

# Can stable and accurate neural networks always be computed?

**Matthew Colbrook** (Cambridge, [m.colbrook@damtp.cam.ac.uk](mailto:m.colbrook@damtp.cam.ac.uk))

Joint work with: **Vegard Antun** (Oslo), **Anders Hansen** (Cambridge)

**Based on:** M. Colbrook, V. Antun, A. Hansen, “Can stable and accurate neural networks be computed? - On the barriers of deep learning and Smale’s 18th problem”

**Code:** [www.github.com/Comp-Foundations-and-Barriers-of-AI/firenet](https://www.github.com/Comp-Foundations-and-Barriers-of-AI/firenet)

# Interest in deep learning unprecedented and exponentially growing

Google search (7th Jan) “deep learning” or “machine learning” yields  $\approx 2.5$  billion hits

Contrast with “computational mathematics” which has  $< 150$  million hits

## Machine Learning Arxiv Papers per Year

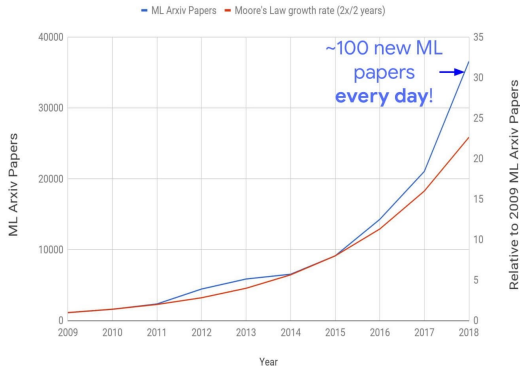


Figure: Source: ‘Deep Learning to Solve Challenging Problems’ (Google AI)

To keep up last year, you would need to continually read a paper every  $< 5$  mins!

# Why is AI suddenly such a big deal?

## **AI techniques are replacing humans in problem solving:**

- ▶ Self-driving vehicles
- ▶ Automated diagnosis in medicine and automated decision processes
- ▶ Automated weapon systems
- ▶ Music composition
- ▶ Call centres
- ▶ Any security system based on face or voice recognition
- ▶ Mathematical proofs

## **AI techniques are replacing established algorithms in science:**

- ▶ Medical imaging (MRI, CT, etc)
- ▶ Microscopy
- ▶ Imaging problems in general
- ▶ Radar, sonar, etc.
- ▶ Methods for solving PDEs

# Will AI replace standard algorithms in medical imaging?

“superior immunity to noise and a reduction in reconstruction artefacts compared with conventional handcrafted reconstruction methods”

nature > letters > article

a natureresearch journal

MENU ▾

nature



Search



E-alert



Submit



Login

We'd like to understand how you use our websites in order to improve them. [Register your interest.](#)

Published: 22 March 2018

## Image reconstruction by domain-transform manifold learning

Bo Zhu, Jeremiah Z. Liu, Stephen F. Cauley, Bruce R. Rosen & Matthew S. Rosen

*Nature* **555**, 487–492(2018) | [Cite this article](#)

17k Accesses | 235 Citations | 197 Altmetric | [Metrics](#)

### Abstract

Image reconstruction is essential for imaging applications across the physical and life sciences, including optical and radar systems, magnetic resonance imaging, X-ray computed tomography, positron emission tomography, ultrasound imaging and radio astronomy<sup>1,2,3</sup>. During image

You have full access to this article via  
**University of Oslo Oslo University Hospital**

Download PDF



### Editorial Summary

#### Machine learning improves image reconstruction

Reconstructing images from data, whether for medical or astronomical purposes, hinges on well-defined steps. The data sensor encodes an intermediate representation of the observed

[show all](#)




# A bold claim?


[nature](#) > [nature methods](#) > [research highlights](#) > [article](#)


a natureresearch journal


MENU ▾

nature methods

 Search

 E-alert

 Submit

 Login

We'd like to understand how you use our websites in order to improve them. [Register your interest.](#)

Published: 27 April 2018

Imaging

## AI transforms image reconstruction

Rita Strack

*Nature Methods* **15**, 309(2018) | [Cite this article](#)

**1254** Accesses | **2** Citations | **8** Altmetric | [Metrics](#)

**A deep-learning-based approach improves the speed, accuracy, and robustness of biomedical image reconstruction.**

Artificial intelligence (AI) and machine learning are poised to revolutionize the way biologists acquire and interact with experimental data. In biomedical imaging, such approaches have largely been focused on improving and automating the analysis of acquired images. For example, machine learning has been used to address the challenging

Download PDF



Sections

References

[References](#)

[Rights and permissions](#)

[About this article](#)

[Further reading](#)

# Very strong confidence in deep learning

**The New Yorker quotes Geoffrey Hinton (April 2017):**

“They should stop training radiologists now.”

BUT...

# DANGER: AI generated hallucinations

BBC

Sign in

News

Sport

Weather

Shop

Reel

Travel

More

Search



machine minds

What is BBC Future?

Latest

Best of..

Machine Minds

Future Now

## The 'weird events' that make machines hallucinate



By Linda Geddes

5 December 2018

Computers can be made to see a sea turtle as a gun or hear a concerto as someone's voice, which is raising concerns about using artificial intelligence in the real world.

# The danger of false positives

## Uber's self-driving car saw the pedestrian but didn't swerve - report

Tuning of car's software to avoid false positives blamed, as US National Transportation Safety Board investigation continues



▲ Uber's modified Volvo XC90 SUV detected but did not react to the crossing pedestrian in first self-driving car fatality, report says. Photograph: Volvo

An **Uber** self-driving test car which killed a woman crossing the street detected her but decided not to react immediately, a report has said.

The car was travelling at 40mph (64km/h) in self-driving mode when it **collided with 49-year-old Elaine Herzberg** at about 10pm on 18 March. Herzberg was pushing a bicycle across the road outside of a crossing. She later died from her injuries.

Although the car's sensors detected Herzberg, its software which decides how it should react was tuned too far in favour of ignoring objects in its path which might be "false positives" (such as plastic bags), according to **a report from the Information**. This meant the modified Volvo XC90 did not react fast enough.

# Deep Fool



**FIGURE 4.** Examples of natural images perturbed with the universal perturbation and their corresponding estimated labels with GoogLeNet. (a)–(h) Images belonging to the ILSVRC 2012 validation set. (i)–(l) Personal images captured by a mobile phone camera. (Figure used courtesy of [22].)

# DL is also unstable in inverse problems!

[Submit](#) [About](#) [Contact](#) [Journal Club](#) [Subscribe](#) [Log in](#)

Proceedings of the National Academy of Sciences of the United States of America

Keyword, Author, or DOI

[Advanced Search](#)

[Home](#) [Articles](#) [Front Matter](#) [News](#) [Podcasts](#) [Authors](#)

**NEW RESEARCH IN**

Physical Sciences

Social Sciences

Biological Sciences

**PHYSICAL SCIENCES**

## On instabilities of deep learning in image reconstruction and the potential costs of AI

Vegard Antun, Francesco Renna, Clarice Poon, Ben Adcock, and Anders C. Hansen

PNAS first published May 11, 2020 <https://doi.org/10.1073/pnas.1907377117>

Edited by David L. Donoho, Stanford University, Stanford, CA, and approved March 12, 2020 (received for review June 4, 2019)

Article

Figures & SI

Info & Metrics

PDF

Article Alerts

Email Article

Citation Tools

Request Permissions

Share

Tweet

Like 52

Mendeley

Submit

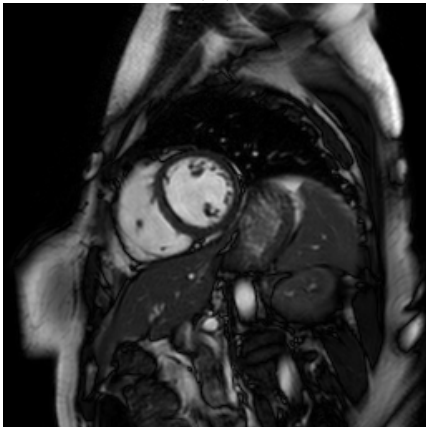
Sign up for Article Alerts

Enter Email Address

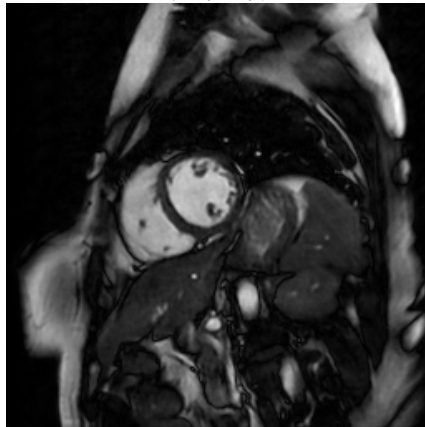
Sign up

# Example

$$|x|$$



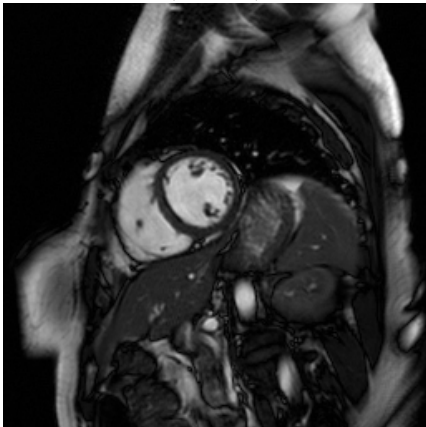
$$|\psi(Ax)|$$



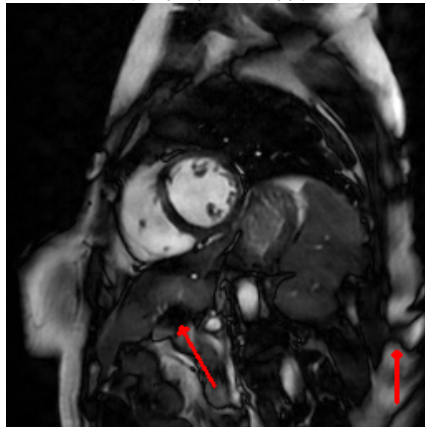
**Network (33% subsampling) from:** J. Schlemper, J. Caballero, J. V. Hajnal, A. Price and D. Rueckert, 'A deep cascade of convolutional neural networks for MR image reconstruction', in International conference on information processing in medical imaging, Springer, 2017, pp. 647–658.  
**Figures from:** Antun, V., Renna, F., Poon, C., Adcock, B., & Hansen, A. C., 'On instabilities of deep learning in image reconstruction and the potential costs of AI'. Proc. Natl. Acad. Sci. USA, 2020..

## Example

$$|x + r_1|$$



$$|\Psi(A(x + r_1))|$$

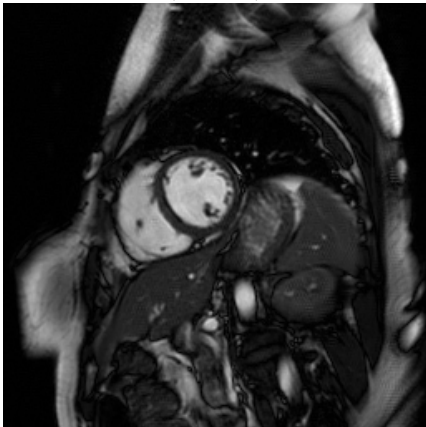


**Network (33% subsampling) from:** J. Schlemper, J. Caballero, J. V. Hajnal, A. Price and D. Rueckert, 'A deep cascade of convolutional neural networks for MR image reconstruction', in International conference on information processing in medical imaging, Springer, 2017, pp. 647–658.  
**Figures from:** Antun, V., Renna, F., Poon, C., Adcock, B., & Hansen, A. C., 'On instabilities of deep learning in image reconstruction and the potential costs of AI'. Proc. Natl. Acad. Sci. USA, 2020..

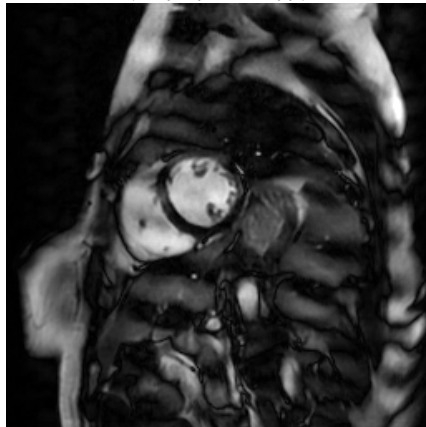


# Example

$$|x + r_2|$$



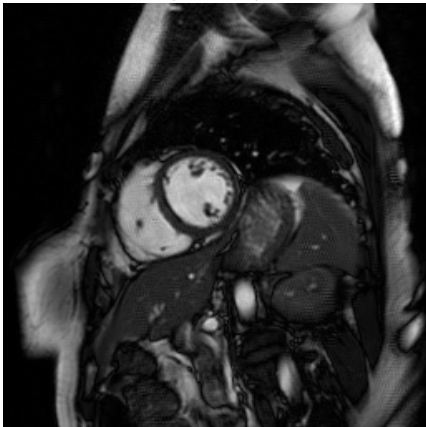
$$|\Psi(A(x + r_2))|$$



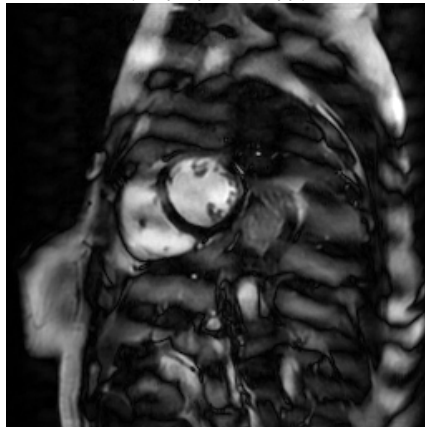
**Network (33% subsampling) from:** J. Schlemper, J. Caballero, J. V. Hajnal, A. Price and D. Rueckert, 'A deep cascade of convolutional neural networks for MR image reconstruction', in International conference on information processing in medical imaging, Springer, 2017, pp. 647–658.  
**Figures from:** Antun, V., Renna, F., Poon, C., Adcock, B., & Hansen, A. C., 'On instabilities of deep learning in image reconstruction and the potential costs of AI'. Proc. Natl. Acad. Sci. USA, 2020..

# Example

$$|x + r_3|$$



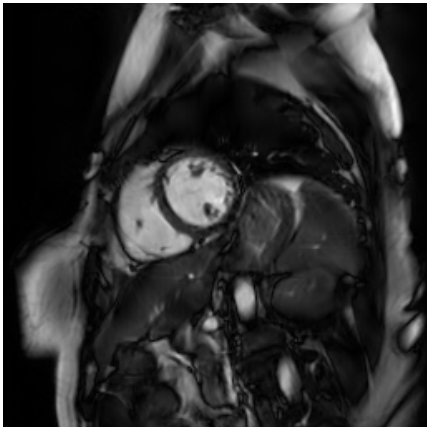
$$|\Psi(A(x + r_3))|$$



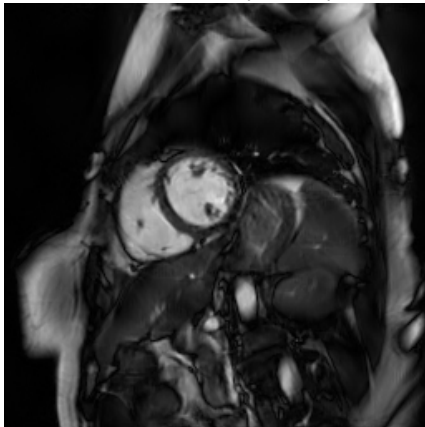
**Network (33% subsampling) from:** J. Schlemper, J. Caballero, J. V. Hajnal, A. Price and D. Rueckert, 'A deep cascade of convolutional neural networks for MR image reconstruction', in International conference on information processing in medical imaging, Springer, 2017, pp. 647–658.  
**Figures from:** Antun, V., Renna, F., Poon, C., Adcock, B., & Hansen, A. C., 'On instabilities of deep learning in image reconstruction and the potential costs of AI'. Proc. Natl. Acad. Sci. USA, 2020..

# Reconstruction using state-of-the-art standard methods

SoA from  $Ax$



SoA from  $A(x + r_3)$

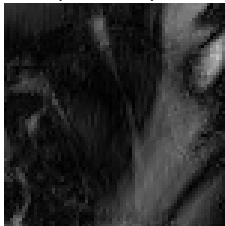


# AI generated hallucinations with random noise

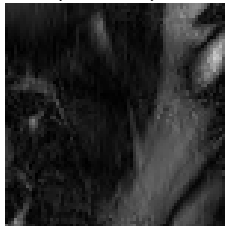
$|x + v_1|$   
(Full image)



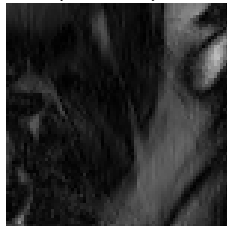
$\Phi(A(x + v_1))$   
(Cropped)



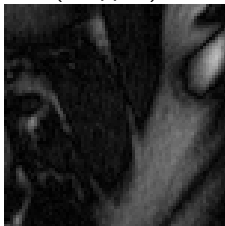
$\Phi(A(x + v_2))$   
(Cropped)



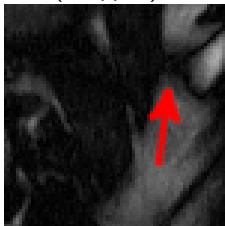
$\Phi(A(x + v_3))$   
(Cropped)



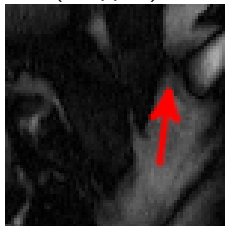
$|x + v_1|$   
(Cropped)



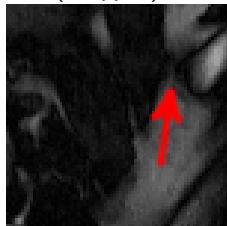
$\Psi(A(x + v_1))$   
(Cropped)



$\Psi(A(x + v_2))$   
(Cropped)



$\Psi(A(x + v_3))$   
(Cropped)



Worst of 100

Worst of 20

Worst of 1

# Facebook and NYU's 2020 FastMRI challenge

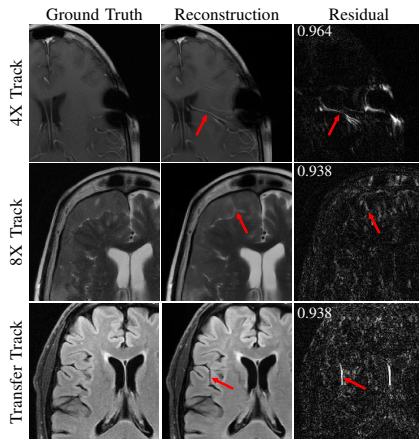


Fig. 6. Examples of reconstruction hallucinations among challenge submissions with SSIM scores over residual plots (residuals magnified by 5). *(top)* A 4X submission from Neurospin generated a false vessel, possibly related to susceptibilities introduced by surgical staples. *(middle)* An 8X submission from ATB introduced a linear bright signal mimicking a cleft of cerebrospinal fluid, as well as blurring of the boundaries of the extra-axial mass. *(bottom)* A submission from ResoNNance introduced a false sulcus or prominent vessel.

# Optimism: Echoes of an old story

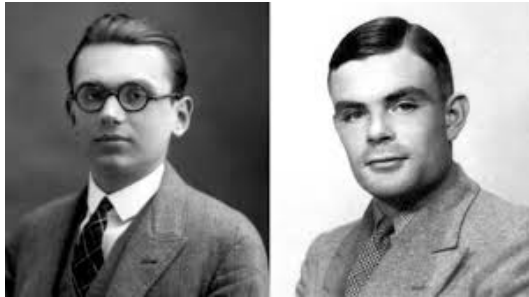
Hilbert's vision at start of 20th century: provide secure foundations for all mathematics.

- ▶ All mathematical statements should be written in a precise formal language, and manipulated according to well defined rules.
- ▶ Completeness: a proof that all true mathematical statements can be proved in the formalism.
- ▶ Consistency: a proof that no contradiction can be obtained in the formalism of mathematics.
- ▶ Decidability: an algorithm for deciding the truth or falsity of any mathematical statement.



**Hilbert's 10th problem:** Provide an algorithm which, for any given Diophantine equation (polynomial equation with integer coefficients and finite number of unknowns), can decide whether the equation has an integer-valued solution.

**Foundations  $\Rightarrow$  better understanding, discover feasible directions for techniques, discover new methods,...**



Gödel (arguably father of **modern logic**) and Turing (arguably father of **modern computer science**) turned Hilbert's optimism upside down:

- ▶ True statements in mathematics that cannot be proven.
- ▶ Problems that cannot be computed by an algorithm.

**Hilbert's 10th problem (Solution in 1970, Matiyasevich):** No such algorithm exists.

# A program for the foundations of DL and AI

**Smale's 18th problem\*:** *What are the limits of artificial intelligence?*

A program determining the foundations/limitations of deep learning and AI is needed:

- ▶ Boundaries of methodologies.
- ▶ Universal/intrinsic boundaries (e.g. no algorithm can do it).

There is a key difference between existence and construction here.

Need to also incorporate two pillars of numerical analysis:

- ▶ Stability
- ▶ Accuracy

**GOAL for rest of talk:** Develop some results in this direction for inverse problems.

\*Steve Smale composed a list of problems for the 21st century in reply to a request of Vladimir Arnold inspired by Hilbert's list.



# Mathematical setup

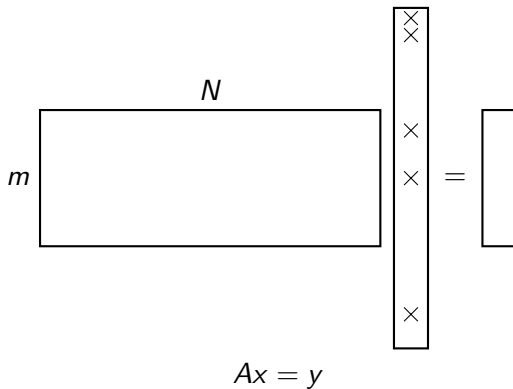
Given measurements  $y = Ax + e$  recover  $x \in \mathbb{C}^N$ .

- ▶  $x \in \mathbb{C}^N$  be an unknown vector,
- ▶  $A \in \mathbb{C}^{m \times N}$  be a matrix ( $m < N$ ) describing modality (e.g. MRI), and
- ▶  $y = Ax + e$  the noisy measurements of  $x$ .

## Outline:

- ▶ Fundamental barriers
- ▶ Sufficient conditions and Fast Iterative REstarted NETworks (FIRENETs)
- ▶ Balancing stability and accuracy

# Sparse linear systems



The diagram illustrates a sparse linear system  $Ax = y$ . It features a large rectangular matrix  $A$  with dimensions  $m$  (height) and  $N$  (width). To the right of  $A$  is a tall, narrow vertical vector  $x$ , which is sparse, indicated by 'x' marks in only a few of its cells. An equals sign follows, leading to a small vertical rectangular vector  $y$ . Below the matrix and vectors, the equation  $Ax = y$  is written.

$$Ax = y$$

We say that a vector  $x \in \mathbb{C}^N$  is **s-sparse**, if it has at most  $s$  non-zero components.

# Sparse solutions of underdetermined systems have many applications!

- ▶ Linear regression in statistics – The LASSO
- ▶ Medical imaging - MRI, CT, microscopy ...
- ▶ Non-linear function approximation
- ▶ Error correction
- ▶ Explainable AI - LIME
- ▶ Dictionary learning and sparse coding
- ▶ Classification

# Can we compute neural networks that solve $(P_j)$ ?

Sparse regularisation (benchmark method):

$$\min_{x \in \mathbb{C}^N} \|x\|_{l^1} \quad \text{subject to} \quad \|Ax - y\|_{l^2} \leq \eta \quad (P_1)$$

$$\min_{x \in \mathbb{C}^N} \lambda \|x\|_{l^1} + \|Ax - y\|_{l^2}^2 \quad (P_2)$$

$$\min_{x \in \mathbb{C}^N} \lambda \|x\|_{l^1} + \|Ax - y\|_{l^2} \quad (P_3)$$

Denote the **minimising** vectors by  $\Xi$ .

- ▶ Avoid bizarre, unnatural & pathological mappings:  $(P_j)$  well-understood & well-used!
- ▶ Simpler solution map than inverse problem  $\Rightarrow$  stronger impossibility results.
- ▶ DL has also been used to speed up sparse regularisation and tackle  $(P_j)$ .

# Approximation qualities of neural nets

## Theorem (Universal Approximation Theorem)

Let  $\rho \in C(\mathbb{R})$  (activation function) and assume that  $\rho$  is not a polynomial. Let  $K \subset \mathbb{R}^d$  be compact,  $f \in C(K)$  and  $\epsilon > 0$ . Then there exists a neural network (with one hidden layer)  $\phi$  such that

$$\sup_{x \in K} |\phi(x) - f(x)| \leq \epsilon.$$

## Theorem (Universal Interpolation Theorem)

Let  $\rho \in C(\mathbb{R})$  and assume that  $\rho$  is not a polynomial. For any  $k$  distinct points  $\{x_j\}_{j=1}^k \subset \mathbb{R}^d$  and associated data  $\{\alpha_j\}_{j=1}^k \subset \mathbb{R}$ . Then there exists a neural network (with one hidden layer)  $\phi$  such that

$$\phi(x_j) = \alpha_j, \quad j = 1, \dots, k.$$

# Approximation qualities of neural nets

A zoo of so-called “universal approximation” theorems. However, this is not enough:

- (a) Other methods (e.g. polynomials, splines, wavelets, etc.) have universal approximation theorems. Why are NNs so effective? E.g., are there useful classes of functions that are efficiently approximated by NNs but not classical methods?
- (b) We want to construct or compute a good neural network. There is a subtle difference between existence and computability (more on this later).

We will focus on point (b). For point (a) (which is largely open) see, for example:

DeVore, R., Hanin, B. and Petrova, G., 2020. *Neural Network Approximation*. arXiv preprint arXiv:2012.14501.

## What could go wrong?

$$\min_{x \in \mathbb{C}^N} \|x\|_{l^1} \quad \text{subject to} \quad \|Ax - y\|_{l^2} \leq \eta \quad (P_1)$$

$$\min_{x \in \mathbb{C}^N} \lambda \|x\|_{l^1} + \|Ax - y\|_{l^2}^2 \quad (P_2)$$

$$\min_{x \in \mathbb{C}^N} \lambda \|x\|_{l^1} + \|Ax - y\|_{l^2} \quad (P_3)$$

- (i) There does not exist a neural network that approximates the function we are interested in.
- (ii)
- (iii)

## What could go wrong?

$$\min_{x \in \mathbb{C}^N} \|x\|_{l^1} \quad \text{subject to} \quad \|Ax - y\|_{l^2} \leq \eta \quad (P_1)$$

$$\min_{x \in \mathbb{C}^N} \lambda \|x\|_{l^1} + \|Ax - y\|_{l^2}^2 \quad (P_2)$$

$$\min_{x \in \mathbb{C}^N} \lambda \|x\|_{l^1} + \|Ax - y\|_{l^2} \quad (P_3)$$

- (i) ~~There does not exist a neural network that approximates the function we are interested in.~~
- (ii)
- (iii)



## What could go wrong?

$$\min_{x \in \mathbb{C}^N} \|x\|_{l^1} \quad \text{subject to} \quad \|Ax - y\|_{l^2} \leq \eta \quad (P_1)$$

$$\min_{x \in \mathbb{C}^N} \lambda \|x\|_{l^1} + \|Ax - y\|_{l^2}^2 \quad (P_2)$$

$$\min_{x \in \mathbb{C}^N} \lambda \|x\|_{l^1} + \|Ax - y\|_{l^2} \quad (P_3)$$

- (i) ~~There does not exist a neural network that approximates the function we are interested in.~~
- (ii) There does exist a neural network that approximates the function, however, there does not exist an algorithm that can construct the neural network.
- (iii)

## What could go wrong?

$$\min_{x \in \mathbb{C}^N} \|x\|_{l^1} \quad \text{subject to} \quad \|Ax - y\|_{l^2} \leq \eta \quad (P_1)$$

$$\min_{x \in \mathbb{C}^N} \lambda \|x\|_{l^1} + \|Ax - y\|_{l^2}^2 \quad (P_2)$$

$$\min_{x \in \mathbb{C}^N} \lambda \|x\|_{l^1} + \|Ax - y\|_{l^2} \quad (P_3)$$

- (i) ~~There does not exist a neural network that approximates the function we are interested in.~~
- (ii) There does exist a neural network that approximates the function, however, there does not exist an algorithm that can construct the neural network.
- (iii) There does exist a neural network that approximates the function, and an algorithm to construct it. However, the algorithm will need prohibitively many samples.

## The set-up

$A \in \mathbb{C}^{m \times N}$  (modality),  $\mathcal{S} = \{y_k\}_{k=1}^R \subset \mathbb{C}^m$  (samples),  $R < \infty$

**Question:** Given a collection  $\Omega$  of  $(A, \mathcal{S})$ , does there exist a neural network approximating  $\Xi$  (solution map of  $(P_j)$ ), and can it be trained by an algorithm?

In practice, the matrix  $A$  is not known exactly or cannot be stored to infinite precision.

**Assume access to:**  $\{y_{k,n}\}_{k=1}^R$  and  $A_n$  (rational approximations, e.g. floats) such that

$$\|y_{k,n} - y_k\| \leq 2^{-n}, \quad \|A_n - A\| \leq 2^{-n}, \quad \forall n \in \mathbb{N}.$$

And  $\{x_{k,n}\}_{k=1}^R$  such that  $\inf_{x^* \in \Xi(A_n, y_{k,n})} \|x_{k,n} - x^*\| \leq 2^{-n}, \quad \forall n \in \mathbb{N}.$

Training set associated with  $(A, \mathcal{S}) \in \Omega$  is

$$\iota_{A, \mathcal{S}} := \{(y_{k,n}, A_n, x_{k,n}) \mid k = 1, \dots, R, \text{ and } n \in \mathbb{N}\}.$$

# Good news - a good neural network exists

## Theorem (Neural networks exist for $\Xi$ )

For  $(P_j)$  and any family  $\Omega$  of  $(A, S)$ , there exists a mapping

$$\mathcal{K}: \iota_{A,S} \rightarrow \varphi_{A,S} \text{ (a neural network)}$$

such that  $\varphi_{A,S}(y)$  solves  $(P_j)$  for each  $y \in S$ . In other words,  $\mathcal{K}$  maps the training data to NNs that solve the optimisation problem  $(P_j)$  for each  $(A, S) \in \Omega$ .

Proof.

Easy - apply universal approximation/interpolation theorems.



# Bad news - can't necessarily approximate such a neural network

## Theorem

For  $(P_j)$ ,  $N \geq 2$  and  $m < N$ . Let  $K > 2$  be a positive integer,  $L \in \mathbb{N}$ . Then there exists a **well-conditioned** class (condition numbers  $\leq 1$ )  $\Omega$  of elements  $(A, S)$  s.t. ( $\Omega$  fixed in what follows):

- (i) **There does not exist any algorithm** that, given a training set  $\iota_{A,S}$ , produces a neural network  $\phi_{A,S}$  with

$$\min_{y \in S} \inf_{x^* \in \Xi(A,y)} \|\phi_{A,S}(y) - x^*\|_{l_2} \leq 10^{-K}, \quad \forall (A, S) \in \Omega. \quad (1)$$

Furthermore, for any  $p > 1/2$ , **no probabilistic algorithm** can produce a neural network  $\phi_{A,S}$  such that (1) holds with probability at least  $p$ .

- (ii) **There exists an algorithm** that produces a neural network  $\phi_{A,S}$  such that

$$\max_{y \in S} \inf_{x^* \in \Xi(A,y)} \|\phi_{A,S}(y) - x^*\|_{l_2} \leq 10^{-(K-1)}, \quad \forall (A, S) \in \Omega.$$

However, for any such algorithm (even probabilistic),  $M \in \mathbb{N}$  and  $p \in \left[0, \frac{N-m}{N+1-m}\right)$ , there exists a training set  $\iota_{A,S}$  such that for all  $y \in S$ ,

$$\mathbb{P}\left(\inf_{x^* \in \Xi(A,y)} \|\phi_{A,S}(y) - x^*\|_{l_2} > 10^{1-K} \text{ or size of training data needed} > M\right) > p.$$

- (iii) **There exists an algorithm** using only  $L$  training data from each  $\iota_{A,S}$  that produces a neural network  $\phi_{A,S}(y)$  such that

$$\max_{y \in S} \inf_{x^* \in \Xi(A,y)} \|\phi_{A,S}(y) - x^*\|_{l_2} \leq 10^{-(K-2)}, \quad \forall (A, S) \in \Omega.$$

## In words...

Nice classes  $\Omega$  where one can prove NNs with great approximation qualities exist. But:

- ▶ No algorithm, even randomised can train (or compute) such a NN accurate to  $K$  digits with probability greater than  $1/2$ .
- ▶ There exists a deterministic algorithm that computes a NN with  $K - 1$  correct digits, but any such (even randomised) algorithm needs arbitrarily many training data.
- ▶ There exists a deterministic algorithm that computes a NN with  $K - 2$  correct digits using no more than  $L$  training samples.

Result **independent of neural network architecture** - a universal barrier.

Existence vs computation (universal approximation/interpolation theorems **not** enough).

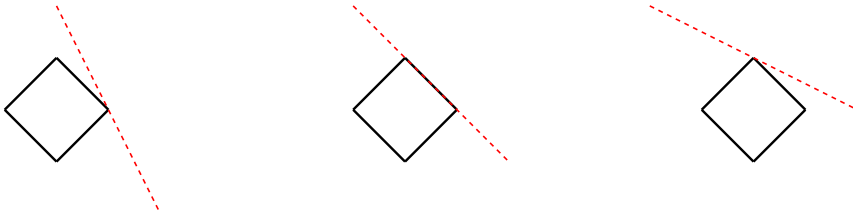
**Conclusion:** Theorems on existence of neural networks may have little to do with the neural networks produced in practice.

## Numerical example: fails with training methods

$\text{dist}(\Psi_{A_n}(y_n), \Xi_3(A, y))$	$\text{dist}(\Phi_{A_n}(y_n), \Xi_3(A, y))$	$\ A_n - A\  \leq 2^{-n}$ $\ y_n - y\ _{l^2} \leq 2^{-n}$	$10^{-K}$	$\Omega_K$
0.2999690	0.2597827	$n = 10$	$10^{-1}$	$K = 1$
0.3000000	0.2598050	$n = 20$	$10^{-1}$	$K = 1$
0.3000000	0.2598052	$n = 30$	$10^{-1}$	$K = 1$
0.0030000	0.0025980	$n = 10$	$10^{-3}$	$K = 3$
0.0030000	0.0025980	$n = 20$	$10^{-3}$	$K = 3$
0.0030000	0.0025980	$n = 30$	$10^{-3}$	$K = 3$
0.0000030	0.0000015	$n = 10$	$10^{-6}$	$K = 6$
0.0000030	0.0000015	$n = 20$	$10^{-6}$	$K = 6$
0.0000030	0.0000015	$n = 30$	$10^{-6}$	$K = 6$

**Table:** (Impossibility of computing the existing neural network to arbitrary accuracy).  $A$  constructed from discrete cosine transform,  $R = 8000$ ,  $N = 20$ ,  $m = 19$ , solutions are 6-sparse. We demonstrate the impossibility statement (i) on FIRENETs  $\Phi_{A_n}$ , and LISTA (learned iterative shrinkage thresholding algorithm) networks  $\Psi_{A_n}$ . The table shows the shortest  $l^2$  distance between the output from the networks, and the true minimizer of the problem ( $P_3$ ), with  $w_l = 1$  and  $\lambda = 1$ , for different values of  $n$  and  $K$ .

## The basic mechanism



Similar phase transitions can be built for  $(P_j)$  in arbitrary dimensions.



# Can we avoid this?

$$\hat{x} = \operatorname{argmin} f(x), \quad f^* = \min f(x)$$

**Question:** Can we find 'good' input classes where

$$f(x) < f^* + \epsilon \implies \|x - \hat{x}\| \lesssim \epsilon$$

We shall see that the answer is yes!

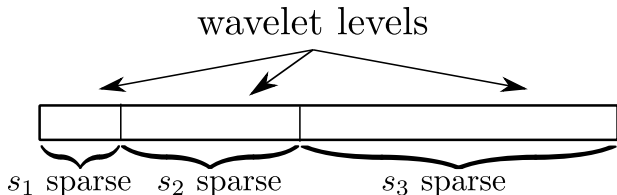
# State-of-the-art model for sparse regularisation

**Definition [Sparsity in levels]:** Let  $\mathbf{M} = (M_1, \dots, M_r) \in \mathbb{N}^r$ , where  $1 \leq M_1 < \dots < M_r = N$ , and  $\mathbf{s} = (s_1, \dots, s_r) \in \mathbb{N}_0^r$ , where  $s_k \leq M_k - M_{k-1}$  for  $k = 1, \dots, r$  and  $M_0 = 0$ . A vector  $x \in \mathbb{C}^N$  is  $(\mathbf{s}, \mathbf{M})$ -sparse in levels if

$$|\text{supp}(x) \cap \{M_{k-1} + 1, \dots, M_k\}| \leq s_k, \quad k = 1, \dots, r.$$

The total sparsity is  $s = s_1 + \dots + s_r$ . We denote the set of  $(\mathbf{s}, \mathbf{M})$ -sparse vectors by  $\Sigma_{\mathbf{s}, \mathbf{M}}$ . We also define the following measure of distance of a vector  $x$  to  $\Sigma_{\mathbf{s}, \mathbf{M}}$  by

$$\sigma_{\mathbf{s}, \mathbf{M}}(x)_{l_w^1} = \inf \{ \|x - z\|_{l_w^1} : z \in \Sigma_{\mathbf{s}, \mathbf{M}} \}.$$



# The robust nullspace property

**Definition [weighted rNSP in levels]:** Let  $(\mathbf{s}, \mathbf{M})$  be local sparsities and sparsity levels respectively. For weights  $\{w_i\}_{i=1}^N$  ( $w_i > 0$ ), we say that  $A \in \mathbb{C}^{m \times N}$  satisfies the weighted robust null space property in levels (weighted rNSPL) of order  $(\mathbf{s}, \mathbf{M})$  with constants  $0 < \rho < 1$  and  $\gamma > 0$  if for any  $(\mathbf{s}, \mathbf{M})$  support set  $\Delta$ ,

$$\|x_\Delta\|_{\ell^2} \leq \frac{\rho \|x_{\Delta^c}\|_{\ell_w^1}}{\sqrt{\xi}} + \gamma \|Ax\|_{\ell^2}, \quad \text{for all } x \in \mathbb{C}^N.$$

$$\xi := \sum_{k=1}^r w_{(k)}^2 s_k, \quad \zeta := \min_{k=1, \dots, r} w_{(k)}^2 s_k, \quad \kappa := \frac{\xi}{\zeta}.$$

$$\begin{aligned} \text{rNSPL} \Rightarrow \|z_1 - z_2\|_{\ell^2} &\lesssim \underbrace{\sigma_{\mathbf{s}, \mathbf{M}}(z_2)_{\ell_w^1}}_{\text{"small"}} + \|Az_2 - y\|_{\ell^2} \\ &+ \underbrace{(\lambda \|z_1\|_{\ell_w^1} + \|Az_1 - y\|_{\ell^2} - \lambda \|z_2\|_{\ell_w^1} - \|Az_2 - y\|_{\ell^2})}_{F_3^A(z_1, y, \lambda) - F_3^A(z_2, y, \lambda)}, \end{aligned}$$

# Main result

**Simplified version of Theorem:** *We provide an algorithm such that:*

Input: *Sparsity parameters  $(\mathbf{s}, \mathbf{M})$ , weights  $\{w_i\}_{i=1}^N$ ,  $A \in \mathbb{C}^{m \times N}$  (with the input  $A$  given by  $\{A_l\}$ ) satisfying the rNSPL with constants  $0 < \rho < 1$  and  $\gamma > 0$ ,  $n \in \mathbb{N}$  and positive  $\{\delta, b_1, b_2\}$ .*

Output: *A neural network  $\phi_n$  with  $\mathcal{O}(n)$  layers and the following property.*

*For any  $x \in \mathbb{C}^N$  and  $y \in \mathbb{C}^m$  with*

$$\underbrace{\sigma_{\mathbf{s}, \mathbf{M}}(x)_{l_w^1}}_{\text{distance to sparse in levels vectors}} + \underbrace{\|Ax - y\|_{l^2}}_{\text{noise of measurements}} \lesssim \delta, \quad \|x\|_{l^2} \lesssim b_1, \quad \|y\|_{l^2} \lesssim b_2,$$

*we have the following **stable** and **exponential convergence** guarantee in  $n$*

$$\|\phi_n(y) - x\|_{l^2} \lesssim \delta + e^{-n}.$$

## Comments

- ▶ Architecture inspired by restarted & reweighted unrolling of primal-dual algorithm for:

$$(P_3) \quad \operatorname{argmin}_{x \in \mathbb{C}^N} F_3^A(x, y, \lambda) := \lambda \|x\|_{l_w^1} + \|Ax - y\|_{l^2}.$$

- ▶ As well as stability, rNSPL allows exponential convergence.
- ▶ Even ignoring stability, naive unrolling of iterative methods only gives slow convergence  $\mathcal{O}(\delta + n^{-1})$  (and in certain regimes  $\mathcal{O}(\delta + n^{-2})$ ).
- ▶ If we do not know  $\rho$  or  $\gamma$  (constants for rNSPL), can perform log-scale grid search for suitable parameters (increase width of layers by a factor of  $\log(n)$ ). Sometimes (see below) we know  $\rho$  and  $\gamma$  with probabilistic bounds.
- ▶ Also bound error when approximately applying the nonlinear maps of the NNs, we show these errors can only accumulate slowly as  $n$  increases  $\Rightarrow$  numerical stability.

# Examples in image recovery ( $\Psi = \text{Haar Wavelet transform}$ )

## Theorem

Consider recovering wavelet coeffs.  $x = \Psi c$  of  $c \in \mathbb{C}^{K^d}$  from subsampled noisy Fourier or Walsh measurements  $y = DP_{\mathcal{I}} Vc + e$ . Let  $A = DP_{\mathcal{I}} V\Psi^*$ ,  $m = |\mathcal{I}|$ ,  $\epsilon_{\mathbb{P}} \in (0, 1)$ .

(i) If  $\mathcal{I}$  is a random sampling pattern drawn according to strategy in paper, and

$$m \gtrsim (s_1 + \cdots + s_r) \cdot \mathcal{L}.$$

Then with prob.  $1 - \epsilon_{\mathbb{P}}$ ,  $A$  satisfies wrNSPL of order  $(\mathbf{s}, \mathbf{M})$ ,  $(\rho, \gamma) = (1/2, \sqrt{2})$ .

(ii) For any  $\delta \in (0, 1)$ , let  $\mathcal{J}(\delta, \mathbf{s}, \mathbf{M}, w)$  be the set of all  $y = Ax + e \in \mathbb{C}^m$  where

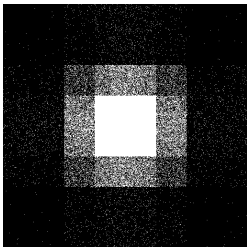
$$\|x\|_{l^2} \leq 1, \quad \max \{ \sigma_{\mathbf{s}, \mathbf{M}}(x)_{l_w^1}, \|e\|_{l^2} \} \leq \delta. \quad (2)$$

We provide an algorithm that constructs a NN  $\phi$  with  $\mathcal{O}(\log(\delta^{-1}))$  hidden layers (width bounded by  $2(N + m)$ ) s.t. with probability at least  $1 - \epsilon_{\mathbb{P}}$ ,

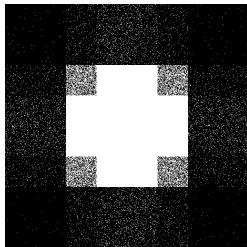
$$\|\phi(y) - c\|_{l^2} \lesssim \delta, \quad \forall y = Ax + e \in \mathcal{J}(\delta, \mathbf{s}, \mathbf{M}, w).$$

## Fourier sampling patterns

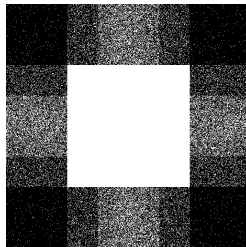
15%



25%

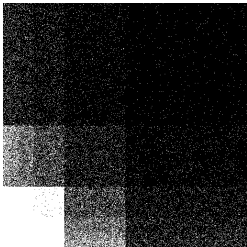


40%

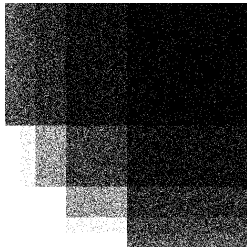


## Walsh sampling patterns

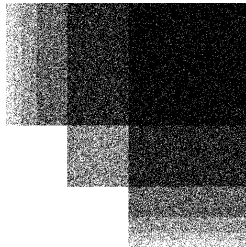
15%



25%



40%



# Demonstration of convergence

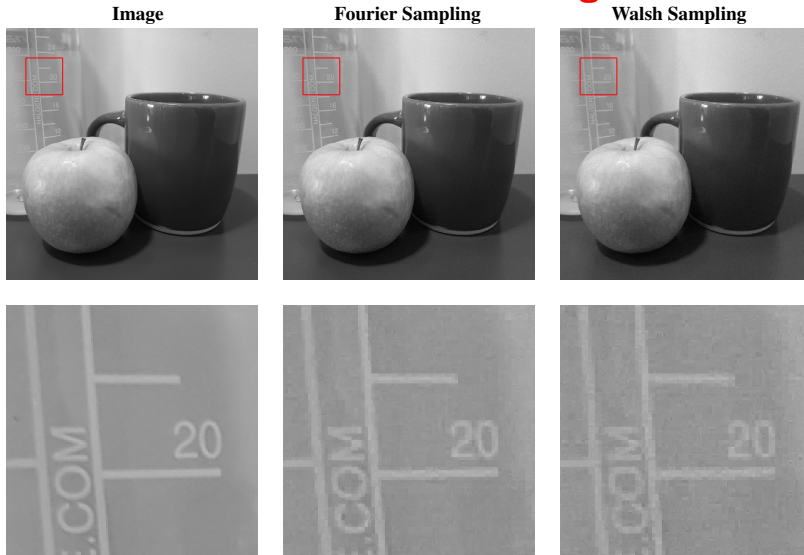
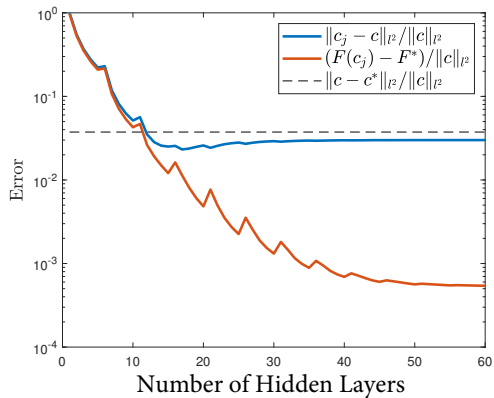


Figure: Images corrupted with 2% Gaussian noise and reconstructed using 15% sampling.

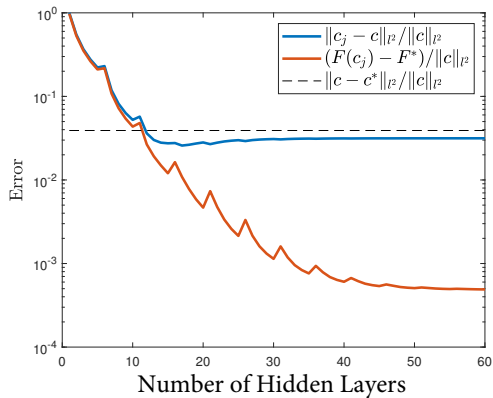


# Demonstration of convergence

## Convergence, Fourier Sampling

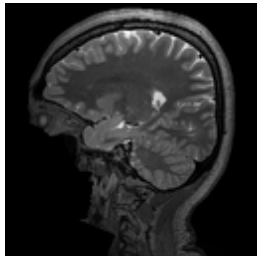


## Convergence, Walsh Sampling

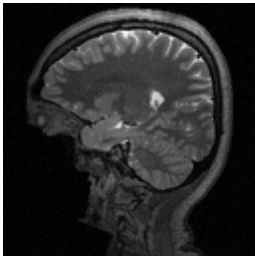


# Stable? AUTOMAP $\times$

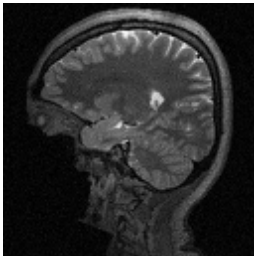
Original  $x$



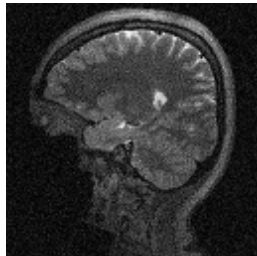
$|x + r_1|$



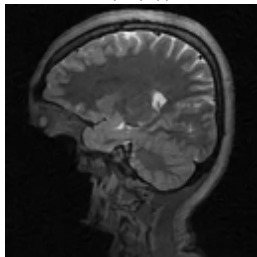
$|x + r_2|$



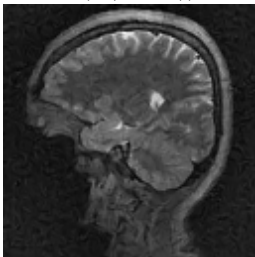
$|x + r_3|$



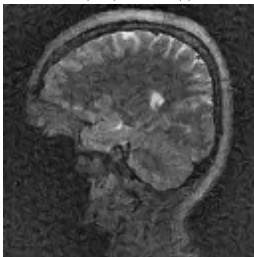
$\Psi(A(x))$



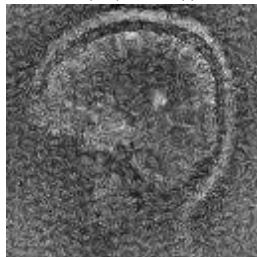
$\Psi(A(x + r_1))$



$\Psi(A(x + r_2))$

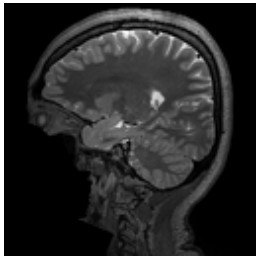


$\Psi(A(x + r_3))$

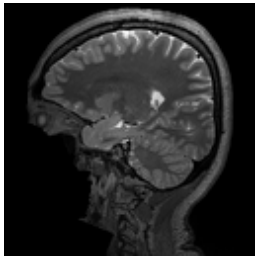


# Stable? FIRENETs ✓

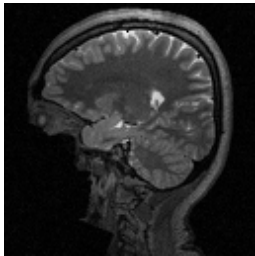
Original  $x$



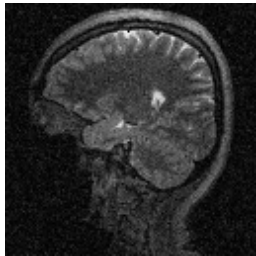
$|x + v_1|$



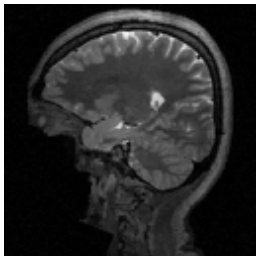
$|x + v_2|$



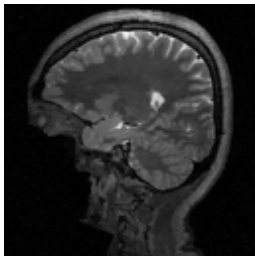
$|x + v_3|$



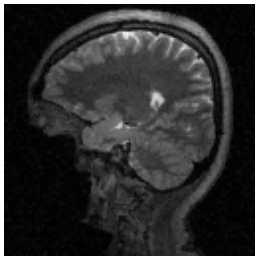
$\Phi(A(x))$



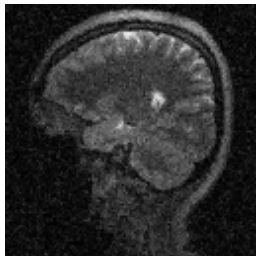
$\Phi(A(x + v_1))$



$\Phi(A(x + v_2))$

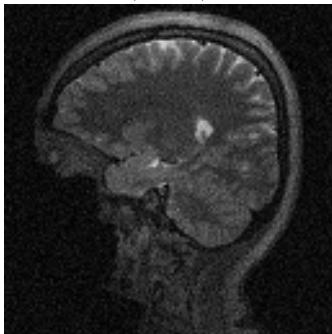


$\Phi(A(x + v_3))$



# Adding FIRENET layers stabilises AUTOMAP

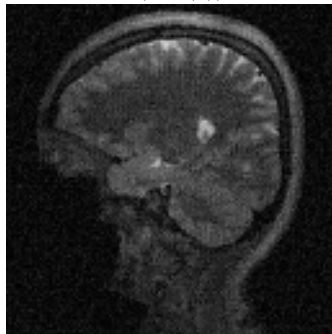
$$|x + r_3|$$



$$\Psi(\tilde{y}), \tilde{y} = A(x + r_3)$$

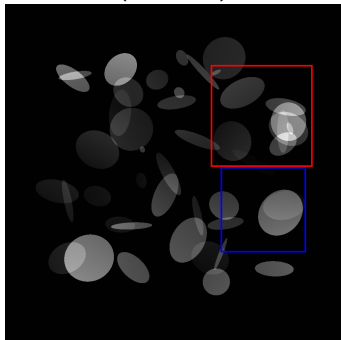


$$\Phi(\tilde{y}, \Psi(\tilde{y}))$$

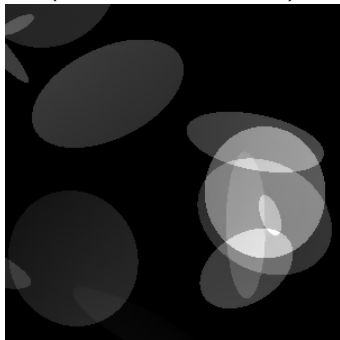


# Stability and accuracy, and false negative

Original  $x$   
(full size)



Original  
(cropped, red frame)

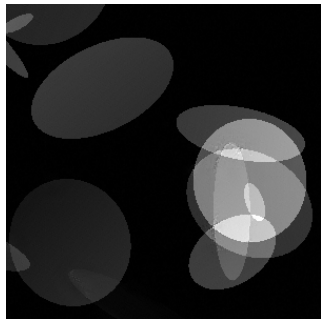


Original + detail ( $x + h_1$ )  
(cropped, blue frame)

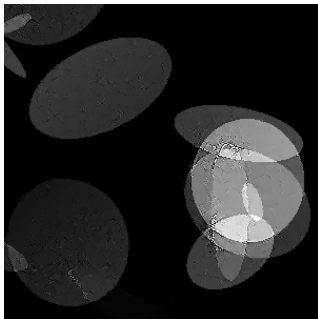


## U-net trained without noise

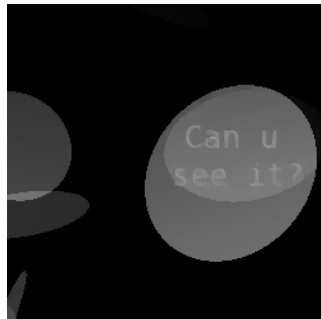
Orig. + worst-case noise



Rec. from worst-case noise

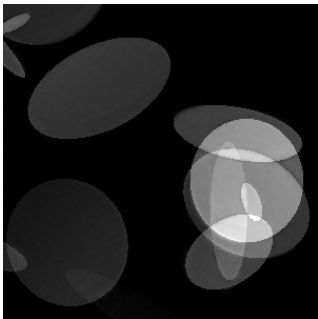


Rec. of detail

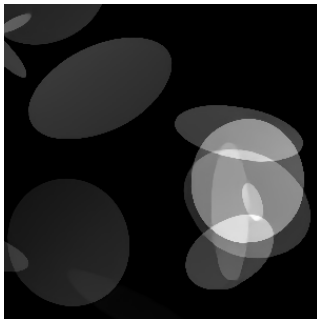


## U-net trained with noise

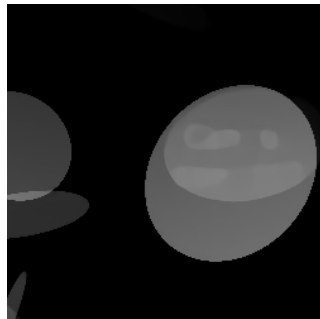
Orig. + worst-case noise



Rec. from worst-case noise



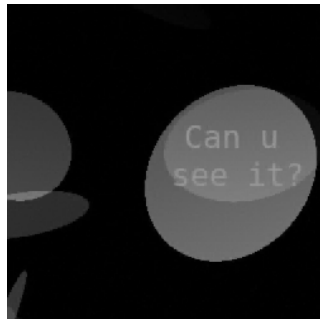
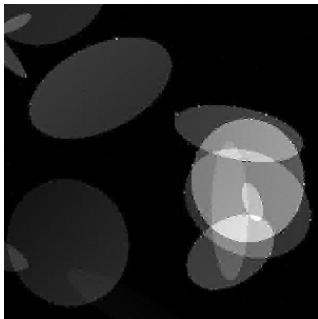
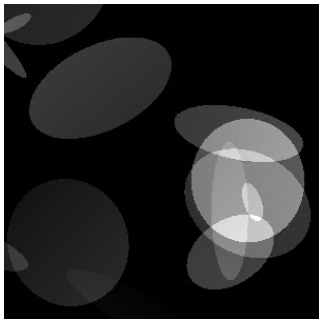
Rec. of detail



# FIRENET

Orig. + worst-case noise Rec. from worst-case noise

Rec. of detail





## Concluding remarks

There is a **need for foundations** in AI/deep learning. Our results:

- ▶ There are well-conditioned problems where mappings from training data to suitable NNs exist, but no training algorithm (even randomised) can approximate them.
- ▶ Existence of algorithms depends on desired accuracy.  $\forall K \in \mathbb{Z}_{\geq 3}, \exists$  well-conditioned problems where simultaneously:
  - (i) Algorithms may compute NNs to  $K - 1$  digits of accuracy, but not  $K$ .
  - (ii) Achieving  $K - 1$  digits of accuracy requires arbitrarily many training data.
  - (iii) Achieving  $K - 2$  correct digits requires only one training datum.
- ▶ Under specific conditions, there are algorithms that compute stable NNs. E.g., Fast Iterative REstarted NETworks (FIRENETs) converge exponentially in the number of hidden layers. We prove FIRENETs withstand adversarial attacks.
- ▶ There is a trade-off between stability and accuracy in deep learning.

**Question:** How do we optimally traverse the stability & accuracy trade-off? FIRENETs provide a balance but are likely not the end of the story.

Hopefully this talk has inspired you to build on these results and take up the challenge!