DAEs for Linear Inverse Problems: Improved Recovery with Provable Guarantees

Jasjeet Dhaliwal and Kyle Hambrook
Department of Mathematics, San Jose State University, San Jose, U.S.A.

Keywords: Autoencoders, Generative Priors, Compressive Sensing, Inpainting, Superresolution.

Abstract: Generative priors have been shown to provide improved results over sparsity priors in linear inverse problems. However, current state of the art methods suffer from one or more of the following drawbacks: (a) speed of recovery is slow; (b) reconstruction quality is deficient; (c) reconstruction quality is contingent on a computationally expensive process of tuning hyperparameters. In this work, we address these issues by utilizing Denoising Auto Encoders (DAEs) as priors and a projected gradient descent algorithm for recovering the original signal. We provide rigorous theoretical guarantees for our method and experimentally demonstrate its superiority over existing state of the art methods in compressive sensing, inpainting, and super-resolution. We find that our algorithm speeds up recovery by two orders of magnitude (over 100x), improves quality of reconstruction by an order of magnitude (over 10x), and does not require tuning hyperparameters.

1 INTRODUCTION

Linear inverse problems can be formulated mathematically as $y = Ax + e$ where $y \in \mathcal{R}^m$ is the observed vector, $A \in \mathcal{R}^{m \times N}$ is the measurement process, $e \in \mathcal{R}^m$ is a noise vector, and $x \in \mathcal{R}^N$ is the original signal. The problem is to recover the signal x, given the observation y and the measurement matrix A. Such problems arise naturally in a wide variety of fields including image processing, seismic and medical tomography, geophysics, and magnetic resonance imaging. In this paper, we focus on three linear inverse problems encountered in image processing: compressive sensing, inpainting, and super-resolution. We motivate our method using the compressive sensing problem.

Sparsity Prior. The problem of compressive sensing assumes the matrix $A \in \mathcal{R}^{m \times N}$ is fat, i.e. $m < N$. Even when no noise is present ($y = Ax$), the system is under determined and the recovery problem is intractable. However, it has been shown that if the matrix A satisfies certain conditions such as the Restricted Isometry Property (RIP) and if x is known to be approximately sparse in some fixed basis, then x can typically be recovered even when $m \ll N$ (Tibshirani, 1996; Donoho et al., 2006; Candes et al., 2006).

Sparsity (or approximate sparsity) is a very restrictive condition to impose on the signal as it limits the applicability of recovery methods to a small subset of input domains. There has been considerable effort in using other forms of structured priors such as structured sparsity (Baraniuk et al., 2010), sparsity in tree-structured dictionaries (Peyre, 2010), and low-rank mixture of Gaussians (Chen et al., 2010). Although these efforts improve on the sparsity prior, they do not cater to signals that are not naturally sparse or structured-sparse.

Generative Prior. Bora et al. (Bora et al., 2017) address this issue by replacing the sparsity prior on x with a generative prior. In particular, the authors first train a generative model $f : \mathcal{R}^k \mapsto \mathcal{R}^N$ with $k < N$ that maps a lower dimensional latent space to the higher dimensional ambient space. This model is referred to as the generator. Next, they impose the prior that the original signal x lies in (or near) the range of f. Hence, the recovery problem reduces to finding the best approximation to x in $f(\mathcal{R}^k)$.

The quality of the generative prior depends on how well the training set captures the data distribution. Bora et al.(Bora et al., 2017) used a Generative Adversarial Network (GAN) as the generator, $G : \mathcal{R}^k \mapsto \mathcal{R}^N$, where $k < N$, to model the distribution of the training data and posed the following non-convex optimization problem $\hat{z} = \arg\min_{z \in \mathcal{R}^k} \left(\|AG(z) - y\|^2 + \lambda \|z\|^2 \right)$[1]. such that $G(\hat{z})$ is

[1] We use $\|.\|$ to denote the ℓ_2-norm throughout the paper

Dhaliwal, J. and Hambrook, K.
DAEs for Linear Inverse Problems: Improved Recovery with Provable Guarantees.
DOI: 10.5220/0010804500003124
In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2022) - Volume 4: VISAPP*, pages 97-105
ISBN: 978-989-758-555-5; ISSN: 2184-4321

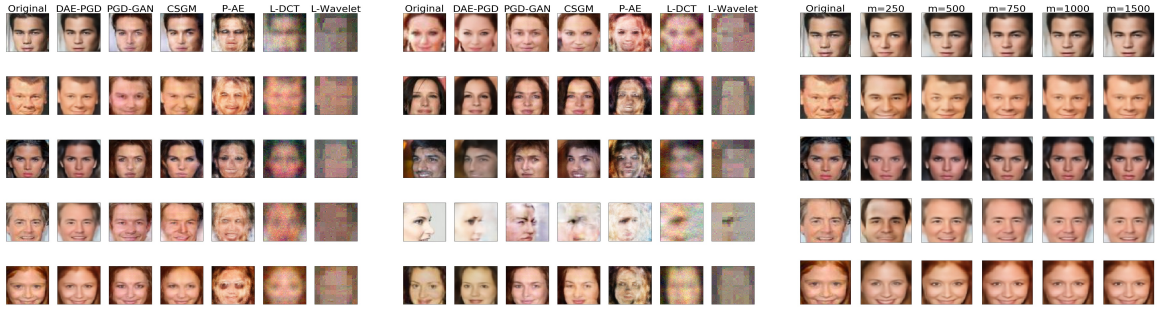

Figure 1: CS on CelebA without noise for $m = 1000$ (left), CS on CelebA with noise for $m = 1000$ (middle), CS on CelebA for various m using DAE-PGD (right). The left and middle images qualitatively capture the 10x improvement in reconstruction error. The right image shows how DAE-PGD reconstructions capture finer grained details as m increases.

treated as the approximation to x. The authors provided recovery guarantees for their methods and validated the efficacy of using generative priors by showing that their method required 5-10x fewer measurements than Lasso (with a sparsity constraint) (Tibshirani, 1996) while yielding the same accuracy in recovery. However, since the problem is non-convex and requires a search over $\mathcal{R}^k$, it is computationally expensive and the reconstruction quality depends on the initialization vector $z \in \mathcal{R}^k$.

Since then, there have been significant efforts to improve recovery results using neural networks as generative priors (Adler and Öktem, 2017; Fan et al., 2017; Gupta et al., 2018; Liu et al., 2017; Mardani et al., 2018; Metzler et al., 2017; Mousavi et al., 2017; Rick Chang et al., 2017a; Shah and Hegde, 2018; Yeh et al., 2017; Raj et al., 2019; Heckel and Hand, 2018). Shah et al. (Shah and Hegde, 2018) extended the work of (Bora et al., 2017) by training a generator G and using a projected gradient descent algorithm that consists of a gradient descent step $w_t = x_t - \eta A^T(Ax_t - y)$ followed by a projection step $x_{t+1} = G(\arg\min_{z \in \mathcal{R}^k} \|G(z) - w_t\|^2)$The core idea being that the estimate w_t is improved by projecting it onto the range of G. However, since their method requires solving a non-convex optimization problem at every update step, it also leads to slow recovery.

Raj et al. (Raj et al., 2019) enhanced the results of (Shah and Hegde, 2018) by eliminating the expensive non-convex optimization based projection step with one that is an order of magnitude cheaper. In particular, they trained a GAN G to model the data distribution and also trained a pseudo-inverse GAN $G^\ddagger$ that learned a mapping from the ambient space to the latent space. Next, they used the projection step: $x_{t+1} = G(G^\ddagger(w_t))$. By eliminating the need to solve a non-convex optimization problem to update x_{t+1}, they were able to attain a significant speed up in the running time of the recovery algorithm.

However, the recovery algorithm of (Raj et al., 2019) has two main drawbacks. First, training two networks: G and $G^\ddagger$ makes the training process and the projection step unnecessarily convoluted. Second, their recovery guarantees only hold when the learning rate $\eta = \frac{1}{\beta}$, where β is a RIP-style constant of the matrix A. Since it is NP-hard to estimate the constant β (Bandeira et al., 2013), it follows that setting $\eta = \frac{1}{\beta}$ is NP-hard as well. [2].

DAE Prior. In an effort to address the aforementioned issues, we propose to use a DAE (Vincent et al., 2008) prior in lieu of the generative prior introduced by Bora et al. (Bora et al., 2017). It has previously been shown that DAEs not only capture useful structure of the data distribution (Vincent et al., 2010) but also implicitly capture properties of the data-generating density (Alain and Bengio, 2014; Bengio et al., 2013). Moreover, as DAEs are trained to remove noise from vectors sampled from the input distribution, they integrate naturally with gradient descent algorithms that lead to noisy approximations at each time step.

We replace the generator G used in Bora et al. (Bora et al., 2017) with a DAE $F : \mathcal{R}^N \mapsto \mathcal{R}^N$ such that the range of F contains the vectors from the original data generating distribution. We then impose the prior that the original signal x lies in the range of F and utilize Algorithm 1 to recover an approximation to x. We provide theoretical recovery guarantees and find that our framework speeds up recovery by two orders of magnitude (over 100x), improves quality of reconstruction by an order of magnitude (over 10x), and does not require tuning hyperparameters. We note that Peng et al. (Peng et al., 2020) have recently utilized Auto Encoders (AE) instead of DAEs as in our

[2]We observed this problem when trying to reproduce the experimental results of (Raj et al., 2019). Specifically, we tried an exhaustive grid-search for η but each value led to poor reconstruction quality.

approach. However, unlike our work, their theoretical results rely on the measurement matrix being Gaussian and we find their experimental results are inferior to those of Algorithm 1 (Section 3.2).

2 ALGORITHM AND RESULTS

2.1 Denoising Auto Encoder

A DAE is a non-linear mapping $F : \mathcal{R}^N \mapsto \mathcal{R}^N$ that can be written as a composition of two non-linear mappings - an encoder $E : \mathcal{R}^N \mapsto \mathcal{R}^k$ where $k < N$ and a decoder $D : \mathcal{R}^k \mapsto \mathcal{R}^N$. Therefore, $F(x) = (D \circ E)(x)$. Given a set of n samples from a domain of interest $\{x_i\}_{i=1}^n$, the training set X is created by adding Gaussian noise to the original samples. That is, $X = \{x_i'\}_{i=1}^n$, where $x_i' = x_i + e_i$ and $e_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$.

The loss function for training F is the Mean Squared Error (MSE) loss defined as : $L_F(X) = \frac{1}{n}\sum_{i=1}^n \|F(x_i') - x_i\|^2$. The training procedure uses gradient descent to minimize $L_F(X)$ with back-propagation.

2.2 Algorithm

Recall that in the linear inverse problem $y = Ax + e$, our goal is to recover an approximation $\hat{x}$ to x such that $\hat{x}$ lies in the range of F. Thus we aim to find $\hat{x}$ such that $\hat{x} = \underset{z \in F(\mathcal{R}^N)}{\arg\min} \|Az - y\|^2$ As in (Shah and Hegde, 2018; Raj et al., 2019), we use a projected gradient descent algorithm. Given an estimate x_t at iteration t, we compute a gradient descent step for solving the unrestricted problem: $\underset{z \in \mathcal{R}^N}{\text{minimize}} \|Az - y\|^2$ as: $w_t \leftarrow x_t - \eta A^T(Ax_t - y)$ Next we project w_t onto the range of F to satisfy our prior: $x_{t+1} = F(w_t)$ Note that, compared to (Shah and Hegde, 2018; Raj et al., 2019), the projection step does not require solving a non-convex optimization problem.

Now suppose that the domain of interest is represented by the set $D \subseteq \mathcal{R}^N$. Then, given a vector $x' = x + e$, where $x \in D$, and $e \in \mathcal{R}^N$ is an unknown noise vector, the success of our method depends on how small the error $\|F(x') - x\|$ is. If the training set X captures the domain of interest well and if the training procedure utilizes a diverse enough set of noise vectors $\{e_i\}_{i=1}^N$, then we expect $\|F(x') - x\|$ to be small. Consequently, we expect the projection step of Algorithm 1 to yield vectors in or close to D. We provide the complete algorithm below.

Algorithm 1: DAE-PGD.

Input: $y \in \mathcal{R}^m, A \in \mathcal{R}^{m \times N}, f : \mathcal{R}^N \to \mathcal{R}^N$, $T \in \mathbb{Z}_+, \eta \in \mathcal{R}_{>0}$
Output: x_T

$t \leftarrow 0, x_0 \leftarrow 0$
while $t < T$ **do**
  $w_t \leftarrow x_t - \eta A^T(Ax_t - y)$
  $x_{t+1} \leftarrow f(w_t)$
return x_T

2.3 Theoretical Results

We begin by introducing two standard definitions required to provide recovery guarantees.

Definition 1 (RIP(S, δ))**.** *Given $S \subseteq \mathcal{R}^N$ and $\delta > 0$, a matrix $A \in \mathcal{R}^{m \times N}$ satisfies the RIP(S, δ) property if*

$$(1-\delta)\|x_1 - x_2\|^2 \leq \|A(x_1 - x_2)\|^2 \leq (1+\delta)\|x_1 - x_2\|^2$$

for all $x_1, x_2 \in S$.

A variation of the RIP(S, δ) property for sparse vectors was first introduced by Candes et al. in (Candes and Tao, 2005) and has been shown to be a sufficient condition in proving recovery guarantees using ℓ_1-minimization methods (Foucart and Rauhut, 2017). Next, we define an Approximate Projection (AP) property and provide an interpretation that elucidates its role in the results of Theorem 6. [3].

Definition 2 (AP(S, α))**.** *Let $\alpha \geq 0$. A mapping $f : \mathcal{R}^N \to S \subseteq \mathcal{R}^N$ satisfies AP(S, α) if*

$$\|w - f(w)\|^2 \leq \|w - x\|^2 + \alpha\|f(w) - x\|$$

for every $w \in \mathcal{R}^N$ and $x \in S$.

We now explain the significance of Def. 5. Let $x^* = \underset{z \in S}{\arg\min} \|w - z\|$ and observe

$$\|w - f(w)\|^2 \leq (\|w - x^*\| + \|f(w) - x^*\|)^2 \qquad (1)$$

Hence, $\alpha \leq \|f(w) - x^*\| + 2\|w - x^*\|$ is needed to ensure the RHS of Def. 5 is bounded by the RHS of (1). In other words, for α to be small, the projection error $\|f(w) - x^*\|$ as well as distance of w to S need to be small. Since the DAE F learns to minimize $\|F(w) - x^*\|^2$ (Section 2.1), we expect a small projection error.

Theorem 3. *Let $f : \mathcal{R}^N \to S \subseteq \mathcal{R}^N$ satisfy AP(S, α) and let $A \in \mathcal{R}^{m \times N}$ be a matrix with $\|A\|^2 \leq M$ that*

[3] Various flavors of the AP(S, α) property have been used in previous works, such as Shah et al. (Shah and Hegde, 2018) and Raj et al. (Raj et al., 2019).

satisfies RIP(S,δ)*. If* $y = Ax$ *with* $x \in S$*, the recovery error of Algorithm 1 is bounded as:*

$$\|x_T - x\| \leq (2\gamma)^T \|x_0 - x\| + \alpha\left(\frac{1-(2\gamma)^T}{1-(2\gamma)}\right) \quad (2)$$

where $\gamma = \sqrt{\eta^2 M(1+\delta) + 2\eta(\delta-1)+1}$.

Theorem 6 tells us that, if $\gamma < \frac{1}{2}$, then for large T, the recovery error is essentially $\alpha/(1-2\gamma)$. Note that the requirement $\gamma < \frac{1}{2}$ is satisfied for a large range of values of η as long as δ is sufficiently small [4]. Hence, as long as the value of α is small, we expect to see a small recovery error.

We now compare the above results to Theorem 1 of (Raj et al., 2019), Theorem 2.2 of (Shah and Hegde, 2018) and Theorem 1 of (Peng et al., 2020). As mentioned in Section 1, convergence in Theorem 1 of (Raj et al., 2019) is only guaranteed when $\eta = \frac{1}{\beta}$, which is a much more restrictive condition on η than Theorem 6 provides. In fact, β is a RIP-style constant that is NP-hard to find (Bandeira et al., 2013) which makes setting the value of $\eta = \frac{1}{\beta}$ NP-hard as well. The results of Theorem 2.2 from (Shah and Hegde, 2018) require a less restrictive constraint on η but do require a stricter constraint on $\|A\|^2 \leq \omega$, where ω is a RIP-style constant for A. In contrast, the results of Theorem 6 do not impose a strict condition on $\|A\|^2$. Finally, the proof of Theorem 1 of (Peng et al., 2020) relies on the matrix A being Gaussian. We do not impose such a constraint.

3 EXPERIMENTS

We provide experimental results for the problems of compressive sensing, inpainting, and super-resolution. We refer to the results of Algorithm 1 as DAE-PGD and compare its results to the methods of Bora et al. (Bora et al., 2017) (CSGM), and Shah et al. (Shah and Hegde, 2018), (PGD-GAN), and Peng et al (Peng et al., 2020) (P-AE). Although the work of Raj et al. (Raj et al., 2019) is the closest to our method, we do not include comparisons to their work as we were unable to reproduce their results [5].

[4]For instance, random Gaussian matrices yield small values for δ with high probability (Foucart and Rauhut, 2017)

[5]We used their code, their trained models, their recovery algorithm, and a grid search for η but the reconstructed images were of very poor quality. We also reached out to the authors but they did not have the exact values of η that were used in their experiments.

3.1 Setup

Datasets. Our experiments are conducted on the MNIST (LeCun,) and CelebA (Liu et al., 2015) datasets. The MNIST dataset consists of 28×28 greyscale images of digits with 50,000 training and 10,000 test samples. We report results for a random subset of the test set. The CelebA dataset consists of more than 200,000 celebrity images. We pre-processes each image to a size of $64 \times 64 \times 3$ and use the first 160, 000 images as the training set and a random subset of the remaining 40,000+ images as the test set.

Network Architecture. The network architectures for our DAEs are inspired by the Variational Auto Encoder architecture from Fig 2. of (Hou et al., 2017) with a few key changes. We replace the Leaky Relu activation with Relu, we add the two outputs of the encoder to get the latent representation z, and we alter the kernel sizes as well as the convolution strides of the network as described in the Appendix.

Training. We use the Adam optimizer (Kingma and Ba, 2014) to minimize the MSE loss function with learning rate 0.01 and a batch size of 128. We train the CelebA network for 400 epochs and the MNIST network for 100 epochs.

In an effort to ensure that $\|A(x') - x\|$ defined in Section 2.2 is small, we split the training set into 5 equal sized subsets. For each distinct subset, we sample the noise vectors from a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ with a distinct value for σ for each subset. The five different values for σ that we use are $\{0.25, 0.5, 0.75, 1.0, 1.25\}$.

All of our experiments were conducted on a Tesla M40 GPU with 12 GB of memory using Keras (Chollet, 2015) and Tensorflow (Abadi et al., 2015) libraries. The code to reproduce our results is available here.

3.2 Compressive Sensing

We consider the problem of compressive sensing without noise: $y = Ax$ and with noise: $y = Ax + e$, with $e \sim \mathcal{N}(0, 0.25)$. We use m to denote the number of observed measurements in our results (i.e. $y \in \mathcal{R}^m$). As done in previous works (Bora et al., 2017; Shah and Hegde, 2018; Raj et al., 2019), the matrix $A \in \mathcal{R}^{m \times N}$ is chosen to be a random Gaussian matrix with $A_{ij} \sim \mathcal{N}(0, \frac{1}{m})$. Finally, we set the learning rate of Algorithm 1 as $\eta = 1$. Note that in both (with and w/out noise) cases, we also include recovery results for the Lasso algorithm (Tibshirani, 1996) with a DCT basis (L-DCT) and with a wavelet basis (L-

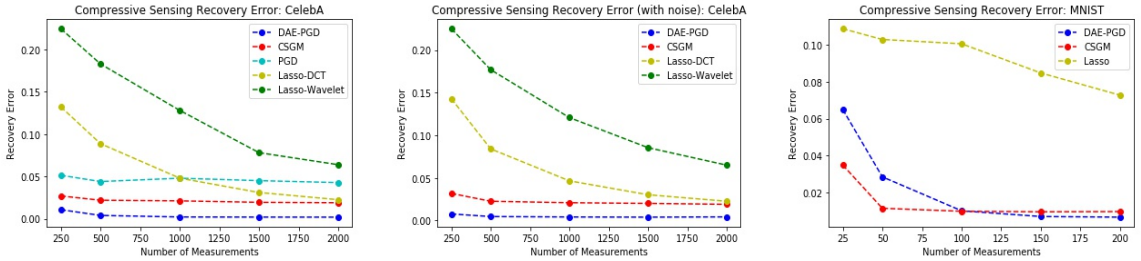

Figure 2: Compressive Sensing recovery error: $\|x-\hat{x}\|^2$. Left: CelebA without noise - DAE-PGD shows over 10x improvement. Middle: CelebA with noise - DAE-PGD shows over 10x improvement. Right: MNIST without noise - DAE-PGD beats CSGM for $m > 100$.

Wavelet).

We begin with CelebA. Figure 1 provides a qualitative comparison of reconstruction results for $m = 1000$. We observe that DAE-PGD provides the best quality reconstructions and is able to reproduce even fine grained details of the original images such as eyes, nose, lips, hair, texture, etc. Indeed the high quality reconstructions support the case that the DAE has a small α as per Def. 5. For a quantitative comparison, we turn to Figure 2 which plots the average squared reconstruction error $\|x-\hat{x}\|^2$ for each algorithm at different values of m. Note that DAE-PGD provides more than 10x improvement in the squared reconstruction error.

In order to capture how the quality of reconstruction degrades as the number of measurements decrease, we refer to Figure 1, which shows reconstructions for different values of m. We observe that even though reconstructions with a small number of measurements capture the essence of the original images, the fine grained details are captured only as the number of measurements increase.

We show a similar comparison for MNIST in Figure 3

Table 1: Average running times (in seconds) for the Compressive Sensing problem (w/out noise) on the CelebA dataset.

m	CGSM	PGD-GAN	DAE-PGD	Speedup
250	53.78	48.40	0.07	692x
500	59.81	48.46	0.09	538x
1000	81.08	48.46	0.11	440x
1500	92.68	48.50	0.14	346x
2000	107.41	48.56	0.21	230x

We now turn to the speed of reconstruction. Table 1 shows that our method provides speedups of over 100x as compared to PGD-GAN and CSGM [6].

[6]CSGM is executed for 500 max iterations with 2 restarts and PGD-GAN is executed for 100 max iterations and 1 restart.

3.3 Inpainting

Inpainting is the problem of recovering the original image, given an occluded version of it. Specifically, the observed image y consists of occluded (or masked) regions created by applying a pixel-wise mask A to the original image x. We use m to refer to the size of mask that occludes a $m \times m$ region of the original image x.

We present recovery results for CelebA with $m = 10$ in Figure 4 and observe that DAE-PGD is able to recovery a high quality approximation to the original image and outperforms CSGM in all cases. Figure 4 also captures how recovery is affected by different mask sizes. As in the compressive sensing problem, we find that DAE-PGD reconstructions capture the fine-grained details of each image. Figure 4 also reports the result for the MNIST dataset. Even though DAE-PGD outperforms CSGM, we see that the recovery quality of DAE-PGD degrades considerably when $m = 15$. We hypothesize this is due to the structure of MNIST images. In particular, since MNIST images are grayscale with most of the pixels being black, putting a 15×15 black patch on the small area displaying the number makes the reconstruction problem considerably more difficult. This causes considerable degradation in reconstruction quality for larger mask sizes.

3.4 Super-resolution

Super-resolution is the problem of recovering the original image from a smaller and lower-resolution version. We create this smaller and lower-resolution image by taking the spatial averages of $f \times f$ pixel values where f is the ratio of downsampling. This results in blurring a $f \times f$ region followed by downsampling the image. We test our algorithm with $f = 2,3,4$ corresponding to $4\times, 9\times$, and $16\times$ smaller image sizes, respectively.

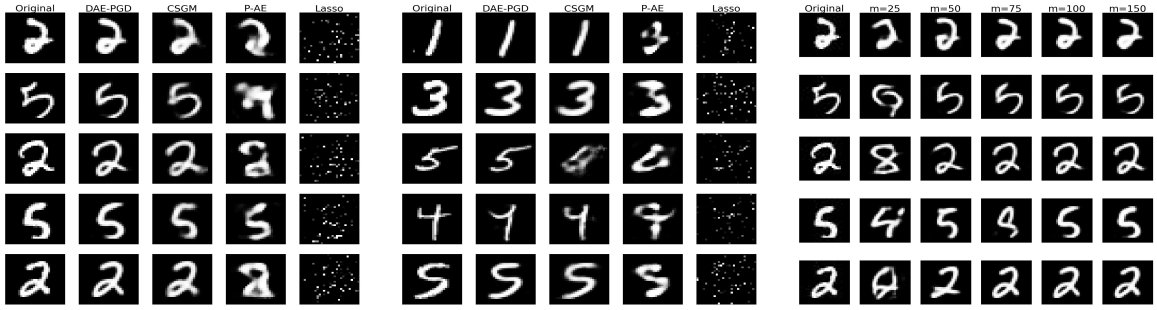

Figure 3: CS on MNIST without noise for $m = 100$ (left), CS on MNIST with noise for $m = 100$ (middle), CS on CelebA for various m using DAE-PGD (right). The left and middle images qualitatively capture the 100x improvement in reconstruction error. The right image shows how DAE-PGD reconstructions capture finer grained details as m increases.

Figure 4: Inpainting. Left: CelebA reconstructions for $m = 10$. Middle-Left: DAE-PGD CelebA reconstructions for different m. Middle-Right: MNIST reconstructions for $m = 5$. Right: DAE-PGD MNIST reconstructions for different m.

The reconstruction results are provided in 5. We see that DAE-PGD provides higher quality reconstruction for $f = 2$ for both CelebA and MNIST. Moreover, reconstruction quality degrades gracefully for CelebA for increasing values of f. However, in the case of MNIST, reconstruction quality degrades considerably when $f = 4$. Noting that $f = 4$ only gives 16 measurements (i.e. $y \in \mathcal{R}^{16}$), we hypothesize that 16 measurements may not contain enough signal [7] to accurately reconstruct the original images.

4 RELATED WORK

Compressive Sensing. The field of compressive sensing was initiated with the work of (Candès et al., 2006) and (Donoho et al., 2006) where provided recovery results for sparse signals with a random measurement matrix. Some of the earlier work in extending compressive sensing to perform stable recovery with deterministic matrices was done by (Candes and Tao, 2005) and (Candes et al., 2006), where a sufficient condition for recovery was satisfaction of a restricted isometry hypothesis. (Blumensath and Davies, 2009) introduced IHT as an algorithm to recover sparse signals which was later modified in (Baraniuk et al., 2010) to reduce the search space as long as the sparsity was structured.

Generative Priors. Following the lead of (Bora et al., 2017), there have been significant efforts to improve on previous recovery results using neural networks as generative models (Adler and Öktem, 2017; Fan et al., 2017; Gupta et al., 2018; Liu et al., 2017; Mardani et al., 2018; Metzler et al., 2017; Mousavi et al., 2017; Rick Chang et al., 2017a; Shah and Hegde, 2018; Yeh et al., 2017; Raj et al., 2019; Heckel and Hand, 2018). One line of work (Jagatap and Hegde, 2019; Heckel and Hand, 2018) extends the efforts of Bora et al. (Bora et al., 2017) by fixing a seed z and finding the weights $\hat{w}$ of an untrained neural network G in the optimization problem $\hat{w} = \arg\min_{w \in \mathcal{R}^l} \|AG(w,z) - y\|^2$. However, the optimization problem is highly non-convex and requires a large number of iterations with multiple restarts. Another line of work, (Mousavi et al., 2017; Mousavi and Baraniuk, 2017) trains a neural network to model the transformation $f(y) = \hat{x}$ where $\hat{x}$ is the approximation to the original input x. This approach is limited as a) the inverse mapping is non-trivial to learn and b) will only work for a fixed measurement mechanism.

Denoisers in Linear Inverse Problems. Given the success of denoisers in image processing tasks such as image denoising (Wang et al., 2018; Guo

[7] Consider compressive sensing with sparsity constraints where recovery guarantees hold when $m \geq Cs\ln(\frac{N}{s})$ (Foucart and Rauhut, 2017).

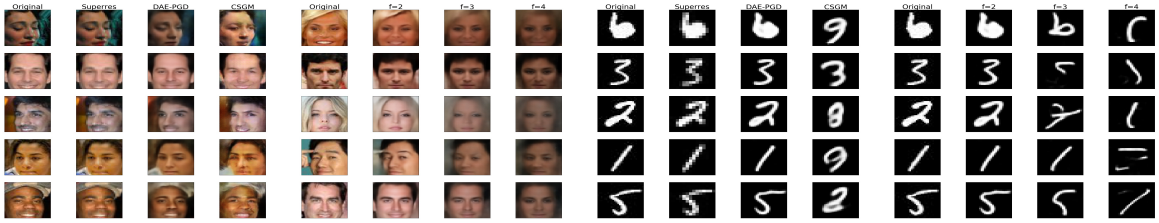

Figure 5: Super-resolution. Left: CelebA reconstructions for $f = 2$. Middle-Left: DAE-PGD CelebA reconstructions for different f. Middle-Right: MNIST reconstructions for $f = 2$. Right: DAE-PGD MNIST reconstructions for different f.

et al., 2019; Rick Chang et al., 2017b) and image super-resolution (Sønderby et al., 2016) to yield good results, (Venkatakrishnan et al., 2013) introduced denoisiers as plug-and-play (PnP) proximal operators in solving linear inverse problems via alternating directions method of multipliers (ADMM). (Ryu et al., 2019) extended this work by investigating convergence properties of ADMM methods asked and showed that if the denoiser was close to the identity map, then PnP methods are contractive iterations that converge with bounded error.

(Rick Chang et al., 2017b) showed that neural network based denoisers (such as DAEs) with ADMM could achieve state of the art results for a wide array of linear inverse problems. They also showed that if the gradient of the proximal operator (denoiser) is Lipschitz continuous, ADMM has a fixed point. (Xu et al., 2020) analyzed convergence results for minimum mean squared error (MMSE) denoisers used in iterative shrinkage/thresholding algorithm (ISTA). They showed that the iterates produced by ISTA with an MMSE denoiser converge to a stationary point of some global cost function. (Meinhardt et al., 2017) demonstrated that using a fixed denoising network as a proximal operator in the primal-dual hybrid gradient (PDHG) method yields state-of-the-art results. (González et al., 2021) used variational auto encoders (VAEs) as priors defined an optimization method JPMAP that performs Joint Posterior Maximization using an the VAE prior. They showed theoretical and experimental evidence that the proposed objective function satisfies a weak bi-convexity property which is sufficient to guarantee that the optimization scheme converges to a stationary point.

5 CONCLUSION

We introduced DAEs as priors for general linear inverse problems and provided experimental results for the problems of compressive sensing, inpainting, and super-resolution on the CelebA and MNIST datasets. Utilizing a projected gradient descent algorithm for recovery, we provided rigorous theoretical guarantees for our framework and showed that our recovery algorithm does not impose strict constraints on the learning rate and hence eliminates the need to tune hyperparameters. We compared our framework to state of the art methods experimentally and found that our recovery algorithm provided a speed up of over two orders of magnitude and an order of magnitude improvement in reconstruction quality.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Adler, J. and Öktem, O. (2017). Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007.

Alain, G. and Bengio, Y. (2014). What regularized auto-encoders learn from the data-generating distribution. *The Journal of Machine Learning Research*, 15(1):3563–3593.

Bandeira, A. S., Dobriban, E., Mixon, D. G., and Sawin, W. F. (2013). Certifying the restricted isometry property is hard. *IEEE transactions on information theory*, 59(6):3448–3450.

Baraniuk, R. G., Cevher, V., Duarte, M. F., and Hedge, C. (2010). Model-based compressive sensing. *IEEE Transactions on Information Theory*, 56(4):1982–2001.

Bengio, Y., Yao, L., Alain, G., and Vincent, P. (2013). Generalized denoising auto-encoders as generative models. *Advances in neural information processing systems*, 26:899–907.

Blumensath, T. and Davies, M. E. (2009). Iterative hard thresholding for compressed sensing. *Applied and computational harmonic analysis*, 27(3):265–274.

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

Candes, E. and Tao, T. (2005). Decoding by linear programming. *arXiv preprint math/0502327*.

Candès, E. J., Romberg, J., and Tao, T. (2006). Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509.

Candes, E. J., Romberg, J. K., and Tao, T. (2006). Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 59(8):1207–1223.

Chen, M., Silva, J., Paisley, J., Wang, C., Dunson, D., and Carin, L. (2010). Compressive sensing on manifolds using a nonparametric mixture of factor analyzers: Algorithm and performance bounds. *IEEE Transactions on Signal Processing*, 58(12):6140–6155.

Chollet, F. (2015). keras. `https://github.com/fchollet/keras`.

Donoho, D. L. et al. (2006). Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306.

Fan, K., Wei, Q., Carin, L., and Heller, K. A. (2017). An inner-loop free solution to inverse problems using deep neural networks. *Advances in Neural Information Processing Systems*, 30:2370–2380.

Foucart, S. and Rauhut, H. (2017). *A Mathematical Introduction to Compressive Sensing*.

González, M., Almansa, A., and Tan, P. (2021). Solving inverse problems by joint posterior maximization with autoencoding prior. *arXiv preprint arXiv:2103.01648*.

Guo, B., Han, Y., and Wen, J. (2019). Agem: Solving linear inverse problems via deep priors and sampling. volume 32, pages 547–558.

Gupta, H., Jin, K. H., Nguyen, H. Q., McCann, M. T., and Unser, M. (2018). Cnn-based projected gradient descent for consistent ct image reconstruction. *IEEE transactions on medical imaging*, 37(6):1440–1453.

Heckel, R. and Hand, P. (2018). Deep decoder: Concise image representations from untrained non-convolutional networks. *arXiv preprint arXiv:1810.03982*.

Hou, X., Shen, L., Sun, K., and Qiu, G. (2017). Deep feature consistent variational autoencoder. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1133–1141. IEEE.

Jagatap, G. and Hegde, C. (2019). Algorithmic guarantees for inverse imaging with untrained network priors. In *Advances in Neural Information Processing Systems*, pages 14832–14842.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

LeCun, Y. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*.

Liu, D., Wen, B., Liu, X., Wang, Z., and Huang, T. S. (2017). When image denoising meets high-level vision tasks: A deep learning approach. *arXiv preprint arXiv:1706.04284*.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738.

Mardani, M., Sun, Q., Donoho, D., Papyan, V., Monajemi, H., Vasanawala, S., and Pauly, J. (2018). Neural proximal gradient descent for compressive imaging. In *Advances in Neural Information Processing Systems*, pages 9573–9583.

Meinhardt, T., Moller, M., Hazirbas, C., and Cremers, D. (2017). Learning proximal operators: Using denoising networks for regularizing inverse imaging problems. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1781–1790.

Metzler, C., Mousavi, A., and Baraniuk, R. (2017). Learned d-amp: Principled neural network based compressive image recovery. In *Advances in Neural Information Processing Systems*, pages 1772–1783.

Mousavi, A. and Baraniuk, R. G. (2017). Learning to invert: Signal recovery via deep convolutional networks. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 2272–2276. IEEE.

Mousavi, A., Dasarathy, G., and Baraniuk, R. G. (2017). Deepcodec: Adaptive sensing and recovery via deep convolutional neural networks. *arXiv preprint arXiv:1707.03386*.

Peng, P., Jalali, S., and Yuan, X. (2020). Solving inverse problems via auto-encoders. *IEEE Journal on Selected Areas in Information Theory*, 1(1):312–323.

Peyre, G. (2010). Best basis compressed sensing. *IEEE Transactions on Signal Processing*, 58(5):2613–2622.

Raj, A., Li, Y., and Bresler, Y. (2019). Gan-based projector for faster recovery with convergence guarantees in linear inverse problems. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5602–5611.

Rick Chang, J., Li, C.-L., Poczos, B., Vijaya Kumar, B., and Sankaranarayanan, A. C. (2017a). One network to solve them all–solving linear inverse problems using deep projection models. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5888–5897.

Rick Chang, J. H., Li, C.-L., Poczos, B., Vijaya Kumar, B. V. K., and Sankaranarayanan, A. C. (2017b). One network to solve them all – solving linear inverse problems using deep projection models. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Ryu, E., Liu, J., Wang, S., Chen, X., Wang, Z., and Yin, W. (2019). Plug-and-play methods provably converge with properly trained denoisers. In *International Conference on Machine Learning*, pages 5546–5557. PMLR.

Shah, V. and Hegde, C. (2018). Solving linear inverse problems using gan priors: An algorithm with provable guarantees. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4609–4613. IEEE.

Sønderby, C. K., Caballero, J., Theis, L., Shi, W., and Huszár, F. (2016). Amortised map inference for image super-resolution. *arXiv preprint arXiv:1610.04490*.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Venkatakrishnan, S. V., Bouman, C. A., and Wohlberg, B. (2013). Plug-and-play priors for model based reconstruction. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 945–948. IEEE.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.-A., and Bottou, L. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12).

Wang, Y., Liu, Q., Zhou, H., and Wang, Y. (2018). Learning multi-denoising autoencoding priors for image super-resolution. *Journal of Visual Communication and Image Representation*, 57:152–162.

Xu, X., Sun, Y., Liu, J., Wohlberg, B., and Kamilov, U. S. (2020). Provable convergence of plug-and-play priors with mmse denoisers. *IEEE Signal Processing Letters*, 27:1280–1284.

Yeh, R. A., Chen, C., Yian Lim, T., Schwing, A. G., Hasegawa-Johnson, M., and Do, M. N. (2017). Semantic image inpainting with deep generative models. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5485–5493.

APPENDIX

We begin by introducing two standard definitions required to provide recovery guarantees.

Definition 4 (RIP(S,δ))**.** *Given $S \subseteq \mathcal{R}^N$ and $\delta > 0$, a matrix $A \in \mathcal{R}^{m\times N}$ satisfies the RIP(S,δ) property if*

$$(1-\delta)\|x_1-x_2\|^2 \le \|A(x_1-x_2)\|^2 \le (1+\delta)\|x_1-x_2\|^2$$

for all $x_1, x_2 \in S$.

A variation of the RIP(S,δ) property for sparse vectors was first introduced by Candes et al. in (Candes and Tao, 2005) and has been shown to be a sufficient condition in proving recovery guarantees using ℓ_1-minimization methods (Foucart and Rauhut, 2017). Next, we define an Approximate Projection (AP) property and provide an interpretation that elucidates its role in the results of Theorem 6. [8].

Definition 5 (AP(S, α))**.** *Let $\alpha \ge 0$. A mapping $f\colon \mathcal{R}^N \to S \subseteq \mathcal{R}^N$ satisfies AP(S,α) if*

$$\|w-f(w)\|^2 \le \|w-x\|^2 + \alpha\|f(w)-x\|$$

for every $w \in \mathcal{R}^N$ and $x \in S$.

[8] Various flavors of the AP(S,α) property have been used in previous works, such as Shah et al. (Shah and Hegde, 2018) and Raj et al. (Raj et al., 2019).

Table 2: Network Architectures for CelebA and MNIST. C-K, C-S, M-K, and M-S report CelebA Kernel Sizes, CelebA Strides, MNIST Kernel Sizes, and MNIST strides respectively.

Layer	C-K	C-S	M-K	M-S
Conv2D 1	9×9	2	5×5	2
Conv2D 2	7×7	2	5×5	2
Conv2D 3	5×5	2	3×3	2
Conv2D 4	5×5	1	3×3	1
TransConv2d 1	5×5	2	3×3	1
TransConv2d 2	5×5	2	3×3	2
TransConv2d 3	7×7	2	5×5	2
TransConv2d 4	9×9	1	5×5	2

Theorem 6. *Let $f : \mathcal{R}^N \to S \subseteq \mathcal{R}^N$ satisfy AP(S, α) and let $A \in \mathcal{R}^{m\times N}$ be a matrix with $\|A\|^2 \le M$ that satisfies RIP(S,δ). If $y = Ax$ with $x \in S$, the recovery error of Algorithm 1 is bounded as:*

$$\|x_T - x\| \le (2\gamma)^T \|x_0 - x\| + \alpha\left(\frac{1-(2\gamma)^T}{1-(2\gamma)}\right) \quad (3)$$

where $\gamma = \sqrt{\eta^2 M(1+\delta) + 2\eta(\delta-1) + 1}$.

Proof of Theorem 6. Using the notation of Algorithm 1 and the fact that f satisfies $AP(S,\alpha)$ we have

$$\|(w_t - x) - (x_{t+1} - x)\|^2 \le \|w_t - x\|^2 + \alpha\|x_{t+1} - x\|.$$

Noting $\|a-b\|^2 = \|a\|^2 + \|b\|^2 - 2\langle a,b\rangle$ and rearranging terms we get

$$\|x_{t+1} - x\|^2 \le 2\langle (w_t - x), (x_{t+1} - x)\rangle + \alpha\|x_{t+1} - x\|.$$

Now we expand the inner product using $w_t = x_t - \eta A^T(Ax_t - y)$ and $y = Ax$ to get

$$\|x_{t+1} - x\|^2 \le 2\langle (I - \eta A^T A)(x_t - x), (x_{t+1} - x)\rangle + \alpha\|x_{t+1} - x\|. \quad (4)$$

Using the Cauchy–Schwarz inequality we have

$$\begin{aligned} &|\langle (I - \eta A^T A)(x_t - x), (x_{t+1} - x)\rangle| \\ &\le \left\|(I - \eta A^T A)(x_t - x)\right\| \|(x_{t+1} - x)\| \end{aligned} \quad (5)$$

By setting $u = x_t - x$, expanding, and using the RIP(S,α) property of A, we see that

$$\begin{aligned} \left\|(I - \eta A^T A)u\right\|^2 &= \|u\|^2 - 2\eta\|Au\|^2 + \eta^2\left\|A^T(Au)\right\|^2 \\ &\le \|u\|^2 - 2\eta(1-\delta)\|u\|^2 \\ &\quad + \eta^2(1+\delta)M\|u\|^2 \\ &= \gamma^2\|u\|^2 \end{aligned} \quad (6)$$

We substitute the results of (5) and (6) into (4) and divide both sides by $\|x_{t+1} - x\|$ to get

$$\|x_{t+1} - x\| \le 2\gamma\|x_t - x\| + \alpha \quad (7)$$

Using induction on (7) gives (3). □