DENOISING AND REGULARIZATION VIA EXPLOITING THE STRUCTURAL BIAS OF CONVOLUTIONAL GENERATORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Convolutional Neural Networks (CNNs) have emerged as highly successful tools for image generation, recovery, and restoration. This success is often attributed to large amounts of training data. On the contrary, a number of recent experimental results suggest that a major contributing factor to this success is that convolutional networks impose strong prior assumptions about natural images. A surprising experiment that highlights this structural bias towards simple, natural images is that one can remove various kinds of noise and corruptions from a corrupted natural image by simply fitting (via gradient descent) a randomly initialized, over-parameterized convolutional generator to this single image. While this over-parameterized model can eventually fit the corrupted image perfectly, surprisingly after a few iterations of gradient descent one obtains the uncorrupted image, without using any training data. This intriguing phenomena has enabled state-of-the-art CNN-based denoising as well as regularization in linear inverse problems such as compressive sensing. In this paper we take a step towards demystifying this experimental phenomena by attributing this effect to particular architectural choices of convolutional networks, namely fixed convolutional operations. We then formally characterize the dynamics of fitting a two layer convolutional generator to a noisy signal and prove that early-stopped gradient descent denoises/regularizes. This results relies on showing that convolutional generators fit the structured part of an image significantly faster than the corrupted portion.

1 INTRODUCTION

Convolutional neural networks are extremely popular for image generation: The majority of image generating networks is convolutional, ranging from the Deep Convolutional Generative Adversarial Networks (DC-GANs) Radford et al. (2015) to the U-Net (Ronneberger et al., 2015). It is well known that convolutional neural networks incorporate implicit assumption about the signals they generate, such as pixels that are close being related. This makes them particularly well suited for representing sets of images or modeling distributions of images. It is less known, however, that those prior assumptions build into the architecture are so strong that convolutional neural networks are useful even without ever being exposed to training data.

The fact that convolutional networks are useful without ever being trained was first shown by the so called Deep Image Prior (DIP) experiment in Ulyanov et al. (2018). This paper demonstrated that over-paramterized convolutional neural networks can be used to perform image restoration tasks such as denoising with state-of-the-art performance based on the observation that un-trained convolutional auto-decoders fit a single natural image faster when optimized with gradient descent (i.e., with significantly fewer iterations) than pure noise. A more recent paper Heckel & Hand (2019) proposed a much simpler image generating network, termed the deep decoder. This network can be seen as the relevant part of a convolutional generator architecture to function as a prior, and can be obtained from the convolutional autoencoder architectures used in the paper Ulyanov et al. (2018) by removing the encoder, the skip connections, and the trainable convolutional filters of spacial extent larger than one. Crucially, the deep decoder does not use learned convolutional filters such as conventional convolutional networks, but has convolutions with fixed convolutional kernels through bi-linear upsampling operations, further high-lighting the importance of a particular architecture.

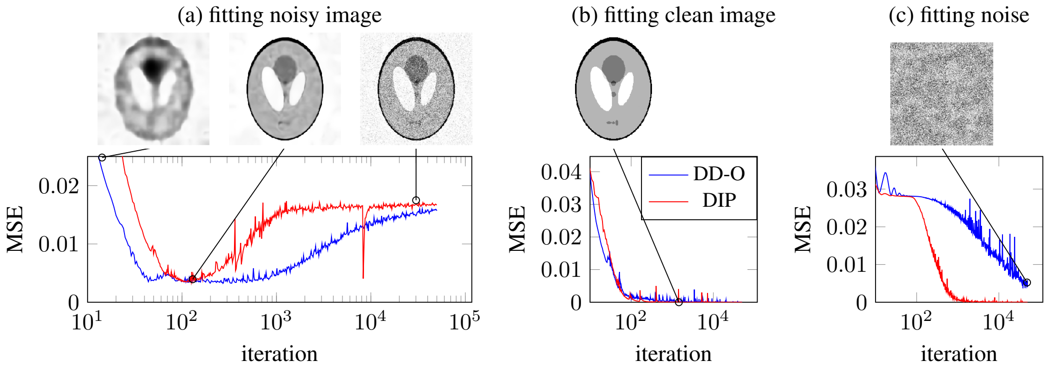


Figure 1: Fitting an over-parameterized Deep Decoder (DD-O) and the deep image prior (DIP) to a (a) noisy image, (b) clean image, and (c) pure noise. Here, MSE denotes Mean Square Error of the network output with respect to the clean image in (a) and fitted images in (b) and (c). While the network can fit the noise due to over-parameterization, it fits structured/natural images in significantly fewer iterations. Hence, when fitting a noisy image, the image is fitted faster than the noise leading to denoising via early stopping.

In this paper, we study a simple untrained convolutional network that only consists of convolutional-like operations, such as the deep image prior and the deep decoder. We consider the over-parameterized regime where the network has sufficiently many parameters to represent an arbitrary image (including noise) perfectly and show that:

Fitting convolutional generators via early stopped gradient descent provably denoises "natural" images.

To prove this denoising capability we characterize how the network architecture governs the dynamics of fitting over-parameterzed networks to a single (noisy) image. In particular we prove:

Randomly initialized convolutional generators optimized with gradient descent fit natural images faster than complex ones.

Here, by complex images, we mean unstructured images that consist of a large number of edges or variations such as noise.

We depict this phenomena in Figure 1 where we fit a randomly initialized over-parameterized convolutional generator to a signal via running gradient descent on the objective $\mathcal{L}(\mathbf{C}) = \|G(\mathbf{C}) - \mathbf{y}\|_2^2$. Here, $G(\mathbf{C})$ is the convolutional generator with weight parameters $\mathbf{C}$, and $\mathbf{y}$ is either a noisy image, a clean image, or noise. This experiment demonstrates that over-parameterized convolutional networks fit a structured/natural image (Figure 1b) much faster than pure noise (Figure 1c). Thus, when fitting the noisy image (Figure 1a), early stopping the optimization enables image denoising. Interestingly, this effect is so strong that it gives state-of-the-art denoising performance, outperforming the BM3D algorithm (Dabov et al., 2007), which is the next-best method that requires no training data (see Figure 2). Beyond denoising, this effect also enables significant improvements in regularizing a variety of inverse problems such as compressive sensing Veen et al. (2018); Heckel (2019).

1.1 Contributions and overview of results

In this paper we take a step towards demystifying why fitting a convolutional generator via early stopped gradient descent leads to such surprising denoising and regularization capabilities.

- We show experimentally that this denoising phenomena can be primarily attributed to fixed convolutional filters, typically implicitly implemented by bi-linear upsampling operations in the convolutional network.
- We then study over-parameterized convolutional networks (with fixed convolutional filters) theoretically. Specifically, we show that fitting such a network via early stopped gradient

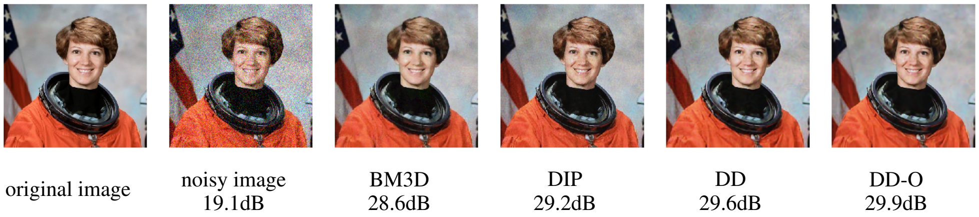


Figure 2: Denoising with BM3D and various convolutional generators. The relative ranking of algorithms is representative and maintained on a large test set of images. DIP and DD-O are over-parameterized generators and with early stopping outperform the BM3D algorithm, the next best method that does not require additional training data.

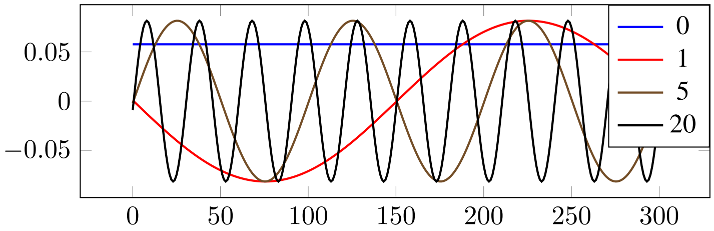


Figure 3: The 0st, 1st, 5th, and 20th trigonometric basis functions in dimension $n = 300$.

descent to a signal provably denoises it. Specifically, let $\mathbf{x} \in \mathbb{R}^n$ be a smooth signal that can be represented exactly as a linear combination of the p orthogonal trigonometric functions of lowest frequency (defined in equation (4), see Figure 3 for a depiction), and suppose our goal is to obtain an estimate of this signal from a noisy signal $\mathbf{y} = \mathbf{x} + \mathbf{z}$ where $\mathbf{z} \sim \mathcal{N}(0, \frac{\varsigma^2}{n}\mathbf{I})$. Note that natural images are often smooth. Let $\hat{\mathbf{y}} = G(\mathbf{C}^t)$ be the estimate obtained from early stopping the fitting of a two-layer convolutional generator to $\mathbf{y}$. We prove this estimate achieves the denoising rate

$$\|\hat{\mathbf{y}} - \mathbf{x}\|_2^2 \leq c\frac{p}{n}\varsigma^2,$$

with c a constant. We note that this rate is optimal up to a constant factor.

- Our denoising result follows from a detailed analysis of gradient descent applied to fitting a two-layer convolutional generator $G(\mathbf{C})$ with fixed convolutional filters to a noisy signal $\mathbf{y} = \mathbf{x} + \mathbf{z}$ with $\mathbf{x}$ representing the signal and $\mathbf{z}$ the noise. Specifically, let $\widetilde{\mathbf{x}} \in \mathbb{R}^n$ and $\widetilde{\mathbf{z}} \in \mathbb{R}^n$ be the coefficients of the signal and noise in terms of trigonometric basis functions $\mathbf{w}_1, \ldots, \mathbf{w}_n \in \mathbb{R}^n$ (precisely defined later, see Figure 3 for a depiction). We show that there is a dual filter $\sigma \in \mathbb{R}^n$, depending only on the convolutional filter used, whose entries typically obey $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n > 0$ and can be thought of as weights associated with each of those basis function. These weights in turn determine the speed at which the associated components of the signal are fitted. Specifically, we show that the dynamics of gradient descent are approximately given by

$$G(\mathbf{C}_\tau) - \mathbf{x} \approx \underbrace{\sum_{i=1}^{n} \mathbf{w}_i \widetilde{x}_i (1 - \eta\sigma_i^2)^\tau}_{\text{error in fitting signal}} + \underbrace{\sum_{i=1}^{n} \mathbf{w}_i \widetilde{z}_i ((1 - \eta\sigma_i^2)^\tau - 1)}_{\text{fit of noise}}.$$

The filters commonly used are typically such that the dual filter decays rather quickly, implying that low-frequency components in the trigonometric expansion are fitted significantly faster than high frequency ones. So when the signal mostly consists of low-frequency components, we can choose an early stopping time such that the error in fitting the signal is very low, and thus the signal part is well described, whereas at the same time only a small part of the noise, specifically the part aligned with the low-frequency components has been fitted.

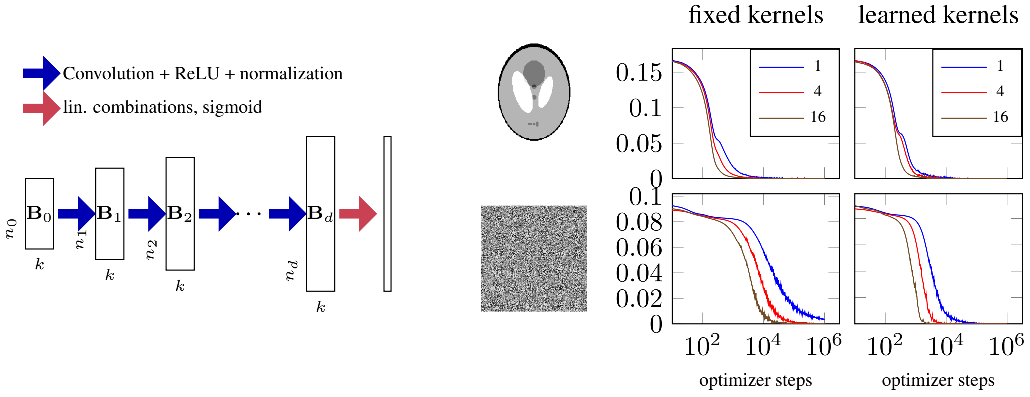


Figure 4: **Left panel:** Convolutional generators. The output is generated through repeated convolutional layers, channel normalization, and applying ReLU non-linearities. **Right panel:** Fitting the phantom MRI and noise with different architectures of depth $d = 5$, for different number of over-parameterization factors (1,4, and 16). Gradient descent on convolutional generators involving fixed convolutional matrixes fit an image significantly faster than noise.

2 Convolutional generators

A convolutional generator maps an input tensor $\mathbf{B}_0$ to an image only using upsampling/no-upsampling and convolutional operations, followed by channel normalization (a special case of batch normalization) and applications of non-linearities, see Figure 4. All previously mentioned convolutional generator networks Radford et al. (2015); Ronneberger et al. (2015) including the networks considered in the DIP paper Ulyanov et al. (2018) primarily consist of those operations. Before discussing the specifics of the convolutional generators studied in this paper we first demonstrate in Section 2.1 that fixed convolutional generators are critical to the denoising performance with early stopping. In particular, we empirically demonstrate that what makes them fit natural or simple images significantly faster than complex images or noise are the presence of *fixed* convolutional kernels. Then in Section 2.2 we formally introduce the convolutional generators studied in this paper. Finally, in Section 2.3 we introduce a minimal convolutional architecture which will be the focus of our theoretical results.

2.1 The importance of fixed convolutional filters

In this section we demonstrate that using fixed convolutional kernel is critical for achieving good denoising performance. To do this we begin by noting that the upsampling operation present in most architectures actually implicitly incorporates a convolution with a *fixed* convolutional (interpolation) filter. Precisely this operation is critical for the phenomena that structured/natural images are fitted significantly faster than noise. To see this, consider the experiment in Figure 4 in which we fit an image and noise by minimizing the least-squares loss via gradient descent with i) a convolutional generator with only fixed convolutional filters (see Section 2.2 below for a precise description) and ii) a conventional convolutional generator where the convolutional filters are also learned (essentially the architecture from the popular DC-GAN generators, see Appendix A for details and additional numerical evidence). These results show that a generator with *fixed filters* fits an image much faster than noise, whereas the network with learned convolutional filters, only fits it slightly faster, and the effect goes away as the network becomes highly overparameterized. Thus, fixed convolutional filters are what enables un-trained convolutional networks to function as highly effective image priors.

2.2 Architecture of convolutional generator with fixed convolutions

In the previous section we found that convolutional generators with *fixed* convolutional operators simple/structured images significantly faster than complex images or noise. In these architectures (e.g. deep decoder Heckel & Hand (2019)), the channels in the $(i+1)$-th layer are given by

$$\mathbf{B}_{i+1} = \mathrm{cn}(\mathrm{ReLU}(\mathbf{U}_i \mathbf{B}_i \mathbf{C}_i)), \quad i = 0, \ldots, d-1,$$

and finally, the output of the d-layer network is formed as

$$\mathbf{x} = \mathbf{B}_d \mathbf{C}_{d+1}.$$

Here, the coefficient matrices $\mathbf{C}_i \in \mathbb{R}^{k \times k}$ and $\mathbf{C}_{d+1} \in \mathbb{R}^{k \times k_{\text{out}}}$ contain the weights of the network. Note the number of channels, k, determines the number of weight parameters of the network, given by $dk^2 + k_{\text{out}}k$. Each column of the tensor $\mathbf{B}_i\mathbf{C}_i \in \mathbb{R}^{n_i \times k}$ is formed by taking linear combinations of the channels of the tensor $\mathbf{B}_i$ in a way that is consistent across all pixels. Then, $\mathrm{cn}(\cdot)$ performs the channel normalization operation which normalizes each channel individually and can be viewed as a special case of the popular batch normalization operation (Ioffe & Szegedy, 2015).

The operator $\mathbf{U}_i \in \mathbb{R}^{n_{i+1} \times n_i}$ is a tensor implementing an upsampling and most importantly a convolution operation with a fixed kernel. This fixed kernel was chosen in all experiments above as a triangular kernel so that $\mathbf{U}$ performs bi-linear 2x upsampling (this is the standard implementation in the popular packages pytorch and tensorflow). As mentioned earlier this convolution with a *fixed kernel* is critical for fitting natural images faster than complex ones.

2.3 Two layer convolutional generator studied theoretically in this paper

The simplest model to study the denoising capability of convolutional generators and the phenomena that a natural image is fitted faster than a complex one theoretically is a network with only one hidden layers and one output channel i.e., $G(\mathbf{C}) \in \mathbb{R}^n$. Then, the generator becomes

$$G(\mathbf{C}) = \mathrm{ReLU}(\mathbf{U}\mathbf{B}_1\mathbf{C}_1)\mathbf{c}_2,$$

where $\mathbf{U} \in \mathbb{R}^{n \times n}$ is a circulant matrix that implements a convolution with a filter $\mathbf{u}$. In this paper we consider the highly over-parameterized regime where $k \geq n$. In this regime, using that the input $\mathbf{B}_1$ is random, with probability one, the matrix $\mathbf{B}_1$ has full column rank and thus spans $\mathbb{R}^n$. It follows that optimizing over $\mathbf{C}_1$ and $\mathbf{c}_2$ is equivalent to optimizing over the parameter $\mathbf{C} \in \mathbb{R}^{n \times k}$ in

$$G(\mathbf{C}) = \mathrm{ReLU}(\mathbf{U}\mathbf{C})\mathbf{v}, \tag{1}$$

where $\mathbf{v} = [1, \ldots, 1, -1, \ldots, -1]/\sqrt{k}$ is fixed and $\mathbf{C} \in \mathbb{R}^{n \times k}$ is the new coefficient matrix we optimize over. Figure **??** in the appendix shows that even for this simple model, the convolutional generator fits a simple image faster than pure noise. This is the simplest model in which the phenomena that a convolutional networks fits structure faster than noise can reliably observed. This is consistent with our previous finding that fixed convolutional operations (for example due to upsampling) are at the heart of the effect. As a consequence, the dynamics of training the model (1) is the focus of the remainder of this paper.

3 Warmup: Dynamics of gradient descent on least squares

As a prelude for fitting convolutional generators via a *non-linear* least square problem, in this section we study the dynamics of gradient descent applied to a *linear* least squares problem which corresponds to a linear network. We demonstrate how early stopping can lead to denoising capabilities even in this simple linear model. Specifically, we consider a least-squares problem of the form

$$\mathcal{L}(\mathbf{c}) = \frac{1}{2}\|\mathbf{y} - \mathbf{J}\mathbf{c}\|_2^2.$$

We study gradient descent with a constant step size η starting at $\mathbf{c}_0 = \mathbf{0}$. In this case the updates are given by

$$\mathbf{c}_{\tau+1} = \mathbf{c}_\tau - \eta\nabla\mathcal{L}(\mathbf{c}_\tau), \quad \nabla\mathcal{L}(\mathbf{c}) = \mathbf{J}^T(\mathbf{J}\mathbf{c} - \mathbf{y}).$$

The following simple proposition characterizes the trajectory of gradient descent.

Proposition 1. *Let $\mathbf{J} \in \mathbb{R}^{n \times m}$ whose left singular vectors are given by a matrix $\mathbf{W} \in \mathbb{R}^{n \times n}$ $\mathbf{J}$ and the corresponding singular values given by $\sigma_1 \geq \sigma_2 \geq \ldots, \sigma_n$. Then the residual after τ steps, $\mathbf{r}_\tau = \mathbf{y} - \mathbf{J}\mathbf{c}_\tau$, takes the form*

$$\mathbf{r}_\tau = \sum_{i=1}^{n} \mathbf{w}_i \langle \mathbf{w}_i, \mathbf{y} \rangle (1 - \eta\sigma_i^2)^\tau.$$

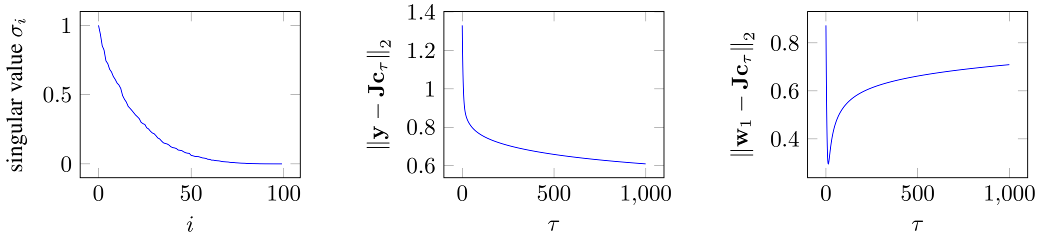


Figure 5: Gradient descent on the least squares problem of minimizing $\|\mathbf{y} - \mathbf{J}\mathbf{c}_\tau\|^2$, where $\mathbf{J} \in \mathbb{R}^{100\times 100}$ has decaying singular values (left panel) and the observation is the sum of a signal component, equal to the leading singular vector $\mathbf{w}_1$ of $\mathbf{J}$, and a noisy component $\mathbf{z} \sim \mathcal{N}(0, (1/n)\mathbf{I})$, i.e., $\mathbf{y} = \mathbf{w}_1 + \mathbf{z}$. The signal component $\mathbf{w}_1$ is fitted significantly faster than the other components (right panel), thus early stopping enables denoising.

Suppose that the observation $\mathbf{y}$ lies in the column span of $\mathbf{J}$, and that the stepsize is chosen sufficiently small ($\eta \leq 1/\|\mathbf{J}\|^2$). Then (by Proposition 1) gradient descent converges to a zero-loss solution and thus fits the signal perfectly. More interestingly, the proposition shows that gradient descent fits the components of $\mathbf{y}$ corresponding to large singular values faster than it fits the components corresponding to small singular values.

To explicitly show how this simple observation enables regularization via early stopped gradient descent, suppose our goal is to find a good estimate of a signal $\mathbf{x}$ from a noisy observation

$$\mathbf{y} = \mathbf{x} + \mathbf{z},$$

where the signal $\mathbf{x}$ lies in a signal subspace that is spanned by the p leading left-singular vectors of $\mathbf{J}$. In this case, by Proposition 1, the signal estimate after τ iterations, $\mathbf{J}\mathbf{c}_\tau$, obeys

$$\|\mathbf{J}\mathbf{c}_\tau - \mathbf{x}\|_2 \leq (1 - \eta\sigma_p^2)^\tau \|\mathbf{x}\|_2 + E(\mathbf{z}), \quad E(\mathbf{z}) := \sqrt{\sum_{i=1}^{n} ((1 - \eta\sigma_i^2)^\tau - 1)^2 \langle \mathbf{w}_i, \mathbf{z}\rangle^2}\,. \qquad (2)$$

Thus, after a few iterations most of the signal has been fitted (i.e., $(1 - \eta\sigma_p)^\tau$ is small). Furthermore, if we assume that the ratio σ_{p+1}/σ_p is sufficiently large so that the spread between the two singular values separating the signal subspace from the rest is sufficiently large, most of the noise outside the signal subspace has not been fitted (i.e., $((1 - \eta\sigma_i^2)^\tau - 1)^2 \approx 0$ for $i = p+1, \ldots, n$). In particular suppose the noise vector has a Gaussian distribution given by $\mathbf{z} \sim \mathcal{N}(0, \frac{\varsigma^2}{n}\mathbf{I})$. Then $E(\mathbf{z}) \approx \varsigma\sqrt{\frac{p}{n}}$ so that after order $\tau = \log(\epsilon)/\log(1 - \eta\sigma_p^2)$ iterations, with high probability,

$$\|\mathbf{J}\mathbf{c}_\tau - \mathbf{x}\|_2 \leq \epsilon \|\mathbf{x}\|_2 + c\varsigma\sqrt{\frac{p}{n}}.$$

This demonstrates, that provided the signal lies in a subspace spanned by the leading singular vectors, early stoped gradient descent reaches the optimal denoising rate of $\varsigma\sqrt{p/n}$ after a few iterations. See Figure 5 for a numerical example demonstrating this phenomena.

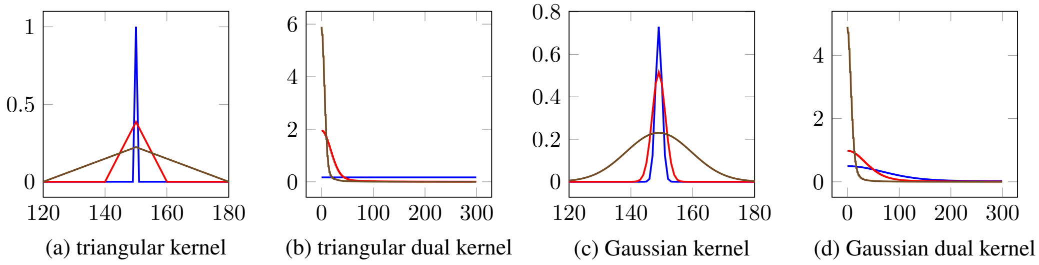
4 Dynamics of gradient descent on convolutional generators

In this section we study the implicit bias of gradient descent towards natural/structured images theoretically. Consider a two-layer decoder network (introduced in Section 2.3) of the form

$$G(\mathbf{C}) = \mathrm{ReLU}(\mathbf{U}\mathbf{C})\mathbf{v},$$

with the ReLU activation function given by $\mathrm{ReLU}(t) = \max(0, t)$, $\mathbf{v} = [1, \ldots, 1, -1, \ldots, -1]/\sqrt{k}$, and with weight parameter $\mathbf{C} \in \mathbb{R}^{n\times k}$, and recall that $\mathbf{U}$ is a circulant matrix implementing a convolution with a kernel $\mathbf{u} \in \mathbb{R}^n$. We consider the least squares objective

$$\mathcal{L}(\mathbf{C}) = \frac{1}{2}\|\mathbf{y} - G(\mathbf{C})\|_2^2 \qquad (3)$$



(a) triangular kernel (b) triangular dual kernel (c) Gaussian kernel (d) Gaussian dual kernel

Figure 6: Triangular and Gaussian kernels and the weights associated to low-frequency trigonometric functions they induce for a generator network of output dimension $n = 300$. The wider the kernels are, the more the weights are concentrated towards the low-frequency components of the signal.

and study early-stopped gradient descent with a constant stepsize η starting at a random initialization of the weights applied to this non-linear least squares problem.

In our warmup section on linear least squares we saw that the singular vectors and values of the feature matrix $\mathbf{J}$ determine the speed at which different components of the noisy signal $\mathbf{y}$ are fitted by gradient descent. The main insight that allows us to extend this intuition to the nonlinear case is that the role of the feature matrix $\mathbf{J}$ can be replaced with the Jacobian of the generator defined as $\mathcal{J}(\mathbf{C}) := \frac{\partial}{\partial \mathbf{C}} G(\mathbf{C})$. Contrary to the linear least squares problem however, in the nonlinear case, the Jacobian is not constant and changes across iterations. Nevertheless, we show that the eigenvalues/vectors of the Jacobian at the random initialization govern the dynamics of fitting the network throughout the iterative updates. For the two layer convolutional generator of the form studied in this paper the left eigenvectors of the Jacobian mapping can be well approximated throughout the updates. Interestingly, the form of these eigenvectors only depends on the network architecture and not the convolutional kernel used. We formalize these eigenvectors below.

Definition 1. *Define the trigonometric basis functions* $\mathbf{w}_1, \ldots, \mathbf{w}_n$

$$[\mathbf{w}_i]_j = \frac{1}{\sqrt{n}} \begin{cases} 1 & k = 0 \\ \sqrt{2}\cos(2\pi ji/n) & k = 1, \ldots, n/2-1 \\ (-1)^j & k = n/2 \\ \sqrt{2}\sin(2\pi ji/n) & k = n/2+1, \ldots, n-1 \end{cases} . \tag{4}$$

As mentioned earlier, Figure 3 depicts some of these eigenvectors.

In addition to the left eigenvectors we can also approximate the spectrum of the Jacobian throughout the updates (for the purposes of denoising) by a dual filter/kernel that only depends on the original filter/kernel used in the network architecture. We formalize this dual kernel below.

Definition 2 (Dual kernel)**.** *Associated with a kernel* $\mathbf{u} \in \mathbb{R}^n$ *we define the dual kernel* $\sigma \in \mathbb{R}^n$ *as*

$$\sigma = \|\mathbf{u}\|_2 \sqrt{\left| \mathbf{F} g\left(\frac{\mathbf{u} \circledast \mathbf{u}}{\|\mathbf{u}\|_2^2} \right) \right|} \quad \textit{with} \quad g(t) = \frac{1}{2}\left(1 - \frac{\cos^{-1}(t)}{\pi}\right) t.$$

Here, for two vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$, $\mathbf{u} \circledast \mathbf{v}$ *denotes their circular convolution, the scalar non-linearity* g *is applied entrywise, and* $\mathbf{F}$ *is the discrete Fourier transform matrix.*

In Figure 6, we depict two commonly used interpolation kernels $\mathbf{u}$, namely a triangular and a Gaussian kernel (recall that the standard upsampling operator is a convolution with a triangle), along with the induced dual kernel σ. The figure shows that the dual filter σ induced by these kernels is concentrated with a few large values associated with the low frequency trigonometric functions, and with the rest of the values being rather small. This in turn implies that the Jacobian spectrum throughout training decreases rapidly.

With these definitions in place we are ready to state our main denoising results. We note that we have slightly simplified the statement of this result for exposition purposes. The appendix contains more detailed statements at the expense of easy interpretability.

Theorem 1 (Denoising with early stopping)**.** *Consider a noisy signal $\mathbf{y} \in \mathbb{R}^n$ given by $\mathbf{y} = \mathbf{x} + \mathbf{z}$, with $\mathbf{x} \in \mathbb{R}^n$ denoting the signal and $\mathbf{z} \in \mathbb{R}^n$ the noise. Assume that $\mathbf{x}$ lies in the span of the first p trigonometric basis functions, and that the noise is Gaussian with distribution $\mathcal{N}(\mathbf{0}, \frac{\varsigma^2}{n}\mathbf{I})$. To denoise this signal we fit a two layer generator network with k channels of the form $G(\mathbf{C}) = \mathrm{ReLU}(\mathbf{U}\mathbf{C})\mathbf{v}$ with $\mathbf{U}$ a circulant matrix obtained from a convolutional kernel $\mathbf{u} \in \mathbb{R}^n$. Let $\sigma \in \mathbb{R}^n$ denote the associated dual kernel associated and $\hat{\mathbf{u}}$ denote the discrete Fourier transform of $\mathbf{u}$. Also assume the number of channels obeys*

$$k \geq C_{\mathbf{u}} n \tag{5}$$

with $C_{\mathbf{u}}$ a constant only depending on the convolutional kernel $\mathbf{u}$. We fit the neural generator $G(\mathbf{C})$ to the noisy signal $\mathbf{y} \in \mathbb{R}^n$ by minimizing the least-squares loss (3) *via running gradient descent iterations $\mathbf{C}_{\tau+1} = \mathbf{C}_\tau - \eta \nabla \mathcal{L}(\mathbf{C}_\tau)$, starting from $\mathbf{C}_0$ with i.i.d. $\mathcal{N}(0, \omega^2)$, entries, $\omega \propto \frac{\|\mathbf{y}\|_2}{\sqrt{n}}$, and step size obeying $\eta \leq \frac{1}{\|\hat{\mathbf{u}}\|_\infty^2}$. Then, with probability at least $1 - e^{-k^2} - \frac{1}{n^2}$, at stopping time $\tau = \log(1 - \sqrt{p}/n)/\log(1 - \eta\sigma_{p+1}^2)$, the reconstruction error is bounded by*

$$\|G(\mathbf{C}_\tau) - \mathbf{x}\|_2 \leq (1 - \eta\sigma_p^2)^\tau \|\mathbf{x}\|_2 + C\varsigma\sqrt{\frac{p}{n}}, \tag{6}$$

for C a fixed numerical constant.

Note that for this choice of stopping time, provided that dual kernel decays sharply around the p-th singular value, the first term in the bound (6) (i.e., $(1 - \eta\sigma_p^2)^\tau \approx 0$) essentially vanishes and the error bound becomes $O(\varsigma\sqrt{\frac{p}{n}})$. The dual kernel decays sharply around the leading eigenvalues provided the kernel is for example a sufficiently wide triangular or Gaussian kernel (see Figure 6).

This result demonstrates that when the noiseless signal $\mathbf{x}$ is sufficiently structured (e.g. contains only the p lowest frequency components in the trigonometric basis) and the convolutional generator has sufficiently many channels, then early stopped gradient descent achieves a near optimal denoising performance proportional to $\varsigma\sqrt{\frac{p}{n}}$. This theorem is obtained from a more general result stated in the appendix which characterizes the evolution of the reconstruction error obtained by the convolutional generator.

Theorem 2 (Reconstruction dynamics of convolutional generators)**.** *Consider the setting and assumptions of Theorem 1 but now with a fixed noise vector $\mathbf{z}$, and without an explicit stopping time. Then, for all iterates τ obeying $\tau \leq \frac{1}{\eta\sigma_p^2}$ and provided that $k \geq C_{\mathbf{u}} n/\epsilon^4$, for some $\epsilon > 0$, with probability at least $1 - e^{-k^2} - \frac{1}{n^2}$, the reconstruction error obeys*

$$\|G(\mathbf{C}_\tau) - \mathbf{x}\|_2 \leq (1 - \eta\sigma_p^2)^\tau \|\mathbf{x}\|_2 + \sqrt{\sum_{i=1}^n ((1 - \eta\sigma_i^2)^\tau - 1)^2 \langle \mathbf{w}_i, \mathbf{z} \rangle^2} + \epsilon.$$

This theorem characterizes the reconstruction dynamics of convolutional generators throughout the updates. In particular, it helps explains why convolutional generators fit a natural signal significantly faster than noise, and thus early stopping enables denoising and regularization. To see this note that as mentioned previously each of the basis functions $\mathbf{w}_i$ have a (positive) weight, singular value, or dual kernel element $\sigma_i > 0$ associated with them that only depend on the convolutional kernel used in the architecture (through the definition of the dual kernel). These weights determine how fast the different components of the noisy signal is fitted by gradient descent. As we demonstrated earlier in Figure 6 for typical convolutional filters those weights decay very quickly from low to high frequency basis functions. As a result when the signal $\mathbf{x}$ is sufficiently structured (i.e. lies in the range of the p trigonometric functions with lowest frequencies), after a few iterations most of the signal is fitted (i.e., $(1 - \eta\sigma_p^2)^\tau$ is small), while most of the noise has not been fitted (i.e., $((1 - \eta\sigma_i^2)^\tau - 1)^2 \approx 0$ for $i = p + 1, \ldots, n$). Thus, early stopping achieves denoising.

4.1 The spectrum of the Jacobian for multilayer networks

Our theoretical results show that for single hidden-layer networks, the leading singular vectors of the Jacobian are the trigonometric functions throughout all iterations, and that the associated weights,

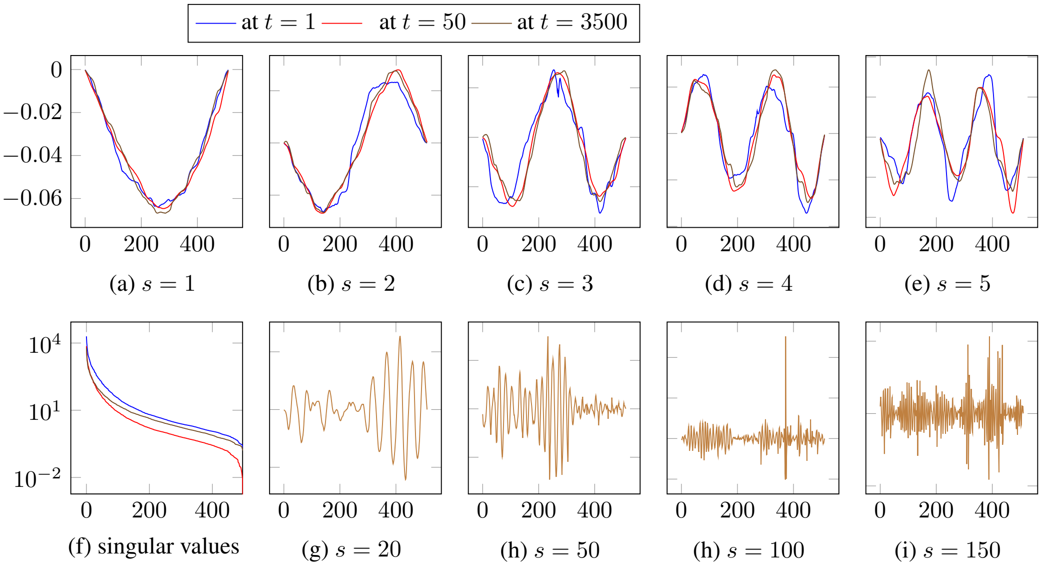


Figure 7: The Singular value distribution of the Jacobian of a four-layer deep decoder after $t = 50$ and $t = 3500$ iterations of gradient descent (panel (f)), along with the corresponding singular vectors/function. The singular functions corresponding to the large singular vectors are close to the low-frequency Fourier modes and do not change significantly through training.

singular values, or dual kernel values are concentrated towards the low frequency components. In this section, we show that for a multilayer network, the spectrum of the Jacobian is concentrated towards singular vectors/functions that are similar to the low-frequency components. We also show that throughout training those functions do vary, albeit the low frequency components do not change significantly and the spectrum remains concentrated towards the low frequency components. This shows that the implications of our theory continue to apply to muli-layer networks.

In more detail, we take a standard one dimensional deep decoder with $d = 4$ layers with output in $\mathbb{R}^{512}$ and with $k = 64$ channels in each layer. Recall that the standard one dimensional decoder obtains layer $i+1$ from layer i by linearly combining the channels of layer i with learnable coefficients followed by linear upsampling (which involves convolution with the triangular kernel $[1/2, 1, 1/2]$). The number of parameters is $d \times k^2 = 32 \cdot 512$, so the network is over-parameterized by a factor of 32. In Figure 7, we display the singular values as well as the leading singular vectors/function of the Jacobian at initialization ($t = 1$) and after $t = 50$ and $t = 3500$ iterations of gradient descent. As can be seen the leading singular vectors ($s = 1-6$) are close to the trigonometric basis functions and do not change dramatically throughout training. The singular vectors corresponding to increasingly smaller singular values ($s = 20, 50, 100, 150$) contain increasingly higher frequency components but are far from the high-frequency trigonometric basis functions.

5 Related literature

As mentioned before, the DIP paper Ulyanov et al. (2018) was the first to show that over-parameterized convolutional network enable solving denoising, inpainting, and super-resolution problems well even without any training data. Subsequently, the paper Heckel & Hand (2019) proposed a much simpler image generating network, termed the deep decoder. Subsequently the papers Veen et al. (2018); Heckel (2019); Jagatap & Hegde (2019) have shown that the DIP and the deep decoder also enables solving/regularizing compressive sensing problems and other inverse problems.

Since the convolutional generators considered here are image-generating deep networks, our work is also related to methods that rely on trained deep image models. Deep learning based methods are either trained end-to-end for tasks ranging from compression (Toderici et al., 2016; Agustsson et al., 2017; Theis et al., 2017; Burger et al., 2012; Zhang et al., 2017) to denoising (Burger et al., 2012; Zhang et al., 2017), or are based on learning a generative image model (by training an autoencoder or GAN (Hinton & Salakhutdinov, 2006; Goodfellow et al., 2014)) and then using the resulting model to solve inverse problems such as compressed sensing (Bora et al., 2017; Hand & Voroninski, 2018), denoising (Heckel et al., 2018), or phase retrieval (Hand et al., 2018; Shamshad & Ahmed, 2018), by minimizing an associated loss. In contrast to the method studied here, where the optimization is over the weights of the network, in all the aforementioned methods, the weights are adjusted only during training and then are fixed upon solving the inverse problem.

Our technical proofs rely on showing that the dynamics of gradient descent on an over-parameterized network can be related to that of a linear network or a kernel problem. This proof technique has been utilized in a variety of recent publication Du et al. (2018); Oymak & Soltanolkotabi (2019); Arora et al. (2019); Oymak et al. (2019). Two recent publication have used this proof technique to show that functions are learned at different rates: Basri et al. (2019) have shown that functions of different frequencies are learned at different speeds, and Arora et al. (2019) has provided a theoretical explanation of the empirical observation that a simple 2-layer network fits random labels slower than actual labels in the context of classification. None of these publications however study denoising, convolutional generators, or focus on the effect of early stopping. A recent publication Li et al. (2019) focuses on demonstrating how early stopping leads to robust classification in the presence of label corruption under a cluster model for the input data. Neither of the aforementioned publication however, does address denoising in a regression setting or fitting convolutional generators of the form studied in this paper.

References

Eirikur Agustsson, Fabian Mentzer, Michael Tschannen, Lukas Cavigelli, Radu Timofte, Luca Benini, and Luc V Gool. Soft-to-hard vector quantization for end-to-end learning compressible representations. In *Advances in Neural Information Processing Systems*, pp. 1141–1151, 2017.

Sanjeev Arora, Simon S. Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *International Conference on Machine Learning*, 2019.

Ronen Basri, David Jacobs, Yoni Kasten, and Shira Kritchman. The convergence rate of neural networks for learned functions of different frequencies. *arXiv:1906.00425 [cs, eess, stat]*, 2019.

A. Bora, A. Jalal, E. Price, and A. G. Dimakis. Compressed sensing using generative models. *arXiv:1703.03208*, 2017.

H. C. Burger, C. J. Schuler, and S. Harmeling. Image denoising: Can plain neural networks compete with BM3d? In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2392–2399, 2012.

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007.

Amit Daniely, Roy Frostig, and Yoram Singer. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. In *Advances in Neural Information Processing Systems*, pp. 2253–2261, 2016.

Simon S. Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. In *International Conference on Learning Representations*, 2018.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pp. 2672–2680. 2014.

Paul Hand and Vladislav Voroninski. Global guarantees for enforcing deep generative priors by empirical risk. In *Conference on Learning Theory*, 2018. arXiv:1705.07576.

Paul Hand, Oscar Leong, and Vladislav Voroninski. Phase retrieval under a generative prior. *arXiv preprint arXiv:1807.04261*, 2018.

Reinhard Heckel. Regularizing linear inverse problems with convolutional neural networks. *arXiv:1907.03100 [cs, math, stat]*, 2019.

Reinhard Heckel and Paul Hand. Deep decoder: Concise image representations from untrained non-convolutional networks. In *ICLR*, 2019.

Reinhard Heckel, Wen Huang, Paul Hand, and Vladislav Voroninski. Deep denoising: Rate-optimal recovery of structured signals with a deep prior. *arXiv:1805.08855*, 2018.

G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pp. 448–456, 2015.

Gauri Jagatap and Chinmay Hegde. Algorithmic guarantees for inverse imaging with untrained network priors. *arXiv:1906.08763 [cs, stat]*, 2019.

Mingchen Li, Mahdi Soltanolkotabi, and Samet Oymak. Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. *arXiv:1903.11680 [cs, stat]*, 2019. URL `http://arxiv.org/abs/1903.11680`.

Samet Oymak and Mahdi Soltanolkotabi. Towards moderate overparameterization: Global convergence guarantees for training shallow neural networks. *arXiv:1902.04674*, 2019.

Samet Oymak, Zalan Fabian, Mingchen Li, and Mahdi Soltanolkotabi. Generalization guarantees for neural networks via harnessing the low-rank structure of the jacobian. *arXiv:1906.05392*, 2019.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv:1511.06434 [cs]*, 2015.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. Lecture Notes in Computer Science, 2015.

Fahad Shamshad and Ali Ahmed. Robust compressive phase retrieval via deep generative priors. *arXiv preprint arXiv:1808.05854*, 2018.

Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. Lossy image compression with compressive autoencoders. *arXiv:1703.00395*, 2017.

George Toderici, Sean M. OMalley, Sung Jin Hwang, Damien Vincent, David Minnen, Shumeet Baluja, Michele Covell, and Rahul Sukthankar. Variable rate image compression with recurrent neural networks. In *International Conference on Learning Representations*, 2016.

Joel A. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, Aug 2011.

D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. In *Conference on Computer Vision and Pattern Recognition*, 2018.

Dave Van Veen, Ajil Jalal, Mahdi Soltanolkotabi, Eric Price, Sriram Vishwanath, and Alexandros G. Dimakis. Compressed sensing with deep image prior and learned regularization. *arXiv:1806.06438 [cs, math, stat]*, 2018.

K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.

A A NUMERICAL STUDY OF THE IMPLICIT BIAS OF CONVOLUTIONAL NETWORKS

To demonstrate that fixed convolutional filters (such as those in the bi-linear upsampling operation) are what matters for our effect, we study numerically the following four closely related architectural choices, which differ in the upsampling/no-upsampling and convolutional operations which generate the activations in the $(i+1)$-st layer, $\mathbf{B}_{i+1}$, from the activations in the i-th layer, $\mathbf{B}_i$:

i) **Bilinear upsampling and linear combinations.** Layer $i+1$ is obtained by linearly combining the channels of layer i with learnable coefficients (i.e., performing one-times-one convolutions), followed by bi-linear upsampling. This is the deep decoder architecture from (Heckel et al., 2018).

ii) **Fixed interpolation kernels and linear combinations.** Layer $i+1$ is obtained by linearly combining the channels of layer i with learnable coefficients followed by convolving each channel with the same 4x4 interpolation kernel that is used in the linear upsampling operator.

iii) **Parameterized convolutions:** Layer $i+1$ is obtained from layer i though a convolutional layer.

iv) **Deconvolutional network:** Layer $i+1$ is obtained from layer i though a deconvolution layer. This is essentially the DC-GAN generator architecture.

To emphasize that architectures i)-iv) are structurally extremely similar operations, we recall that each operation consists only of upsampling and convolutional operations. Let $\mathbf{T}(\mathbf{c})\colon \mathbb{R}^n \to \mathbb{R}^n$ be the convolutional operator with kernel $\mathbf{c}$, let $\mathbf{u}$ the linear upsampling kernel (equal to $\mathbf{u} = [0.5, 1, 0.5]$ in the one-dimensional case), and let $\mathbf{U}\colon \mathbb{R}^n \to \mathbb{R}^{2n}$ be an upsampling operator, that in the one dimensional case transforms $[x_1, x_2, \ldots, x_n]$ to $[x_1, 0, x_2, 0, \ldots, x_n, 0]$. In each of the architectures i)-iv), the ℓ-th channel of layer $i+1$ is obtained from the channels in the i-th layer as: $\mathbf{b}_{i+1,\ell} = \mathrm{ReLU}\left(\sum_{j=1}^{k} \mathbf{M}(\mathbf{c}_{ij\ell})\mathbf{b}_i\right)$, where the linear operator $\mathbf{M}$ is defined as follows for the four architectures

$$\text{i) } \mathbf{M}(c) = c\mathbf{T}(\mathbf{u})\mathbf{U}, \quad \text{ii) } \mathbf{M}(c) = c\mathbf{T}(\mathbf{u}), \quad \text{iii) } \mathbf{M}(\mathbf{c}) = \mathbf{T}(\mathbf{c}), \quad \text{iv) } \mathbf{M}(\mathbf{c}) = \mathbf{T}(\mathbf{c})\mathbf{U}\,.$$

The coefficients associated with the i-th layer are given by $\mathbf{C}_i = \{\mathbf{c}_{ij\ell}\}$, and all coefficients of the networks are $\mathbf{C} = \{\mathbf{c}_{ij\ell}\}$. Note that here, the coefficients or parameters of the networks are the weights and not the input to the network.

A.1 DEMONSTRATING IMPLICIT BIAS OF CONVOLUTIONAL GENERATORS

We next show that convolutional generators, in particular those with *fixed* convolutional operations, fit natural or simple images significantly faster than complex images or noise. Throughout this section, for each image or signal $\mathbf{x}^*$ we fit weights by minimizing the loss

$$\mathcal{L}(\mathbf{C}) = \|G(\mathbf{C}) - \mathbf{x}^*\|_2^2$$

with respect to the network parameters $\mathbf{C}$ using plain gradient descent with a fixed stepsize.

In order to exemplify the effect, we fit the phantom MRI image as well as noise for each of the architectures above for a 5-layer network. We choose the number of channels, k, such that the over-parameterization factor (i.e., the ratio of number of parameters of the network over the output dimensionality) is $1, 4$, and 16, respectively. The results show that for architectures i) and ii) involving *fixed convolutional operations*, gradient descent requires more than one order of magnitude fewer iterations to obtain a good fit of the phantom MRI image relative to noise. For architectures ii) and iii), with trainable convolutional filters, we see a smaller effect, but the effect essentially vanishes when the network is highly over-parameterized.

This effect continues to exist for natural images in general, as demonstrated by Figure 10 which depicts the average and standard deviation of the loss curves of 100 randomly chosen images from the imagenet dataset.

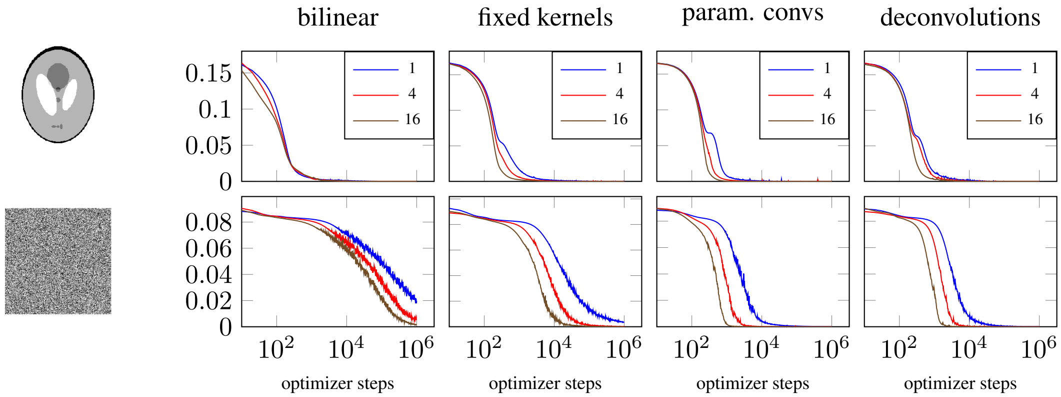


Figure 8: Fitting the phantom MRI and noise with different architectures of depth $d = 5$, for different number of over-parameterization factors (1,4, and 16). Gradient descent on convolutional generators involving fixed convolutional matrixes fit an image significantly faster than noise.

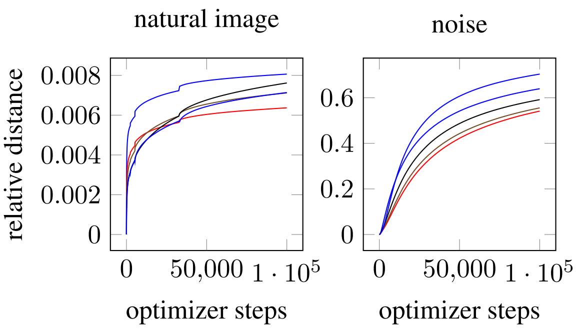


Figure 9: The relative distances of the weights in each layer from its random initialization. The weights need to change significantly more to fit the noise, compared to an image, thus a natural image lies closer to a random initialization than noise.

We also note that the effect continues to exist in the sense that highly structured images with a huge number of discontinuities are difficult to fit. An example is the checkerboard image in which each pixel alternates between 1 and 0; this image leads to the same loss curves as noise.

In our next experiment, we highlight that the distance between final and initial network weights is a key feature that determines the difference of fitting a natural image and noise. Towards this goal, we again fit the phantom MRI image and noise for the architecture i) and an over-parameterization factor of 4 and record, for each layer i the relative distance $\|\mathbf{C}_i^{(t)} - \mathbf{C}_i^{(0)}\| / \|\mathbf{C}_i^{(0)}\|$, where $\mathbf{C}_i^{(0)}$ are the weights at initialization (we initialize randomly), and $\mathbf{C}_i^{(t)}$ are the weights at the optimizer step t. The plot shows that to fit the noise, the weights have to change significantly, while for fitting a natural image they only change very slightly.

B The spectrum of the Jacobian of the deep decoder and deep image prior

Our theoretical results predict that for over-parameterized networks, the parts of the signal that is aligned with the leading singular vectors of the Jacobian at initialization is fitted fastest. In this section we briefly show that natural images are much more aligned with the leading singular vectors than with Gaussian noise, which is equally aligned with each of the singular vectors.

Towards this goal, we compute the norm of the product of the Jacobian at a random initalization, $\mathbf{C}_0$, with a signal $\mathbf{y}$ as this measures the extent to which the signal is aligned with the leading singular

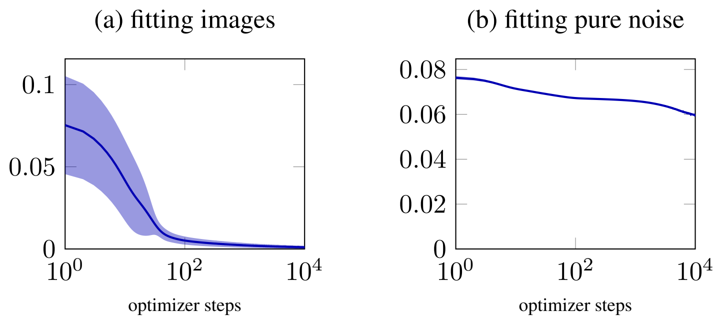
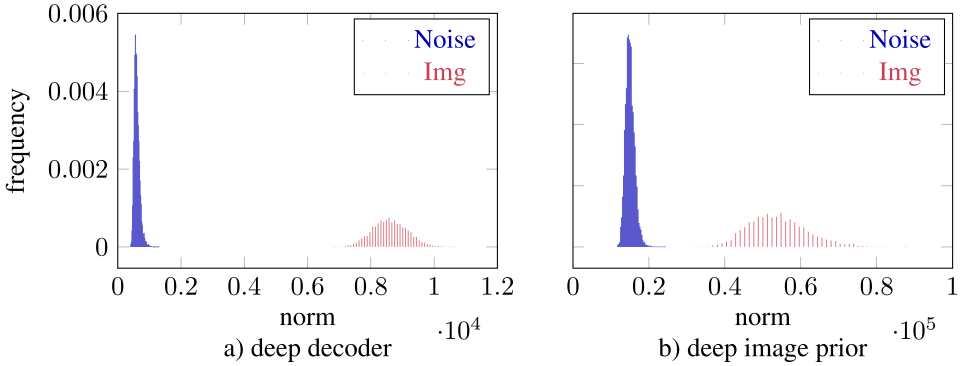


Figure 10: The loss curves for architecture i), a convolutional generator with linear upsampling operations, averaged over 100 $3\times512\times512$ (color) images from the Imagenet dataset. Convolutional generators fit natural images significantly faster than noise.



a) deep decoder

b) deep image prior

Figure 11: The distribution of the norm of the product of the Jacobian at initialization, $\mathcal{J}(\theta_0)$, with an image and noise of equal norm. For both the deep decoder and the deep image prior, the norm of product of the Jacobian and the noise (i.e., $\left\|\mathcal{J}^T(\theta_0)\mathbf{z}\right\|$) is significantly smaller than that with a natural image (i.e., $\left\|\mathcal{J}^T(\theta_0)\mathbf{y}^*\right\|$), it follows that a structured image is much better aligned with the leading singular vectors of the Jacobian than the noise. Thus, the Jacobian has an approximate low-rank structure, with natural images lying in the space spanned by the leading singularvectors.

vectors, due to

$$\left\|\mathcal{J}^T(\mathbf{C}_0)\mathbf{y}\right\|^2 = \left\|\mathbf{V}\Sigma\mathbf{W}^T\mathbf{y}\right\|^2 = \sum_i \sigma_i^2 \left\langle \mathbf{w}_i, \mathbf{y}\right\rangle^2,$$

where $\mathcal{J}(\mathbf{C}_0) = \mathbf{W}\Sigma\mathbf{V}^T$ is the singular value decomposition of the Jacobian.

Figure 11 depicts the distribution of the norm of the product of the Jacobian at initialization, $\mathcal{J}(\mathbf{C}_0)$, with an image $\mathbf{y}^*$ or noise $\mathbf{z}$ of equal norm ($\|\mathbf{y}^*\| = \|\mathbf{z}\|$). Since for both the deep decoder and the deep image prior, the norm of product of the Jacobian and the noise (i.e., $\left\|\mathcal{J}^T(\mathbf{C}_0)\mathbf{z}\right\|$) is significantly smaller than that with a natural image (i.e., $\left\|\mathcal{J}^T(\mathbf{C}_0)\mathbf{y}^*\right\|$), it follows that a structured image is much better aligned with the leading singular vectors of the Jacobian than the Gaussian noise, which is approximately equally aligned with any of the singular vectors. Thus, the Jacobian at random initialization has an approximate low-rank structure, with natural images lying in the space spanned by the leading singularvectors.

C Proofs and formal statement of results

The results stated in the main text are obtained from a slightly more general result which applies beyond convolutional networks. Specifically, we consider neural network generators of the form

$$G(\mathbf{C}) = \mathrm{ReLU}(\mathbf{U}\mathbf{C})\mathbf{v}, \tag{7}$$

with $\mathbf{C} \in \mathbb{R}^{n\times k}$, and $\mathbf{U} \in \mathbb{R}^{n\times n}$ an arbitrary fixed matrix, and $\mathbf{v} \in \mathbb{R}^k$, with half of the entries of $\mathbf{v}$ equal to $+1/\sqrt{k}$ and the other half equal to $-1/\sqrt{k}$.

The (transposed) Jacobian of $\sigma(\mathbf{U}\mathbf{c})$ is $\mathbf{U}^T \mathrm{diag}(\sigma'(\mathbf{U}\mathbf{c}))$. Thus the Jacobian of $G(\mathbf{C})$ is given by

$$\mathcal{J}^T(\mathbf{C}) = \begin{bmatrix} v_1 \mathbf{U}^T \mathrm{diag}(\mathrm{ReLU}'(\mathbf{U}\mathbf{c}_1)) \\ \vdots \\ v_k \mathbf{U}^T \mathrm{diag}(\mathrm{ReLU}'(\mathbf{U}\mathbf{c}_k)) \end{bmatrix} \in \mathbb{R}^{nk\times n}, \tag{8}$$

where ReLU' is the derivative of the activation function. Next we define the neural tangent kernel associated with this generator.

Definition 3 (Neural Tangent Kernel (NTK))**.** *Associated with a generator $G(\mathbf{C})$ we define the neural tangent kernel*

$$\mathbf{\Sigma}(\mathbf{U}) := \mathbb{E}\left[\mathcal{J}(\mathbf{C})\mathcal{J}^T(\mathbf{C})\right],$$

where expectation is over $\mathbf{C}$ with iid $\mathcal{N}(0,1)$ entries.

Consider the eigenvalue decomposition of the NTK given by

$$\mathbf{\Sigma}(\mathbf{U}) = \sum_{i=1}^{n} \sigma_i^2 \mathbf{w}_i \mathbf{w}_i^T.$$

Our results depend on the largest and smallest eigenvalue of the NTK, defined throughout as

$$\alpha = \sigma_n^2, \quad \beta = \sigma_1^2 = \|\mathbf{U}\|.$$

With these definitions in place we are now ready to state our result about neural generators.

Theorem 3. *Consider a noisy signal $\mathbf{y} \in \mathbb{R}^n$ given by*

$$\mathbf{y} = \mathbf{x} + \mathbf{z},$$

where $\mathbf{x} \in \mathbb{R}^n$ is assumed to lie in the signal subspace spanned by the p leading singular vectors $\mathbf{w}_1, \ldots, \mathbf{w}_n$ of $\mathbf{\Sigma}(\mathbf{U})$, and $\mathbf{z} \in \mathbb{R}^n$ is an arbitrary noise vector. Suppose that the number of channels obeys

$$k \geq 2^{34} n (\beta/\alpha)^{28} \xi^{-4} (\beta/\alpha + \xi\Gamma)^2 \tag{9}$$

for an error tolerance $0 < \xi \leq \sqrt{8\log\left(\frac{2n}{\delta}\right)}$ and for a constant Γ obeying $1 \leq \Gamma \leq 32\frac{\beta^6}{\xi\alpha^6}$. We fit the neural generator $G(\mathbf{C})$ to the noisy signal $\mathbf{y} \in \mathbb{R}^n$ by minimizing a loss of the form

$$\mathcal{L}(\mathbf{C}) = \frac{1}{2}\|G(\mathbf{C}) - \mathbf{y}\|_2^2. \tag{10}$$

via running gradient descent iterations $\mathbf{C}_{\tau+1} = \mathbf{C}_\tau - \eta\nabla\mathcal{L}(\mathbf{C}_\tau)$, starting from $\mathbf{C}_0$ with i.i.d. $\mathcal{N}(0,\omega^2)$, entries, $\omega = \frac{\|\mathbf{y}\|_2}{\sqrt{n}\beta}\xi\frac{\alpha^2}{\beta^2}$, and step size obeying $\eta \leq 1/\beta^2$. Then, with probability at least $1 - e^{-k^2} - \delta$, for all iterations $\tau \leq \frac{\Gamma}{\eta\alpha^2}$,

$$\|\mathbf{x} - G(\mathbf{C}_\tau)\| \leq (1 - \eta\sigma_p^2)^\tau \|\mathbf{x}\|_2 + \sqrt{\sum_{i=1}^{n} ((1 - \eta\sigma_i^2)^\tau - 1)^2 \langle \mathbf{w}_i, \mathbf{z}\rangle^2} + 2\xi\frac{\alpha^2}{\beta^2}\sqrt{8\log(2n/\delta)}\|\mathbf{y}\|_2. \tag{11}$$

C.1 Proof of Theorems 1 and 2

Theorems 1 and 2 stated in the main text follow directly from the theorem above as follows. We first note that for a convolutional generator (where $\mathbf{U}$ implements a convolution and thus is circulant) the eigenvectors of the NTK are given by the trigonometric basis functions per Definition 1 and the eigenvalues are the square of the entries of the dual kernel (Definition 2). To see this, we note that as detailed in Section G, the neural tangent kernel is given by

$$[\mathbf{\Sigma}(\mathbf{U})]_{ij} = \frac{1}{2}\left(1 - \cos^{-1}\left(\frac{\langle \mathbf{u}_i, \mathbf{u}_j \rangle}{\|\mathbf{u}_i\| \|\mathbf{u}_j\|}\right)/\pi\right)\langle \mathbf{u}_i, \mathbf{u}_j \rangle. \tag{12}$$

Since the matrix $\mathbf{U}$ implements a convolution with a kernel $\mathbf{u}$ that is equal to its first column, the neural tangent kernel $\mathbf{\Sigma}(\mathbf{U})$ is again a circulant matrix and is also Hermitian. Thus its spectrum is given by the Fourier transform of the first column of the circulant matrix, and its left-singular vectors are given as cosines and sines of different frequencies, as depicted in Figure 3.

Furthermore, using the fact that the eigenvalues of a circulant matrix are given by its discrete Fourier transform we can substitute $\beta = \|\mathbf{U}\| = \|\hat{\mathbf{u}}\|_\infty$ and $\alpha = \sigma_n$ to conclude that $C_{\mathbf{u}} \propto \left(\frac{\|\hat{\mathbf{u}}\|_\infty}{\sigma_n}\right)^{30}$. This yields Theorem 2.

Finally, we note that to obtain the simplified final expression in Theorem 1 we also used the fact that for a Gaussian vector $\mathbf{z}$, the vector $\mathbf{W}^T\mathbf{z}$ is also Gaussian. Furthermore, by the concentration of Lipschitz functions of Gaussians with high probability we have

$$\begin{aligned}\sum_{i=1}^{n}((1-\eta\sigma_i^2)^\tau - 1)^2 \langle \mathbf{w}_i, \mathbf{z}\rangle^2 &\approx \mathbb{E}\left[\sum_{i=1}^{n}((1-\eta\sigma_i^2)^\tau - 1)^2 \langle \mathbf{w}_i, \mathbf{z}\rangle^2\right] \\ &\overset{\text{(i)}}{=} \frac{\varsigma^2}{n}\sum_{i=1}^{n}((1-\eta\sigma_i^2)^\tau - 1)^2 \\ &\overset{\text{(ii)}}{\leq} \varsigma^2\frac{2p}{n}.\end{aligned}$$

Here, (i) follows from $\langle \mathbf{w}_i, \mathbf{z}\rangle$ being zero mean Gaussian with variance ς^2/n (since $\mathbf{z} \sim \mathcal{N}(0, (\varsigma^2/n)\mathbf{I})$, and $\|\mathbf{w}_i\|_2 = 1$). Finally, (ii) follows by choosing the early stop time so that it obeys $\tau \leq \log(1-\sqrt{p}/n)/\log(1-\eta\sigma_{p+1})$ which in turn implies that $(1-\eta\sigma_i^2)^\tau \geq 1-\sqrt{p}/n$, for all $i > p$, yielding that $((1-\eta\sigma_i^2)^\tau - 1)^2 \leq p/n^2$, for all $i > p$.

D The dynamics of linear and nonlinear least-squares

The results stated in the main text for convolutional generators build on a more general result on the dynamics of a non-linear least squares problem which applies beyond convolutional networks. We start by proving this result and then derive our main result on neural network-based denoising, specifically Theorem 3. We then show how Theorems 1 and 2 in the main body follow from Theorem 3.

In this section we discuss results for gradient descent applied to a nonlinear least-squares problem. Specifically, we focus on denoising/removing corruption from a corrupted image/signal $\mathbf{y} \in \mathbb{R}^n$ via a nonlinear least-squares fitting problem of the form

$$\mathcal{L}(\theta) = \frac{1}{2}\|f(\theta) - \mathbf{y}\|_2^2.$$

Here, $f\colon \mathbb{R}^p \to \mathbb{R}^n$ is a non-linear model with parameters $\theta \in \mathbb{R}^p$. To solve this problem, we run gradient descent with a fixed stepsize η, starting from an initial point θ_0 with updates of the form

$$\theta_{\tau+1} = \theta_\tau - \eta\nabla\mathcal{L}(\theta_\tau) \quad \text{where} \quad \nabla\mathcal{L}(\theta) = \mathcal{J}^T(\theta)(f(\theta) - \mathbf{y}). \tag{13}$$

Here, $\mathcal{J}(\theta) \in \mathbb{R}^{n\times p}$ is the Jacobian associated with the nonlinear map f with entries given by $[\mathcal{J}(\theta)]_{i,j} = \frac{\partial f_i(\theta)}{\partial \theta_j}$. To study the properties of the gradient descent iterates (13) we will connect the non-linear least squares problem with a linear one around the initialization. We note that similar strategies have been utilized in a variety of recent publications Du et al. (2018); Oymak &

Soltanolkotabi (2019); Oymak et al. (2019). In particular, we will associate the following linearized least-squares problem with the non-linear least squares problem

$$\mathcal{L}_{\text{lin}}(\theta) = \frac{1}{2} \|f(\theta_0) + \mathbf{J}(\theta - \theta_0) - \mathbf{y}\|_2^2 . \tag{14}$$

Here, $\mathbf{J} \in \mathbb{R}^{n \times p}$, refered to as the reference Jacobian, is a fixed matrix independent of θ that approximates the Jacobian mapping at initialization, $\mathcal{J}(\theta_0)$, and will be formally defined later on in this section. Starting from the same initial point θ_0, gradient descent updates of this linearized problem take the form

$$\begin{aligned} \widetilde{\theta}_{\tau+1} &= \widetilde{\theta}_\tau - \eta \mathbf{J}^T \left(f(\theta_0) + \mathbf{J}(\widetilde{\theta}_\tau - \theta_0) - \mathbf{y} \right) \\ &= \widetilde{\theta}_\tau - \eta \mathbf{J}^T \mathbf{J} \left(\widetilde{\theta}_\tau - \theta_0 \right) - \eta \mathbf{J}^T \left(f(\theta_0) - \mathbf{y} \right). \end{aligned} \tag{15}$$

To show that the non-linear updates (13) are close to the linearized iterates (15), we make the following assumptions:

Assumption 1 (Bounded spectrum). *We assume the singular values of the reference Jacobian obey*

$$\alpha \le \sigma_n \le \sigma_1 \le \beta. \tag{16}$$

Furthermore, we assume that the Jacobian mapping associated with the nonlinear model f obeys

$$\|\mathcal{J}(\theta)\| \le \beta \quad \textit{for all} \quad \theta \in \mathbb{R}^p. \tag{17}$$

Assumption 2 (Closeness of the reference and initialization Jacobians). *We assume the reference Jacobian and the Jacobian of the nonlinearity at initialization $\mathcal{J}(\theta_0)$ are ϵ_0-close in the sense that*

$$\left\|\mathcal{J}(\theta_0)\mathcal{J}^T(\theta_0) - \mathbf{J}\mathbf{J}^T\right\| \le \epsilon_0^2, \quad \textit{and} \quad \|\mathcal{J}(\theta_0) - \mathbf{J}\| \le \epsilon_0. \tag{18}$$

Assumption 3 (Bounded variation of Jacobian around initialization). *We assume that within a radius R around the initialization, the Jacobian varies by no more than ϵ in the sense that*

$$\|\mathcal{J}(\theta) - \mathcal{J}(\theta_0)\| \le \frac{\epsilon}{2}, \quad \textit{for all} \quad \theta \in \mathcal{B}_R(\theta_0), \tag{19}$$

where $\mathcal{B}_R(\theta_0) := \{\theta \colon \|\theta - \theta_0\| \le R\}$ is the ball with radius R around θ_0.

Our first result shows that under these assumptions the nonlinear iterative updates (13) are intimately related to the linear iterative updates (15). Specifically, we show that the misfit or residuals associated with these two problems defined below

$$\text{nonlinear residual:} \quad \mathbf{r}_\tau := f(\theta_\tau) - \mathbf{y} \tag{20}$$

$$\text{linear residual:} \quad \widetilde{\mathbf{r}}_\tau := \left(\mathbf{I} - \eta \mathbf{J}\mathbf{J}^T\right)^\tau \mathbf{r}_0. \tag{21}$$

are close in the proximity of the initialization.

Theorem 4 (Closeness of linear and nonlinear least-squares problems). *Assume the Jacobian mapping $\mathcal{J}(\theta) \in \mathbb{R}^{n \times p}$ associated with the function $f(\theta)$ obeys Assumptions 1, 2, and 3 around an initial point $\theta_0 \in \mathbb{R}^p$ with respect to a reference Jacobian $\mathbf{J} \in \mathbb{R}^{n \times p}$ and with parameters $\alpha, \beta, \epsilon_0, \epsilon$, and R. Furthermore, assume the parameter R is given by*

$$\frac{R}{2} := \left\|\mathbf{J}^\dagger \mathbf{r}_0\right\|_2 + \frac{\kappa}{2}\left(\frac{1}{\alpha^2} + \eta T\right) \|\mathbf{r}_0\|_2 \tag{22}$$

with κ and T constants obeying $\kappa > 0$ and $T \ge 1$. Also assume that $\alpha, \beta, \epsilon_0, \epsilon$, and κ obey

$$\epsilon \le \frac{1}{8}\frac{\kappa\alpha^2}{\beta^2} \quad \textit{and} \quad \epsilon_0 \le \min\left(\frac{1}{4}\kappa, \sqrt{\frac{1}{8}\frac{\kappa\alpha^2}{\beta}}\right). \tag{23}$$

We run gradient descent on the linear and non-linear least squares problem, starting from the same initialization θ_0, and with stepsize obeying $\eta \le \frac{1}{\beta^2}$. Then, for all $\tau \le T \le \frac{1}{2\eta\epsilon^2}$ the iterates of the original and the linearized problems and their corresponding residuals obey

$$\|\mathbf{r}_\tau - \widetilde{\mathbf{r}}_\tau\| \le \frac{1}{2}\frac{\kappa}{\beta}\|\mathbf{r}_0\|_2 \tag{24}$$

$$\left\|\theta_\tau - \widetilde{\theta}_\tau\right\| \le \frac{\kappa}{2}\left(\frac{1}{\alpha^2} + \eta\tau\right)\|\mathbf{r}_0\|_2 \tag{25}$$

Moreover, for all iterates $\tau \le T$,

$$\|\theta_\tau - \theta_0\| \le \frac{R}{2}. \tag{26}$$

The above theorem formalizes that in a (small) radius around the initialization, the non-linear problem behaves similarly as its linearization. Thus to characterize the performance of the nonlinear problem for denoising it suffices to characterize the denoising capability of the linearized problem. This is the subject of our next theorem.

Theorem 5. *Consider a linearized denoising reference problem of the form* (14) *with a reference Jacobian* $\mathbf{J} \in \mathbb{R}^{n \times p}$*. Let* $\mathbf{J} = \mathbf{W}\mathbf{\Sigma}\mathbf{V}^T \in \mathbb{R}^{n \times p} = \sum_{i=1}^{n} \sigma_i \mathbf{w}_i \mathbf{v}_i^T$ *be the singular value decomposition of this Jacobian with the singular values defined in decreasing order. Then using the linearized updates* (15) *the linearized residual iterates* $\widetilde{\mathbf{r}}_\tau$ *can be written as*

$$\widetilde{\mathbf{r}}_\tau = \sum_{i=1}^{n} \left(1 - \eta\sigma_i^2\right)^\tau \mathbf{w}_i \left\langle \mathbf{w}_i, \mathbf{r}_0 \right\rangle . \tag{27}$$

Moreover, using a step size satisfying $\eta \leq \frac{1}{\sigma_1^2}$*, the linearized iterates* (15) *obey*

$$\left\| \widetilde{\theta}_\tau - \theta_0 \right\|_2^2 \leq \eta^2 \sum_{i=1}^{n} \left(\left\langle \mathbf{w}_i, \mathbf{r}_0 \right\rangle \frac{1 - (1 - \eta\sigma_i^2)^\tau}{\eta\sigma_i} \right)^2 . \tag{28}$$

In the next section we will show we can combine these two general theorems to provide guarantees for denoising using general neural networks.

D.1 Proof of Theorem 4 (closeness of linear and non-linear least-squares)

The proof is by induction. We suppose the statement, in particular the bounds (24), (25), and (26) hold for iterations $t \leq \tau - 1$. We then show that they continue to hold for iteration τ in four steps. We begin by showing that a weaker version of (26) holds. In particular, in Step I we show that $\|\theta_\tau - \theta_0\|_2 \leq R$. Then in Steps II and III we show that the bounds (24) and (25) hold, respectively. Finally, in Step IV we utilize Steps I-III to complete the proof of equation (26).

Step I: Next iterate obeys $\theta_\tau \in \mathcal{B}_R(\theta_0)$. To prove $\theta_\tau \in \mathcal{B}_R(\theta_0)$, first note that by the triangle inequality and the induction assumption (26) we have

$$\begin{aligned}
\|\theta_\tau - \theta_0\|_2 &\leq \|\theta_\tau - \theta_{\tau-1}\|_2 + \|\theta_{\tau-1} - \theta_0\|_2 , \\
&\leq \|\theta_\tau - \theta_{\tau-1}\|_2 + \frac{R}{2}.
\end{aligned}$$

So to prove $\|\theta_\tau - \theta_0\|_2 \leq R$ it suffices to prove $\|\theta_\tau - \theta_{\tau-1}\|_2 \leq R/2$. To this aim note that

$$\begin{aligned}
\|\theta_\tau - \theta_{\tau-1}\|_2 &= \eta \left\|\nabla\mathcal{L}(\theta_{\tau-1})\right\|_2 \\
&= \eta \left\|\mathcal{J}^T(\theta_{\tau-1})\mathbf{r}_{\tau-1}\right\|_2 \\
&\leq \eta \left(\left\|\mathcal{J}^T(\theta_{\tau-1})\widetilde{\mathbf{r}}_{\tau-1}\right\|_2 + \|\mathcal{J}(\theta_{\tau-1})\| \left\|\mathbf{r}_{\tau-1} - \widetilde{\mathbf{r}}_{\tau-1}\right\|_2 \right) \\
&\leq \eta \left(\left\|\mathbf{J}^T\widetilde{\mathbf{r}}_{\tau-1}\right\|_2 + \|\mathcal{J}(\theta_{\tau-1}) - \mathbf{J}\| \left\|\widetilde{\mathbf{r}}_{\tau-1}\right\|_2 + \|\mathcal{J}(\theta_{\tau-1})\| \left\|\mathbf{r}_{\tau-1} - \widetilde{\mathbf{r}}_{\tau-1}\right\|_2 \right) \\
&\overset{\text{(i)}}{\leq} \eta \left\|\mathbf{J}^T\widetilde{\mathbf{r}}_{\tau-1}\right\|_2 + \eta(\epsilon + \epsilon_0) \left\|\widetilde{\mathbf{r}}_{\tau-1}\right\|_2 + \eta\beta \left\|\mathbf{r}_{\tau-1} - \widetilde{\mathbf{r}}_{\tau-1}\right\|_2 \\
&\overset{\text{(ii)}}{\leq} \eta \left\|\mathbf{J}^T\widetilde{\mathbf{r}}_{\tau-1}\right\|_2 + \eta\frac{1}{2}\kappa \left\|\widetilde{\mathbf{r}}_{\tau-1}\right\|_2 + \eta\frac{1}{2}\kappa \|\mathbf{r}_0\|_2 \\
&\overset{\text{(iii)}}{\leq} \eta \left\|\mathbf{J}^\dagger\mathbf{r}_0\right\|_2 + \eta\kappa \|\mathbf{r}_0\|_2 \\
&\leq \frac{R}{2}.
\end{aligned} \tag{29}$$

In the above (i) follows from Assumptions 1, 2, and 3, (ii) from $\epsilon_0 + \epsilon \leq \frac{1}{2}\kappa$ (which is a consequence of (23)) combined with with the induction hypothesis (24), and (iii) follows from $\|\widetilde{\mathbf{r}}_{\tau-1}\| \leq \|\mathbf{r}_0\|$

as well as from the bound

$$\begin{aligned}\left\|\mathbf{J}^T\widetilde{\mathbf{r}}_{\tau-1}\right\|_2 &= \left\|\mathbf{J}^T(\mathbf{I}-\eta\mathbf{J}\mathbf{J}^T)^{\tau-1}\mathbf{r}_0\right\|_2\\ &= \left\|\boldsymbol{\Sigma}(\mathbf{I}-\eta\boldsymbol{\Sigma}^2)^{\tau-1}\mathbf{W}^T\mathbf{r}_0\right\|_2\\ &\leq \sqrt{\sum_{j=1}^{n}\sigma_j^2\langle\mathbf{w}_j,\mathbf{r}_0\rangle^2}\\ &= \left\|\mathbf{J}^{\dagger}\mathbf{r}_0\right\|_2.\end{aligned}$$

Finally, the last inequality follows by definition of R in (22) together with the fact that $T \geq 1$.

Step II: Original and linearized residuals are close: In this step, we bound the deviation of the residuals of the original and linearized problem defined as

$$\mathbf{e}_\tau := \mathbf{r}_\tau - \widetilde{\mathbf{r}}_\tau.$$

This step relies on the following lemma from Oymak et al. (2019).

Lemma 1 (Bound on growth of perturbations, (Oymak et al., 2019, Lem. 6.7))**.** *Suppose that Assumptions 1, 2, and 3 hold and that $\theta_\tau, \theta_{\tau+1} \in \mathcal{B}_R(\theta_0)$. Then, provided the stepsize obeys $\eta \leq 1/\beta^2$, the deviation of the residuals obeys*

$$\|\mathbf{e}_{\tau+1}\|_2 \leq \eta\left(\epsilon_0^2+\epsilon\beta\right)\|\widetilde{\mathbf{r}}_\tau\|_2 + \left(1+\eta\epsilon^2\right)\|\mathbf{e}_\tau\|_2. \tag{30}$$

By the previous step, $\theta_\tau, \theta_{\tau+1} \in \mathcal{B}_R(\theta_0)$. We next bound the two terms on the right hand side of (30). Regarding the first term, we note that an immediate consequence of Theorem 5 is the following bound on the residual of the linearized iterates:

$$\|\widetilde{\mathbf{r}}_\tau\|_2 \leq \left(1-\eta\alpha^2\right)^\tau\|\mathbf{r}_0\|_2. \tag{31}$$

In order to bound the second term in (30), namely, $\|\mathbf{e}_\tau\|_2$, we used the following lemma, proven later in Section D.1.1.

Lemma 2. *Suppose that for positive scalars $\alpha, \eta, \rho, \xi > 0$, $\eta \leq 1/\alpha^2$, the sequences $\widetilde{r}_\tau$ and e_τ obey*

$$\begin{aligned}\widetilde{r}_\tau &\leq \left(1-\eta\alpha^2\right)^\tau\rho\\ e_\tau &\leq \left(1+\eta\epsilon^2\right)e_{\tau-1}+\eta\xi\widetilde{r}_{\tau-1}\end{aligned}$$

Then, for all $\tau \leq \frac{1}{2\eta\epsilon^2}$, we have that

$$e_\tau \leq 2\xi\frac{\Gamma}{\alpha^2}. \tag{32}$$

With these lemmas in place we now have all the tools to prove that the original and linear residuals are close. In particular, from Step I, we know that $\theta_\tau \in \mathcal{B}_R(\theta_0)$ so that the assumptions of Lemma 1 are satisfied. Thus, combining Lemmas 1, equation (31), and Lemma 2 above we conclude that for all $\tau \leq \min\left(\frac{1}{2\eta\epsilon^2}, \Gamma\frac{1}{\eta\alpha^2}\right)$

$$\|\mathbf{e}_\tau\|_2 \leq 2\frac{\left(\epsilon_0^2+\epsilon\beta\right)}{\alpha^2}\|\mathbf{r}_0\|_2 \leq \frac{\kappa}{2\beta}\|\mathbf{r}_0\|_2, \tag{33}$$

where in the last inequality we used the assumption that $\frac{2(\epsilon_0^2+\epsilon\beta)}{\alpha^2} \leq \frac{\kappa}{2\beta}$. This concludes the proof of (24).

Step III: Original and linearized parameters are close: First note that by the triangle inequality and Assumption 3 we have

$$\|\mathcal{J}(\theta_\tau)-\mathbf{J}\| \leq \|\mathcal{J}(\theta_\tau)-\mathcal{J}(\theta_0)\| + \|\mathcal{J}(\theta_0)-\mathbf{J}\| \leq \epsilon_0+\epsilon. \tag{34}$$

The difference between the parameter of the original iterate θ and the linearized iterate $\widetilde{\theta}$ obey

$$\begin{aligned}
\frac{1}{\eta}\left\|\theta_\tau-\widetilde{\theta}_\tau\right\|_2 &\leq \left\|\sum_{t=0}^{\tau-1}\nabla\mathcal{L}(\theta_t)-\nabla\mathcal{L}_{\mathrm{lin}}(\widetilde{\theta}_t)\right\|_2\\
&=\left\|\sum_{t=0}^{\tau-1}\mathcal{J}^T(\theta_t)\mathbf{r}_t-\mathbf{J}^T\widetilde{\mathbf{r}}_t\right\|_2\\
&\leq\sum_{t=0}^{\tau-1}\left\|(\mathcal{J}^T(\theta_t)-\mathbf{J}^T)\widetilde{\mathbf{r}}_t\right\|_2+\left\|\mathcal{J}^T(\theta_t)(\mathbf{r}_t-\widetilde{\mathbf{r}}_t)\right\|_2\\
&\overset{\text{(i)}}{\leq}\sum_{t=0}^{\tau-1}(\epsilon_0+\epsilon)\left\|\widetilde{\mathbf{r}}_t\right\|_2+\beta\left\|\mathbf{e}_t\right\|_2\\
&\overset{\text{(ii)}}{\leq}(\epsilon_0+\epsilon)\sum_{t=0}^{\tau-1}\left(1-\eta\alpha^2\right)^{\tau-1}\left\|\mathbf{r}_0\right\|_2+\beta\sum_{t=0}^{\tau-1}\left\|\mathbf{e}_t\right\|_2\\
&=(\epsilon_0+\epsilon)\frac{1-(1-\eta\alpha^2)^\tau}{\eta\alpha^2}\left\|\mathbf{r}_0\right\|_2+\beta\sum_{t=0}^{\tau-1}\left\|\mathbf{e}_t\right\|_2\\
&\overset{\text{(iii)}}{\leq}\frac{(\epsilon_0+\epsilon)}{\eta\alpha^2}\left\|\mathbf{r}_0\right\|_2+\frac{\kappa\tau}{2}\left\|\mathbf{r}_0\right\|_2\\
&\overset{\text{(iv)}}{\leq}\frac{\kappa}{2\eta\alpha^2}\left\|\mathbf{r}_0\right\|_2+\frac{\kappa\tau}{2}\left\|\mathbf{r}_0\right\|_2
\end{aligned}$$

where (i) follows from (34) combined with Assumption 1, (ii) from (31), (iii) from $\eta\leq 1/\beta^2$ which implies $(1-\eta\alpha^2)\geq 0$ and (33). Finally, where (iv) follows from the fact that $\epsilon\leq\frac{1}{8}\kappa$ and $\epsilon_0\leq\frac{1}{4}\kappa$ per (23). This concludes the proof of (25).

Step IV: Completing the proof of (26)**:** By the triangle inequality

$$\begin{aligned}
\left\|\theta_\tau-\theta_0\right\|_2 &\leq\left\|\widetilde{\theta}_\tau-\theta_0\right\|_2+\left\|\theta_\tau-\widetilde{\theta}_\tau\right\|_2\\
&\overset{\text{(i)}}{\leq}\left\|\mathbf{J}^\dagger\mathbf{r}_0\right\|_2+\frac{\kappa}{2}\left(\frac{1}{\alpha^2}+\eta\tau\right)\left\|\mathbf{r}_0\right\|_2\\
&\overset{\text{(ii)}}{\leq}R/2,
\end{aligned}$$

where inequality (i) follows from the bound (25), which we just proved, and the fact that, from equation (35) in Theorem 5,

$$\begin{aligned}
\left\|\widetilde{\theta}_\tau-\theta_0\right\|_2^2 &=\sum_{i=1}^n\langle\mathbf{w}_i,\mathbf{r}_0\rangle^2\frac{(1-(1-\eta\sigma_i^2)^\tau)^2}{\sigma_i^2}\\
&\leq\sum_{i=1}^n\langle\mathbf{w}_i,\mathbf{r}_0\rangle^2/\sigma_i^2\\
&=\left\|\mathbf{J}^\dagger\mathbf{r}_0\right\|^2.
\end{aligned}$$

Moreover, inequality (ii) follows from $\tau\leq T$, by assumption as well as from the definition of R in equation (22).

D.1.1 Proof of Lemma 2

We prove the result by induction. Assume equation (32) holds true for some τ. We prove that then it also holds true for $\tau+1$. By the two assumptions,

$$\begin{aligned}
e_{\tau+1}-e_\tau &\leq\eta\epsilon^2 e_\tau+\eta\xi\widetilde{r}_\tau\\
&\leq\eta\epsilon^2 e_\tau+\eta\xi(1-\eta\alpha^2)^\tau\rho\\
&\overset{\text{(i)}}{\leq}\eta\xi\left(\epsilon^2\frac{2\rho}{\alpha^2}+(1-\eta\alpha^2)^\tau\rho\right),
\end{aligned}$$

where (i) follows from the induction assumption (32). Summing up the difference of the errors gives

$$\begin{aligned}\frac{e_\tau}{\xi} &= \sum_{t=0}^{\tau-1}\frac{(e_{t+1}-e_t)}{\xi}\\ &\le \tau\eta\epsilon^2\frac{2\rho}{\alpha^2}+\eta\rho\sum_{t=0}^{\tau-1}(1-\eta\alpha^2)^t\\ &= \tau\eta\epsilon^2\frac{2\rho}{\alpha^2}+\eta\rho\frac{1-(1-\eta\alpha^2)^\tau}{\eta\alpha^2}\\ &\le \frac{\rho}{\alpha^2}(\eta 2\tau\epsilon^2+1)\\ &\le 2\frac{\rho}{\alpha^2},\end{aligned}$$

where the last inequality follows from the assumption that $\tau \le \frac{1}{\eta\epsilon^2}$.

D.2 Proof of Theorem 5

The proof of identity (27) is equivalent to the proof of Proposition (1). Regarding inequality (28), note that

$$\widetilde{\theta}_\tau-\widetilde{\theta}_0 = -\eta\sum_{t=0}^{\tau-1}\nabla\mathcal{L}_{\text{lin}}(\widetilde{\theta}_t) = -\eta\sum_{t=0}^{\tau-1}\mathbf{J}^T\widetilde{\mathbf{r}}_t = -\eta\mathbf{V}\left(\sum_{t=0}^{\tau-1}\mathbf{\Sigma}\left(\mathbf{I}-\eta\mathbf{\Sigma}^2\right)^t\right)\begin{bmatrix}\langle\mathbf{u}_1,\mathbf{r}_0\rangle\\ \vdots\\ \langle\mathbf{u}_n,\mathbf{r}_0\rangle\end{bmatrix}.$$

Thus

$$\left\langle\mathbf{v}_i,\widetilde{\theta}_\tau-\widetilde{\theta}_0\right\rangle = -\eta\sigma_i\langle\mathbf{u}_i,\mathbf{r}_0\rangle\left(\sum_{t=0}^{\tau-1}(1-\eta\sigma_i^2)^t\right) = -\eta\sigma_i\langle\mathbf{u}_i,\mathbf{r}_0\rangle\frac{1-(1-\eta\sigma_i^2)^\tau}{\eta\sigma_i^2}. \qquad (35)$$

This in turn implies that

$$\left\|\widetilde{\theta}_\tau-\widetilde{\theta}_0\right\|_2^2 \le \sum_{i=1}^{n}\left(\langle\mathbf{u}_i,\mathbf{r}_0\rangle\frac{1-(1-\eta\sigma_i^2)^\tau}{\sigma_i}\right)^2.$$

E Proofs for Neural Network Generators (Proof of Theorem 3)

In the previous sections we proved general results for the trajectory of gradient descent applied to a non-linear least-squares problem, and for a linear least squares problem. In this section, we build on those results (specifically Theorems 4 and 5) by proving general result for denoising via early stopping neural network generators. The proof of Theorem 3 relies on the fact (as formalized below) that the Jacobian of the non-linear least squares problem around the initialization behaves like a linearized problem related to the expected value of the Jacobian at initialization. In particular, we utilize the fact that in a ball around the initialization, by Theorem 4, the trajectory of the non-linear least squares problem is close to that of the linear least-squares problem whose convergence behavior is characterized in Theorem 5.

We apply Theorem 4 with the nonlinearity $G(\mathbf{C}) = \text{ReLU}(\mathbf{UC})\mathbf{v}$ with the parameter given by $\theta = \mathbf{C}$, and $\mathbf{v}$ a fixed vector with half of the entries $1/\sqrt{k}$, and the other half $-1/\sqrt{k}$. Similarly, we shall apply Theorem 5 with the reference Jacobian matrix $\mathbf{J} := \left(\mathbb{E}\left[\mathcal{J}(\mathbf{C})\mathcal{J}^T(\mathbf{C})\right]\right)^{\frac{1}{2}}$ so that it obeys $\mathbf{J}\mathbf{J}^T = \mathbb{E}\left[\mathcal{J}(\mathbf{C})\mathcal{J}^T(\mathbf{C})\right]$. To apply these theorems we need to show the conditions of these theorems hold. To this aim we state a few lemmas proven in Appendix H. The first lemma controls the perturbation of the Jacobian matrix around a random initialization.

Lemma 3 (Jacobian perturbation around initialization)**.** *Consider $G(\mathbf{C}) = \text{ReLU}(\mathbf{UC})\mathbf{v}$. Let $\mathbf{C}_0$ be a matrix with i.i.d. $N(0,\omega^2)$ entries. Then, for all $\mathbf{C}$ obeying*

$$\|\mathbf{C}-\mathbf{C}_0\| \le \omega\widetilde{R} \quad \textit{with} \quad \widetilde{R} \le \frac{1}{2}\sqrt{k},$$

the Jacobian mapping associated with this generator (given by (8)*) obeys*

$$\|\mathcal{J}(\mathbf{C}) - \mathcal{J}(\mathbf{C}_0)\| \leq \|\mathbf{v}\|_\infty 2(k\widetilde{R})^{1/3} \|\mathbf{U}\|,$$

with probability at least $1 - ne^{-\frac{1}{2}\widetilde{R}^{4/3}k^{7/3}}$.

In the next lemma we show that the Jacobian has bounded spectrum.

Lemma 4 (Spectral norm of Jacobian)**.** *Consider* $G(\mathbf{C}) = \mathrm{ReLU}(\mathbf{UC})\mathbf{v}$ *with the associated Jacobian* $\mathcal{J}(\mathbf{C})$ *given by* (8)*, and let* $\mathbf{J}$ *be any matrix obeying* $\mathbf{J}\mathbf{J}^T = \mathbb{E}\left[\mathcal{J}(\mathbf{C})\mathcal{J}^T(\mathbf{C})\right]$*, where the expectation is over a matrix* $\mathbf{C}$ *with iid* $\mathcal{N}(0, \omega^2)$ *entries. Then*

$$\|\mathcal{J}(\mathbf{C})\| \leq \|\mathbf{v}\|_2 \|\mathbf{U}\| \quad \text{and} \quad \|\mathbf{J}\| \leq \|\mathbf{v}\|_2 \|\mathbf{U}\|.$$

The next lemma shows that the Jacobian times Jacobian transpose at initialization is close to its expected value, which is the neural tangent kernel (see Definition 3). Using the formula for the Jacobian 8, this can be written as

$$\mathbf{\Sigma}(\mathbf{U}) = \|\mathbf{v}\|_2^2 \, \mathbb{E}_{\mathbf{c}\sim\mathcal{N}(0,\omega^2)} \left[\mathrm{ReLU}'(\mathbf{Uc})\mathrm{ReLU}'(\mathbf{Uc})^T\right] \odot \mathbf{U}\mathbf{U}^T,$$

where $\odot$ ist the entrywise product of two vectors/matrices.

Lemma 5 (Concentration lemma)**.** *Consider* $G(\mathbf{C}) = \mathrm{ReLU}(\mathbf{UC})\mathbf{v}$ *with the associated Jacobian* $\mathcal{J}(\mathbf{C})$ *given by* (8)*. Also let* $\mathbf{C} \in \mathbb{R}^{n\times k}$ *be generated at random with i.i.d.* $\mathcal{N}(0, \omega^2)$ *entries. Then,*

$$\left\|\mathcal{J}(\mathbf{C})\mathcal{J}^T(\mathbf{C}) - \mathbf{\Sigma}(\mathbf{U})\right\| \leq \|\mathbf{U}\|^2 \sqrt{\log\left(\frac{2n}{\delta}\right) \sum_{\ell=1}^{k} v_\ell^4},$$

holds with probability at least $1 - \delta$.

Note that Lemma 5 only ensures the first condition in (18). To see that it also implies the second condition in (18), we used the following lemma which establishes that the first condition implies the second.

Lemma 6 ((Oymak et al., 2019, Lem. 6.4))**.** *Let* $\mathbf{X} \in \mathbb{R}^{n\times p}$*,* $p \geq n$ *and let* $\mathbf{B}$ *be* $n \times n$ *psd matrix obeying* $\left\|\mathbf{X}\mathbf{X}^T - \mathbf{B}\right\| \leq \epsilon^2$*, for a scalar* $\epsilon \geq 0$*. Then there exists a matrix* $\mathbf{Y} \in \mathbb{R}^{n\times p}$ *obeying* $\mathbf{B} = \mathbf{Y}\mathbf{Y}^T$ *such that*

$$\|\mathbf{Y} - \mathbf{X}\| \leq 2\epsilon.$$

Finally, we need the following lemma which bounds the initial misfit/residual.

Lemma 7 (Initial residual)**.** *Consider* $G(\mathbf{C}) = \mathrm{ReLU}(\mathbf{UC})\mathbf{v}$*, and let* $\mathbf{C} \in \mathbb{R}^{n\times k}$ *be generated at random with i.i.d.* $\mathcal{N}(0, \omega^2)$ *entries. Suppose half of the entries of* $\mathbf{v}$ *are* $\nu/\sqrt{k}$ *and the other half are* $-\nu/\sqrt{k}$*, for some constant* $\nu > 0$*. Then, with probability at least* $1 - \delta$*,*

$$\|G(\mathbf{C})\|_2 \leq \nu\omega\sqrt{8\log(2n/\delta)} \|\mathbf{U}\|_F.$$

With these lemmas in place we now have all the elements to verify the conditions of Theorem 4. We thus turn our attention to verifying Assumptions 1, 2, 3, and conditions of (23) with parameters given by $\nu = 1$ and

$$\alpha = \sigma_n(\mathbf{\Sigma}(\mathbf{U})), \quad \beta = \|\mathbf{U}\|, \quad \kappa = \xi\frac{\alpha^2}{\beta}, \quad \epsilon_0 = 2\|\mathbf{U}\|\left(\frac{\log(2n/\delta)}{k}\right)^{1/4}, \quad \omega = \frac{\|\mathbf{y}\|_2}{\sqrt{n}\beta}\xi\frac{\alpha^2}{\beta^2}.$$

First note that with those choices, the initial residual can be upper bounded as

$$\|\mathbf{r}_0\|_2 \leq \|\mathbf{y}\|_2 + \|G(\mathbf{C}_0)\|_2 \leq 2\|\mathbf{y}\|_2, \tag{36}$$

where we used that, by Lemma 7,

$$\|G(\mathbf{C}_0)\|_2 \leq \omega\sqrt{8\log(2n/\delta)} \|\mathbf{U}\|_F \leq \|\mathbf{y}\|_2\xi\frac{\alpha^2}{\beta^2}\sqrt{8\log(2n/\delta)}, \tag{37}$$

together with the bound $\xi \leq 1/\sqrt{8\log(2n/\delta)}$ and $\alpha/\beta \leq 1$.

Verifying Assumption 1: To verify Assumption 1 note that by definition $\mathbf{J}\mathbf{J}^T = \mathbf{\Sigma}(\mathbf{U})$ thus trivially $\sigma_n\left(\mathbf{\Sigma}(\mathbf{U})\right) \geq \alpha$ holds. Furthermore, Lemma 4 combined with the fact that $\left\|\mathbf{v}\right\|_2 = 1$ implies that $\|\mathbf{J}\| \leq \beta$ and $\|\mathcal{J}(\mathbf{C})\| \leq \beta$ for all $\mathbf{C}$. This completes the verification of Assumption 1.

Verifying Assumption 2: We begin by verifying the first condition of the assumption, inequality (18). To this aim note that using the fact that $\sum_{\ell}^{k} v_\ell^4 = \frac{\nu^4}{k}$ by Lemma 5 we have

$$\left\|\mathcal{J}(\mathbf{C}_0)\mathcal{J}^T(\mathbf{C}_0) - \mathbf{\Sigma}(\mathbf{U})\right\| \leq \|\mathbf{U}\|^2 \sqrt{\frac{\log\left(\frac{2n}{\delta}\right)}{k}} \leq \epsilon_0^2, \tag{38}$$

holds with probability at least $1-\delta$. Thus, the first condition in (18) is satisfied. Furthermore, combining the first inequality in (38) with Lemma 6 we conclude that the second condition in (18) also holds for the chosen value of ϵ_0.

Verifying Assumption 3: First, note that the radius in the theorem, defined in equation (22), obeys

$$\begin{aligned}
R &= 2\left\|\mathbf{J}^{\dagger}\mathbf{r}_0\right\|_2 + \kappa\left(\frac{1}{\alpha^2} + \eta T\right)\|\mathbf{r}_0\|_2 \\
&\overset{\text{(i)}}{\leq} \left(\frac{2}{\alpha} + \frac{\kappa}{\alpha^2} + \eta\kappa T\right)\|\mathbf{r}_0\|_2 \\
&\overset{\text{(ii)}}{\leq} 4\left(\frac{1}{\alpha} + \frac{\xi}{\beta}T\right)\|\mathbf{y}\|_2 \\
&\overset{\text{(iii)}}{\leq} \omega\frac{\xi^3}{2^{15}}\frac{\alpha^{12}}{\beta^{12}}\sqrt{k} \\
&:= \omega\widetilde{R}.
\end{aligned}$$

Here, (i) follows from the fact that $\left\|\mathbf{J}_{\mathcal{S}}^{\dagger}\mathbf{r}_0\right\|_2 \leq \frac{1}{\alpha}\|\mathbf{r}_0\|_2$, (ii) from the bound on the initial residual (36), and finally (iii) follows from the assumption (9).

Note that since $\Delta = \frac{\xi^3}{2^{15}}\frac{\alpha^{12}}{\beta^{12}} \leq \frac{1}{2}$ for this choice of radius by Lemma 3 we have

$$\|\mathcal{J}(\mathbf{C}) - \mathcal{J}(\mathbf{C}_0)\| \leq \|\mathbf{v}\|_\infty 2(k\widetilde{R})^{1/3}\|\mathbf{U}\| = \frac{\xi}{16}\|\mathbf{U}\|\frac{\alpha^4}{\beta^4},$$

holds with probability at least

$$1 - ne^{-\frac{1}{2^{21}}\xi^4\frac{\alpha^{16}}{\beta^{16}}k^{25/6}} \overset{\text{(i)}}{\geq} 1-\delta,$$

where in (i) we used (9) together with $\xi \leq \sqrt{8\log\left(\frac{2n}{\delta}\right)}$. Therefore, Assumption 3 holds with high probability by choosing

$$\epsilon = \frac{\xi}{8}\|\mathbf{U}\|\frac{\alpha^4}{\beta^4}.$$

Verifying the conditions of (23)**:** We now turn our attention to verifying that the bounds on ϵ and ϵ_0 in (23) are satisfied. We begin by showing the bound on ϵ holds. To this aim note that our choice of ϵ can alternatively be written as $\epsilon = \frac{\xi}{8}\frac{\alpha^4}{\beta^3}$ so that the condition $\epsilon \leq \frac{1}{8}\frac{\kappa\alpha^2}{\beta^2}$ on ϵ is satisfied. To verify the second condition in (23) note that

$$\epsilon_0 = 2\nu\|\mathbf{U}\|\sqrt[4]{\frac{\log(2n/\delta)}{k}} = 2\beta\sqrt[4]{\frac{\log(2n/\delta)}{k}} \overset{\text{(i)}}{\leq} \frac{1}{4}\kappa \overset{\text{(ii)}}{=} \min\left(\frac{1}{4}\kappa, \sqrt{\frac{1}{8}\frac{\kappa\alpha^2}{\beta}}\right),$$

where inequality (i) holds by (9) which implies $8\sqrt[4]{\log(2n/\delta)/k} \leq \xi\frac{\alpha^2}{\beta^2} = \frac{\kappa}{\beta}$ (ii) holds by the assumption $\xi \leq 2$.

Verifying the bound on number of iterations: Finally we note that the constraint on the number of iterations, T, $T \leq \frac{1}{2\eta\epsilon^2}$, from Theorem 4 is satisfied under the number of constraints of Theorem 3 by $\frac{1}{2\eta\epsilon^2} = 32\frac{\beta^6}{\eta\xi^2\alpha^8}$.

Now that we have verified the conditions of Theorem 4 we can apply this theorem. This allows us to conclude that

$$\begin{aligned}
\left\|G(\mathbf{C}_\tau) - \mathbf{x}\right\|_2 &= \left\|G(\mathbf{C}_\tau) + \mathbf{z} - \mathbf{y}\right\|_2 \\
&= \left\|\mathbf{r}_\tau + \mathbf{z}\right\|_2 \\
&= \left\|\widetilde{\mathbf{r}}_\tau + \mathbf{z} + \mathbf{r}_\tau - \widetilde{\mathbf{r}}_\tau\right\|_2 \\
&\overset{\text{(i)}}{\leq} \left\|\widetilde{\mathbf{r}}_\tau + \mathbf{z}\right\|_2 + \left\|\mathbf{r}_\tau - \widetilde{\mathbf{r}}_\tau\right\|_2 \\
&\overset{\text{(ii)}}{\leq} \left\|\widetilde{\mathbf{r}}_\tau + \mathbf{z}\right\|_2 + \frac{1}{2}\xi\frac{\alpha^2}{\beta^2}\left\|\mathbf{r}_0\right\|_2 \\
&\overset{\text{(iii)}}{=} \left\|\mathbf{W}\left(\mathbf{I} - \eta\mathbf{\Sigma}^2\right)^\tau \mathbf{W}^T\mathbf{r}_0 + \mathbf{z}\right\|_2 + \frac{1}{2}\xi\frac{\alpha^2}{\beta^2}\left\|\mathbf{r}_0\right\|_2 \\
&\overset{\text{(iv)}}{=} \left\|\mathbf{W}\left(\mathbf{I} - \eta\mathbf{\Sigma}^2\right)^\tau \mathbf{W}^T(G(\mathbf{C}_0) - \mathbf{x}) - \left(\mathbf{W}\left(\mathbf{I} - \eta\mathbf{\Sigma}^2\right)^\tau \mathbf{W}^T - \mathbf{I}\right)\mathbf{z}\right\|_2 + \frac{1}{2}\xi\frac{\alpha^2}{\beta^2}\left\|\mathbf{r}_0\right\|_2 \\
&\overset{\text{(v)}}{\leq} \left\|\left(\mathbf{I} - \eta\mathbf{\Sigma}^2\right)^\tau \mathbf{W}^T\mathbf{x}\right\|_2 + \left\|\left(\left(\mathbf{I} - \eta\mathbf{\Sigma}^2\right)^\tau - \mathbf{I}\right)\mathbf{W}^T\mathbf{z}\right\|_2 + \left\|G(\mathbf{C}_0)\right\|_2 + \frac{1}{2}\xi\frac{\alpha^2}{\beta^2}\left\|\mathbf{r}_0\right\|_2 \\
&\overset{\text{(vi)}}{\leq} (1 - \eta\sigma_r^2)^\tau\left\|\mathbf{x}\right\|_2 + \sqrt{\sum_{i=1}^n((1 - \eta\sigma_i^2)^\tau - 1)^2\left\langle\mathbf{w}_i, \mathbf{z}\right\rangle^2} + 2\|\mathbf{y}\|_2\xi\frac{\alpha^2}{\beta^2}\sqrt{8\log(2n/\delta)}.
\end{aligned}$$

Here, (i) follows from the triangular inequality, (ii) from Theorem 4 equation (24), (iii) from Theorem 5, (iv) from $\mathbf{r}_0 = G(\mathbf{C}_0) - \mathbf{y} = G(\mathbf{C}_0) - \mathbf{x} - \mathbf{z}$, (v) from the triangular inequality, (vi) from the fact that $\mathbf{x} \in \text{span}\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_r\}$ combined with the fact that $\left\|\mathbf{W}\left(\mathbf{I} - \eta\mathbf{\Sigma}^2\right)^\tau \mathbf{W}^T\right\| \leq 1$. Finally, (vi) follows by bounding the last two terms with the bounds (36) and (37), respectively. This proves the final bound (11), as desired.

F Proof of Proposition 1 and equation (2)

The residual of gradient descent at iteration τ is

$$\begin{aligned}
\mathbf{r}_\tau &= \mathbf{y} - \mathbf{A}\mathbf{x}_\tau \\
&= \mathbf{y} - \mathbf{A}(\mathbf{x}_{\tau-1} - \eta\mathbf{A}^T(\mathbf{A}\mathbf{x}_{\tau-1} - \mathbf{y})) \\
&= (\mathbf{I} - \eta\mathbf{A}\mathbf{A}^T)(\mathbf{y} - \mathbf{A}\mathbf{x}_{\tau-1}) \\
&= (\mathbf{I} - \eta\mathbf{A}\mathbf{A}^T)^\tau(\mathbf{y} - \mathbf{A}\mathbf{x}_0) \\
&= (\mathbf{I} - \eta\mathbf{A}\mathbf{A}^T)^\tau\mathbf{y} \\
&= (\mathbf{I} - \eta\mathbf{U}\Sigma^2\mathbf{U}^T)^\tau\mathbf{y}
\end{aligned}$$

where we used that $\mathbf{x}_0 = 0$ and the SVD $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T$. Expanding $\mathbf{y}$ in terms of the singular vectors $\mathbf{u}_i$ (i.e., the columns of $\mathbf{U}$), as $\mathbf{y} = \sum_i \mathbf{u}_i\left\langle\mathbf{u}_i, \mathbf{y}\right\rangle$, and noting that $(\mathbf{I} - \eta\mathbf{U}\Sigma^2\mathbf{U}^T)^\tau = \sum_i(1 - \alpha\sigma_i^2)^\tau\mathbf{u}_i\mathbf{u}_i^T$ we get

$$\mathbf{r}_\tau = \sum_i(1 - \eta\sigma_i^2)^\tau\mathbf{u}_i\left\langle\mathbf{u}_i, \mathbf{y}\right\rangle,$$

as desired.

Proof of equation (2)**:** By proposition 1 and using that $\mathbf{y}^*$ and lies in the signal subspace

$$\mathbf{y}^* - \mathbf{A}\mathbf{x}_\tau = \sum_{i=1}^p \mathbf{u}_i(1 - \eta\sigma_i^2)^\tau\left\langle\mathbf{u}_i, \mathbf{y}^*\right\rangle + \sum_{i=1}^n \mathbf{u}_i((1 - \eta\sigma_i^2)^\tau - 1)\left\langle\mathbf{u}_i, \mathbf{z}\right\rangle.$$

By the triangle inequality,

$$\begin{aligned}\|\mathbf{y}^* - \mathbf{A}\mathbf{x}_\tau\| &\leq \left\| \sum_{i=1}^{p} \mathbf{u}_i (1-\eta\sigma_i^2)^\tau \langle \mathbf{u}_i, \mathbf{y}^* \rangle \right\| + \left\| \sum_{i=1}^{n} \mathbf{u}_i ((1-\eta\sigma_i^2)^\tau - 1) \langle \mathbf{u}_i, \mathbf{z} \rangle \right\| \\ &\leq (1-\eta\sigma_p^2)^\tau \|\mathbf{y}^*\| + \sqrt{\sum_{i=1}^{n} ((1-\eta\sigma_i^2)^\tau - 1)^2 \langle \mathbf{u}_i, \mathbf{z} \rangle^2},\end{aligned}$$

where the second inequality follows by using orthogonality of the $\mathbf{u}_i$ and by using $(1-\eta\sigma_i^2) \leq 1$, from $\eta \leq 1/\sigma_{\max}^2$. This concludes the proof of equation (2).

G The neural tangent kernel for convolutional generators

We first prove the closed form expression of the neural tangent kernel, i.e., equation (12). By the expression for the Jacobian in equation (8), we have that

$$\begin{aligned}\mathcal{J}(\mathbf{C})\mathcal{J}^T(\mathbf{C}) &= \sum_{\ell=1}^{k} v_\ell^2 \operatorname{diag}(\sigma'(\mathbf{U}\mathbf{c}_\ell)) \mathbf{U}\mathbf{U}^T \operatorname{diag}(\sigma'(\mathbf{U}\mathbf{c}_\ell)) \\ &= \sum_{\ell=1}^{k} v_\ell^2 \sigma'(\mathbf{U}\mathbf{c}_\ell) \sigma'(\mathbf{U}\mathbf{c}_\ell)^T \odot \mathbf{U}\mathbf{U}^T \\ &= \sigma'(\mathbf{U}\mathbf{C}) \operatorname{diag}(v_1^2, \ldots, v_k^2) \sigma'(\mathbf{U}\mathbf{C})^T \odot \mathbf{U}\mathbf{U}^T,\end{aligned}$$

where $\odot$ denotes the entrywise product of the two matrices. Then,

$$\mathbb{E}\left[\mathcal{J}(\mathbf{C})\mathcal{J}^T(\mathbf{C})\right] = \sum_{\ell=1}^{k} v_\ell^2 \mathbb{E}\left[\sigma'(\mathbf{U}\mathbf{c}_\ell)\sigma'(\mathbf{U}\mathbf{c}_\ell)^T\right] \odot \mathbf{U}\mathbf{U}^T. \tag{39}$$

Next, we have with (Daniely et al., 2016, Sec. 4.2) and using that the derivative of the ReLU function is the step function,

$$\left[\mathbb{E}\left[\sigma'(\mathbf{U}\mathbf{c}_\ell)\sigma'(\mathbf{U}\mathbf{c}_\ell)^T\right]\right]_{ij} = \frac{1}{2}\left(1 - \cos^{-1}\left(\frac{\langle \mathbf{u}_i, \mathbf{u}_j \rangle}{\|\mathbf{u}_i\|\|\mathbf{u}_j\|}\right)/\pi\right).$$

Using that $\|\mathbf{v}\|_2 = 1$, we get

$$\left[\mathbb{E}\left[\mathcal{J}(\mathbf{C})\mathcal{J}^T(\mathbf{C})\right]\right]_{ij} = \frac{1}{2}\left(1 - \cos^{-1}\left(\frac{\langle \mathbf{u}_i, \mathbf{u}_j \rangle}{\|\mathbf{u}_i\|\|\mathbf{u}_j\|}\right)/\pi\right)\langle \mathbf{u}_i, \mathbf{u}_j \rangle,$$

where $\mathbf{u}_i$ are the rows of $\mathbf{U}$. This concludes the proof of equation (12).

We next briefly comment on the singular value decomposition of a circulant matrix and explain that the singular vectors are given by Definition 1. Recall, that $\mathbf{U} \in \mathbb{R}^{n\times n}$ is a circulant matrix, implementing the convolution with a filter. Assume for simplicity that n is even. It is well known that the discrete Fourier transform diagonalizes $\mathbf{U}$, i.e.,

$$\mathbf{U} = \mathbf{F}^{-1}\hat{\mathbf{U}}\mathbf{F},$$

where $\mathbf{F} \in \mathbb{C}^{n\times n}$ is the DFT matrix with entries

$$[\mathbf{F}]_{jk} = e^{i2\pi jk/n}, \quad j,k = 0,\ldots,n-1,$$

and $\hat{\mathbf{U}}$ is a diagonal matrix with diagonal $\mathbf{F}\mathbf{u}$, where $\mathbf{u}$ is the first column of the circulant matrix $\mathbf{U}$. From this, we can compute the singular value decomposition of $\mathbf{U}$ by using that $\hat{\mathbf{u}} = \mathbf{F}\mathbf{u}$ is conjugate symmetric (since $\mathbf{u}$ is real) so that

$$[\hat{\mathbf{u}}]_{n-k+2} = \hat{\mathbf{u}}^*, \quad k = 2,\ldots,n/2.$$

Let $\mathbf{U} = \mathbf{W}\Sigma\mathbf{V}^T$ be the singular value decomposition of $\mathbf{U}$. The entries of the left singular vectors are given by the trigonometric basis functions defined in (4), and the singular values are given by the absolute values of $\hat{\mathbf{u}}$.

H Proofs of Lemmas for neural network denoisers (Proofs of auxiliary lemmas in Section E)

H.1 Proof of Lemma 3: Jacobian perturbation around initialization

The proof follows that of (Oymak & Soltanolkotabi, 2019, Lem. 6.9).

1. We start by relating the perturbation of the Jacobian to a perturbation of the activation patterns. For any $\mathbf{C}, \mathbf{C}'$, we have that

$$\|\mathcal{J}(\mathbf{C}) - \mathcal{J}(\mathbf{C}')\| \leq \|\mathbf{v}\|_\infty \|\mathbf{U}\| \max_j \left\|\sigma'(\mathbf{u}_j^T \mathbf{C}) - \sigma'(\mathbf{u}_j^T \mathbf{C}')\right\|^2. \tag{40}$$

To see this, first note that, by (8),

$$\mathcal{J}(\mathbf{C}) - \mathcal{J}(\mathbf{C}') = [\ldots v_j(\operatorname{diag}(\sigma'(\mathbf{U}\mathbf{c}_j)) - \operatorname{diag}(\sigma'(\mathbf{U}\mathbf{c}_j))')\mathbf{U} \ldots].$$

This in turn implies that

$$\begin{aligned}\|\mathcal{J}(\mathbf{C}) - \mathcal{J}(\mathbf{C}')\|^2 &= \left\|(\mathcal{J}(\mathbf{C}) - \mathcal{J}(\mathbf{C}'))(\mathcal{J}(\mathbf{C}) - \mathcal{J}(\mathbf{C}'))^T\right\| \\ &= \left\|(\sigma'(\mathbf{U}\mathbf{C}) - \sigma'(\mathbf{U}\mathbf{C}'))\operatorname{diag}(v_1^2, \ldots, v_k^2)(\sigma'(\mathbf{U}\mathbf{C}) - \sigma'(\mathbf{U}\mathbf{C}'))^T \odot \mathbf{U}\mathbf{U}^T\right\|^2 \\ &\overset{\text{(i)}}{\leq} \|\mathbf{U}\|^2 \max_j \left\|(\sigma'(\mathbf{u}_j^T \mathbf{C}) - \sigma'(\mathbf{u}_j^T \mathbf{C}'))\operatorname{diag}(\mathbf{v})\right\|^2 \\ &\leq \|\mathbf{v}\|_\infty^2 \|\mathbf{U}\|^2 \max_j \left\|\sigma'(\mathbf{u}_j^T \mathbf{C}) - \sigma'(\mathbf{u}_j^T \mathbf{C}')\right\|^2,\end{aligned}$$

where for (i) we used that for two positive semidefinite matrices $\mathbf{A}, \mathbf{B}$, $\lambda_{\max}(\mathbf{A} \odot \mathbf{B}) \leq \lambda_{\max}(\mathbf{A}) \max_i \mathbf{B}_{ii}$. This concludes the proof of equation (40).

2. Step one implies that we only need to control $\sigma'(\mathbf{U}\mathbf{c}_j)$ around a neighborhood of $\mathbf{c}_j'$. Since σ' is the step function, we need to count the number of sign flips between the matrices $\mathbf{U}\mathbf{C}$ and $\mathbf{U}\mathbf{C}'$. Let $|\mathbf{v}|_{\pi(q)}$ be the q-th smallest entry of $\mathbf{v}$ in absolute value.

Lemma 8. *Suppose that, for all i, and $q \leq k$,*

$$\|\mathbf{C} - \mathbf{C}'\| \leq \sqrt{q} \frac{|\mathbf{u}_i^T \mathbf{C}'|_{\pi(q)}}{\|\mathbf{u}_i\|}.$$

Then

$$\max_i \left\|\sigma'(\mathbf{u}_i^T \mathbf{C}) - \sigma'(\mathbf{u}_i^T \mathbf{C}')\right\| \leq \sqrt{2q}$$

Proof. Suppose that $\sigma'(\mathbf{u}_i^T \mathbf{C})$ and $\sigma'(\mathbf{u}_i^T \mathbf{C}')$ have $2q$ many different entries, then the conclusion of the statement would be violated. We show that this implies the assumption is violated as well, proving the statement by contraction. By the contradiction hypothesis,

$$\begin{aligned}\|\mathbf{C} - \mathbf{C}'\|^2 &\geq \left\|\mathbf{u}_i^T(\mathbf{C} - \mathbf{C}')\right\|^2 / \|\mathbf{u}_i\|^2 \\ &\geq q \frac{|\mathbf{u}_i^T \mathbf{C}'|_{\pi(q)}^2}{\|\mathbf{u}_i\|^2},\end{aligned}$$

where the last inequality follows by noting that at least $2q$ many entries have different signs, thus their difference is larger than their individual magnitudes, and at least q many individual magnitudes are lower bounded by the q-th smallest one. □

3. Next, we note that, with probability at least $1 - ne^{-kq^2/2}$, the q-th smallest entry of $\mathbf{u}_i^T \mathbf{C}' \in \mathbb{R}^k$ obeys

$$\frac{|\mathbf{u}_i^T \mathbf{C}'|_{\pi(q)}}{\|\mathbf{u}_i\|} \geq \frac{q}{2k} \nu \quad \text{for all } i = 1, \ldots, n. \tag{41}$$

We note that it is sufficient to prove this result for $\nu = 1$. This follows from anti-concentration of Gaussian random variables. Specifically, with the entries of $\mathbf{C}'$ being iid $\mathcal{N}(0,1)$ distributed, the entries of $\mathbf{g} = \mathbf{u}_i^T \mathbf{C}'/\|\mathbf{u}_i\| \in \mathbb{R}^k$ are iid standard Gaussian random variables as well. We show that with probability at least $1 - e^{-kq^2/2}$, at most q entries are larger than $\frac{q}{2k}$. Let γ_δ be the number for which $\mathrm{P}\left[|g_\ell| \le \gamma_\delta\right] \le \delta$, where g is a standard Gaussian random variable. Note that $\gamma_\delta \ge \sqrt{\pi/2}\delta \ge \delta$. Define the random variable

$$\delta_\ell = \begin{cases} 1 & \text{if } |g_\ell| \le \gamma_\delta, \\ 0 & \text{otherwise.} \end{cases}$$

with $\delta = \frac{q}{2k}$. With $\mathbb{E}\left[\delta_\ell\right] = \delta$, by Hoeffding's inequality,

$$\mathrm{P}\left[\sum_{\ell=1}^{k} \delta_\ell \ge m\right] = \mathrm{P}\left[\sum_{\ell=1}^{k} \delta_\ell - \mathbb{E}\left[\delta_\ell\right] \ge m/2\right] \le e^{-2k(m/2)^2} = e^{-km^2/2}. \tag{42}$$

Thus, with probability at least $1 - ke^{-kq^2/2}$ no more than m entries are smaller than $\gamma_\delta \ge \delta = \frac{q}{2k}$. The results now follows by taking the union bound over all $i = 1, \ldots, n$.

We are now ready to conclude the proof of the lemma. By equation (40),

$$\begin{aligned} \|\mathcal{J}(\mathbf{C}) - \mathcal{J}(\mathbf{C}')\| &\le \|\mathbf{v}\|_\infty \|\mathbf{U}\| \max_j \left\|\sigma'(\mathbf{u}_j^T \mathbf{C}) - \sigma'(\mathbf{u}_j^T \mathbf{C}')\right\| \\ &\le \|\mathbf{v}\|_\infty \|\mathbf{U}\| \sqrt{2q} \end{aligned}$$

provided that

$$\|\mathbf{C} - \mathbf{C}'\| \le \sqrt{q}\frac{q}{2k}\nu,$$

with probability at least $1 - ne^{-kq^2/2}$. Setting $q = (2kR)^{2/3}$ concludes the proof (note that the assumption $R \le \frac{1}{2}\sqrt{k}$ ensures $q \le k$).

H.2 Proof of Lemma 4: Bounded Jacobian

By the expression of the Jacobian in equation (8),

$$\begin{aligned} \|\mathcal{J}(\mathbf{C})\|^2 &= \left\|\mathcal{J}(\mathbf{C})\mathcal{J}(\mathbf{C})^T\right\| \\ &= \left\|\sigma'(\mathbf{U}\mathbf{C})\mathrm{diag}(v_1^2, \ldots, v_k^2)\sigma'(\mathbf{U}\mathbf{C})^T \odot \mathbf{U}\mathbf{U}^T\right\|^2 \\ &\overset{\text{(i)}}{\le} \|\mathbf{U}\|^2 \max_j \left\|\sigma'(\mathbf{u}_j^T \mathbf{C})\mathrm{diag}(\mathbf{v})\right\|_2^2 \\ &\le \|\mathbf{v}\|_2^2 \|\mathbf{U}\|^2, \end{aligned}$$

where for (i) we used that for two positive semidefinite matrices $\mathbf{A}, \mathbf{B}$, $\lambda_{\max}(\mathbf{A} \odot \mathbf{B}) \le \lambda_{\max}(\mathbf{A}) \max_i \mathbf{B}_{ii}$. To prove the second inequality note that

$$\begin{aligned} \left\|\mathbf{J}\mathbf{J}^T\right\| &= \|\mathbf{\Sigma}(\mathbf{U})\| \\ &= \|\mathbf{v}\|_2^2 \left\|\mathbb{E}_{\mathbf{c}\sim\mathcal{N}(0,1)}\left[\mathrm{ReLU}'(\mathbf{U}\mathbf{c})\mathrm{ReLU}'(\mathbf{U}\mathbf{c})^T\right] \odot \left(\mathbf{U}\mathbf{U}^T\right)\right\| \\ &\overset{\text{(i)}}{\le} \|\mathbf{v}\|_2^2 \left\|\mathbf{U}\mathbf{U}^T\right\| \\ &= \|\mathbf{v}\|_2^2 \left\|\mathbf{U}\mathbf{U}^T\right\|. \end{aligned}$$

Here, (i) follows from the fact that for two positive semidefinite matrices $\mathbf{A}, \mathbf{B}$, $\lambda_{\max}(\mathbf{A} \odot \mathbf{B}) \le \lambda_{\max}(\mathbf{A}) \max_i \mathbf{B}_{ii}$.

H.3 Proof of Lemma 5: Concentration lemma

We begin by defining the zero-mean random matrices

$$\mathbf{S}_\ell = v_\ell^2 \left(\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T - \mathbb{E}\left[\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T\right]\right) \odot \left(\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T\right).$$

With this notation,

$$\begin{aligned}\mathcal{J}(\mathbf{C})\mathcal{J}^T(\mathbf{C})-\mathbf{\Sigma}(\widetilde{\mathbf{U}})=&\sum_{\ell=1}^{k}v_\ell^2\left(\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T-\mathbb{E}\left[\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T\right]\right)\odot\left(\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T\right)\\=&\sum_{\ell=1}^{k}\mathbf{S}_\ell.\end{aligned}$$

To show concentration we use the matrix Hoeffding inequality. To this aim note that the summands are centered in the sense that $\mathbb{E}\left[\mathbf{S}_\ell\right]=\mathbf{0}$. Next note that

$$\begin{aligned}\left(\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T-\mathbb{E}\left[\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T\right]\right)\odot\left(\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T\right)\preceq&\left(\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T\right)\odot\left(\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T\right)\\=&\text{diag}\left(\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T\right)\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T\text{diag}\left(\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\right)\\\preceq&B^2\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T\end{aligned}$$

Similarly,

$$\begin{aligned}\left(\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T-\mathbb{E}\left[\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T\right]\right)\odot\left(\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T\right)\succeq&-\mathbb{E}\left[\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)\sigma'(\widetilde{\mathbf{U}}\mathbf{c}_\ell)^T\right]\odot\left(\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T\right)\\\succeq&-B^2\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T.\end{aligned}$$

Thus,

$$-v_\ell^2B^2\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T\preceq\mathbf{S}_\ell\preceq v_\ell^2B^2\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T.$$

Therefore, using $\mathbf{A}_\ell:=v_\ell^2B^2\widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T$ we have

$$\mathbf{S}_\ell^2\preceq\mathbf{A}_\ell^2,$$

and

$$\sigma^2:=\left\|\sum_{\ell=1}^{k}\mathbf{A}_\ell^2\right\|\le B^4\left(\sum_{\ell=1}^{k}v_\ell^4\right)\left\|\widetilde{\mathbf{U}}\right\|^4$$

To continue we will apply matrix Hoeffding inequality stated below.

Theorem 6 (Matrix Hoeffding inequality, Theorem 1.3 Tropp (2011))**.** *Consider a finite sequence $\mathbf{S}_\ell$ of independent, random, self-adjoint matrices with dimension n, and let $\{\mathbf{A}_\ell\}$ be a sequence of fixed self-adjoint matrices. Assume that each random matrix satisfies*

$$\mathbb{E}[\mathbf{S}_\ell]=\mathbf{0}\quad\textit{and}\quad\mathbf{S}_\ell^2\preceq\mathbf{A}_\ell^2\quad\textit{almost surely.}$$

Then, for all $t\ge0$,

$$\mathrm{P}\left[\left\|\sum_{\ell=1}^{k}\mathbf{S}_\ell\right\|\ge t\right]\le2ne^{-\frac{t^2}{8\sigma^2}}\quad\textit{where}\quad\sigma^2:=\left\|\sum_{\ell=1}^{k}\mathbf{A}_\ell^2\right\|.$$

Therefore, applying matrix Hoeffding inequality we get that

$$\mathrm{P}\left[\left\|\sum_{\ell=1}^{k}\mathbf{S}_\ell\right\|\ge t\right]\le2ne^{-\frac{t^2}{B^4\|\mathbf{v}\|_4^4\|\widetilde{\mathbf{U}}\|^4}},$$

which concludes the proof.

H.4 Proof of Lemma 7 (bound on initial residual)

Proof. Without loss of generality we prove the result for $\nu=1$. First note that by the triangle inequality

$$\|\mathbf{r}_0\|_2=\|\sigma(\mathbf{U}\mathbf{C})\mathbf{v}-\mathbf{y}\|_2\le\|\sigma(\mathbf{U}\mathbf{C})\mathbf{v}\|_2+\|\mathbf{y}\|_2.$$

We next bound $\|\sigma(\mathbf{UC})\mathbf{v}\|_2$. Consider the i-th entry of the vector $\sigma(\mathbf{UC})\mathbf{v} \in \mathbb{R}^n$, given by $\sigma(\mathbf{u}_i^T\mathbf{C})\mathbf{v}$, and note that $q_j = (\sigma(\mathbf{u}_i^T\mathbf{c}_j) - \sigma(\mathbf{u}_i^T\mathbf{c}_{n-j}))/\|\mathbf{u}_i\|_2$ is sub-Gaussian with parameter 2, i.e., $\mathrm{P}\left[|q_j| \geq t\right] \leq 2e^{-2t^2}$. It follows that

$$\mathrm{P}\left[\left|\sum_{j=1}^{k/2} q_j\right| \geq \beta\sqrt{k}\right] \leq 2e^{-\frac{\beta^2}{8}}.$$

Thus,

$$\mathrm{P}\left[\left|\sigma(\mathbf{u}_i^T\mathbf{C})\mathbf{v}\right| \geq \|\mathbf{u}_i\|_2 \,\xi\beta\right] \leq 2e^{-\frac{\beta^2}{8}},$$

where we used that $|v_j| = \xi/\sqrt{k}$. Taking a union bound over all n entries,

$$\mathrm{P}\left[\|\sigma(\mathbf{UC})\mathbf{v}\|_2^2 \geq \|\mathbf{U}\|_F^2 \,\xi^2\beta^2\right] \leq 2ne^{-\frac{\beta^2}{8}}.$$

Choosing $\beta = \sqrt{8\log(2n/\delta)}$ concludes the proof. □