8. Appendix A

Lemma 8.1. *Given $S \subseteq \mathbb{R}^n$, $y \in \mathbb{R}^m$, $A \in \mathbb{R}^{m\times n}$, and $\gamma, \delta, \epsilon_1, \epsilon_2 > 0$, if matrix A satisfies the S-REC(S, γ, δ), then for any two $x_1, x_2 \in S$, such that $\|Ax_1 - y\| \leq \epsilon_1$ and $\|Ax_2 - y\| \leq \epsilon_2$, we have*

$$\|x_1 - x_2\| \leq \frac{\epsilon_1 + \epsilon_2 + \delta}{\gamma}.$$

Proof.

$$\begin{aligned}
\|x_1 - x_2\| &\leq \frac{1}{\gamma}\left(\|Ax_1 - Ax_2\| + \delta\right), \\
&= \frac{1}{\gamma}\left(\|(Ax_1 - y) - (Ax_2 - y)\| + \delta\right), \\
&\leq \frac{1}{\gamma}\left(\|(Ax_1 - y)\| + \|(Ax_2 - y)\| + \delta\right), \\
&\leq \frac{\epsilon_1 + \epsilon_2 + \delta}{\gamma}.
\end{aligned}$$

□

8.1. Proof of Lemma 4.1

Definition 2. *A random variable X is said to be* subgamma(σ, B) *if $\forall \epsilon \geq 0$, we have*

$$\mathbb{P}\left(|X - \mathbb{E}[X]| \geq \epsilon\right) \leq 2\max\left(e^{-\epsilon^2/(2\sigma^2)}, e^{-\epsilon/(2B)}\right).$$

Lemma 8.2. *Let $G : \mathbb{R}^k \to \mathbb{R}^n$ be an L-Lipschitz function. Let $B^k(r)$ be the L_2-ball in $\mathbb{R}^k$ with radius r, $S = G(B^k(r))$, and M be a δ/L-net on $B^k(r)$ such that $|M| \leq k\log\left(\frac{4Lr}{\delta}\right)$. Let A be a $\mathbb{R}^{m\times n}$ random matrix with IID Gaussian entries with zero mean and variance $1/m$. If*

$$m = \Omega\left(k\log\frac{Lr}{\delta}\right),$$

then for any $x \in S$, if $x' = \arg\min_{\widehat{x}\in G(M)} \|x - \widehat{x}\|$, we have $\|A(x - x')\| = \mathcal{O}(\delta)$ with probability $1 - e^{-\Omega(m)}$.

Note that for any given point x' in S, if we try to find its nearest neighbor of that point in an δ-net on S, then the difference between the two is at most the δ. In words, this lemma says that even if we consider measurements made on these points, *i.e.* a linear projection using a random matrix A, then as long as there are enough measurements, the difference between measurements is of the same order δ. If the point x' was in the net, then this can be easily achieved by Johnson-Lindenstrauss Lemma. But to argue that this is true for all x' in S, which can be an uncountably large set, we construct a chain of nets on S. We now present the formal proof.

Proof. Observe that for any $x \in \mathbb{R}^n$, $\frac{\|Ax\|^2}{\|x\|^2}$ is subgamma$\left(\frac{1}{\sqrt{m}}, \frac{1}{m}\right)$. Thus, for any $f > 0$,

$$\epsilon \geq 2 + \frac{4}{m}\log\frac{2}{f} \geq \max\left(\sqrt{\frac{2}{m}\log\frac{2}{f}}, \frac{2}{m}\log\frac{2}{f}\right)$$

is sufficient to ensure that

$$\mathbb{P}\left(\|Ax\| \geq (1+\epsilon)\|x\|\right) \leq \mathbb{P}\left(\|Ax\| \geq \sqrt{1+\epsilon}\|x\|\right) \leq f.$$

Now, let $M = M_0 \subseteq M_1 \subseteq M_2, \cdots \subseteq M_l$ be a chain of epsilon nets of $B^k(r)$ such that M_i is a δ_i/L-net and $\delta_i = \delta_0/2^i$, with $\delta_0 = \delta$. We know that there exist nets such that

$$\log|M_i| \leq k\log\left(\frac{4Lr}{\delta_i}\right) \leq ik + k\log\left(\frac{4Lr}{\delta_0}\right).$$

Let $N_i = G(M_i)$. Then due to Lipschitzness of G, N_i's form a chain of epsilon nets such that N_i is a δ_i-net of $S = G(B^k(r))$, with $|N_i| = |M_i|$.

For $i \in \{0, 1, 2\cdots, l-1\}$, let

$$T_i = \{x_{i+1} - x_i \mid x_{i+1} \in N_{i+1}, x_i \in N_i\}.$$

Thus,

$$\begin{aligned}
|T_i| &\leq |N_{i+1}||N_i|. \\
\implies \log|T_i| &\leq \log|N_{i+1}| + |\log|N_i|, \\
&\leq (2i+1)k + 2k\log\left(\frac{4Lr}{\delta_0}\right), \\
&\leq 3ik + 2k\log\left(\frac{4Lr}{\delta_0}\right).
\end{aligned}$$

Now assume $m = 3k\log\left(\frac{4Lr}{\delta_0}\right)$,

$$\log(f_i) = -(m + 4ik),$$

and

$$\begin{aligned}
\epsilon_i &= 2 + \frac{4}{m}\log\frac{2}{f_i}, \\
&= 2 + \frac{4}{m}\log 2 + 4 + \frac{16ik}{m}, \\
&= \mathcal{O}(1) + \frac{16ik}{m}.
\end{aligned}$$

By choice of f_i and ϵ_i, we have $\forall i \in [l-1], \forall t \in T_i$,

$$\mathbb{P}\left(\|At\| > (1+\epsilon_i)\|t\|\right) \leq f_i.$$

Thus by union bound, we have

$$\mathbb{P}\left(\|At\| \leq (1+\epsilon_i)\|t\|, \forall i, \forall t \in T_i\right) \geq 1 - \sum_{i=0}^{l-1} |T_i| f_i.$$

Now,

$$\begin{aligned}
\log(|T_i|f_i) &= \log(|T_i|) + \log(f_i), \\
&\leq -k\log\left(\frac{4Lr}{\delta_0}\right) - ik, \\
&= -m/3 - ik. \\
\implies \sum_{i=0}^{l-1} |T_i| f_i &\leq e^{-m/3} \sum_{i=0}^{l-1} e^{-ik}, \\
&\leq e^{-m/3}\left(\frac{1}{1-e^{-1}}\right), \\
&\leq 2e^{-m/3}.
\end{aligned}$$

Observe that for any $x \in S$, we can write

$$\begin{aligned}
x &= x_0 + (x_1 - x_0) + (x_2 - x_1) \ldots (x_l - x_{l-1}) + x^f. \\
x - x_0 &= \sum_{i=0}^{l-1} (x_{i+1} - x_i) + x^f.
\end{aligned}$$

where $x_i \in N_i$ and $x_f = x - x_l$. We also get $\|x_{i+1} - x_i\| \leq \delta_i$, and $\|x^f\| \leq \delta_l$ due to properties of epsilon-nets.

Since each $x_{i+1} - x_i \in T_i$, with probability at least $1 - 2e^{-m/3}$, we have

$$\begin{aligned}
\sum_{i=0}^{l-1} \|A(x_{i+1} - x_i)\| &= \sum_{i=0}^{l-1} (1+\epsilon_i)\|(x_{i+1} - x_i)\|, \\
&\leq \sum_{i=0}^{l-1} (1+\epsilon_i)\delta_i, \\
&= \delta_0 \sum_{i=0}^{l-1} \frac{1}{2^i}\left(\mathcal{O}(1) + \frac{16ik}{m}\right), \\
&= \mathcal{O}(\delta_0) + \delta_0 \frac{16k}{m} \sum_{i=0}^{l-1} \left(\frac{i}{2^i}\right), \\
&= \mathcal{O}(\delta_0).
\end{aligned}$$

We know that $\|A\| \leq 2 + \sqrt{n/m}$ with probability at least $1 - 2e^{-m/2}$ (Corollary 5.35 (Vershynin, 2010)). By setting $l = \log(n)$, we get that, $\|A\|\|x^f\| \leq \left(2 + \sqrt{\frac{n}{m}}\right)\frac{\delta_0}{2^l} = \mathcal{O}(\delta_0)$ with probability $\geq 1 - 2e^{-m/2}$.

Combining these two results, and noting that it is possible to choose $x' = x_0$, we get that with probability $1 - e^{-\Omega(m)}$,

$$\begin{aligned}
\|A(x - x')\| &= \|A(x - x_0)\|, \\
&\leq \sum_{i=0}^{l-1} \|A(x_{i+1} - x_i)\| + \|Ax^f\|, \\
&= \mathcal{O}(\delta_0) + \|A\|\|x^f\|, \\
&= \mathcal{O}(\delta).
\end{aligned}$$

□

Lemma. *Let $G : \mathbb{R}^k \to \mathbb{R}^n$ be L-Lipschitz. Let*

$$B^k(r) = \{z \mid z \in \mathbb{R}^k, \|z\| \leq r\}$$

be an L_2-norm ball in $\mathbb{R}^k$. For $\alpha < 1$, if

$$m = \Omega\left(\frac{k}{\alpha^2}\log\frac{Lr}{\delta}\right),$$

then a random matrix $A \in \mathbb{R}^{m \times n}$ with IID entries such that $A_{ij} \sim \mathcal{N}\left(0, \frac{1}{m}\right)$ satisfies the S-$REC(G(B^k(r)), 1-\alpha, \delta)$ with $1 - e^{-\Omega(\alpha^2 m)}$ probability.

Proof. We construct a $\frac{\delta}{L}$-net, N, on $B^k(r)$. There exists a net such that

$$\log|N| \leq k\log\left(\frac{4Lr}{\delta}\right).$$

Since N is a $\frac{\delta}{L}$-cover of $B^k(r)$, due to the L-Lipschitz property of $G(\cdot)$, we get that $G(N)$ is a δ-cover of $G(B^k(r))$.

Let T denote the pairwise differences between the elements in $G(N)$, *i.e.*,

$$T = \{G(z_1) - G(z_2) \mid z_1, z_2 \in N\}.$$

Then,

$$\begin{aligned}
|T| &\leq |N|^2, \\
\implies \log|T| &\leq 2\log|N|, \\
&\leq 2k\log\left(\frac{4Lr}{\delta}\right).
\end{aligned}$$

For any $z, z' \in B^k$, $\exists\ z_1, z_2 \in N$, such that $G(z_1), G(z_2)$ are δ-close to $G(z)$ and $G(z')$ respectively. Thus, by triangle inequality,

$$\begin{aligned}
\|G(z) - G(z')\| &\leq \|G(z) - G(z_1)\| + \\
&\quad \|G(z_1) - G(z_2)\| + \\
&\quad \|G(z_2) - G(z')\|, \\
&\leq \|G(z_1) - G(z_2)\| + 2\delta.
\end{aligned}$$

Again by triangle inequality,

$$\begin{aligned}\|AG(z_1) - AG(z_2)\| \leq \|AG(z_1) - AG(z)\| + \\ \|AG(z) - AG(z')\| + \\ \|AG(z') - AG(z_2)\|.\end{aligned}$$

Now, by Lemma 8.2, with probability $1 - e^{-\Omega(m)}$, $\|AG(z_1) - AG(z)\| = \mathcal{O}(\delta)$, and $\|AG(z') - AG(z_2)\| = \mathcal{O}(\delta)$. Thus,

$$\|AG(z_1) - AG(z_2)\| \leq \|AG(z) - AG(z')\| + \mathcal{O}(\delta).$$

By the Johnson-Lindenstrauss Lemma, for a fixed $x \in \mathbb{R}^n$, $\mathbb{P}\left[\|Ax\|^2 < (1-\alpha)\|x\|^2\right] < \exp(-\alpha^2 m)$. Therefore, we can union bound over all vectors in T to get

$$\mathbb{P}(\|Ax\|^2 \geq (1-\alpha)\|x\|^2,\ \forall x \in T) \geq 1 - e^{-\Omega(\alpha^2 m)}.$$

Since $\alpha < 1$, and $z_1, z_2 \in N$, $G(z_1) - G(z_2) \in T$, we have

$$\begin{aligned}(1-\alpha)\|G(z_1) - G(z_2)\| &\leq \sqrt{1-\alpha}\|G(z_1) - G(z_2)\|, \\ &\leq \|AG(z_1) - AG(z_2)\|.\end{aligned}$$

Combining the three results above we get that with probability $1 - e^{-\Omega(\alpha^2 m)}$,

$$\begin{aligned}(1-\alpha)\|G(z) - G(z')\| &\leq (1-\alpha)\|G(z_1) - G(z_2)\| + \mathcal{O}(\delta), \\ &\leq \|AG(z_1) - AG(z_2)\| + \mathcal{O}(\delta), \\ &\leq \|AG(z) - AG(z')\| + \mathcal{O}(\delta).\end{aligned}$$

Thus, A satisfies $S\text{-}REC(S, 1-\alpha, \delta)$ with probability $1 - e^{-\Omega(\alpha^2 m)}$.

□

8.2. Proof of Lemma 4.2

Lemma 8.3. *Consider c different $k-1$ dimensional hyperplanes in $\mathbb{R}^k$. Consider the k-dimensional faces (hereafter called k-faces) generated by the hyperplanes, i.e. the elements in the partition of $\mathbb{R}^k$ such that relative to each hyperplane, all points inside a partition are on the same side. Then, the number of k-faces is $\mathcal{O}(c^k)$.*

Proof. Proof is by induction, and follows (Matoušek, 2002).

Let $f(c,k)$ denote the number of $k-$faces generated in $\mathbb{R}^k$ by c different $(k-1)$-dimensional hyperplanes. As a base case, let $k = 1$. Then $(k-1)$-dimensional hyperplanes are just points on a line. c points partition $\mathbb{R}$ into $c+1$ pieces. This gives $f(c,1) = \mathcal{O}(c)$.

Now, assuming that $f(c, k-1) = \mathcal{O}(c^{k-1})$ is true, we need to show $f(c,k) = \mathcal{O}(c^k)$. Assume we have $(c-1)$ different hyperplanes $H = \{h_1, h_2, \ldots, h_{c-1}\} \subset \mathbb{R}^k$, and a new hyperplane h_c is added. h_c intersects H at $(c-1)$ different $(k-2)$-faces given by $F = \{f_j \mid f_j = h_j \cap h_c, 1 \leq j \leq (c-1)\}$. The $(k-2)$-faces in F partition h_c into $f(c-1, k-1)$ different $(k-1)$-faces. Additionally, each $(k-1)$-face in h_c divides an existing k-face into two. Hence the number of new k-faces introduced by the addition of h_c is $f(c-1, k-1)$. This gives the recursion

$$\begin{aligned}f(c,k) &= f(c-1,k) + f(c-1,k-1), \\ &= f(c-1,k) + \mathcal{O}(c^{k-1}), \\ &= \mathcal{O}(c^k).\end{aligned}$$

□

Lemma. *Let $G : \mathbb{R}^k \to \mathbb{R}^n$ be a d-layer neural network, where each layer is a linear transformation followed by a pointwise non-linearity. Suppose there are at most c nodes per layer, and the non-linearities are piecewise linear with at most two pieces, and let*

$$m = \Omega\left(\frac{1}{\alpha^2} kd \log c\right)$$

for some $\alpha < 1$. Then a random matrix $A \in \mathbb{R}^{m \times n}$ with IID entries $A_{ij} \sim \mathcal{N}(0, \frac{1}{m})$ satisfies the $S\text{-}REC(G(\mathbb{R}^k), 1-\alpha, 0)$ with $1 - e^{-\Omega(\alpha^2 m)}$ probability.

Proof. Consider the first layer of G. Each node in this layer can be represented as a hyperplane in $\mathbb{R}^k$, where the points on the hyperplane are those where the input to the node switches from one linear piece to the other. Since there are at most c nodes in this layer, by Lemma 8.3, the input space is partitioned by at most c different hyperplanes, into $\mathcal{O}(c^k)$ k-faces. Applying this over the d layers of G, we get that the input space $\mathbb{R}^k$ is partitioned into at most c^{kd} sets.

Recall that the non-linearities are piecewise linear, and the partition boundaries were made precisely at those points where the non-linearities change from one piece to another. This means that within each set of the input partition, the output is a linear function of the inputs. Thus $G(\mathbb{R}^k)$ is a union of c^{kd} different k-faces in $\mathbb{R}^n$.

We now use an oblivious subspace embedding to bound the number of measurements required to embed the range of $G(\cdot)$. For a single k-face $S \subseteq \mathbb{R}^n$, a random matrix $A \in \mathbb{R}^{m \times n}$ with IID entries such that $A_{ij} \sim \mathcal{N}\left(0, \frac{1}{m}\right)$ satisfies $S\text{-}REC(S, 1-\alpha, 0)$ with probability $1 - e^{-\Omega(\alpha^2 m)}$ if $m = \Omega(k/\alpha^2)$.

Since the range of $G(\cdot)$ is a union of c^{kd} different k-faces, we can union bound over all of them, such that A satisfies the $S\text{-}REC(G(\mathbb{R}^k), 1-\alpha, 0)$ with probability $1 - c^{kd} e^{-\Omega(\alpha^2 m)}$. Thus, we get that A satisfies the

$S\text{-}REC(G(\mathbb{R}^k), 1-\alpha, 0)$ with probability $1-e^{-\Omega(\alpha^2 m)}$ if

$$m = \Omega\left(\frac{kd\log c}{\alpha^2}\right).$$

□

8.3. Proof of Lemma 4.3

Lemma. *Let $A \in \mathbb{R}^{m\times n}$ by drawn from a distribution that (1) satisfies the $S\text{-}REC(S,\gamma,\delta)$ with probability $1-p$ and (2) has for every fixed $x \in \mathbb{R}^n$, $\|Ax\| \le 2\|x\|$ with probability $1-p$. For any $x^* \in \mathbb{R}^n$ and noise η, let $y = Ax^* + \eta$. Let $\widehat{x}$ approximately minimize $\|y - Ax\|$ over $x \in S$, i.e.,*

$$\|y - A\widehat{x}\| \le \min_{x\in S}\|y - Ax\| + \epsilon.$$

Then

$$\|\widehat{x} - x^*\| \le \left(\frac{4}{\gamma}+1\right)\min_{x\in S}\|x^* - x\| + \frac{1}{\gamma}\left(2\|\eta\| + \epsilon + \delta\right)$$

with probability $1-2p$.

Proof. Let $\overline{x} = \arg\min_{x\in S}\|x^* - x\|$. Then we have by Lemma 8.1 and the hypothesis on $\widehat{x}$ that

$$\begin{aligned}\|\overline{x} - \widehat{x}\| &\le \frac{\|A\overline{x} - y\| + \|A\widehat{x} - y\| + \delta}{\gamma},\\ &\le \frac{2\|A\overline{x} - y\| + \epsilon + \delta}{\gamma},\\ &\le \frac{2\|A(\overline{x} - x^*)\| + 2\|\eta\| + \epsilon + \delta}{\gamma},\end{aligned}$$

as long as A satisfies the S-REC, as happens with probability $1-p$. Now, since $\overline{x}$ and x^* are independent of A, by assumption we also have $\|A(\overline{x} - x^*)\| \le 2\|\overline{x} - x^*\|$ with probability $1-p$. Therefore

$$\|x^* - \widehat{x}\| \le \|\overline{x} - x^*\| + \frac{4\|\overline{x} - x^*\| + 2\|\eta\| + \epsilon + \delta}{\gamma}$$

as desired. □

8.4. Lipschitzness of Neural Networks

Lemma 8.4. *Consider any two functions f and g. If f is L_f-Lipschitz and g is L_g-Lipschitz, then their composition $f \circ g$ is $L_f L_g$-Lipschitz.*

Proof. For any two x_1, x_2,

$$\begin{aligned}\|f(g(x_1)) - f(g(x_2))\| &\le L_f\|g(x_1) - g(x_2)\|,\\ &\le L_f L_g\|x_1 - x_2\|.\end{aligned}$$

□

Lemma 8.5. *If G is a d-layer neural network with at most c nodes per layer, all weights $\le w_{\max}$ in absolute value, and M-Lipschitz non-linearity after each layer, then $G(\cdot)$ is L-Lipschitz with $L = (Mcw_{\max})^d$.*

Proof. Consider any linear layer with input x, weight matrix W and bias vector b. Thus, $f(x) = Wx + b$. Now for any two x_1, x_2,

$$\begin{aligned}\|f(x_1) - f(x_2)\| &= \|Wx_1 + b - Wx_2 + b\|,\\ &= \|W(x_1 - x_2)\|,\\ &\le \|W\|\|(x_1 - x_2)\|,\\ &\le cw_{\max}\|(x_1 - x_2)\|.\end{aligned}$$

Let $f_i(\cdot), i \in [d]$ denote the function for the i-th layer in G. Since each layer is a composition of a linear function and a non-linearity, by Lemma 8.4, have that f_i is $Mcw_{\max}$-Lipschitz.

Since $G = f_1 \circ f_2 \circ \dots f_d$, by repeated application of Lemma 8.4, we get that G is L-Lipschitz with $L = (Mcw_{\max})^d$. □

9. Appendix B

9.1. Noise tolerance

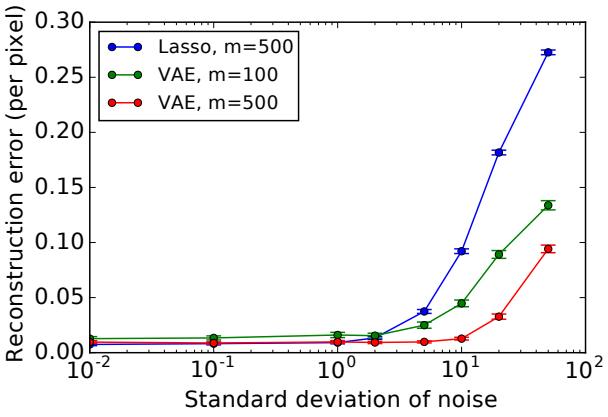
To understand the noise tolerance of our algorithm, we do the following experiment: First we fix the number of measurements so that Lasso does as well as our algorithm. From Fig. 1a, and Fig. 1b we see that this point is at $m = 500$ for MNIST and $m = 2500$ for celebA. Now, we look at the performance as the noise level increases. Hyperparameters are kept fixed as we change the noise level for both Lasso and for our algorithm.

In Fig. 8a, we show the results on the MNIST dataset. In Fig. 8a, we show the results on celebA dataset. We observe that our algorithm has more noise tolerance than Lasso.

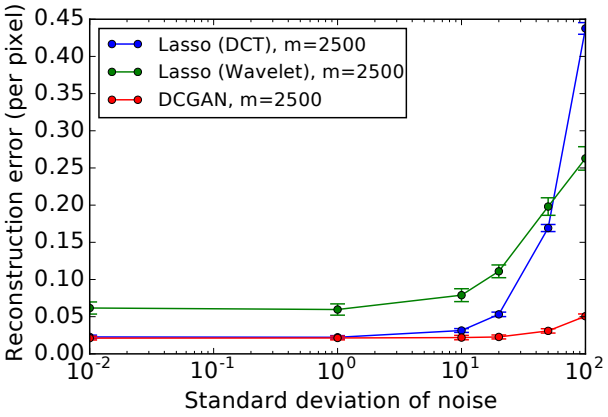
9.2. Scaling with latent dimension

In the experiments in Sec. 6.3, we saw that the representation error was a major component of the total error, and thus a better generative model might be helpful. Recall that a generative model is a function $G : \mathbb{R} \to \mathbb{R}^n$. Thus, one way to make the generative model more powerful is to increase the size of the latent space k.

In this section we present some experiments that investigate how the representation error scales as we use different values of k. We keep the rest of the architecture and hyperparameters fixed as we change k. For comparison, we also plot the representation error of a k-sparse wavelet as we change k. Figure 9 shows the plots for the celebA dataset. We observe that for small values of k, our method is far



(a) Results on MNIST.



(b) Results on celebA.

Figure 8. Noise tolerance. We show a plot of per pixel reconstruction error as we vary the noise level ($\sqrt{\mathbb{E}[\|\eta\|^2]}$). The vertical bars indicate 95% confidence intervals.

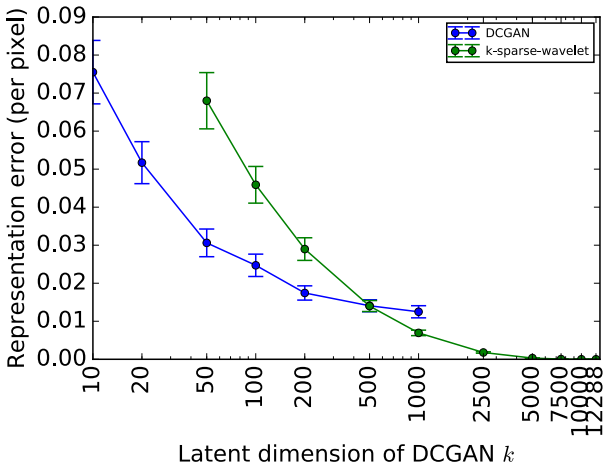


Figure 9. Results on celebA. We show per pixel representation error vs the latent dimension of the generative model. The vertical bars indicate 95% confidence intervals.

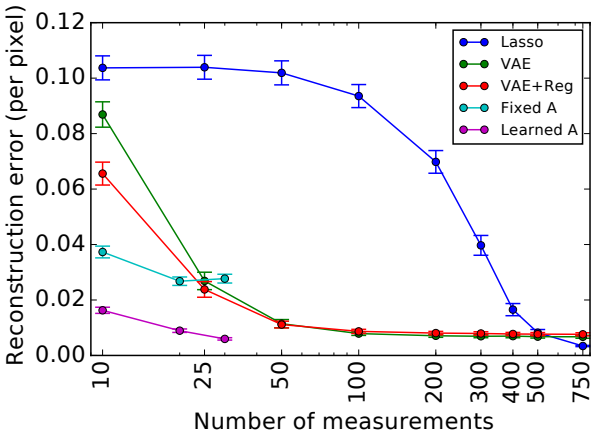


Figure 10. Results for end to end model on MNIST. We show per pixel reconstruction error vs number of measurements. 'Fixed A' and 'Learned A' are two end to end models. The end to end models get noiseless measurements, while the other models get noisy ones. The vertical bars indicate 95% confidence intervals.

superior to k-sparse wavelet. This suggests that neural network based generative models make effective use of the latent space by constructing excellent representations. We see that as we increase k, the error starts to plateau for our method while it goes to zero for k-sparse wavelet model. This suggests that beyond a point, some other factor in our model, such as the architecture of the DCGAN, starts to become the bottleneck. It is possible that the results for our method can be improved by more careful hyperparameter tuning for each k.

9.3. Other models

9.3.1. End to end training on MNIST

Instead of using a generative model to reconstruct the image, another approach is to learn from scratch a mapping that takes the measurements and outputs the original image. A major drawback of this approach is that it necessitates learning a new network if get a different set of measurements.

If we use a random matrix for every new image, the input to the network is essentially noise, and the network does not learn well. Instead we use a fixed measurement matrix. We explore two approaches. First is to randomly sample

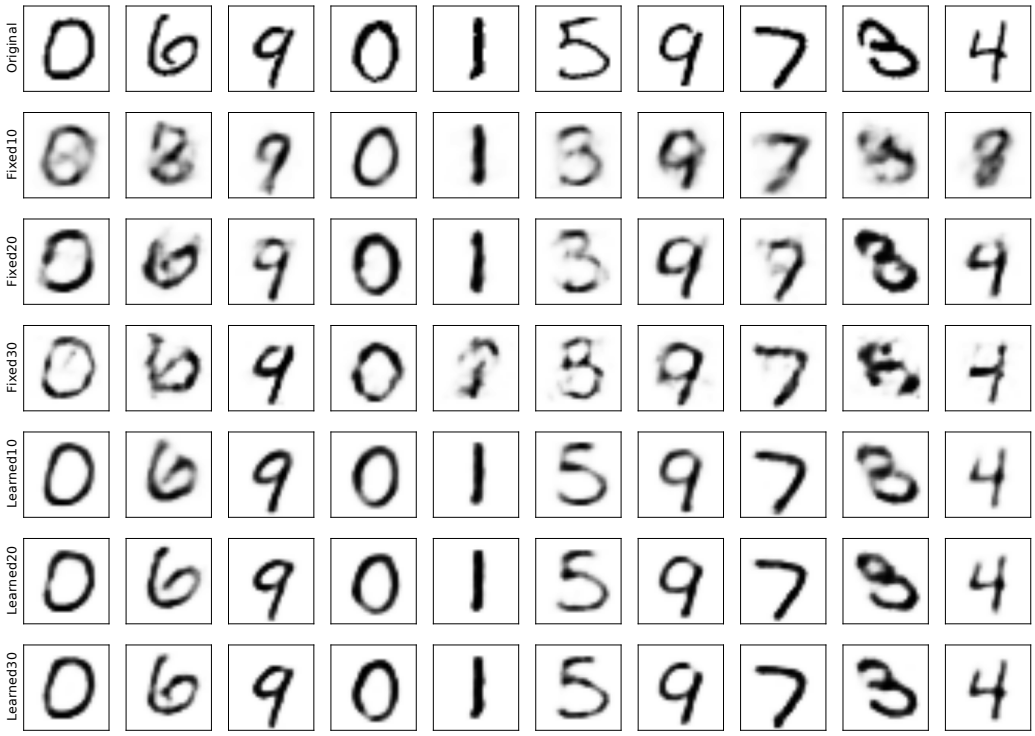


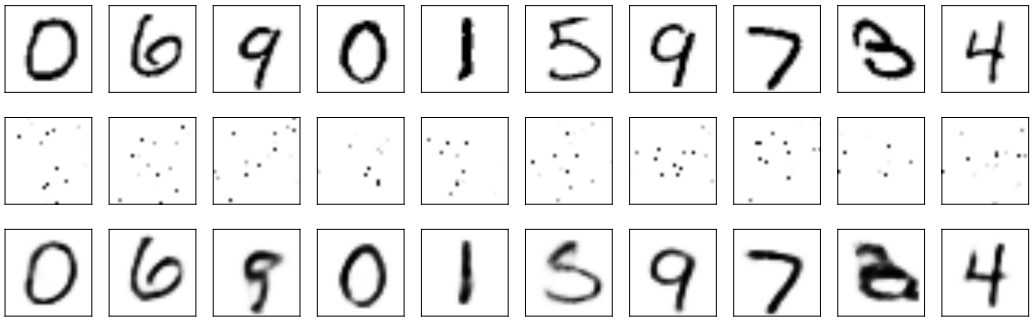
Figure 11. MNIST End to end learned model. Top row are original images. The next three are recovered by model with fixed random A, with 10, 20 and 30 measurements. Bottom three rows are with learned A and 10, 20 and 30 measurements.

and fix the measurement matrix and learn the rest of the mapping. In the second approach, we jointly optimize the measurement matrix as well.

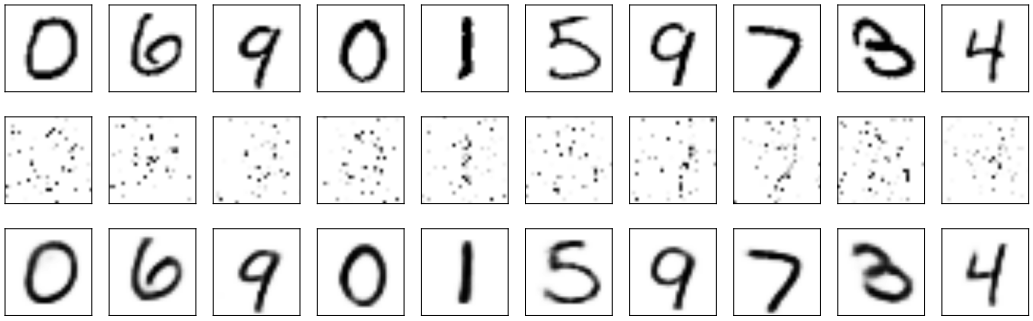
We do this for 10, 20 and 30 measurements for the MNIST dataset. We did not use additive noise. The reconstruction errors are shown in Fig. 10. The reconstructed images can be seen in Fig. 11.

9.4. More results

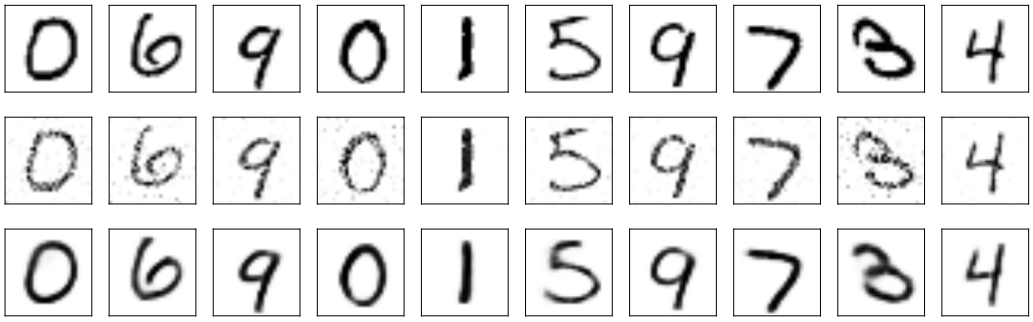
Here, we show more results on the reconstruction task, with varying number of measurements on both MNIST and celebA. Fig. 12 shows reconstructions on MNIST with 25, 100 and 400 measurements. Fig. 13, Fig. 14 and Fig. 15 show results on celebA dataset.



(a) 25 measurements

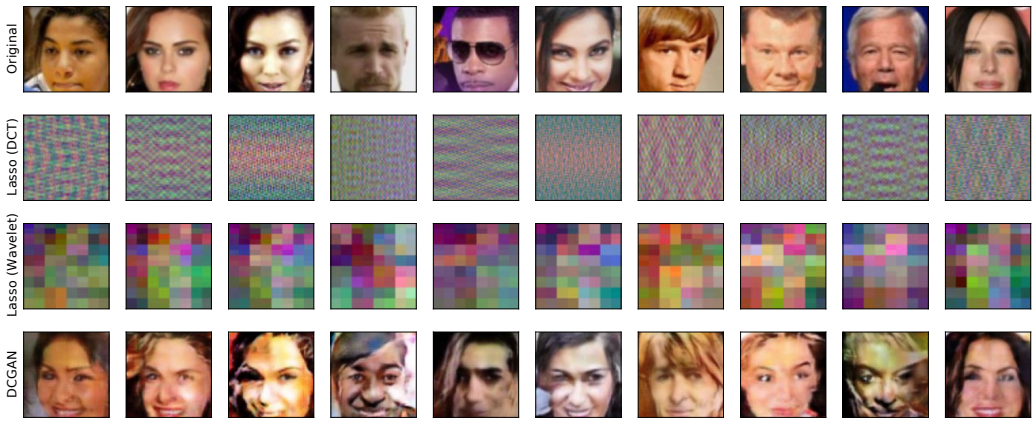


(b) 100 measurements

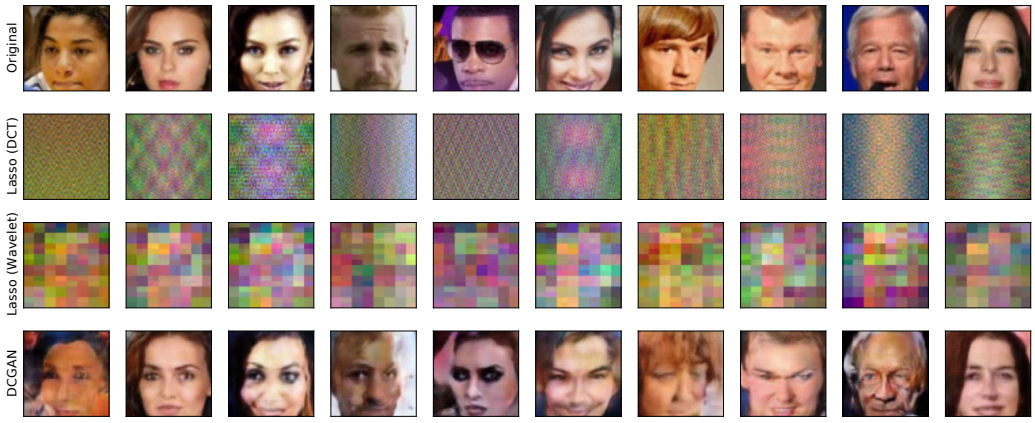


(c) 400 measurements

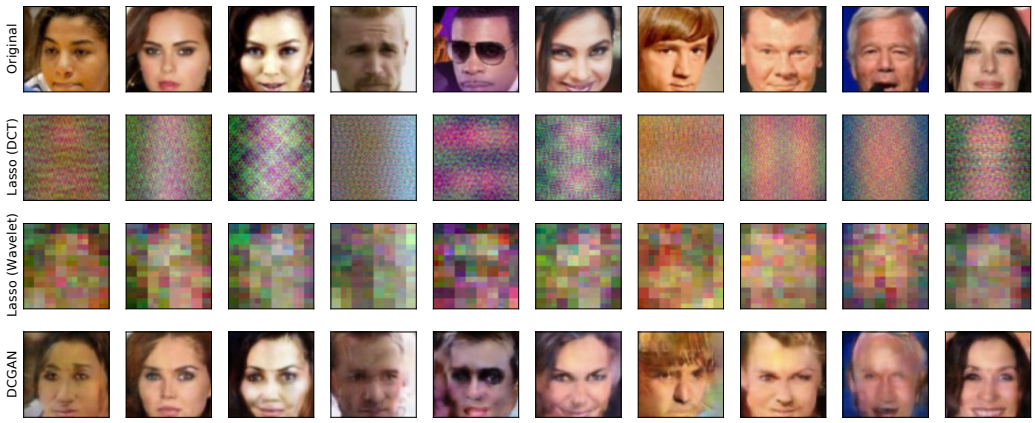
Figure 12. Reconstruction on MNIST. In each image, top row is ground truth, middle row is Lasso, bottom row is our algorithm.



(a) 50 measurements

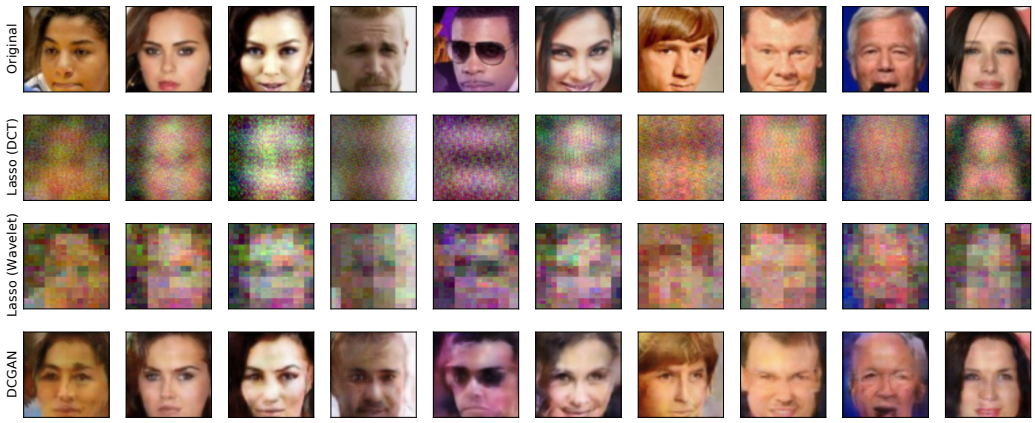


(b) 100 measurements

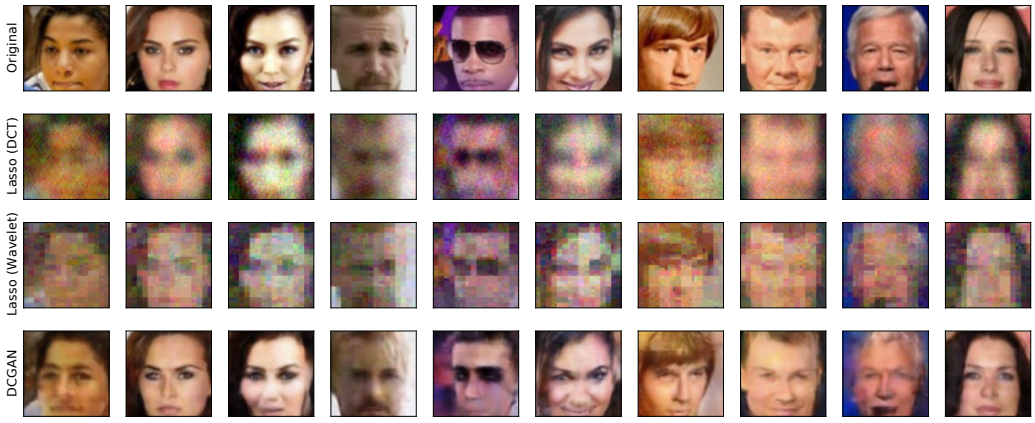


(c) 200 measurements

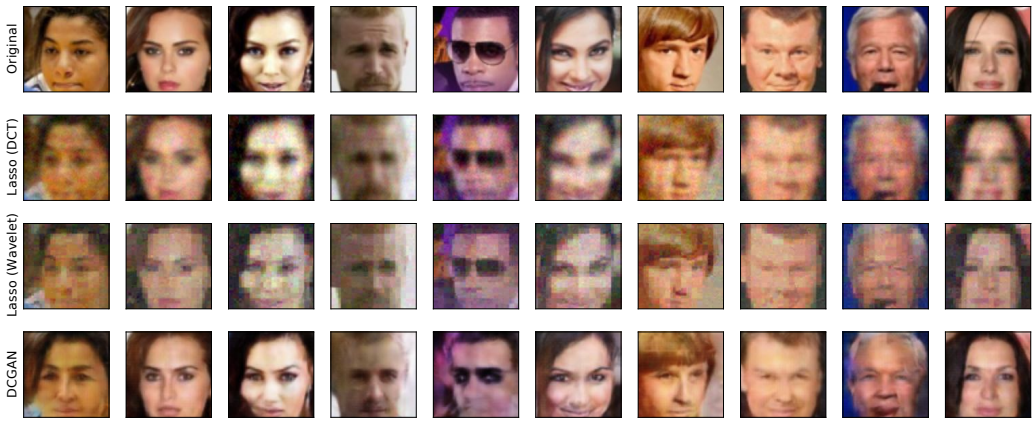
Figure 13. Reconstruction on celebA. In each image, top row is ground truth, subsequent two rows show reconstructions by Lasso (DCT) and Lasso (Wavelet) respectively. The bottom row is the reconstruction by our algorithm.



(a) 500 measurements

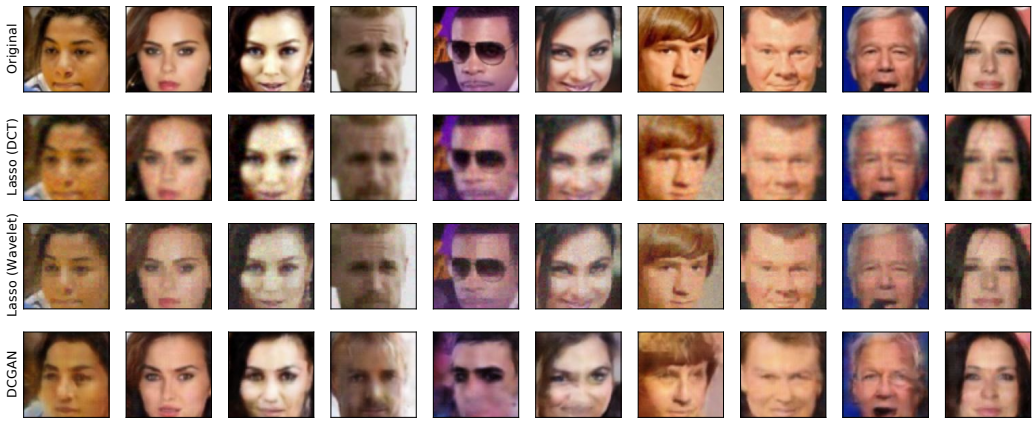


(b) 1000 measurements

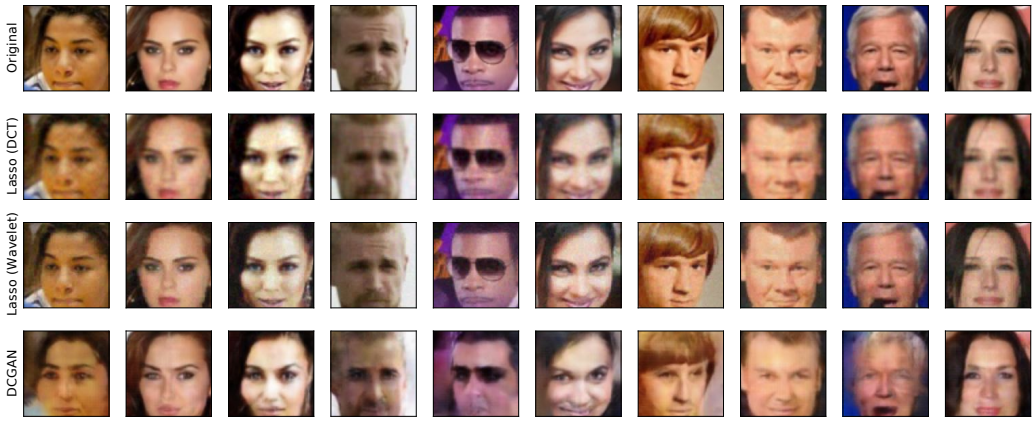


(c) 2500 measurements

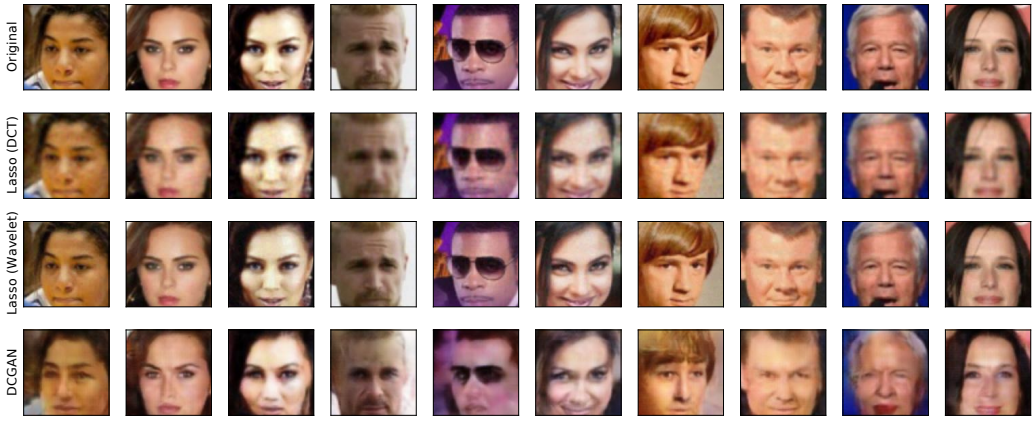
Figure 14. Reconstruction on celebA. In each image, top row is ground truth, subsequent two rows show reconstructions by Lasso (DCT) and Lasso (Wavelet) respectively. The bottom row is the reconstruction by our algorithm.



(a) 5000 measurements



(b) 7500 measurements



(c) 10000 measurements

Figure 15. Reconstruction on celebA. In each image, top row is ground truth, subsequent two rows show reconstructions by Lasso (DCT) and Lasso (Wavelet) respectively. The bottom row is the reconstruction by our algorithm.