On Instabilities of Deep Learning in Image Reconstruction

Part II

Matthew Colbrook
DAMTP, University of Cambridge

May 23, 2019

Recall the set-up

Image $x \in \mathbb{C}^N$, we are given access to measurements of the form

$$y = Ax + e,$$

where $A \in \mathbb{C}^{m \times N}$ represents sampling modality, $m \ll N$.

Task: reconstruct x from the noisy measurements y.

Without additional assumptions, such as sparsity, problem is highly ill-posed.

Might try to solve via

$$\min_{z \in \mathbb{C}^N} \|z\|_1 \quad \text{s.t.} \quad \|Az - y\|_2 \leq \nu,$$

or

$$\min_{z \in \mathbb{C}^N} \lambda \|z\|_1 + \|Az - y\|_2^2,$$

or etc.

Neural networks are FANTASTIC approximators!

Consider the following mapping $\varphi_{A,\nu} : \mathcal{M} \to \mathbb{R}^N$ where

$$\mathcal{M} = \{y_j\}_{j=1}^r \subset \mathbb{R}^m, \quad r < \infty,\, m < N$$

given by

$$\varphi_{A,\nu}(y) = w, \quad w \in \underset{z}{\operatorname{argmin}} \|z\|_1 \text{ subject to } \|Az - y\|_2 \leq \nu.$$

Neural networks are FANTASTIC approximators!

Consider the following mapping $\varphi_{A,\nu} : \mathcal{M} \to \mathbb{R}^N$ where

$$\mathcal{M} = \{y_j\}_{j=1}^r \subset \mathbb{R}^m, \quad r < \infty,\, m < N$$

given by

$$\varphi_{A,\nu}(y) = w, \quad w \in \operatorname*{argmin}_z \|z\|_1 \text{ subject to } \|Az - y\|_2 \leq \nu.$$

Theorem ([Pinkus, 1999])

Let $\nu, \delta \geq 0$. If the non-linear function ρ in each layer is not a polynomial, there exists a neural network Φ, depending on A and $\mathcal{M}$, such that

$$\|\Phi(y) - \varphi_{A,\nu}(y)\|_2 \leq \delta, \quad \forall y \in \mathcal{M}.$$

Neural networks are FANTASTIC approximators!

Consider the following mapping $\varphi_{A,\nu} : \mathcal{M} \to \mathbb{R}^N$ where

$$\mathcal{M} = \{y_j\}_{j=1}^r \subset \mathbb{R}^m, \quad r < \infty,\, m < N$$

given by

$$\varphi_{A,\nu}(y) = w, \quad w \in \operatorname*{argmin}_z \|z\|_1 \text{ subject to } \|Az - y\|_2 \leq \nu.$$

Theorem ([Pinkus, 1999])

Let $\nu, \delta \geq 0$. If the non-linear function ρ in each layer is not a polynomial, there exists a neural network Φ, depending on A and $\mathcal{M}$, such that

$$\|\Phi(y) - \varphi_{A,\nu}(y)\|_2 \leq \delta, \quad \forall y \in \mathcal{M}.$$

But: need a constructive training model.

Constructive?

In reality, given approximations $\{y_{j,n}\}_{j=1}^{r}$, $\{\phi_{j,n}\}_{j=1}^{r}$ and A_n such that:

$$\|y_{j,n} - y_j\|,\ \|\phi_{j,n} - \varphi_{A_n,\nu}(y_{j,n})\|,\ \|A_n - A\| \leq 2^{-n}.$$

This is what we can store on a computer in real life, models irrational A etc. Also models a type of numerical stability.

Constructive?

In reality, given approximations $\{y_{j,n}\}_{j=1}^{r}$, $\{\phi_{j,n}\}_{j=1}^{r}$ and A_n such that:

$$\|y_{j,n} - y_j\|,\ \|\phi_{j,n} - \varphi_{A_n,\nu}(y_{j,n})\|,\ \|A_n - A\| \leq 2^{-n}.$$

This is what we can store on a computer in real life, models irrational A etc. Also models a type of numerical stability.

Training set must be

$$\mathcal{T} := \{(y_{j,n}, \phi_{j,n}, A_n) \,|\, j = 1, \ldots, r, n \in \mathbb{N}\}.$$

Constructive?

In reality, given approximations $\{y_{j,n}\}_{j=1}^{r}$, $\{\phi_{j,n}\}_{j=1}^{r}$ and A_n such that:

$$\|y_{j,n} - y_j\|, \ \|\phi_{j,n} - \varphi_{A_n,\nu}(y_{j,n})\|, \ \|A_n - A\| \leq 2^{-n}.$$

This is what we can store on a computer in real life, models irrational A etc. Also models a type of numerical stability.

Training set must be

$$\mathcal{T} := \{(y_{j,n}, \phi_{j,n}, A_n) \,|\, j = 1, \ldots, r, n \in \mathbb{N}\}.$$

Can we train a neural network that can approximate Φ based on the training set $\mathcal{T}$?

Constructive?

In reality, given approximations $\{y_{j,n}\}_{j=1}^r$, $\{\phi_{j,n}\}_{j=1}^r$ and A_n such that:

$$\|y_{j,n} - y_j\|, \ \|\phi_{j,n} - \varphi_{A_n,\nu}(y_{j,n})\|, \ \|A_n - A\| \leq 2^{-n}.$$

This is what we can store on a computer in real life, models irrational A etc. Also models a type of numerical stability.

Training set must be

$$\mathcal{T} := \{(y_{j,n}, \phi_{j,n}, A_n) \,|\, j = 1, \ldots, r, n \in \mathbb{N}\}.$$

Can we train a neural network that can approximate Φ based on the training set $\mathcal{T}$?

Maybe we expect to be able to do this by unravelling standard (iterative) optimisation algorithms? Like ISTA, FISTA, NESTA,...

Theorem (Impossible in general)

Let $K > 2, L \in \mathbb{N}$ and d be any norm on $\mathbb{C}^N$ where $N \geq 6$. Then there exists a well conditioned class Ω of elements $(A, \mathcal{M})$, such that we have the following three conditions. Consider the neural network Φ from Theorem 1.

(i) *There does not exist any algorithm with $\mathcal{T}$ as input that produces a neural network Ψ that approximates Φ on $(A, \mathcal{M}) \in \Omega$ to K correct digits in the norm d.*

(ii) *There exists an algorithm with $\mathcal{T}$ as input that produces a neural network Ψ that approximates Φ on $(A, \mathcal{M}) \in \Omega$ to $K - 1$ correct digits in the norm d. However, any algorithm producing such a network will need arbitrary many samples of elements from $\mathcal{T}$.*

(iii) *There exists an algorithm using L samples from $\mathcal{T}$ as input that produces a neural network Ψ that approximates Φ on $(A, \mathcal{M}) \in \Omega$ to $K - 2$ correct digits in the norm d.*

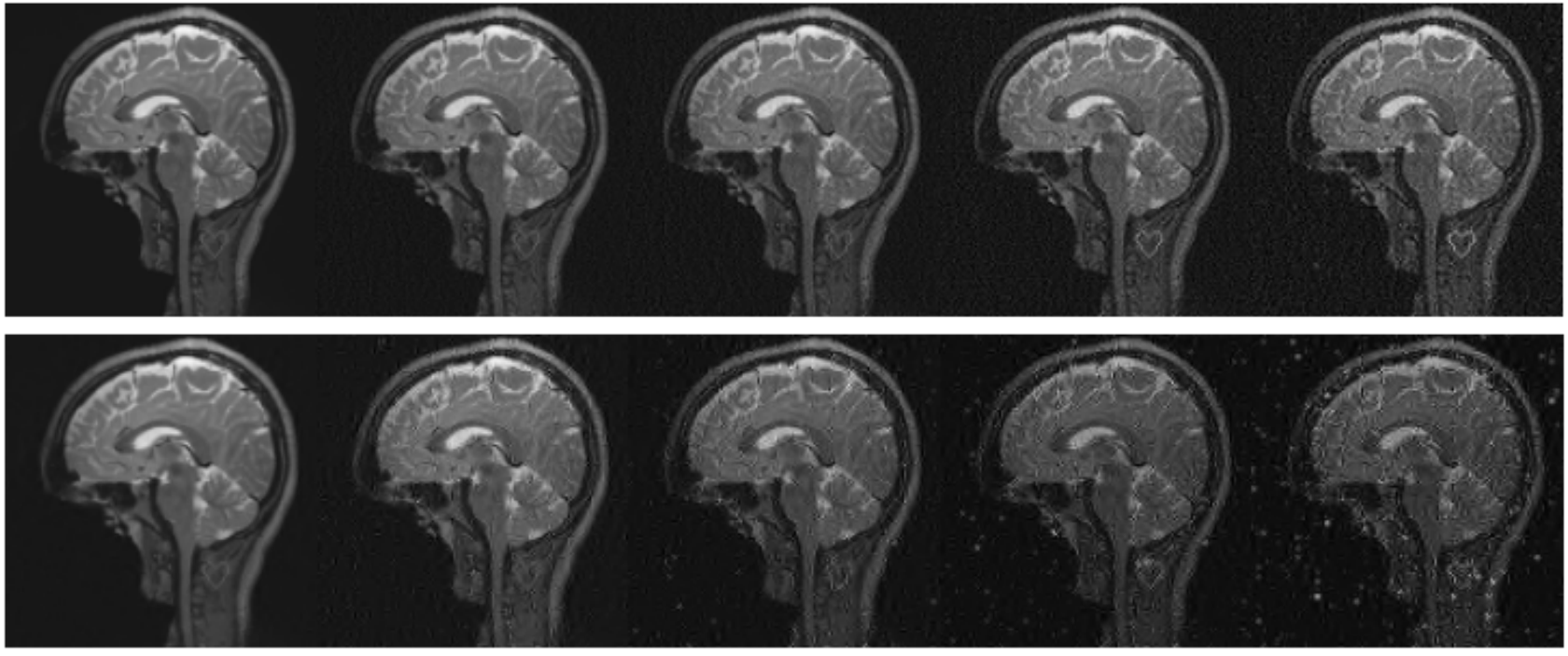
It is NOT enough to just "unravel" your favourite algorithm.

Theorem also holds for other popular optimisation problems such as LASSO.

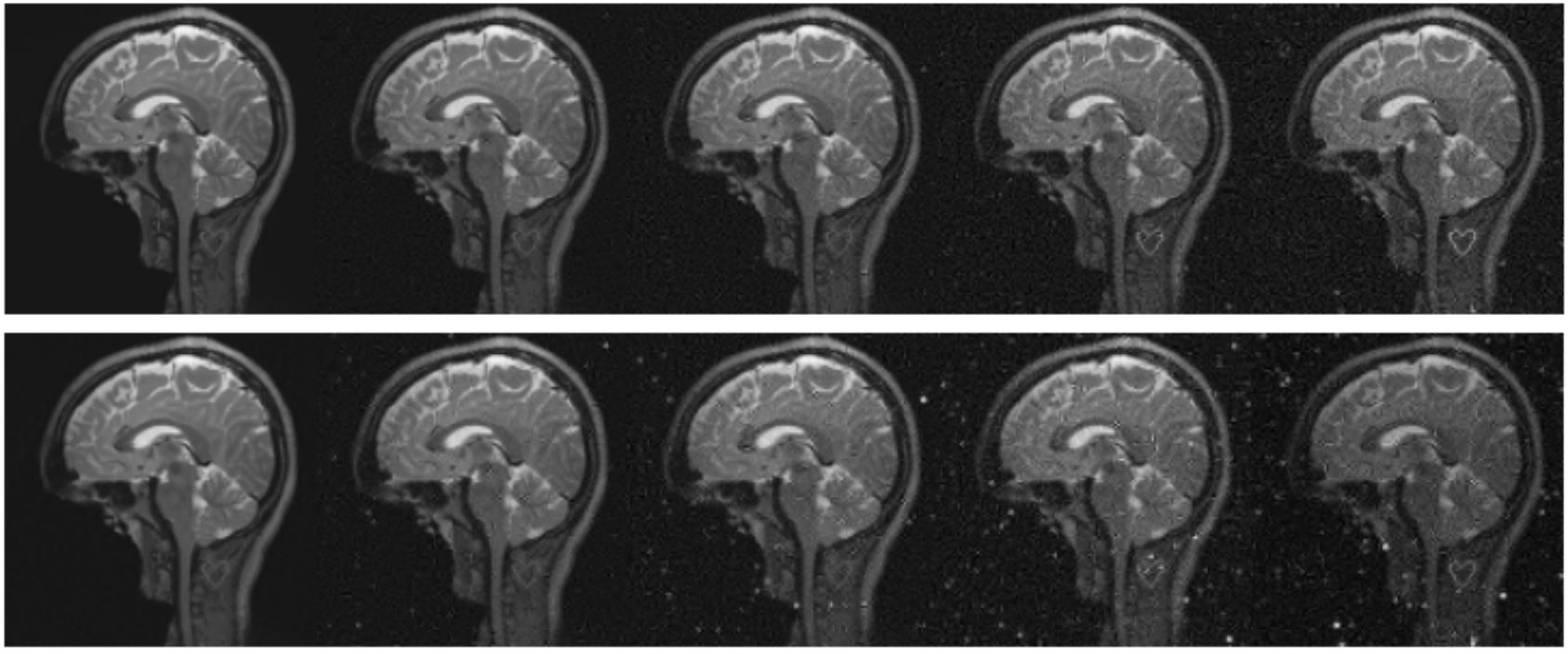
Question: Which functions can be approximated by a neural network that can be computed by an algorithm?

This is only half the story. For numerical purposes and robustness to attack we must have stability!

Solving LASSO with FISTA



Solving basis pursuit with Chambolle-Pock



$P_\Omega : \mathbb{C}^N \to \mathbb{C}^m$ projection onto canonical basis e_j indexed by Ω.

$$A = P_\Omega U$$

where U measurement matrix (d-dimensional discrete FT).

Some ideas from compressed sensing

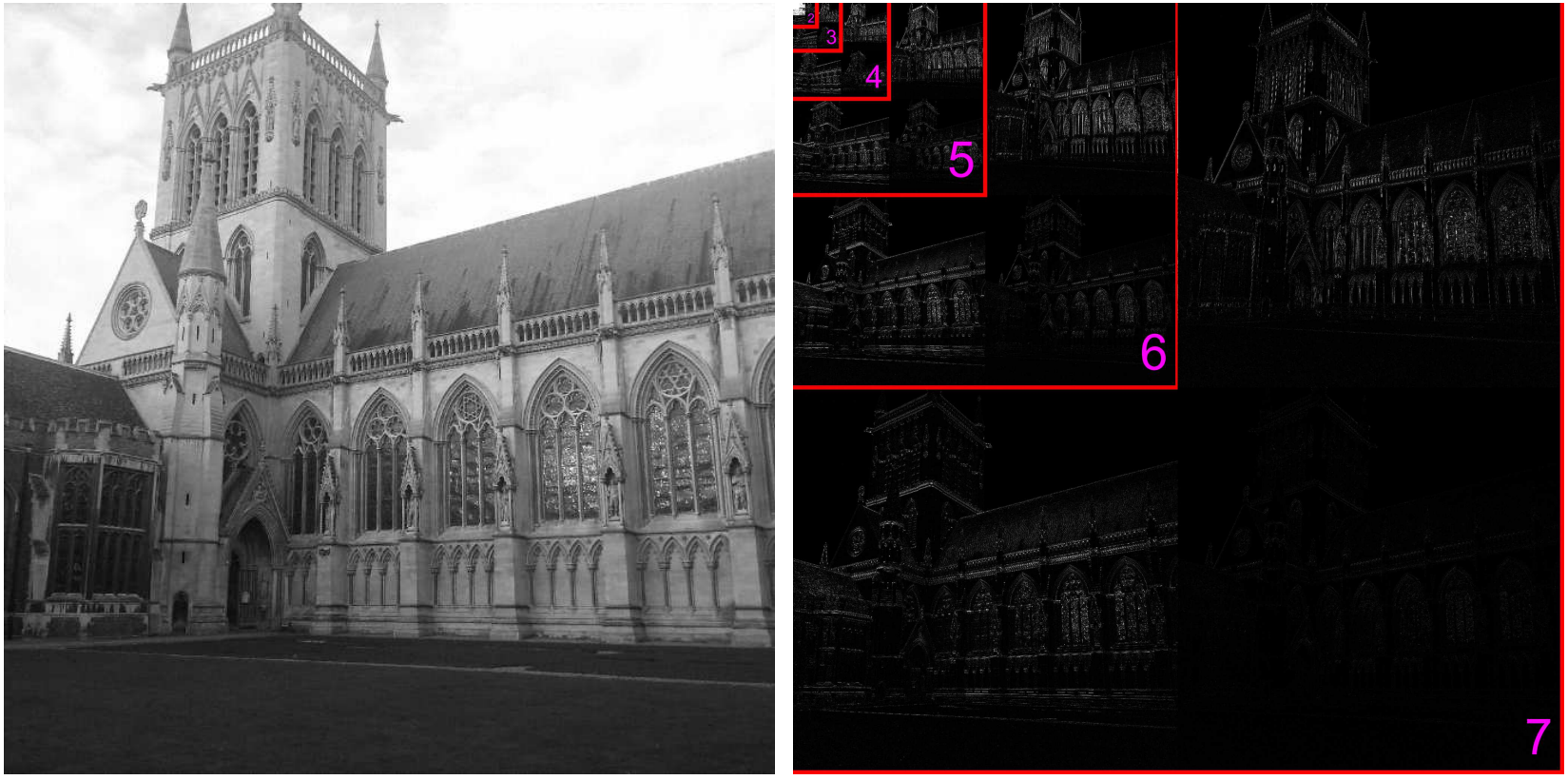


Figure: An image and its wavelet coefficients, where a brighter colour corresponds to a larger value.

Idea: Fully sample rows that correspond to the coarser wavelet levels and subsample the rows that correspond to the finer wavelet levels.

($\mathbf{s}$)-sparse vectors have s_k non-zero elements in each wavelet level. Denote these by $\Sigma_{\mathbf{s}}$.

$$\|x\|_{l^1_w} = \sum_{i=1}^{N} w_i \left|x_i\right|,$$
$$\sigma_{\mathbf{s}}(x)_{l^1_w} = \inf\{\|x - z\|_{l^1_w} : z \in \Sigma_{\mathbf{s}}\}.$$

In practice, expect $\sigma_{\mathbf{s}}(Wx)_{l^1_w}$ to be small for images.

$N = 2^{r \cdot d}$, r wavelet levels

Subsample randomly in dyadic Fourier bands

$$(m_{\mathbf{k}=(k_1,\ldots,k_d)})_{k_1,\ldots,k_d=1}^{r}.$$

Theorem (Stable Neural Networks Exist)

Let $\epsilon_{\mathbb{P}} \in (0,1)$ and $\mathbf{s} = (s_1, ..., s_r)$ describe $(\mathbf{s})$-sparse vectors corresponding wavelet scales (d-dimensional). Suppose

$$m_{\mathbf{k}} \gtrsim \left\{ \sum_{l=1}^{\|\boldsymbol{k}\|_\infty} s_l \prod_{i=1}^{d} 2^{-|k_i - l|} + \sum_{l=\|\boldsymbol{k}\|_\infty + 1}^{r} s_l 2^{-2(l - \|\boldsymbol{k}\|_\infty)} \prod_{i=1}^{d} 2^{-|k_i - l|} \right\} \cdot L,$$

$$L = r^3 \cdot \log(m) \cdot \log^2(rs) + \log(\epsilon_{\mathbb{P}}^{-1}).$$

Then, for each $n \in \mathbb{N}$, we construct a computable neural network ϕ_n^A from $\mathcal{T}$ with $3n$ layers such that with probability at least $1 - \epsilon_{\mathbb{P}}$, the following stable uniform recovery guarantee holds. For any $x \in \mathbb{C}^N$ with $\|x\|_{l^2} \lesssim 1$ and any $y \in \mathbb{C}^m$,

$$\|\phi_n^A(y) - x\|_{l^2} \lesssim \frac{\sigma_{\mathbf{s},\mathbf{M}}(Wx)_{l_w^1}}{\sqrt{s\sqrt{r}}} + \frac{r^{\frac{1}{4}} \|A\|}{n} + r^{\frac{1}{4}} \|Ax - y\|_{l^2}.$$

How to interpret?

- Up to log-factors, equivalent to oracle estimator (as $n \to \infty$).
- For sparse vectors and large n, neural networks are locally Lipschitz so stable.
- Number of samples required in each annular region

$$\sum_{\|\mathbf{k}\|=k} m_{\mathbf{k}} \gtrsim \left(s_k + \sum_{l=1}^{k-1} s_l 2^{-(k-l)} + \sum_{l=k+1}^{r} s_l 2^{-3(l-k)} \right) \cdot L$$

  is (up to logarithmic factors) proportional to s_k + exponentially decaying terms.

Numerical Example

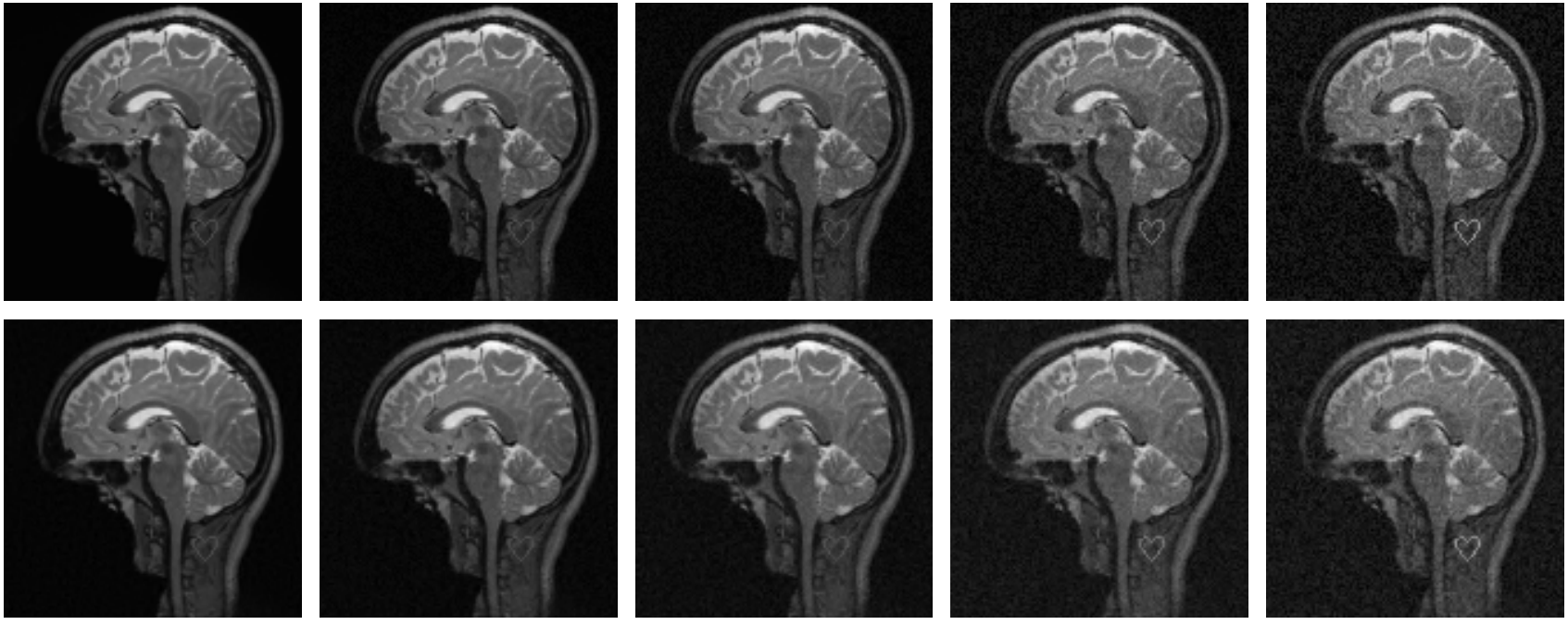


Figure: Stability test for new networks. Top row: original image with perturbations. Bottom row: reconstructions.

STABLE!

Conclusions

- The awesome performance of neural networks may come at a high price in terms of stability. Given the last fifty years of the studying stability via inverse problems, this is an important issue that should not be overlooked.
- There is likely a rich classification theory, stating limits on the performance of stable methods - trade-off.
- One such example was presented with explicitly constructed stable neural networks.
- Current state in compressed sensing only tells half the story. Even standard optimisation methods are susceptible to adversarial attacks!
- Next steps: extensively assessing the performance of these new neural networks, applying these ideas to other problems.

References

Pinkus, A. (1999).
Approximation theory of the mlp model in neural networks.
Acta numerica, 8:143–195.