we could bound all the following quantities using spectral norm,

$$\begin{aligned} \|D_i Q\|_F^2 &= \text{tr}(D_i Q Q^H D_i^H) \\ &= \left| \sum_{j=1}^{s_i} \lambda_j(D_i Q)^2 \right| \\ &\leq s_i \|D_i Q\|_2^2. \end{aligned}$$

Using inequality for trace of products of matrices [36], we have a bound

$$\begin{aligned} |\text{tr}(D_{i_j} Q \dots Q D_{i_1} Q)| &\leq \prod_{l=1}^j \|D_{i_l} Q\|_F \\ &\leq \prod_{l=1}^j \sqrt{s_{i_l}} \|D_{i_l} Q\|_2 \end{aligned}$$

6.3 Proof of Theorem 1

Plugging the result of Lemma 4 into to the Jacobian expansion in (10) and (11), we have the decomposition of Jacobian as follows

$$\mathbb{E}\text{tr}[J_T] = \mathbb{E}\left[\text{tr}\left(\prod_{\tau=0}^T (I - W^H D_{i_\tau} W)\right)\right] \quad (25)$$

$$= \mathbb{E}\left[\sum_{j=0}^T \sum_{i_1, \dots, i_j} (-1)^j \text{tr}(W^H D_{i_j} W W^H \dots W W^H D_{i_1} W)\right] \quad (26)$$

$$= \mathbb{E}\left[\text{tr}(I) - \sum_{i=0}^T \text{tr}(D_{T-\tau} W W^H) + \dots + \sum_{i_1, \dots, i_T} (-1)^T \text{tr}(W^H D_{i_T} W W^H \dots W W^H D_{i_1} W)\right] \quad (27)$$

After some rearrangements the deviation bound from the Lemma 4 would result in

$$\left| \mathbb{E}[\text{tr}(J_T)] - \left(n + \sum_{\mathcal{I}} (-1)^{|\mathcal{I}|} p_{\mathcal{I}}\right) \right| \quad (28)$$

$$\leq \sum \left| \mathbb{E}[\text{tr}(W^H D_{i_j} W W^H \dots W W^H D_{i_1} W)] - p_{\mathcal{I}} \right| \quad (29)$$

$$\leq \sum \prod_{l=1}^j [\sqrt{s_{i_l}} (s_{i_l} - 1) \mu_W] \quad (30)$$

$$\leq \binom{T}{2} \epsilon^2 + \dots + \binom{T}{T} \quad (31)$$

$$= (1 + \epsilon)^T - 1 - \epsilon T. \quad (32)$$

References

- [1] 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. Omnipress, 2010.
- [2] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.
- [3] Stanley H Chan, Xiran Wang, and Omar A Elgendy. Plug-and-play admm for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3(1):84–98, 2017.
- [4] Jian Sun, Huibin Li, Zongben Xu, et al. Deep ADMM-net for compressive sensing MRI. In *Advances in Neural Information Processing Systems*, pages 10–18, 2016.
- [5] Jonas Adler and Ozan Öktem. Learned primal-dual reconstruction. *arXiv preprint arXiv:1707.06474*, 2017.
- [6] Steven Diamond, Vincent Sitzmann, Felix Heide, and Gordon Wetzstein. Unrolled optimization with deep priors. *arXiv preprint arXiv:1705.08041*, 2017.
- [7] Chris Metzler, Ali Mousavi, and Richard Baraniuk. Learned D-AMP: Principled neural network based compressive image recovery. In *Advances in Neural Information Processing Systems*, pages 1770–1781, 2017.
- [8] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1637–1645, 2016.
- [9] Jo Schlemper, Jose Caballero, Joseph V. Hajnal, Anthony Price, and Daniel Rueckert. A deep cascade of convolutional neural networks for MR image reconstruction. In *Proceedings of the 25th Annual Meeting of ISMRM, Honolulu, HI, USA*, 2017.
- [10] Shreyas Vasawanala Vardan Papyan Hatef Monajemi John Pauly Morteza Mardani, Qingyun Sun and David Donoho. Neural proximal gradient descent for compressive imaging. In *Advances in Neural Information Processing Systems*, 2018.
- [11] Shakarim Soltanayev and Se Young Chun. Training deep learning based denoisers without ground truth data. In *Advances in Neural Information Processing Systems*, pages 3257–3267, 2018.
- [12] Christopher A Metzler, Ali Mousavi, Reinhard Heckel, and Richard G Baraniuk. Unsupervised learning with stein’s unbiased risk estimator. *arXiv preprint arXiv:1805.10531*, 2018.
- [13] Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. Compressed sensing using generative models. *arXiv preprint arXiv:1703.03208*, 2017.
- [14] Patrick Putzky and Max Welling. Recurrent inference machines for solving inverse problems. *arXiv preprint arXiv:1706.04008*, 2017.
- [15] Charles M Stein. Estimation of the mean of a multivariate normal distribution. *The annals of Statistics*, pages 1135–1151, 1981.
- [16] David L Donoho and Iain M Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the american statistical association*, 90(432):1200–1224, 1995.
- [17] Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013.
- [18] Joakim Andén and Stéphane Mallat. Deep scattering spectrum. *IEEE Transactions on Signal Processing*, 62(16):4114–4128, 2014.

- [19] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095, 2007.
- [20] Weisheng Dong, Lei Zhang, Guangming Shi, and Xin Li. Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing*, 22(4):1620–1630, 2013.
- [21] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 479–486. IEEE, 2011.
- [22] Singanallur V Venkatakrishnan, Charles A Bouman, and Brendt Wohlberg. Plug-and-play priors for model based reconstruction. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 945–948. IEEE, 2013.
- [23] Yaniv Romano, John Isidoro, and Peyman Milanfar. RAISR: rapid and accurate image super resolution. *IEEE Transactions on Computational Imaging*, 3(1):110–125, 2017.
- [24] Joan Bruna, Pablo Sprechmann, and Yann LeCun. Super-resolution with deep convolutional sufficient statistics. *arXiv preprint arXiv:1511.05666*, 2015.
- [25] Michael Lustig, David Donoho, and John M. Pauly. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine*, 58(6):1182–1195, December 2007.
- [26] Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3):127–239, 2014.
- [27] Yonina C Eldar. Generalized sure for exponential families: Applications to regularization. *IEEE Transactions on Signal Processing*, 57(2):471–481, 2009.
- [28] Bradley Efron. The estimation of prediction error: covariance penalties and cross-validation. *Journal of the American Statistical Association*, 99(467):619–632, 2004.
- [29] David L Donoho and Iain M Johnstone. Threshold selection for wavelet shrinkage of noisy data. In *Proceedings of 16th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 1, pages A24–A25. IEEE, 1994.
- [30] David Donoho, Matan Gavish, et al. Minimax risk of matrix denoising by singular value thresholding. *The Annals of Statistics*, 42(6):2413–2440, 2014.
- [31] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [32] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 416–423. IEEE, 2001.
- [33] Yaniv Romano, Michael Elad, and Peyman Milanfar. The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017.
- [34] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.
- [35] [online] <http://mridata.org/fullysampled/knees.html>.
- [36] Khalid Shebrawi and Hussien Albadawi. Trace inequalities for matrices. *Bulletin of the Australian Mathematical Society*, 87(1):139–148, 2013.