CREATIS

Lyon, Avril 17– 21

DLMI2023

UNIVERSITÉ DE LYON

(A short intro to) Deep learning for image reconstruction

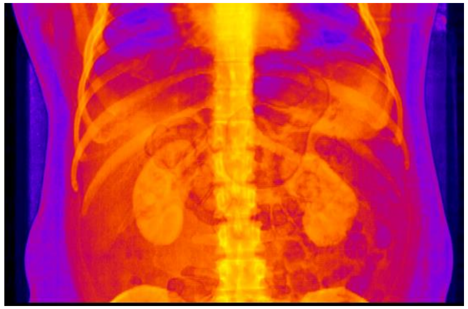
Nicolas Ducros[1, 2]

[1]CREATIS, Univ Lyon, INSA-Lyon, UCB Lyon 1,CNRS, Inserm, CREATIS UMR 5220, U1206, Lyon, France

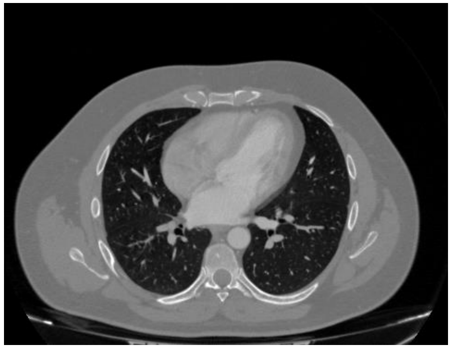
[2]IUF, Institut Universitaire de France

"Natural" vs Reconstructed images

Nat



Rec



Rec

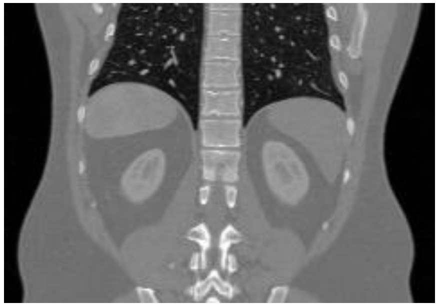
Nat

Nat

Nat

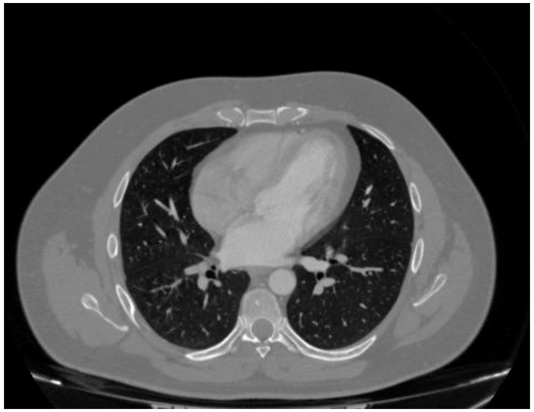
Nat

Rec



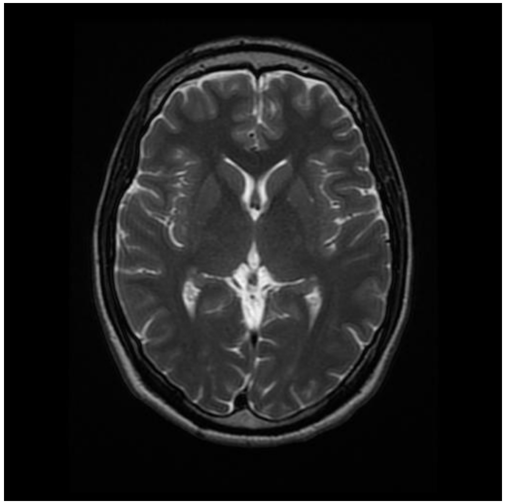
Reconstruction in Medical Imaging

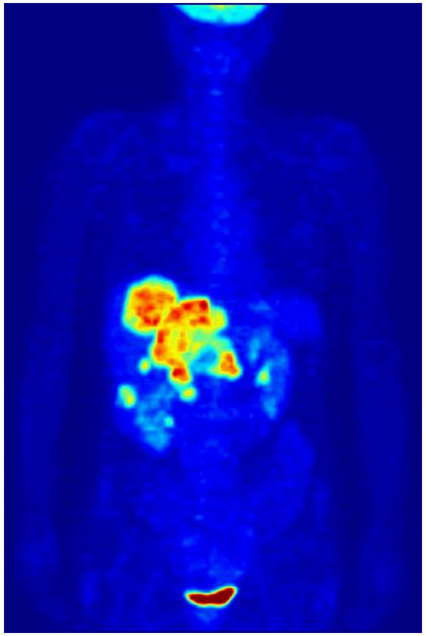
Computerized tomography (CT)



Ultrasound Imaging

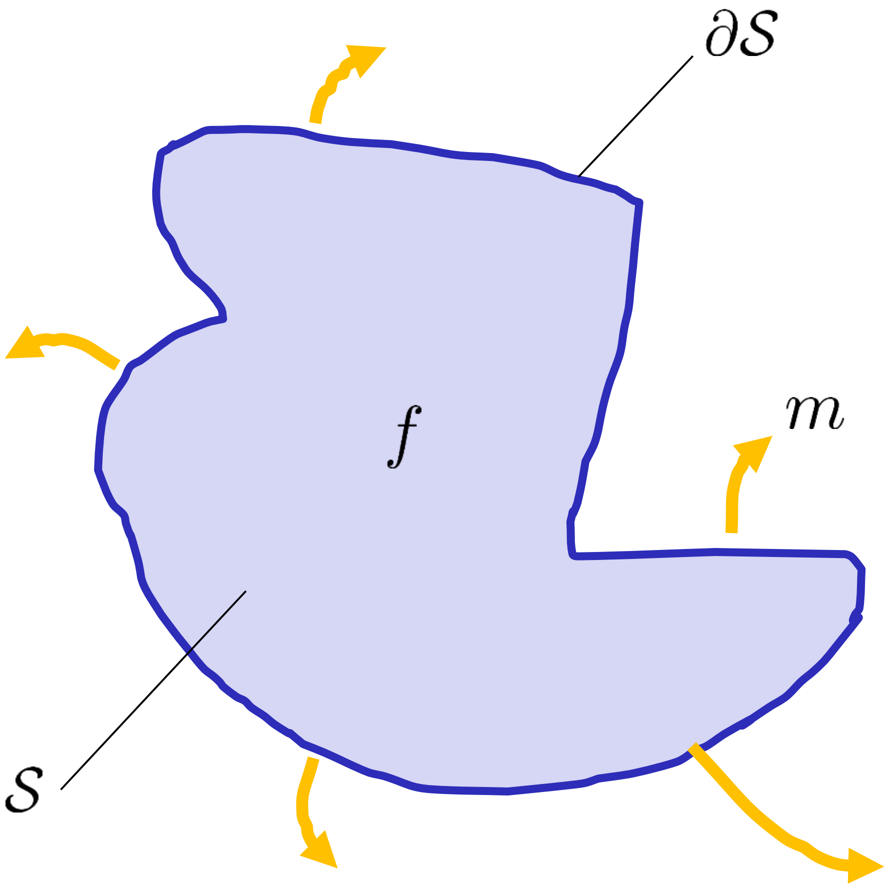
Magnetic Resonance (MRI)





Positron emission tomography (PET)

Inverse Problem



Inverse Problem

- **Internal unknowns from external measurements**

$$m = \mathcal{A}(f)$$

External Measurement

Forward Model

(Unknown) quantity of Interest

- **Most medical imaging problems are**
  - Linear
  - Corrupted by noise

- **In a discrete setting**

$$\boldsymbol{m} = \boldsymbol{A}\boldsymbol{f} + \boldsymbol{\eta}$$

Noise

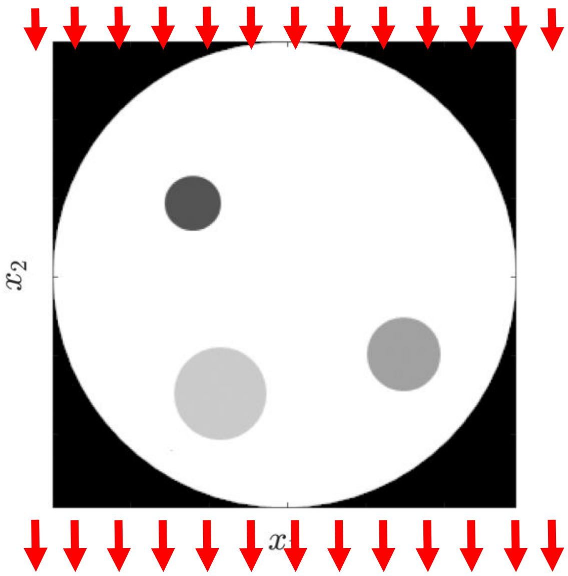
$\boldsymbol{m} \in \mathbb{R}^M$

$\boldsymbol{A} \in \mathbb{R}^{M \times N}$

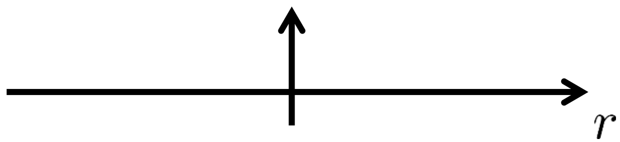
$\boldsymbol{f} \in \mathbb{R}^N$

Computed Tomography (CT)

CT slice (unknown)

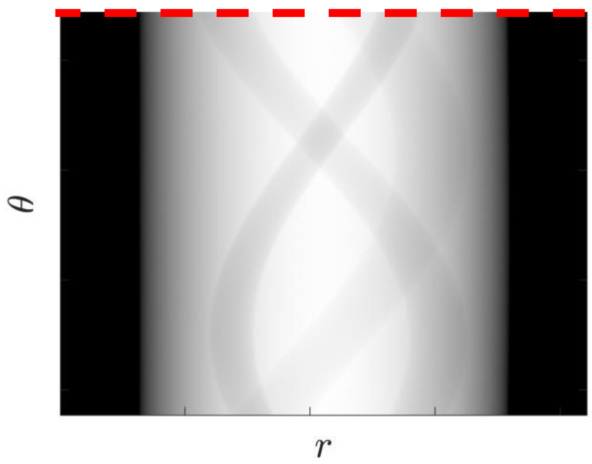


$$m(\theta, r) = \mathcal{R}[f](\theta, r)$$
$$= \int_{\mathcal{L}(\theta,r)} f(x_1, x_2)\,\mathrm{d}\ell$$



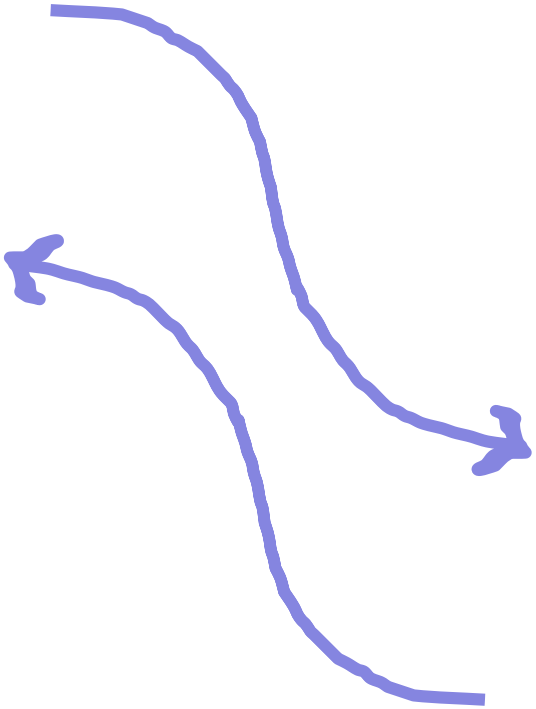
$\theta = 0$

Sinogram (measured)

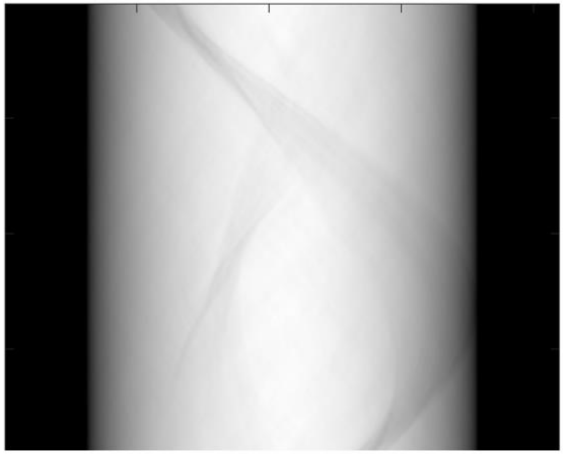


Computed Tomography (CT)

CT slice (unknown)



Sinogram (measured)



Different Options

- **1. Inversion "by hand"**
  - Model the forward and try to invert algebraically

$$\texttt{derive } \mathcal{H} \texttt{ such that } f = \mathcal{H}(m)$$

- **2. Optimization of handcrafted functionals**
  - Build cost function from prior knowledge about the solution/measurements
  - Minimize the cost function

$$\texttt{find and minimize } \mathcal{C} \texttt{ such that } \mathcal{C}(f; m) \texttt{ is small}$$

- **3. "Learn" to reconstruct**
  - (Probably what you expect from this talk)

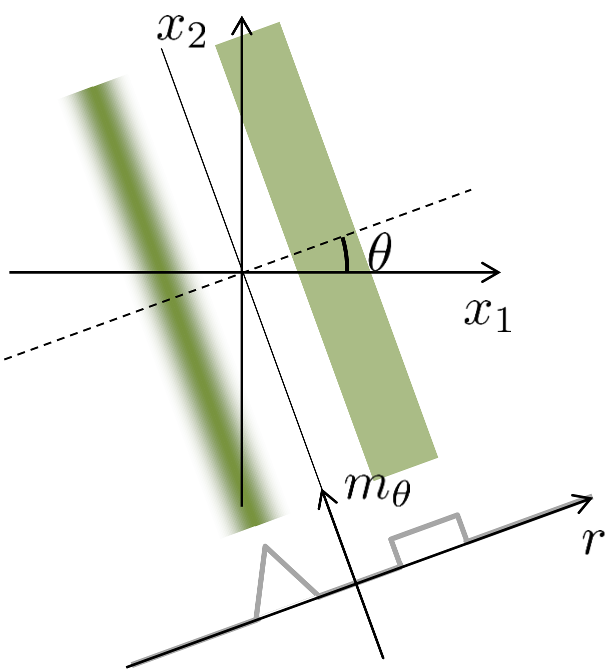
$$\texttt{learn } \mathcal{H}_\omega \texttt{ such that } f = \mathcal{H}_\omega(m)$$

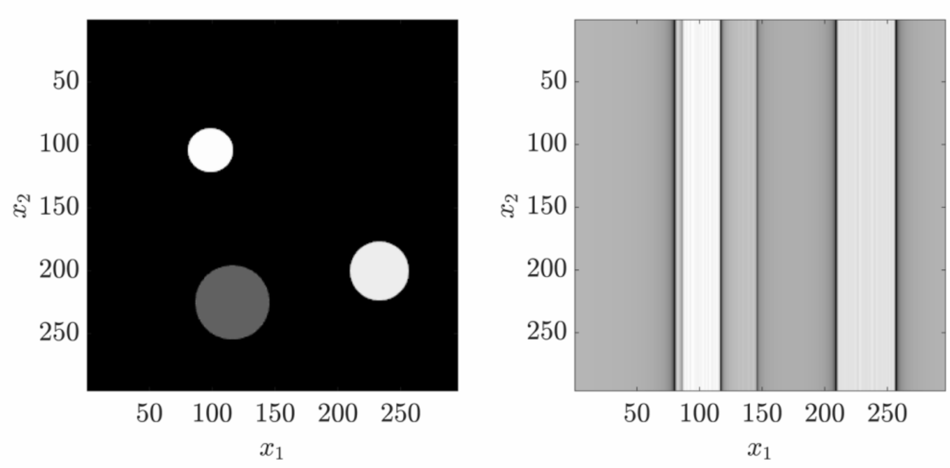
Analytical inversion (option #1)

- **Example with CT (filtered backprojection)**

$$f(x_1, x_2) = \int_0^{\pi} m_\theta^{\text{filt}}(x_1 \cos\theta + x_2 \sin\theta)\, \mathrm{d}\theta$$

$$\hat{m}_\theta^{\text{filt}}(\xi) = |\xi|\, \hat{m}_\theta(\xi)$$





Analytical inversion (option #1)

- **Pros**
  - Elegant
  - Theoretical guarantees
  - Usually fast implementation

- **Cons**
  - Not always possible to derive a solution
  - Influence of noise?
  - What if only few measurements are available?
    - For dose reduction/short scans
    - Short scans are less prone to motion artefacts

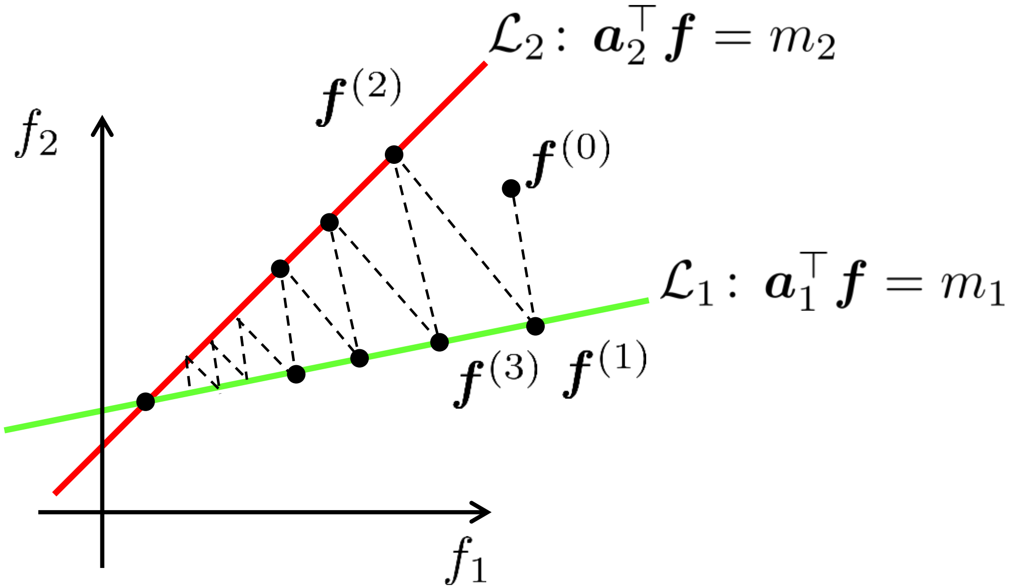
Optimization-based inversion (option #2)

- **Look for an image with small residuals**

$$\boldsymbol{r} = \boldsymbol{m} - \boldsymbol{A}\boldsymbol{f} \approx \boldsymbol{0}$$

- A simple example:

$$\begin{bmatrix} m_1 \\ m_2 \end{bmatrix} = \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}$$



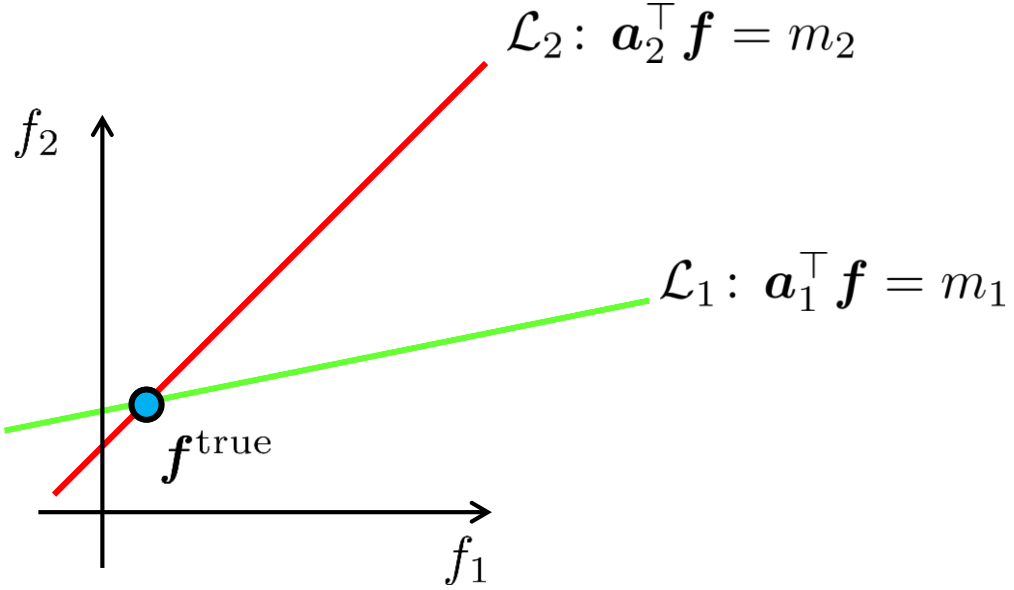
Algebraic reconstruction technique (**ART**) [Gordon R., 1970]

Optimization-based inversion (option #2)

- **Look for an image with small residuals**

$$\boldsymbol{r} = \boldsymbol{m} - \boldsymbol{A}\boldsymbol{f} \approx \boldsymbol{0}$$

- **Influence of noise**

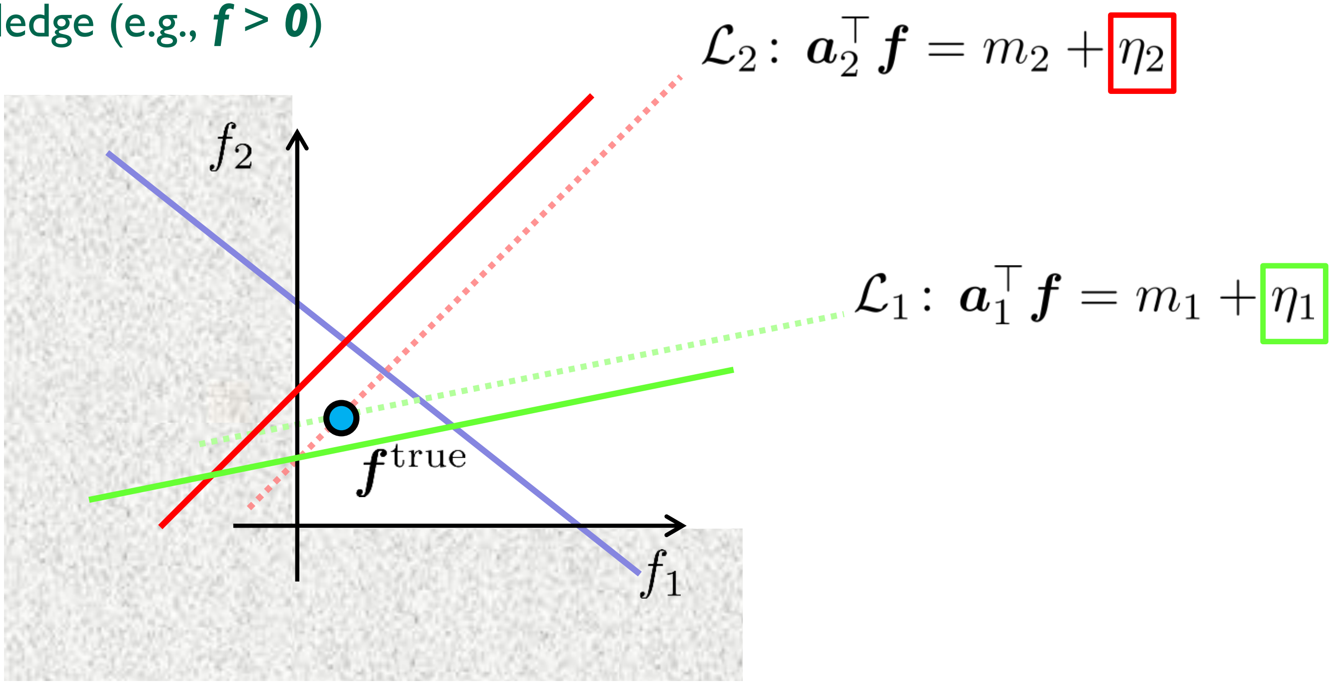


Optimization-based inversion (option #2)

- **Look for an image with small residuals**

$$\boldsymbol{r} = \boldsymbol{m} - \boldsymbol{A}\boldsymbol{f} \approx \boldsymbol{0}$$

- **Influence of noise**
  - More measurements (i.e., *M* > *N*)
  - Prior knowledge (e.g., ***f*** > ***0***)



Optimization-based inversion (option #2)

Typical cost functions

$$\min_{\boldsymbol{f}} \; \mathcal{D}(\boldsymbol{f}, \boldsymbol{m}) + \lambda \mathcal{R}(\boldsymbol{f})$$

Data fidelity — Regularization (prior)

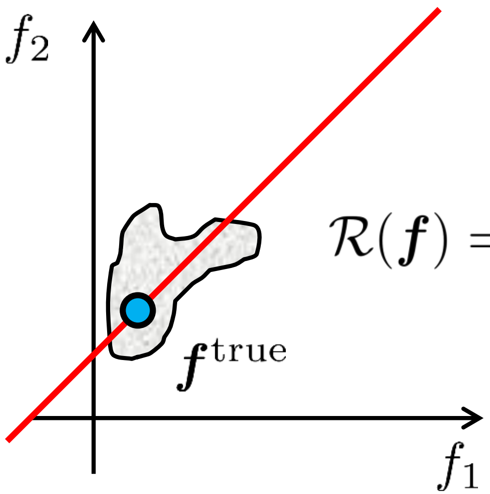
- Data fidelity is related the noise model/measurements confidence. E.g.

$$\mathcal{D}(\boldsymbol{f}; \boldsymbol{m}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|_{\boldsymbol{W}}^2$$

$$\mathcal{D}(\boldsymbol{f}; \boldsymbol{m}) = \mathrm{KL}(\boldsymbol{m}, \boldsymbol{A}\boldsymbol{f})$$

- Regularization convey prior knowledge about the solution.

$$M \ll N$$



$$\mathcal{R}(\boldsymbol{f}) = \begin{cases} 0 \text{ if } \boldsymbol{f} \in \mathcal{X} \\ \infty \text{ otherwise} \end{cases}$$

Optimization-based inversion (option #2)

Typical regularizers

- Quadratic / Tikhonov regularization

$$\mathcal{R}(\boldsymbol{f}) = \|\boldsymbol{f}\|_2^2$$

… leads to

$$\boldsymbol{f}^* = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top + \lambda \boldsymbol{I}_M)^{-1}\boldsymbol{m}$$

- Parsimony-promoting

$$\mathcal{R}(\boldsymbol{f}) = \|\boldsymbol{\Psi}\boldsymbol{f}\|_1$$

… requires iterative algorithms

$$\boldsymbol{z}^{(k)} = \boldsymbol{f}^{(k-1)} - \eta \boxed{\boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{f}^{(k-1)} - \boldsymbol{m})}$$

$$\boldsymbol{f}^{(k)} = \boxed{\mathtt{prox}_{\lambda\mathcal{R}}}(\boldsymbol{z}^{(k)})$$

gradient of data fidelity

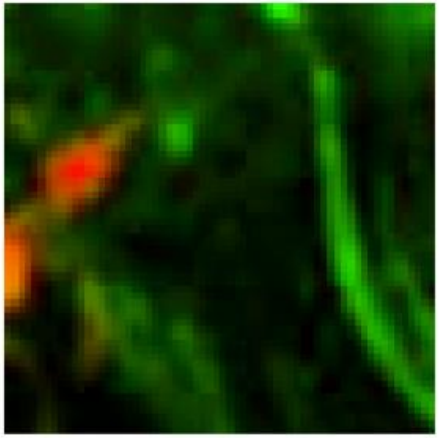
proximal operator of regularizer

Optimization-based inversion (option #2)

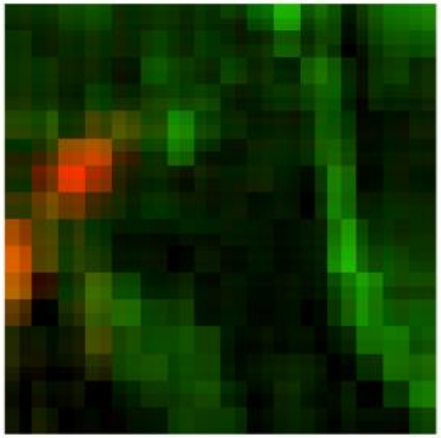
➢ **Illustrative results**

N = 64 × 64 image
M = 333 measurements
N / *M* ≈ 8%

Ground-Truth



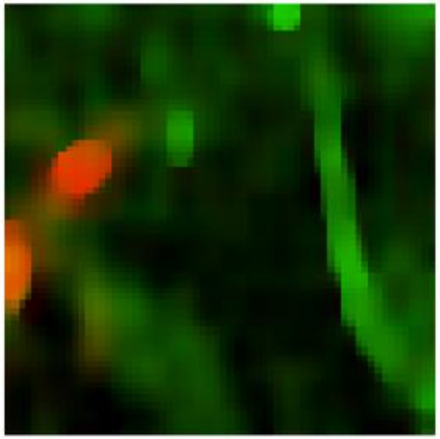
$\mathcal{R}(\boldsymbol{f}) = \|\boldsymbol{f}\|_2^2$



++ Analytical solution (fast computation)

- - Image quality

$\mathcal{R}(\boldsymbol{f}) = \|\nabla \boldsymbol{f}\|_1$



- - Iterative algorithms (time consuming)

++ Image quality

Learning Approaches (option #3)

- **Our dream is to find**

$$\mathcal{H}^* : \mathbb{R}^M \mapsto \mathbb{R}^N \text{ such that } \mathcal{H}^*(\boldsymbol{m}) = \boldsymbol{f}^{\text{true}}$$

… able to reconstruct well any image, i.e., something like

$$\mathcal{H}^* \in \arg\min_{\mathcal{H}} \frac{1}{L} \sum_{\ell} \|\mathcal{H}(\boldsymbol{m}^{\ell}) - \boldsymbol{f}^{\ell}\|_2^2$$

Minimum mean square error (MMSE) estimator

… A scary problem

- **We have to reduce the dimension of the solution space**
  - E.g.,

$$\mathcal{H}(\boldsymbol{m}) = \boldsymbol{W}\boldsymbol{m} + \boldsymbol{b},$$

Linear MMSE estimator

Learning Approaches (option #3)

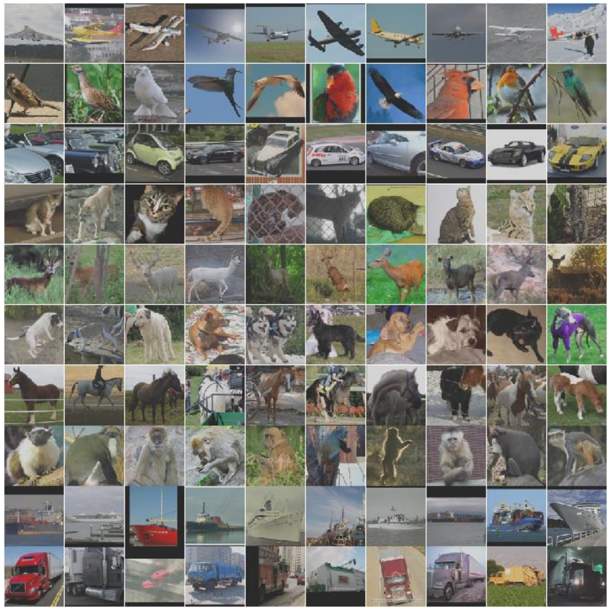
- **Linear MMSE**

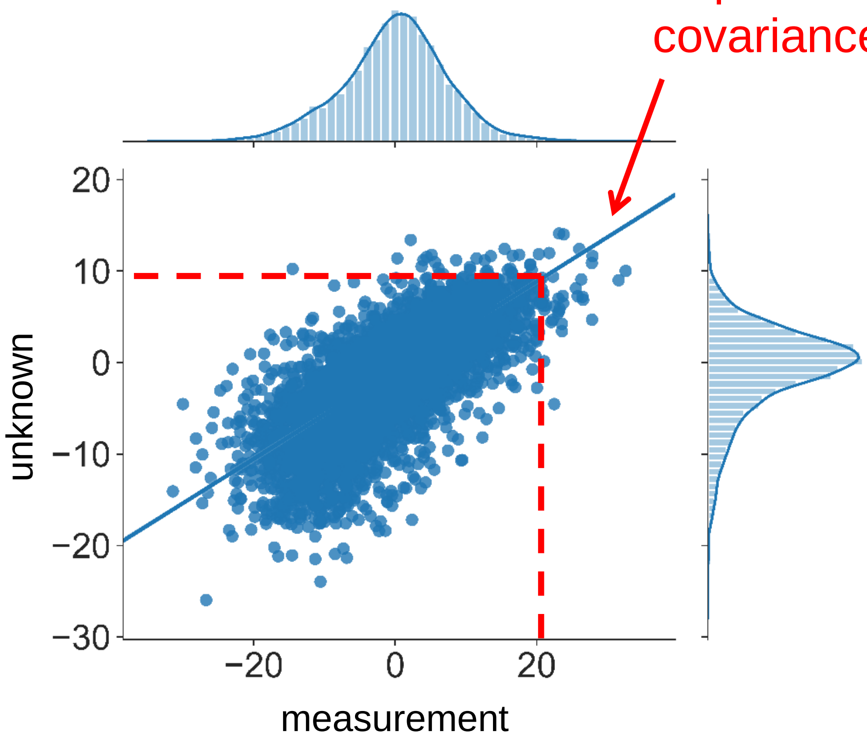
$$\boldsymbol{f}^* = \bar{\boldsymbol{f}} + \boldsymbol{\Sigma}_{f,m}\boldsymbol{\Sigma}_m^{-1}(\boldsymbol{m} - \bar{\boldsymbol{m}})$$

Covariance between measurements and unknowns

Covariance of measurements

slope → covariance





Learning Approaches (option #3)

- **Learning approaches *only* reduce the dimension of the solution space to a family of *non linear* mappings**

$$\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \frac{1}{L} \sum_{\ell} \|\mathcal{H}(\boldsymbol{\theta}; \boldsymbol{m}^{\ell}) - \boldsymbol{f}^{\ell}\|_2^2$$

- Training phase
  - Image-measurement pairs $\{\boldsymbol{f}^{(\ell)}; \boldsymbol{m}^{(\ell)})\}_{1 \leq \ell \leq L}$
  - Loss (e.g., mse)
  - Optimization machinery (i.e., through PyTorch/TensorFlow)

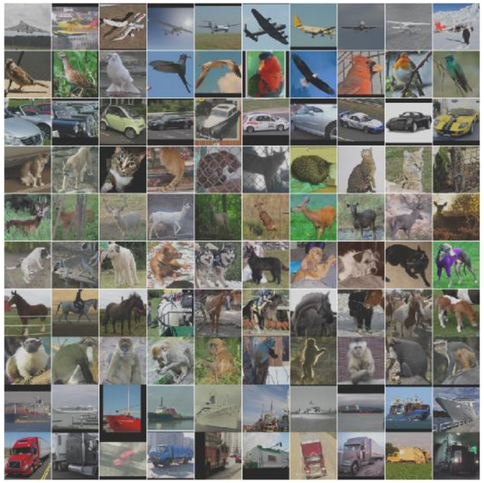
D.P. Kingma and J.L Ba, ICRL, 2015 (> 215k citations)

A. Paszke *et al.*, NEURIPS, 2019 (> 22k citations)

- Reconstruction phase

$$\boldsymbol{f}^* = \mathcal{H}_{\boldsymbol{\theta}^*}(\boldsymbol{m})$$

STL-10 dataset



Learning Approaches (option #3)

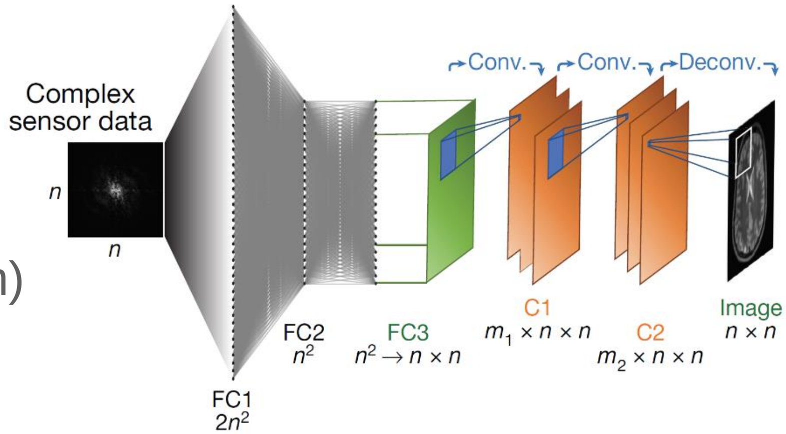
➢ Pros

- Reconstruction performance
  - Empirically excellent (i.e., almost always outperform optimization-based approaches)
- Computation times
  - Training phase is slow, i.e., several hours or days
  - Inference is fast, i.e., tens or hundreds of milliseconds

➢ Cons

- No reconstruction guarantees (*mathematicians don't like it*)
- Black box (*radiologists don't like it*)
- Practical issues
  - How to choose the model?

"Automap"
(> 6.10^9 param)



[B. Zhu et al., Nature Letters, 2018] (> 1.5k citations)

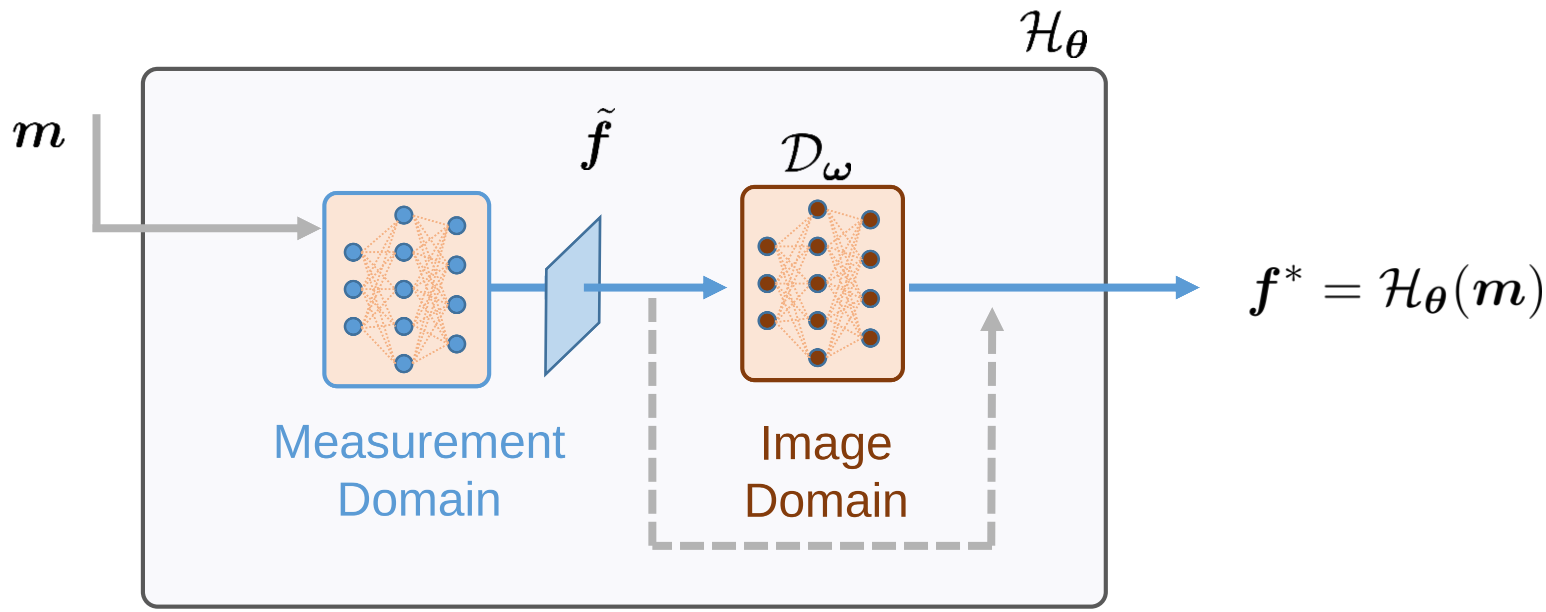
Learning Approaches (option #3)

- **Direct methods**

$$\tilde{\boldsymbol{f}} = \tilde{\boldsymbol{A}}^{-1}\boldsymbol{m}$$

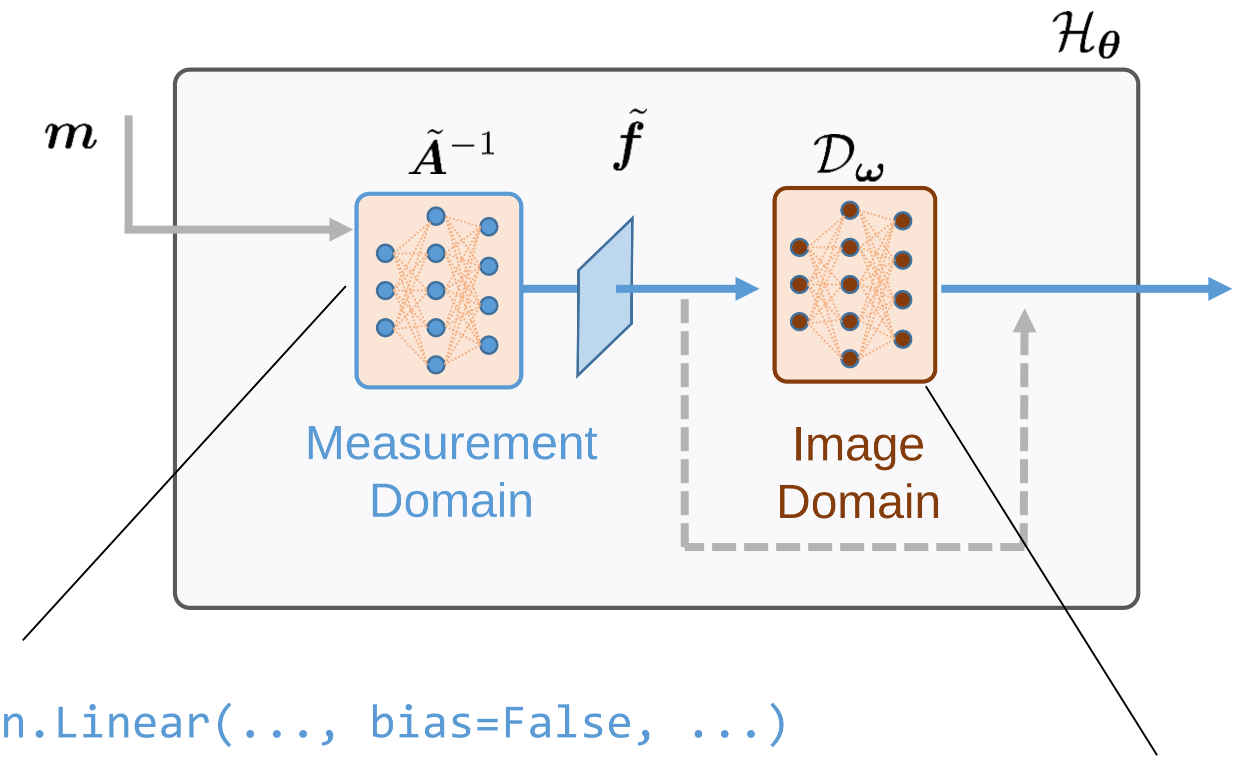
$$\boldsymbol{f}^* = \mathcal{D}_{\boldsymbol{\omega}}(\tilde{\boldsymbol{f}}) + \tilde{\boldsymbol{f}}$$

where $\tilde{\boldsymbol{A}^{-1}}$ is an approximate inverse of the forward, i.e., $\tilde{\boldsymbol{A}}^{-1}\boldsymbol{A}\boldsymbol{f} \approx \boldsymbol{f}$



Learning Approaches (option #3)

➢ **Direct methods with frozen layers**



network parameters

$$\boldsymbol{f}^* = \mathcal{H}_{\boldsymbol{\theta}}(\boldsymbol{m})$$

E.g., $\boldsymbol{\theta} = \boldsymbol{\omega}$.

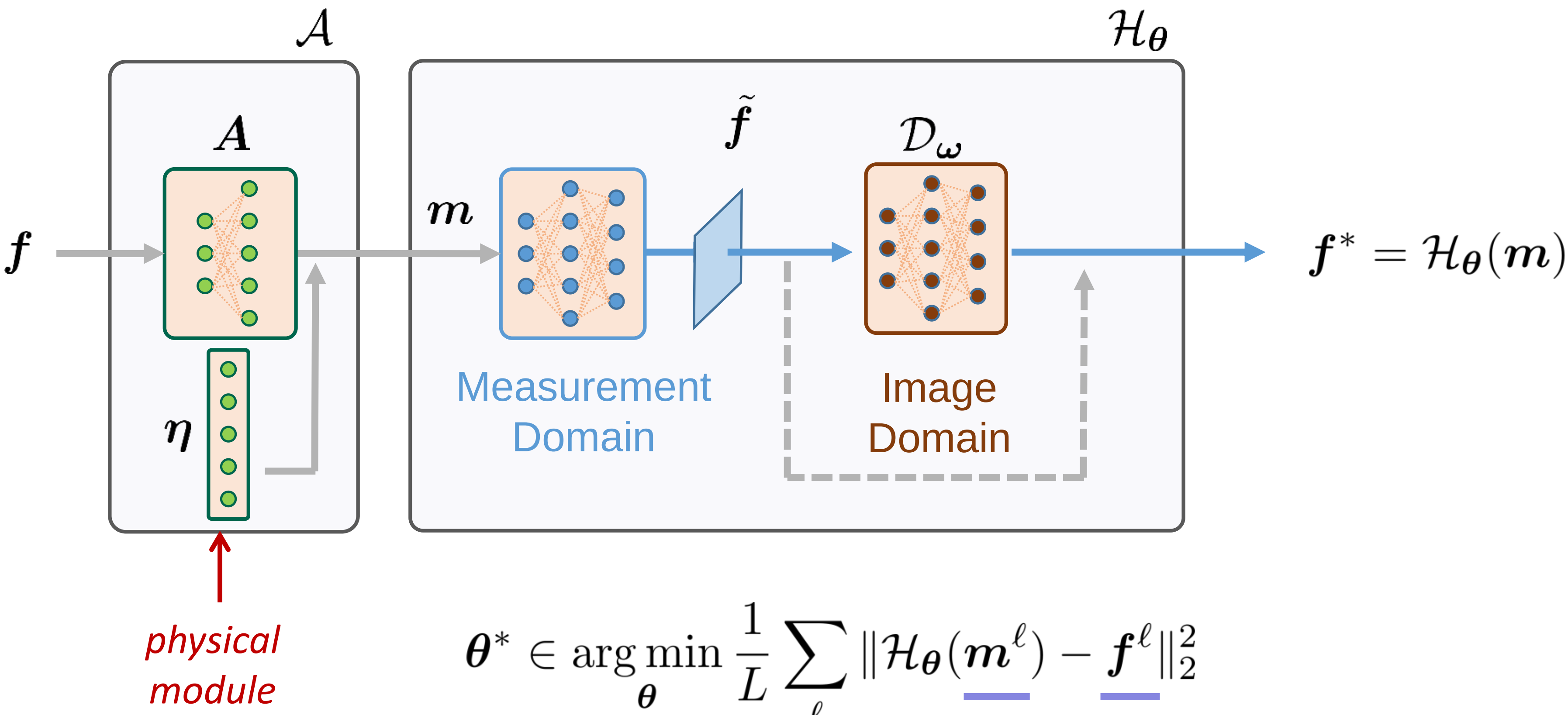
```
Atilde = nn.Linear(..., bias=False, ...)
Atilde.weight.requires_grad = False
```

$$\tilde{\boldsymbol{A}}^{-1} = \boldsymbol{A}^\top$$
$$\tilde{\boldsymbol{A}}^{-1} = \boldsymbol{A}^\dagger = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1}$$
$$\tilde{\boldsymbol{A}}^{-1} = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top + \boldsymbol{I})^{-1}$$

```
D = nn.Module(...)
requires_grad = True
```

Learning Approaches (option #3)

- **Direct methods with a physical module (no need for meas/image pairs)**



$$\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \frac{1}{L} \sum_{\ell} \|\mathcal{H}_{\boldsymbol{\theta}}(\boldsymbol{m}^{\ell}) - \boldsymbol{f}^{\ell}\|_2^2$$

$$\mathcal{A}(\boldsymbol{f}) = \boldsymbol{A}\boldsymbol{f} + \boldsymbol{\eta}$$

$$\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \frac{1}{L} \sum_{\ell} \|(\mathcal{H}_{\boldsymbol{\theta}} \circ \mathcal{A})(\boldsymbol{f}^{\ell}) - \boldsymbol{f}^{\ell}\|_2^2$$

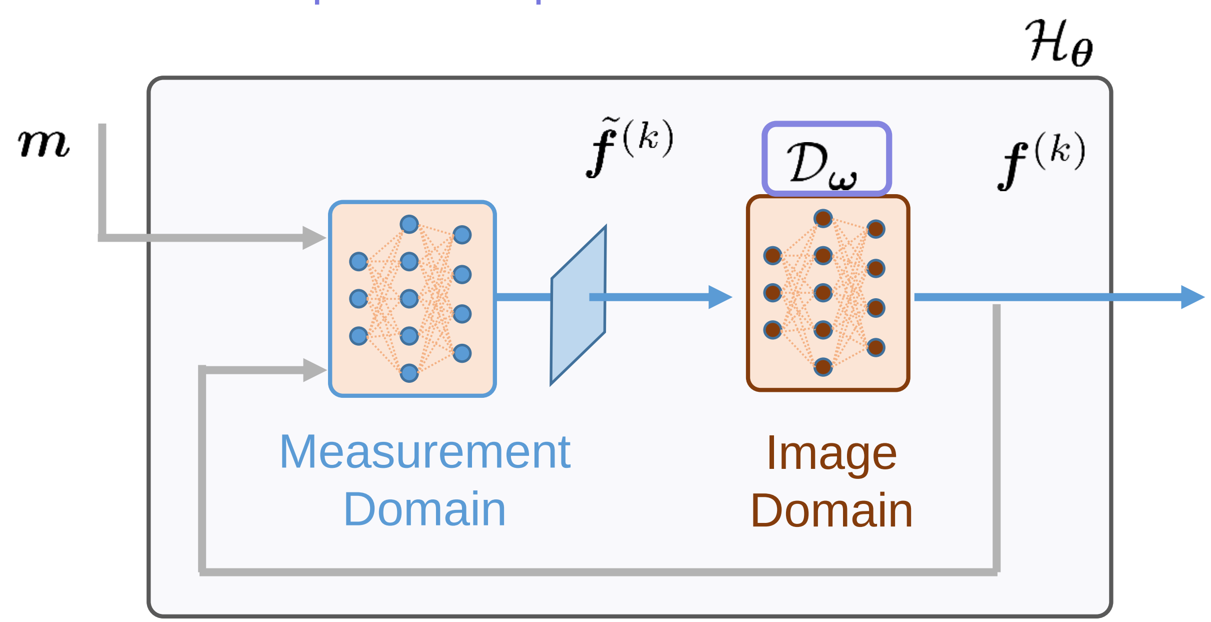
Learning Approaches (option #3)

➢ Unrolled / Plug&Play methods

$$\boldsymbol{z}^{(k)} = \boldsymbol{f}^{(k-1)} - \eta \boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{f}^{(k-1)} - \boldsymbol{m})$$
$$\boldsymbol{f}^{(k)} = \mathtt{prox}_{\lambda\mathcal{R}}(\boldsymbol{z}^{(k)})$$

data fidelity

proximal operator



network parameters

$$\boldsymbol{f}^* = \mathcal{H}_{\boldsymbol{\theta}}(\boldsymbol{m}) = \boldsymbol{f}^{(K)}$$

Parameters can be shared across iterations or not

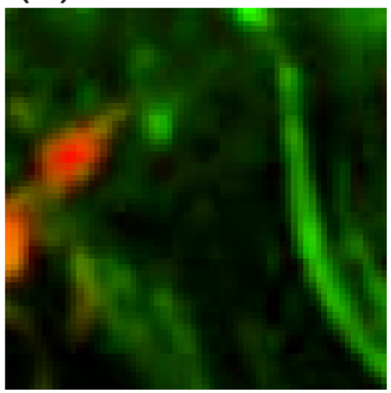
E.g., $\boldsymbol{\theta} = \boldsymbol{\omega}$,
or $\boldsymbol{\theta} = [\boldsymbol{\omega}^{(1)}, \ldots, \boldsymbol{\omega}^{(K)}]$.

Learning Approaches (option #3)

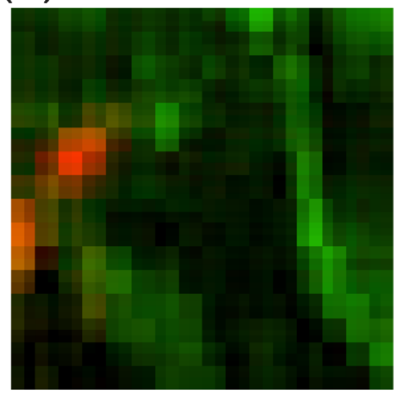
➢ **Illustrative results (option #2 vs option #3)**

N = 64 × 64 image
M = 333 measurements
N / M ≈ 8%

(a) Ground-Truth



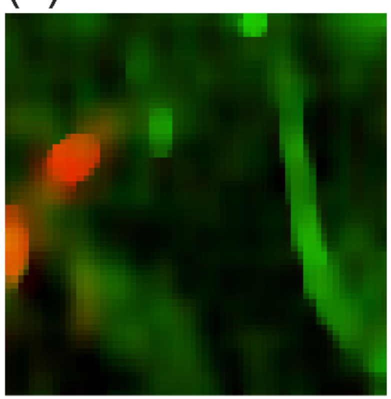
(b) Pseudo Inverse



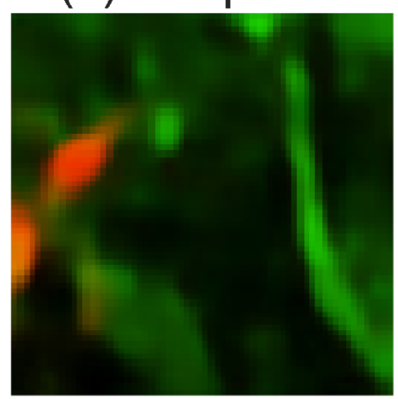
$$\mathcal{R}(\boldsymbol{f}) = \|\boldsymbol{f}\|_2^2$$

(c) Total Variation

$$\mathcal{R}(\boldsymbol{f}) = \|\nabla \boldsymbol{f}\|_1$$



(d) compNET

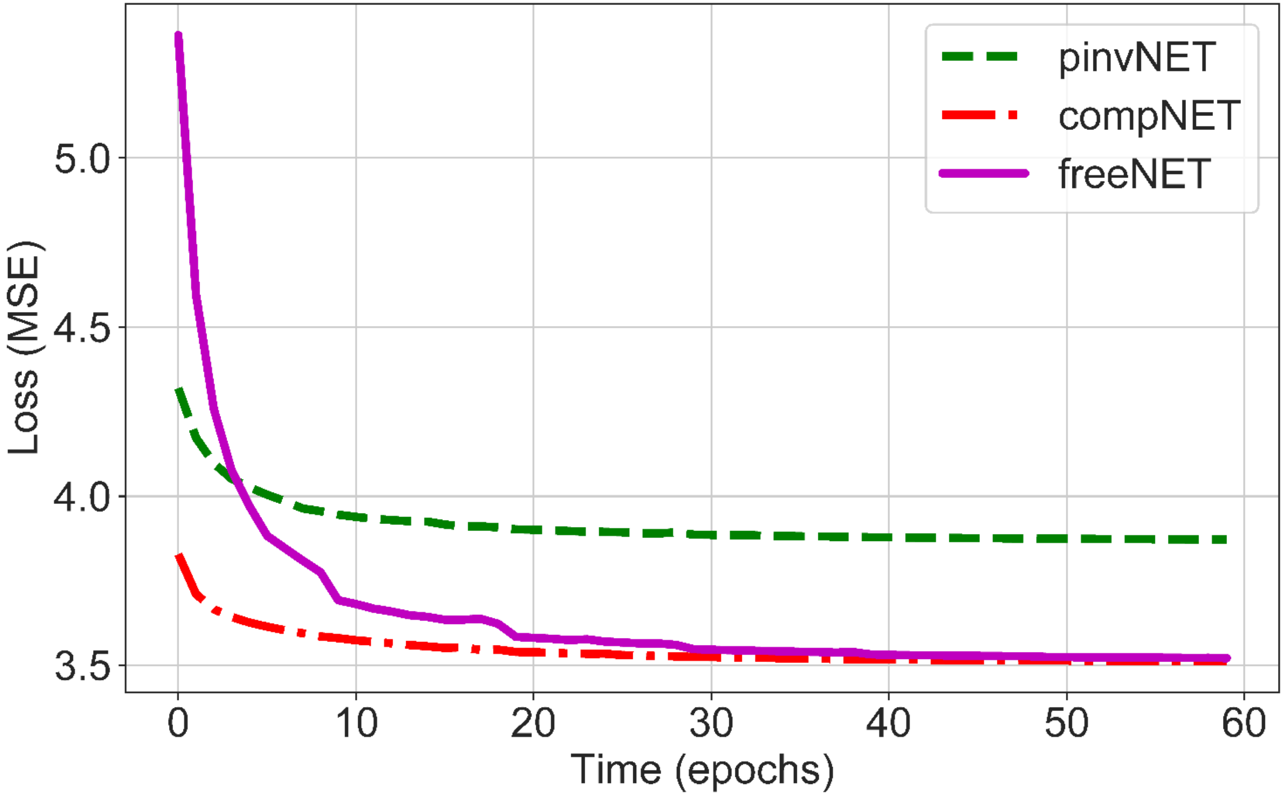


$$\boldsymbol{f}^* = \mathrm{CNN}(\boldsymbol{A}_{\mathrm{mmse}}\boldsymbol{m})$$

Learning Approaches (option #3)

- **STL-10 (training: ~100k images; test: 8k images)**

$$\sum_{\ell \in \mathcal{I}_{\text{test}}} \|\boldsymbol{f}^{(\ell)} - \mathcal{H}_{\boldsymbol{\theta}}(\boldsymbol{m}^{(\ell)})\|^2$$



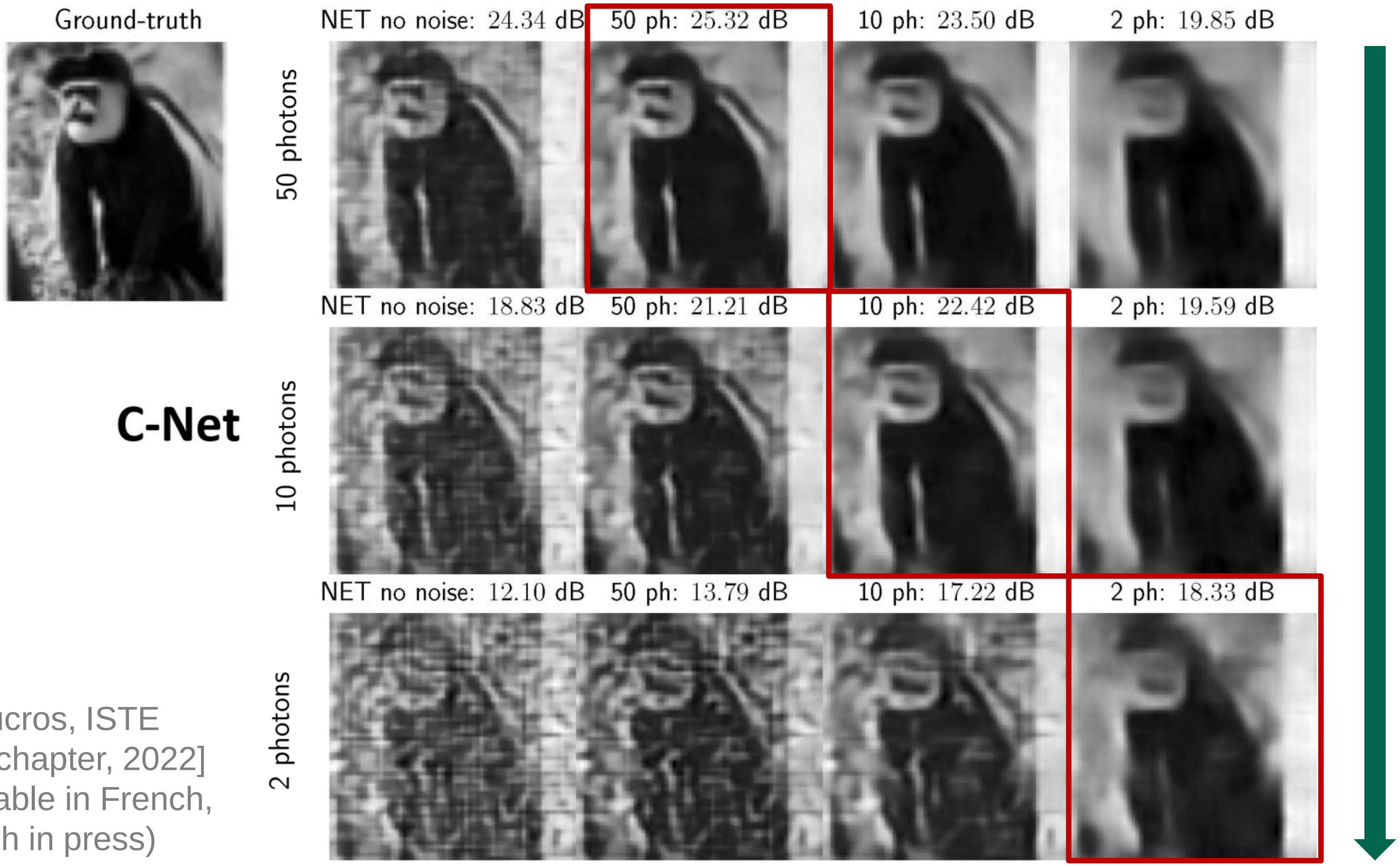
Learning Approaches (option #3)

- **Noise robustness**

Increasing training noise

Ground-truth

C-Net



Increasing test noise

[N. Ducros, ISTE Book chapter, 2022] (Available in French, English in press)

Conclusions

- **Data driven approaches for image reconstruction based on DL are**
  - Powerful!
  - No longer black boxes

- **Unrolled algorithms**
  - Usually require fewer parameters than their direct counterparts
  - More interpretable

- **Warning**
  - Noise is still an issue.
    - Train with noise
    - Evaluate the robustness to noise level deviations

Hands-on session at 2 pm!