Component separation with learnlets on the sphere

Victor Bonjean

with A. Gkogkou, J.-L. Starck and P. Tsakalides

Cosmostat days - February 12th 2026

Illustration

Component separation

$$X = S_0 + S_1 + S_2$$

Illustration

Component separation

Observation

$$X = S_0 + S_1 + S_2$$

Illustration

Component separation

Observation

$$X = S_0 + S_1 + S_2$$

3 components

Illustration

Component separation

Multiple frequencies

Observation

$$X_i = a_i . S_0 + b_i . S_1 + c_i . S_2$$

3 components

Illustration

Component separation

$$\begin{pmatrix} X_0 \\ X_1 \\ \dots \\ X_n \end{pmatrix} = \begin{pmatrix} a_0 & b_0 & c_0 \\ a_1 & b_1 & c_1 \\ \dots & \dots & \dots \\ a_n & b_n & c_n \end{pmatrix} \cdot S_0 \, S_1 \, S_2$$

n frequencies

Illustration

Component separation

Observations

nx3: « **Mixing matrix** »

$$\begin{Bmatrix} X_0 \\ X_1 \\ \dots \\ X_n \end{Bmatrix} = \begin{Bmatrix} a_0 & b_0 & c_0 \\ a_1 & b_1 & c_1 \\ \dots & \dots & \dots \\ a_n & b_n & c_n \end{Bmatrix} \cdot S_0 \, S_1 \, S_2$$

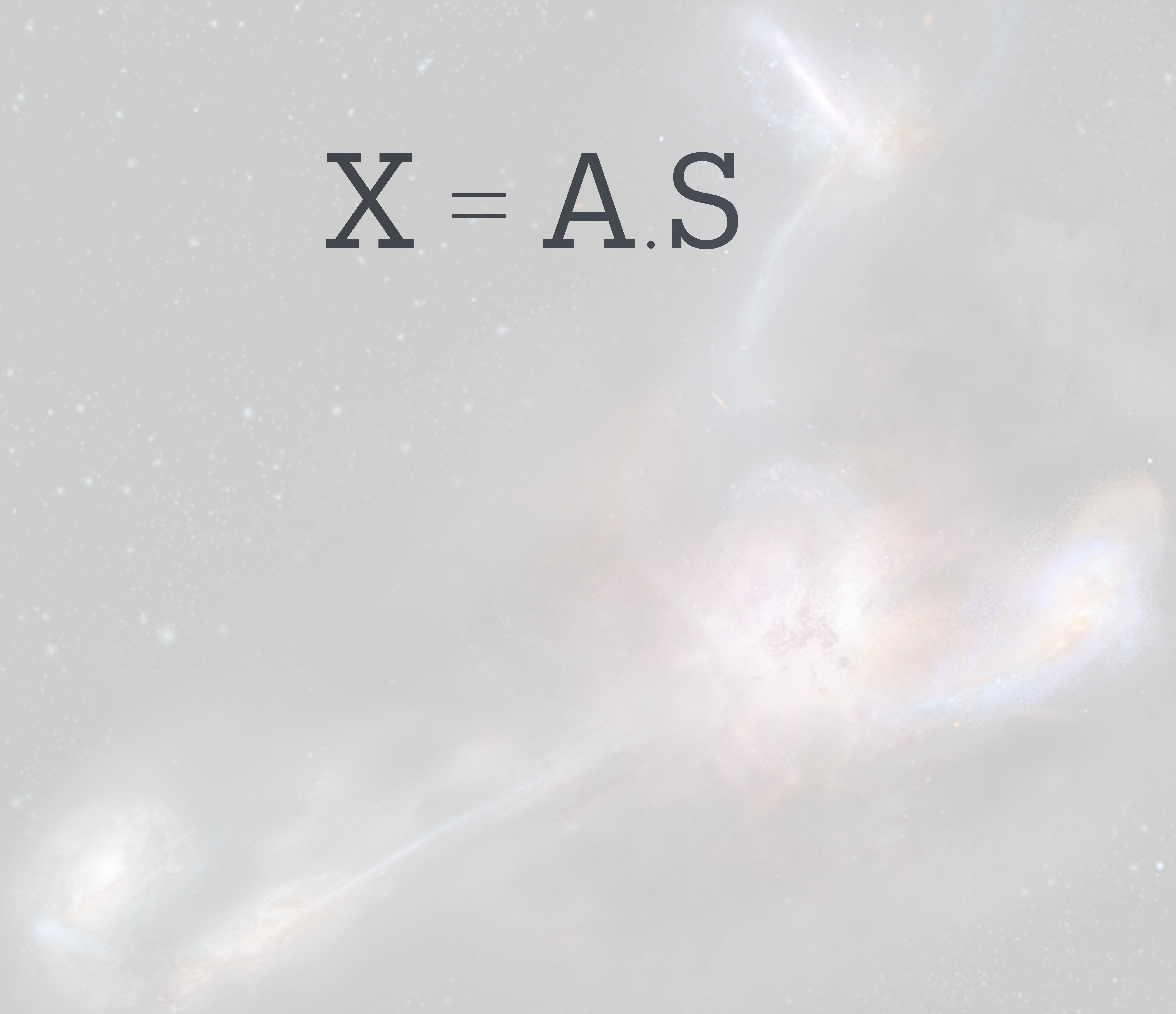
Components

n frequencies

Illustration

Component separation

$$X = A.S$$



Illustration

Component separation

$$\mathbf{X} = \boxed{\mathbf{A}}.\mathbf{S}$$

1. We know **A**

« No problem »

$$\mathbf{S} = \mathbf{A}^{-1}.\mathbf{X}$$

- Never the case

Illustration

Component separation

$$\mathbf{X} = \boxed{\mathbf{A}}.\mathbf{S} + \mathbf{N}$$

Blind Source Separation (BSS)

1. We know **A**

« No problem »

$\mathbf{S} = \mathbf{A}^{-1}.\mathbf{X}$

- Never the case

2. We don't know **A**

Problem

→ Prior on A or S

- Sparsity on S: GMCA (Bobin, Starck et al., 2007)

Illustration

Sparsity-based algorithms

BSS: $\mathbf{X} = \mathbf{A}.\mathbf{S} + \mathbf{N}$

Illustration

Sparsity-based algorithms

BSS: $\mathbf{X} = \mathbf{A}.\mathbf{S} + \mathbf{N}$

Two steps iterating algorithm
(i.e. GMCA):

1) $\min_{\mathbf{S}} ||\mathbf{Y} - \hat{\mathbf{A}} \cdot \mathbf{S}||_2^2 + \lambda||\mathcal{L}_{\mathbf{S}}||_{\mathrm{p}}$ → Impose sparsity on S

2) $\min_{\mathbf{A}} ||\mathbf{Y}^{\mathrm{no-coarse}} - \mathbf{A} \cdot \mathcal{L}_{\hat{\mathbf{S}}}{}^{\mathrm{no-coarse}}||_2^2$

Sparsity-based algorithms

BSS: $\mathbf{X} = \mathbf{A}.\mathbf{S} + \mathbf{N}$

Two steps iterating algorithm
(i.e. GMCA):

1) $\min_{\mathbf{S}} ||\mathbf{Y} - \hat{\mathbf{A}} \cdot \mathbf{S}||_2^2 + \lambda||\mathcal{L}_{\mathbf{S}}||_{\mathrm{p}}$ → Impose sparsity on S

2) $\min_{\mathbf{A}} ||\mathbf{Y}^{\text{no-coarse}} - \mathbf{A} \cdot \mathcal{L}_{\hat{\mathbf{S}}}{}^{\text{no-coarse}}||_2^2$

→ With any denoiser:

- Starlets (GMCA)
- UNet
- Transformers
- Starlets-transformers
- Learnlets

Sparsity-based algorithms

BSS: $\mathbf{X} = \mathbf{A}.\mathbf{S} + \mathbf{N}$

Two steps iterating algorithm
(i.e. GMCA):

1) $\min_{\mathbf{S}} ||\mathbf{Y} - \hat{\mathbf{A}} \cdot \mathbf{S}||_2^2 + \lambda||\mathcal{L}_{\mathbf{S}}||_{\mathrm{p}}$ → Impose sparsity on S

2) $\min_{\mathbf{A}} ||\mathbf{Y}^{\mathrm{no-coarse}} - \mathbf{A} \cdot \mathcal{L}_{\hat{\mathbf{S}}}{}^{\mathrm{no-coarse}}||_2^2$

→ With any denoiser:

- Starlets (GMCA)
- UNet
- Transformers
- Starlets-transformers
- **Learnlets**

$\mathscr{L}_{\mathbf{S}_i}$: Learnlets are trained priorly on each components i or global

What are learnlets?

(Ramzi et al., 2021, Bonjean et al., 2026)

An hybrid **denoiser** combining expressivity of **deep learning** and mathematical properties of **wavelets**

Illustration

What are learnlets?

(Ramzi et al., 2021, Bonjean et al., 2026)

An hybrid **denoiser** combining expressivity of **deep learning** and mathematical properties of **wavelets**

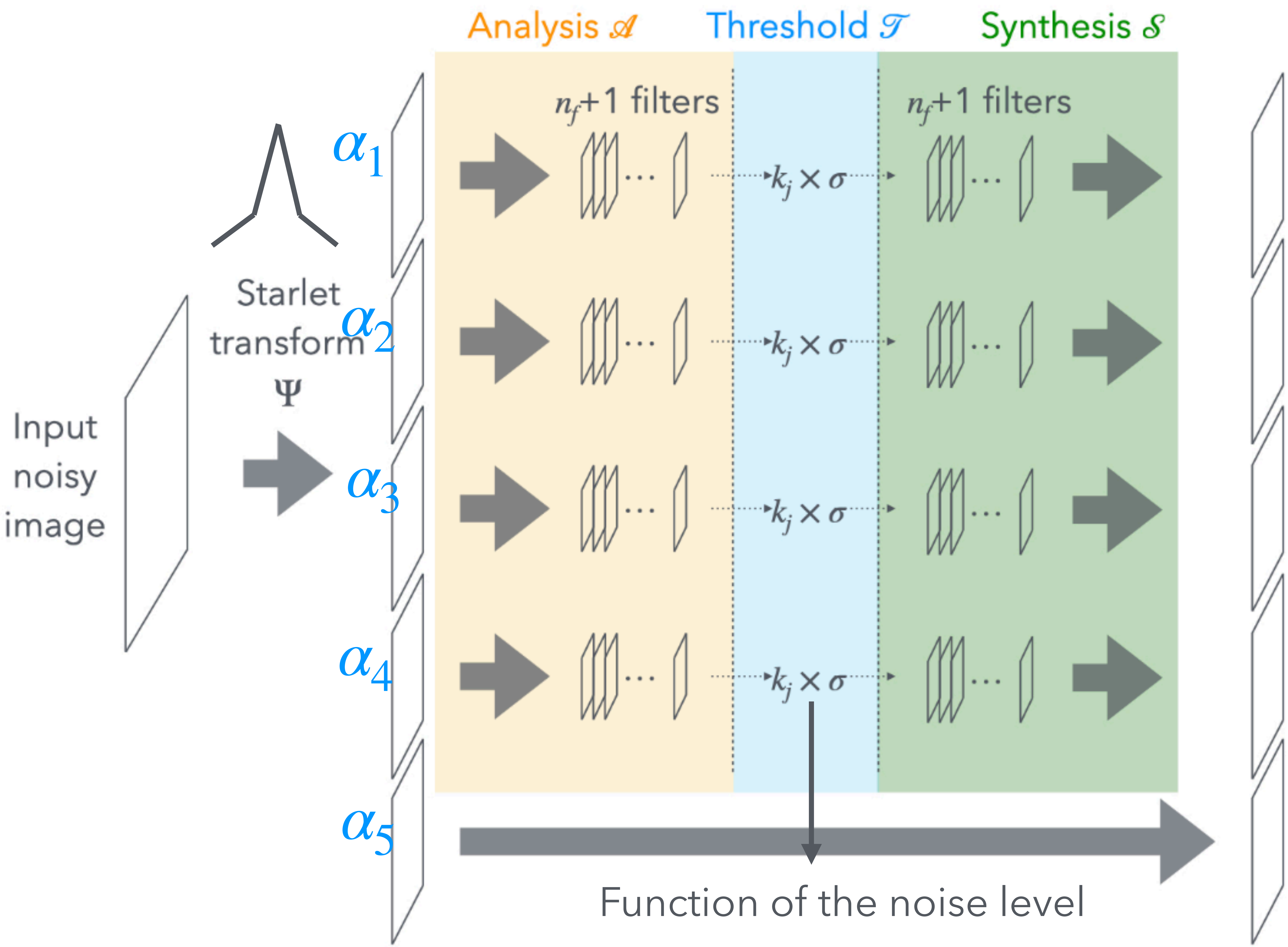
→ Extension of wavelets (sparse)

→ Filters are learned (CNN)

→ Mathematical frame (component separation)

Illustration

Learnlet network architecture: a denoiser

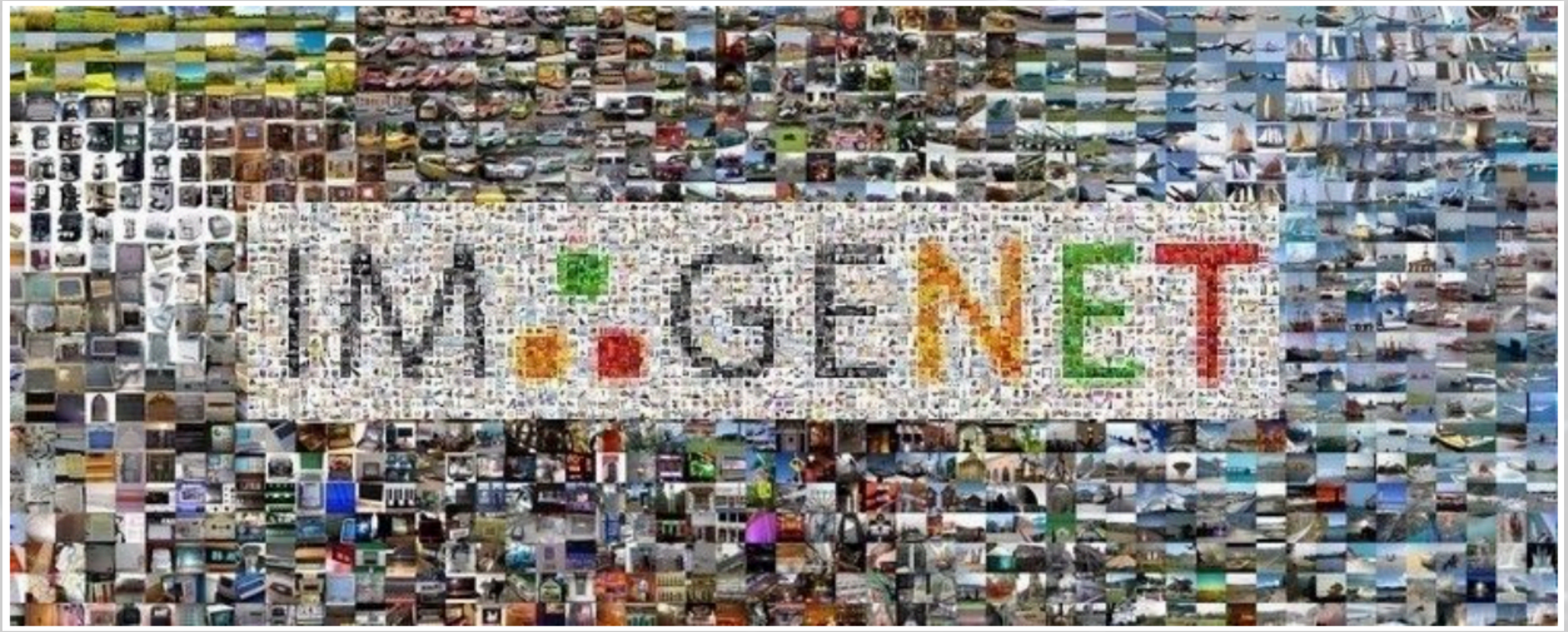


if σ=0: α_1

$$\hat{\mathbf{Y}} = \alpha_J + \sum_{j=1}^{J-1} \mathcal{S}(\mathcal{T}(\mathcal{A}(\alpha_j), k_j \times \sigma)).$$

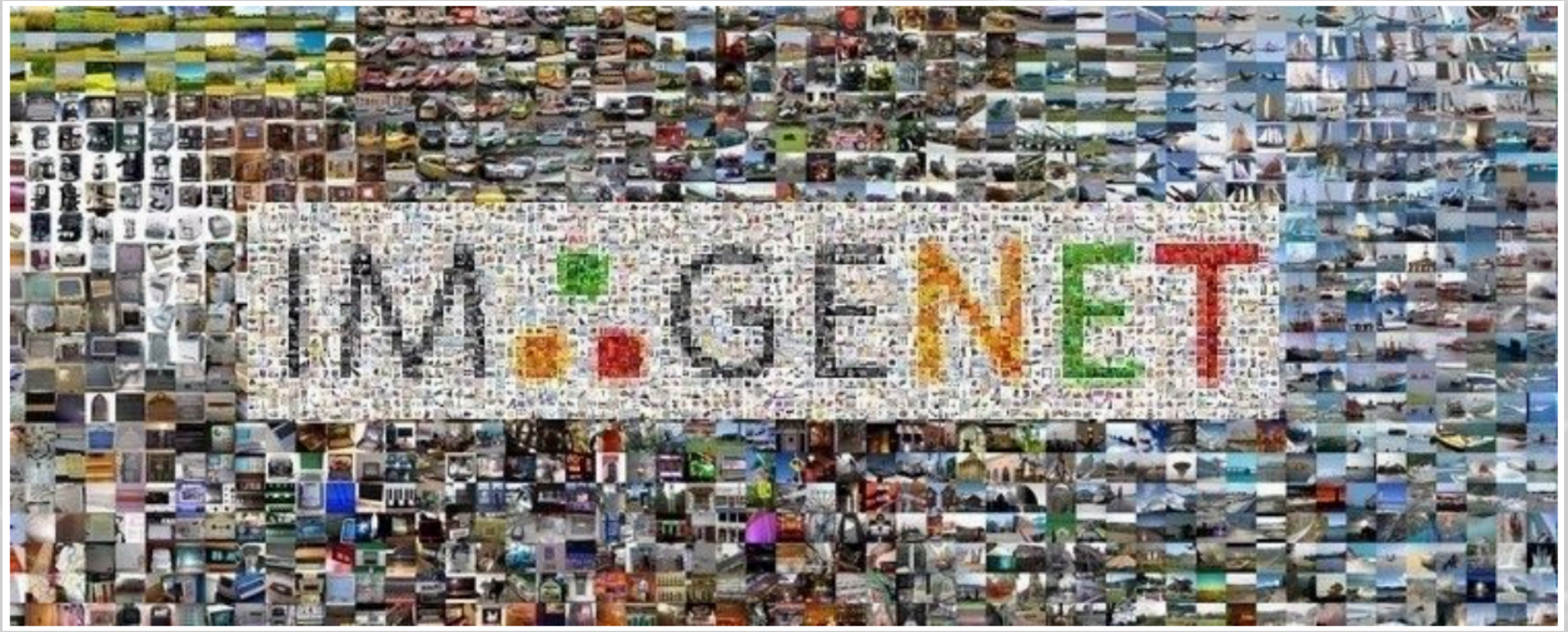
Bonjean et al., 2026

Training learnlets

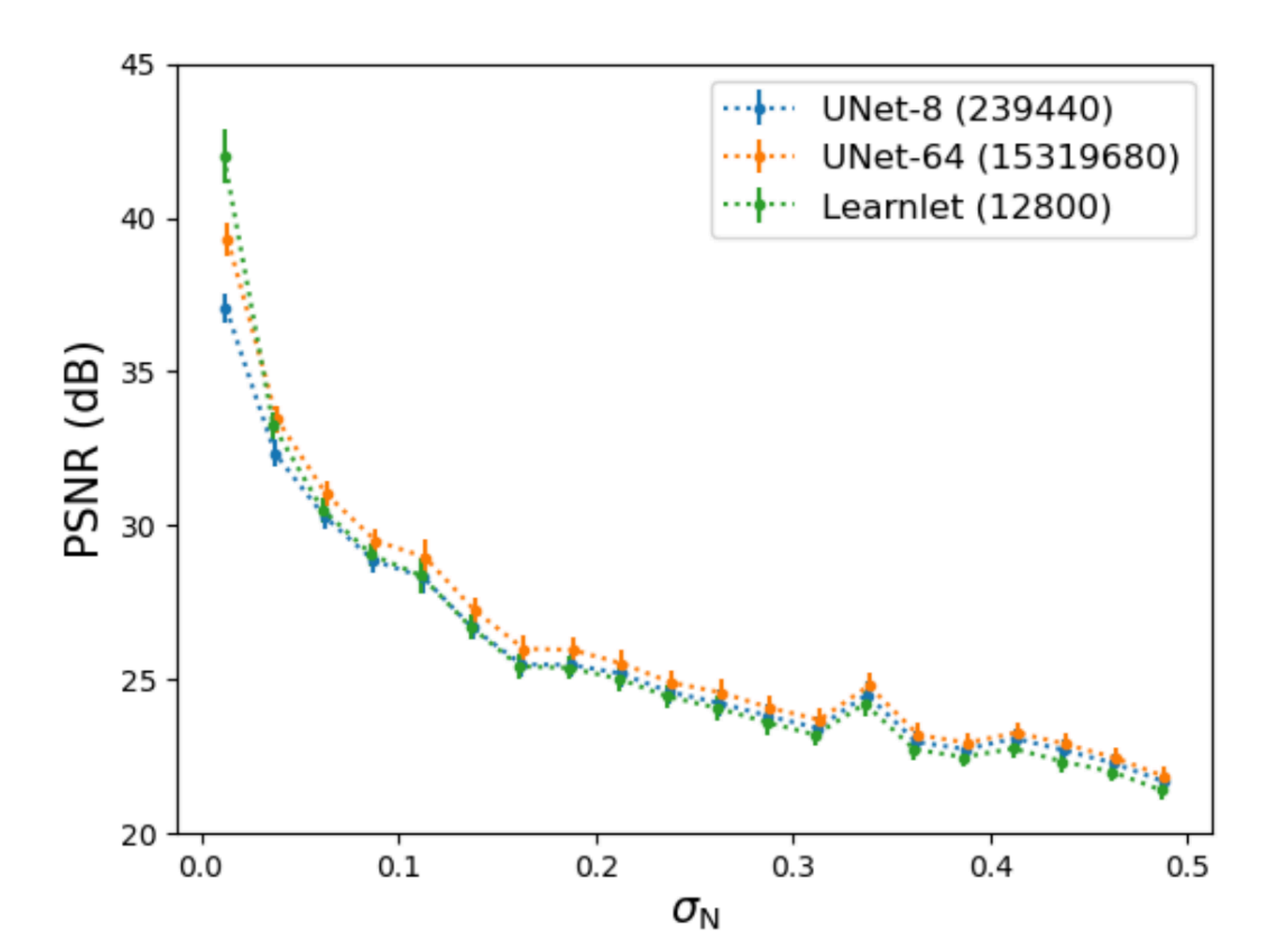


10.000 images from ImageNet:
8.000 training
1.000 validation
1.000 test

Training learnlets

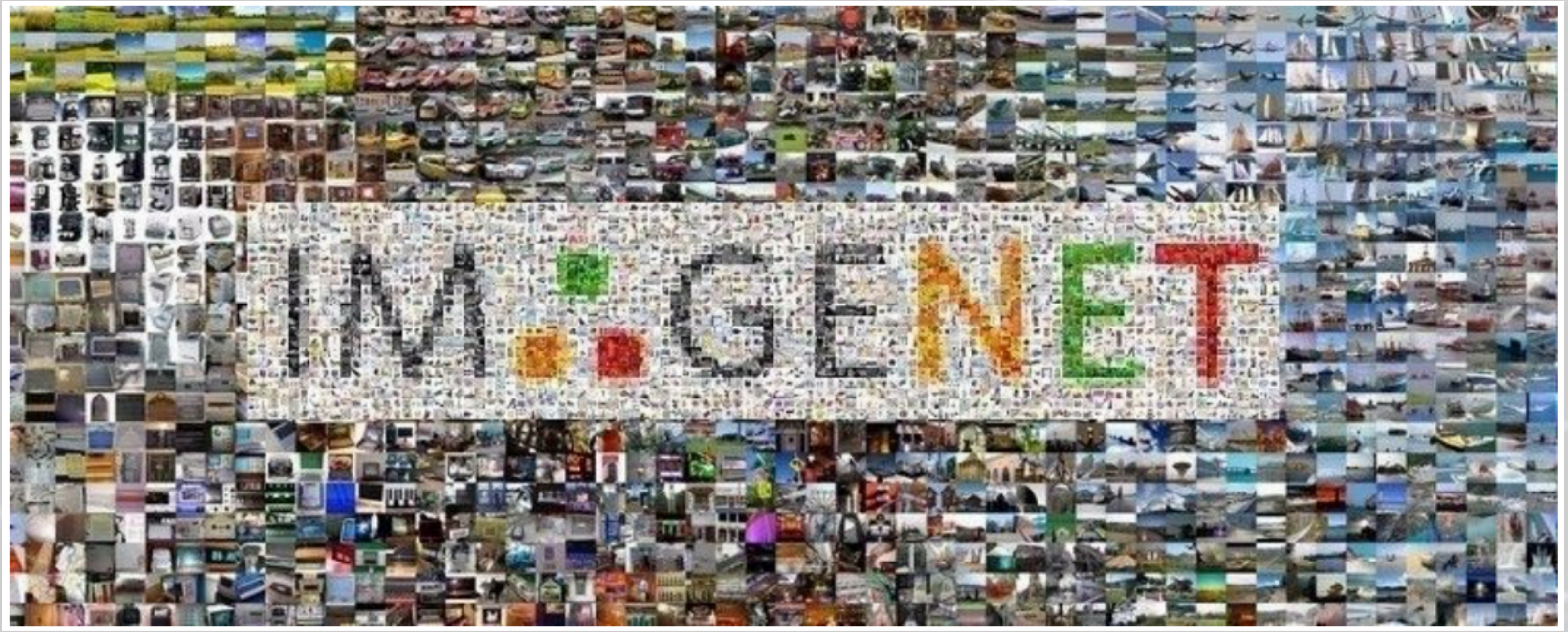


10.000 images from ImageNet:
8.000 training
1.000 validation
1.000 test



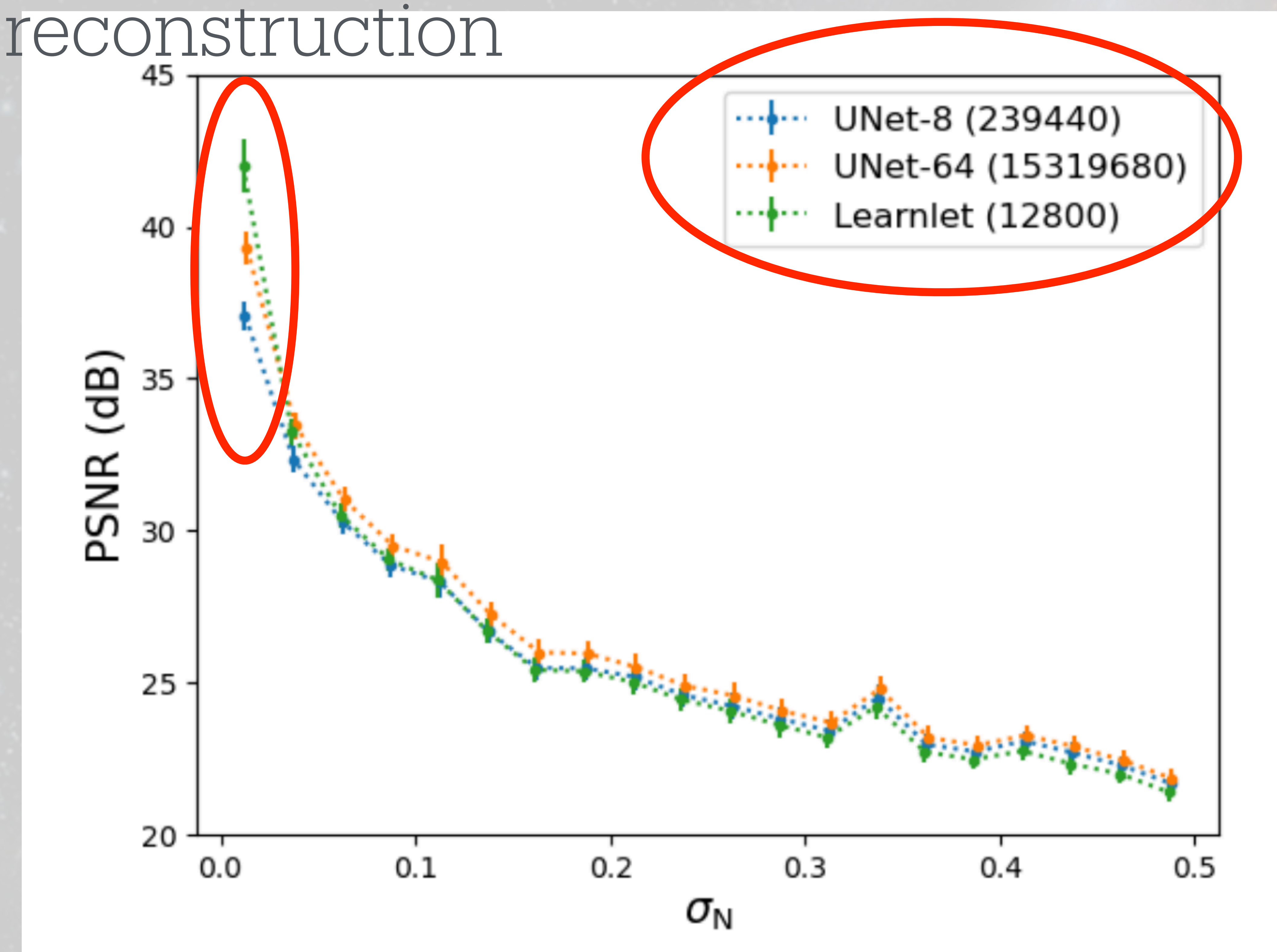
Performance of networks on the test set

Training learnlets



10.000 images from ImageNet:
8.000 training
1.000 validation
1.000 test

Exact reconstruction

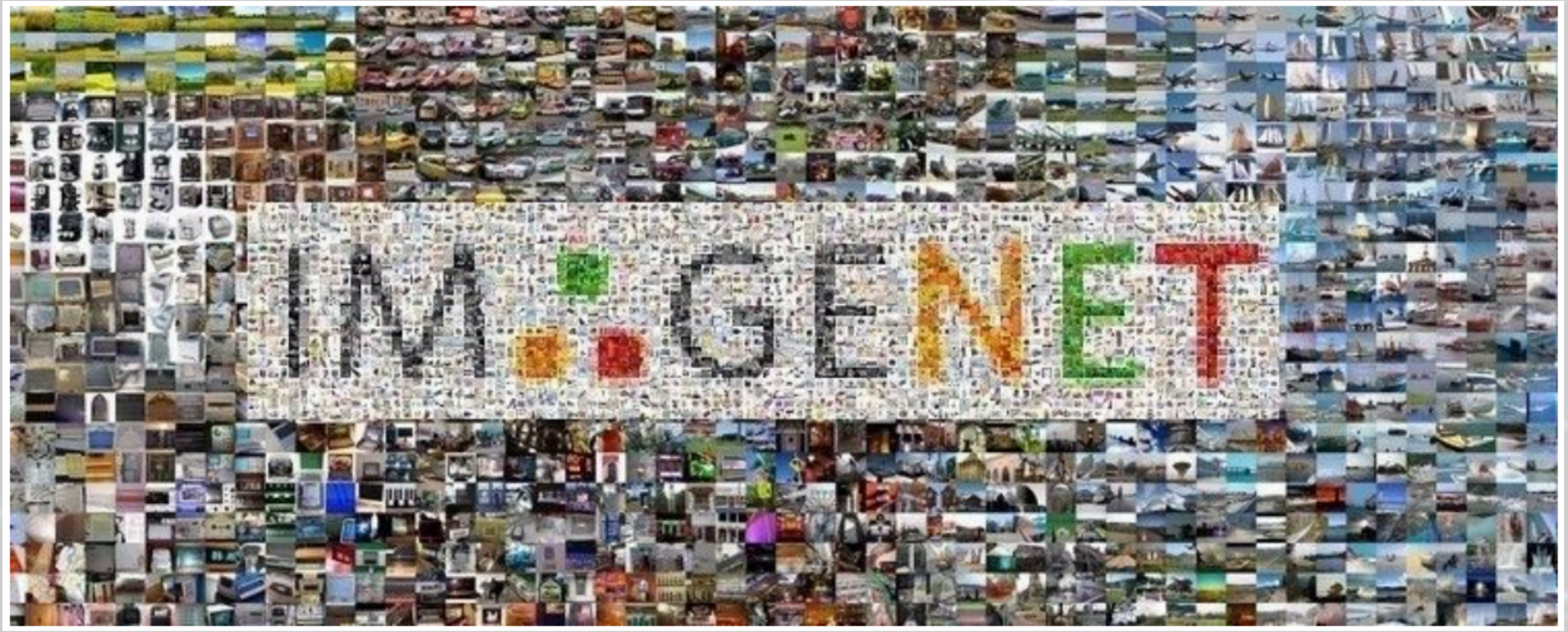


Significant lower number of free parameters

Performance of networks on the test set

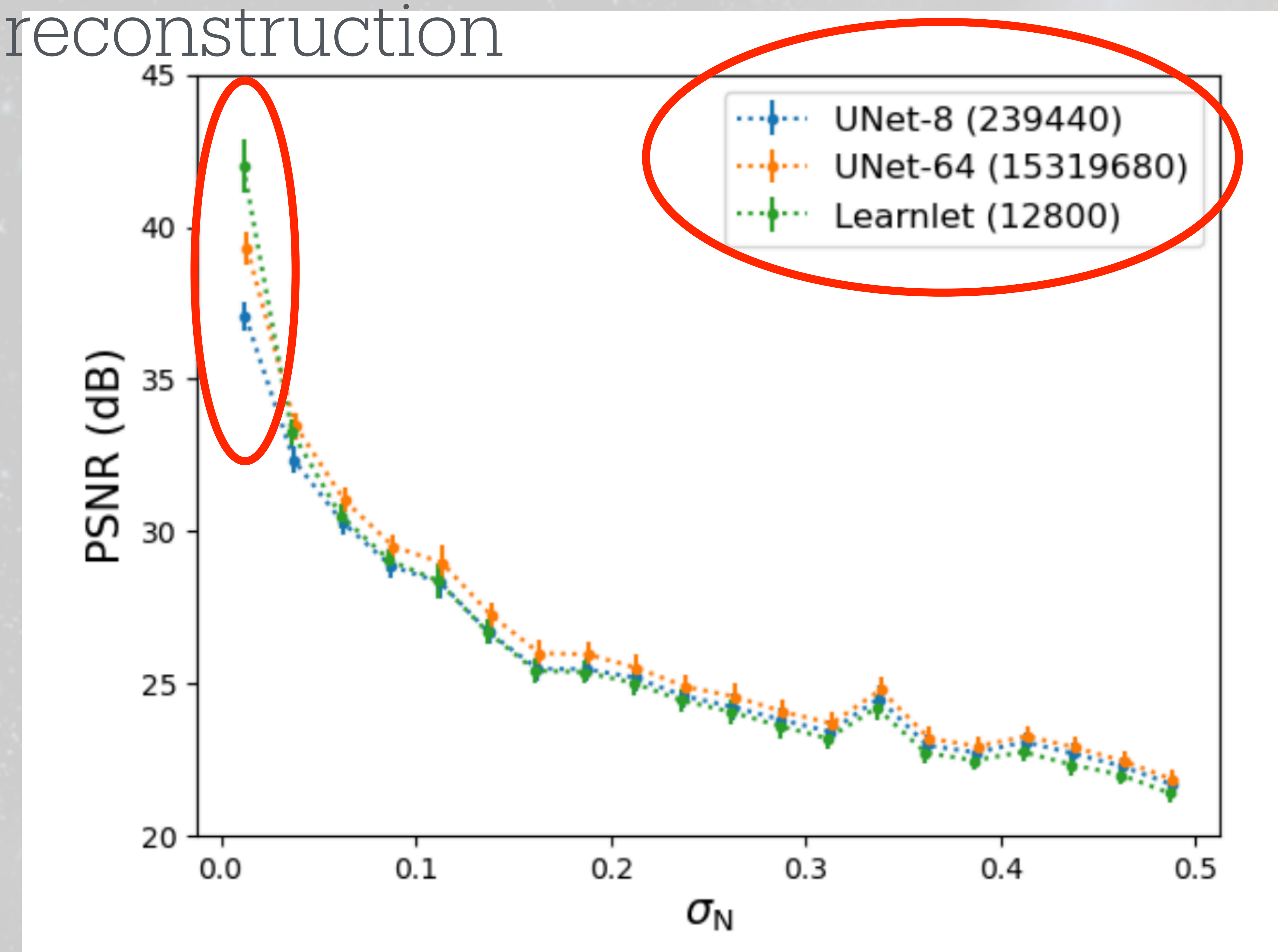
Illustration

Training learnlets



10.000 images from ImageNet:
8.000 training
1.000 validation
1.000 test

Exact reconstruction



Significant lower number of free parameters

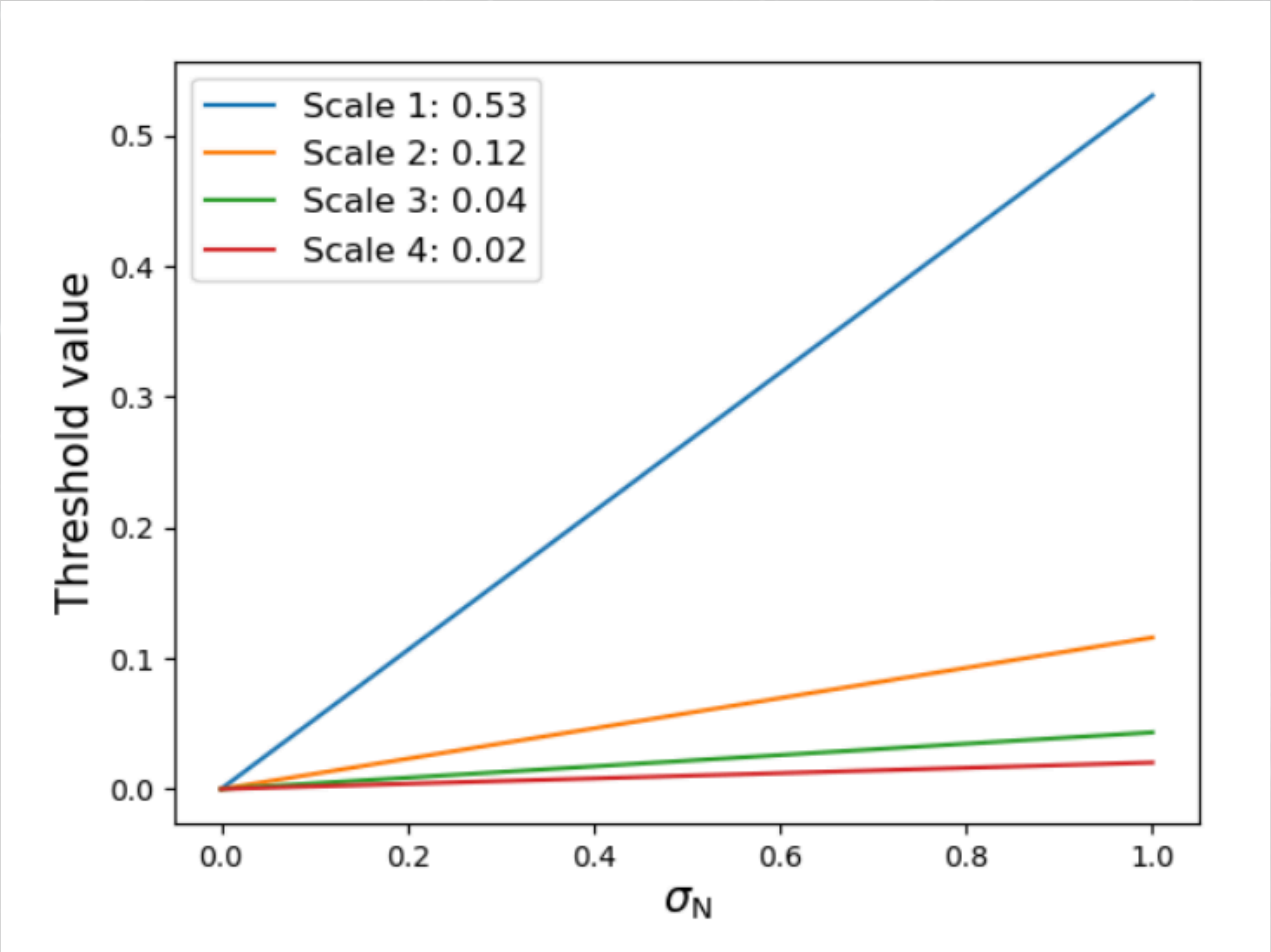
Performance of networks on the test set

GitHub:
https://github.com/vicbonj/learnlet.git

(PyTorch, pre-trained loaded weights)

Training learnlets

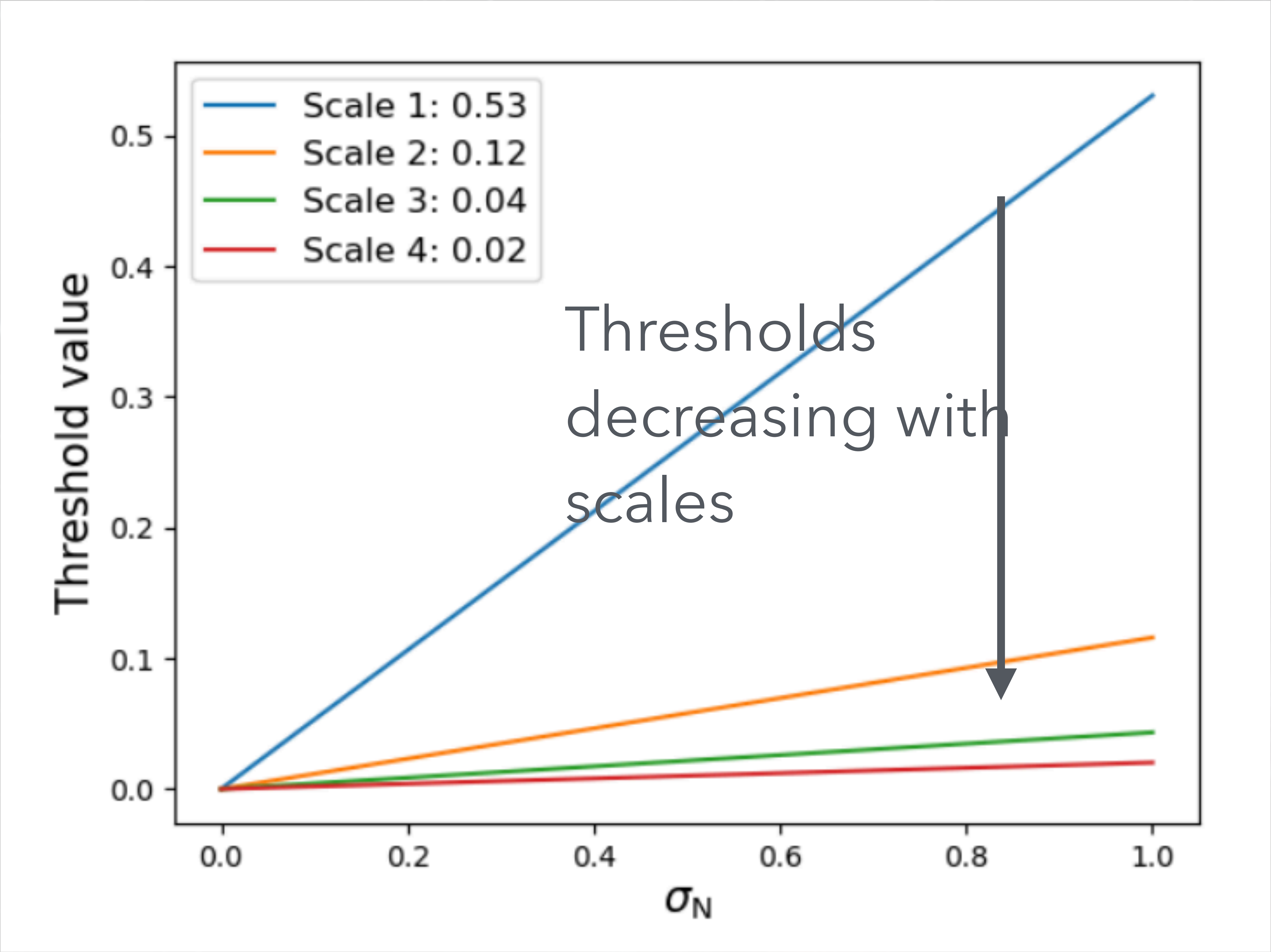
Thresholds k_j learned



Illustration

Training learnlets

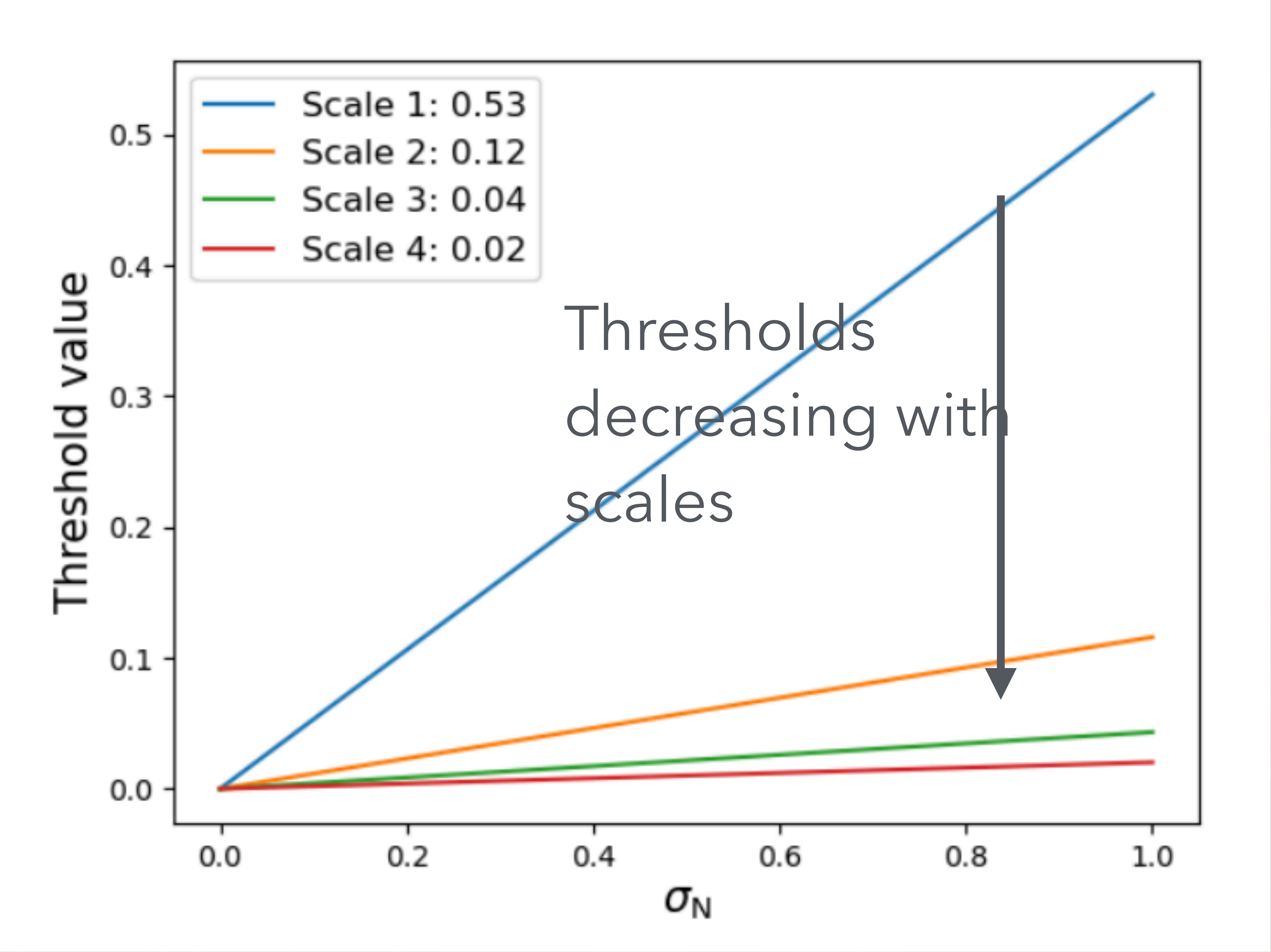
Thresholds k_j learned



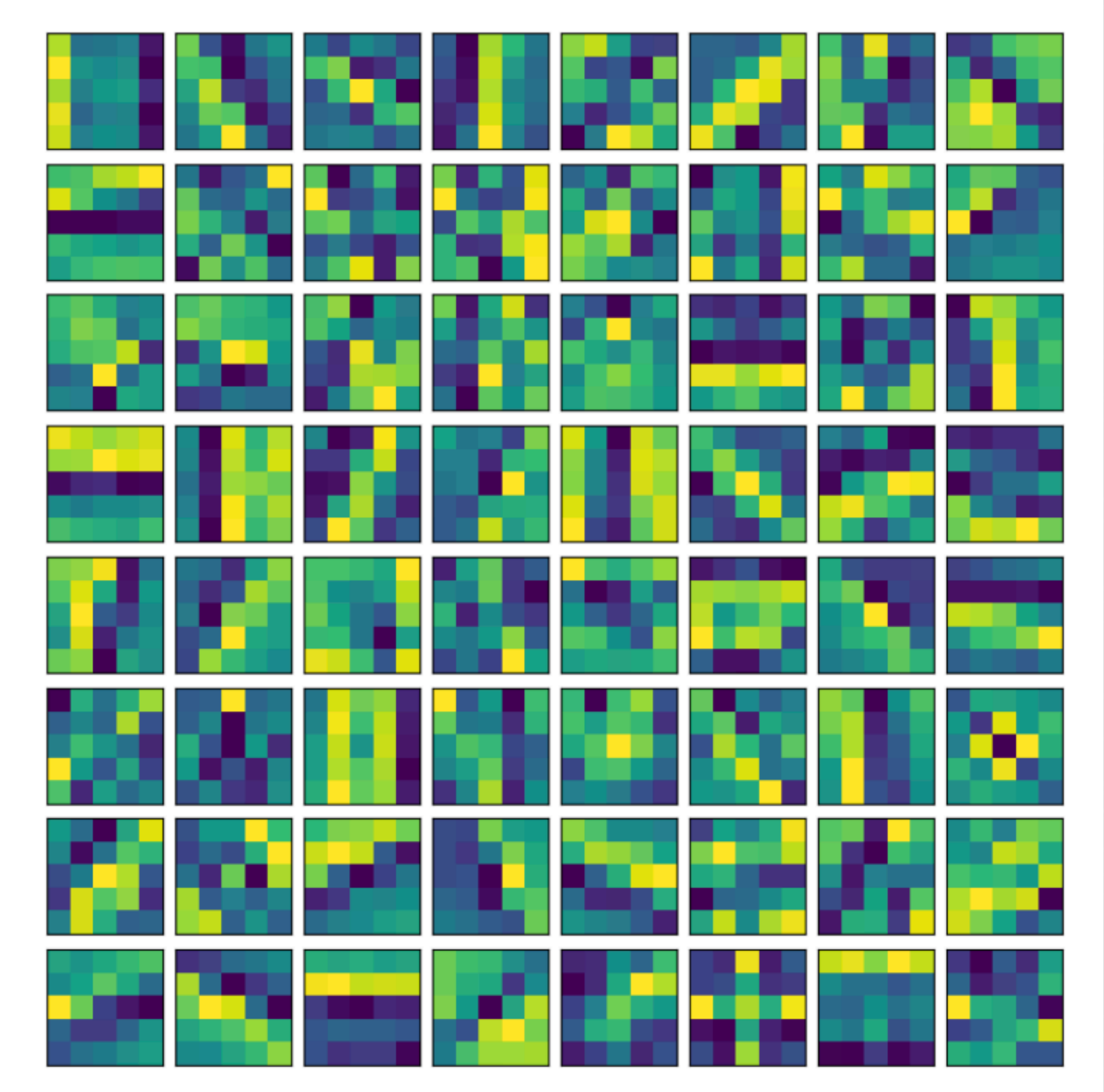
Thresholds decreasing with scales

Training learnlets

Thresholds k_j learned

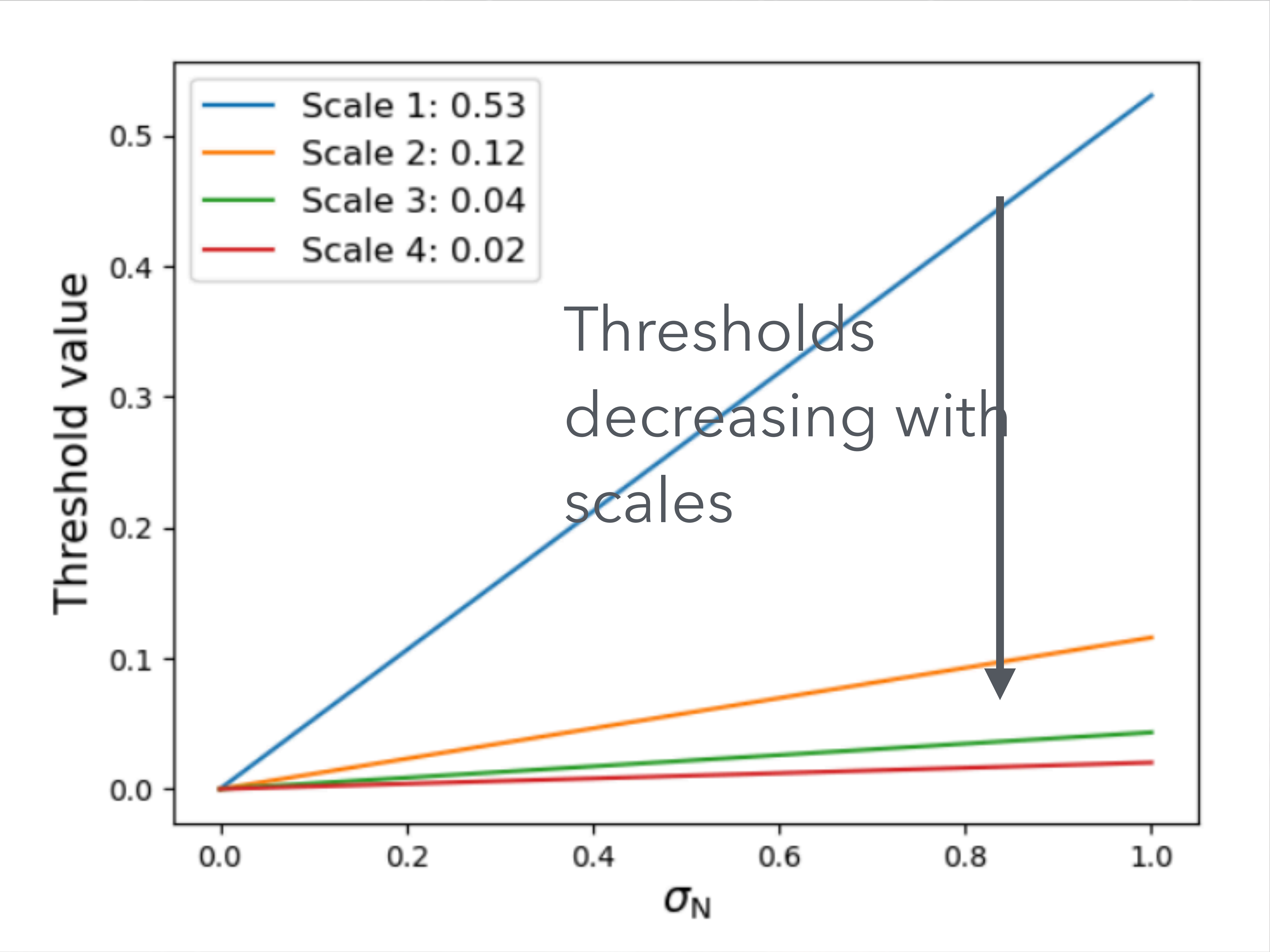


1st scale filters learned

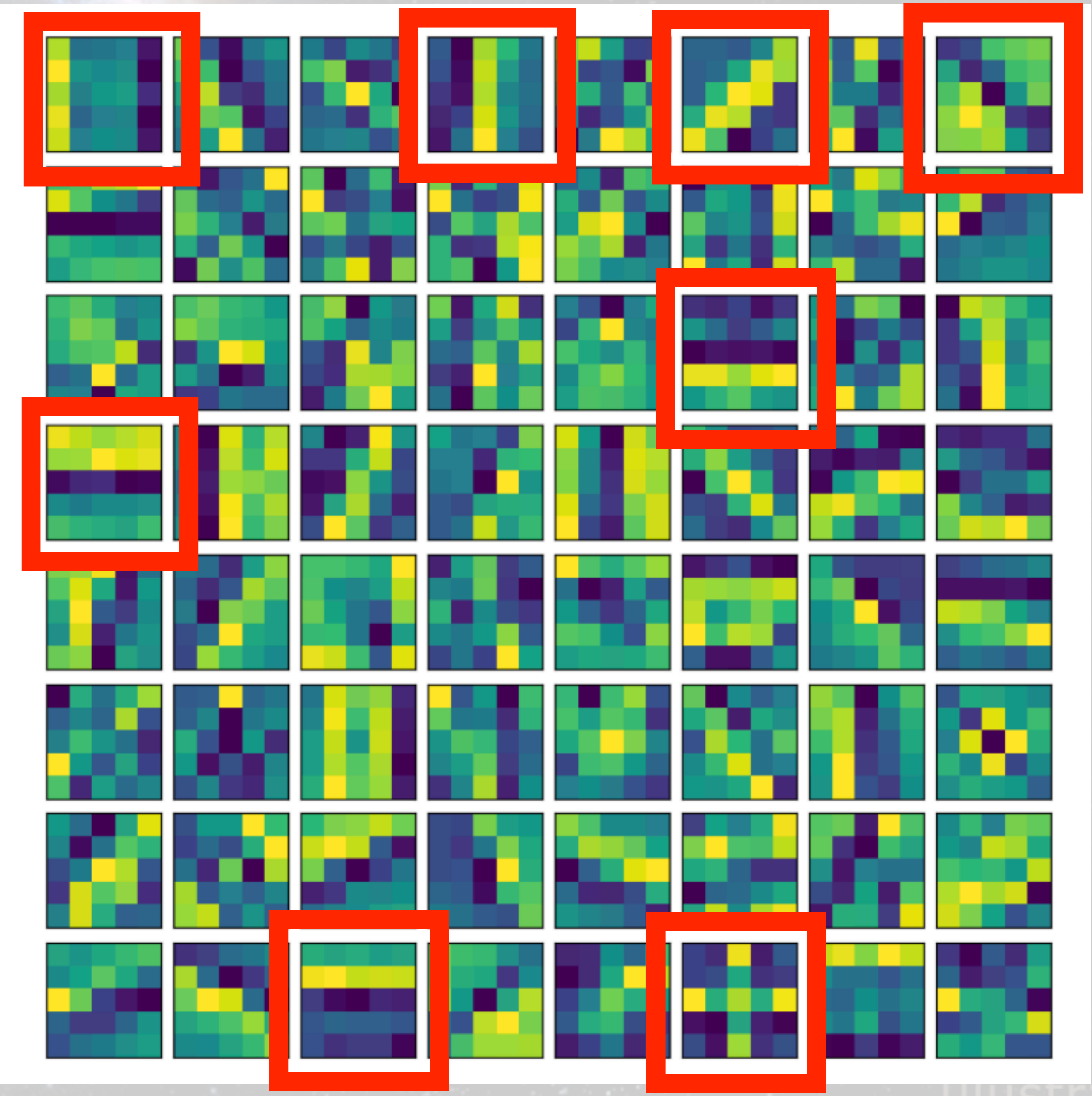


Training learnlets

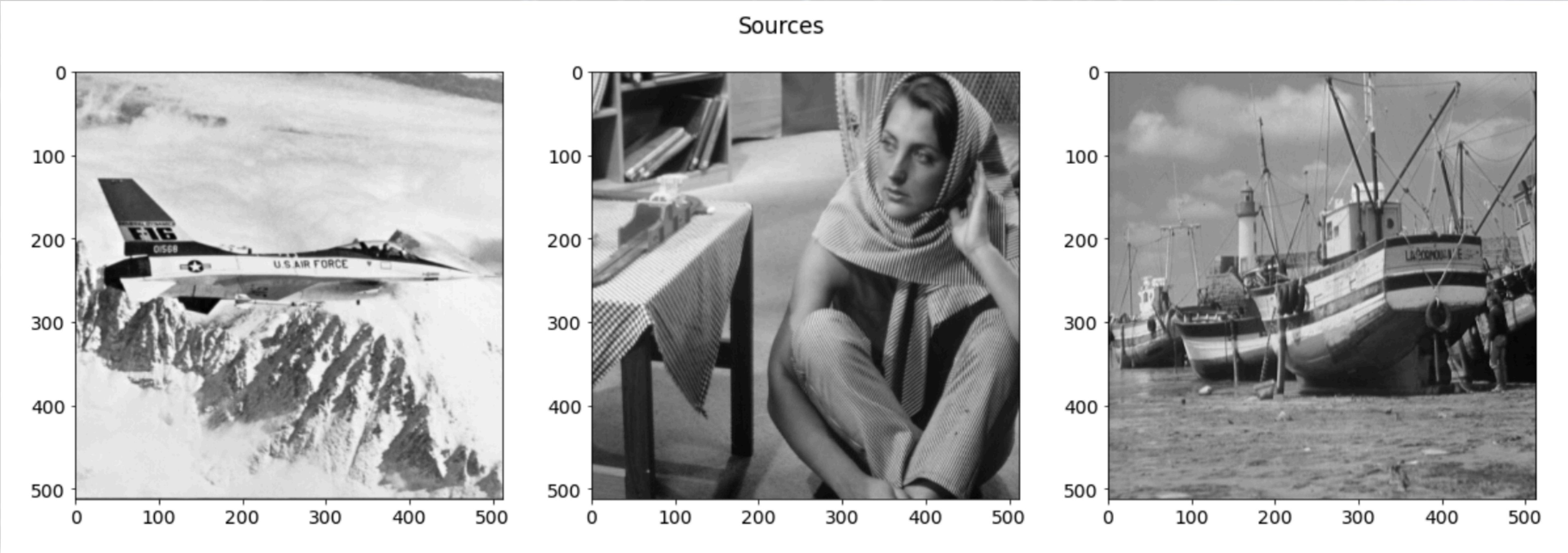
Thresholds k_j learned



1st scale filters learned



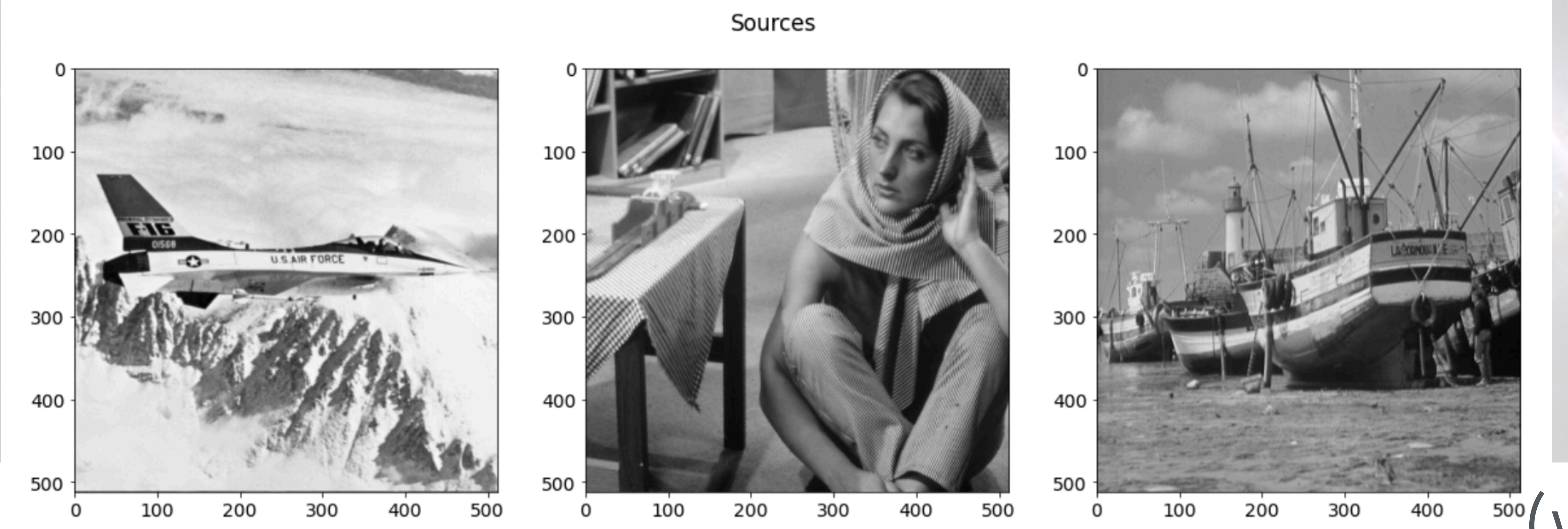
Learnlet Component Separator (LCS): results



Toy model with multiple realizations of A (6 channels) and N (gaussian white) with evolving σ

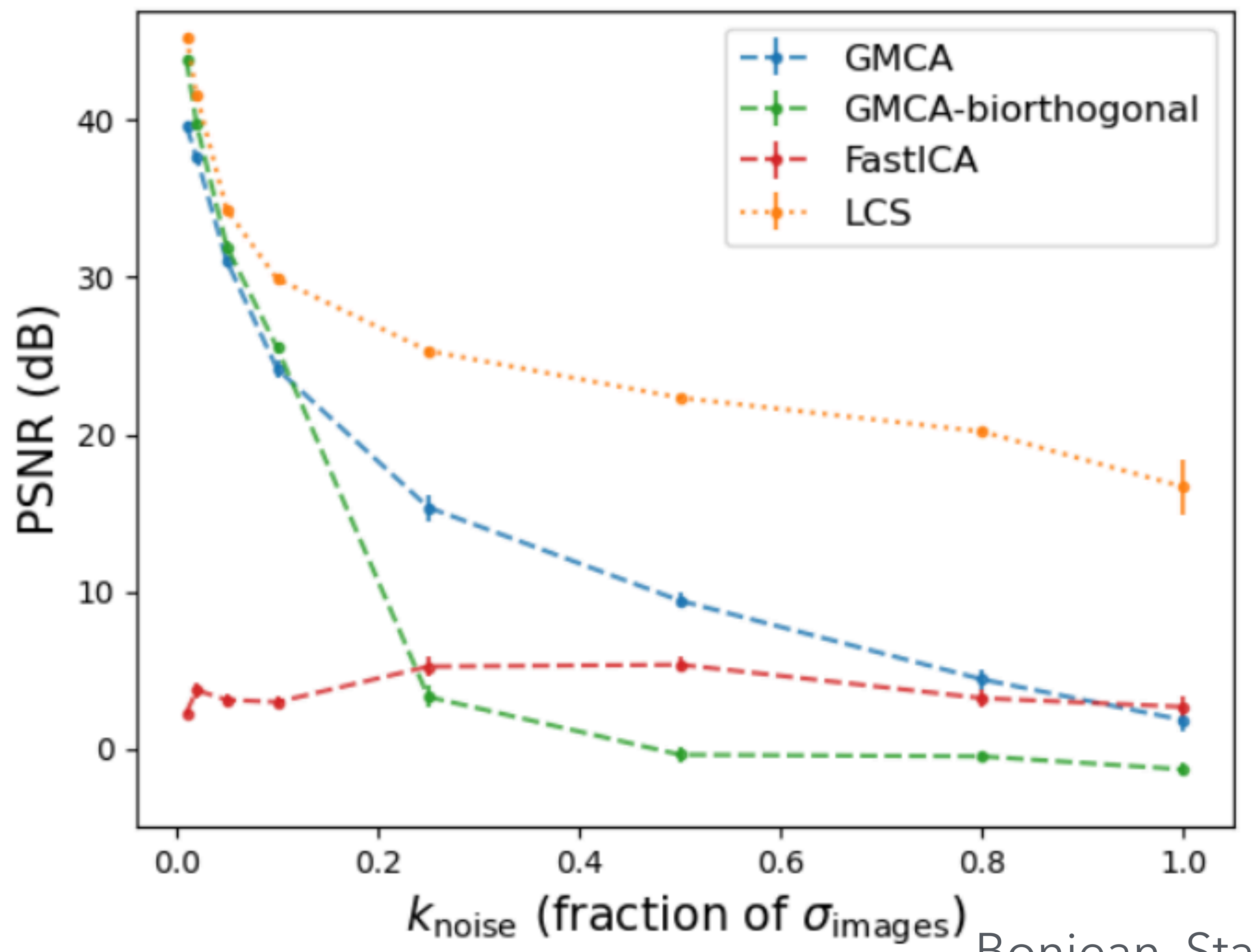
Illustration

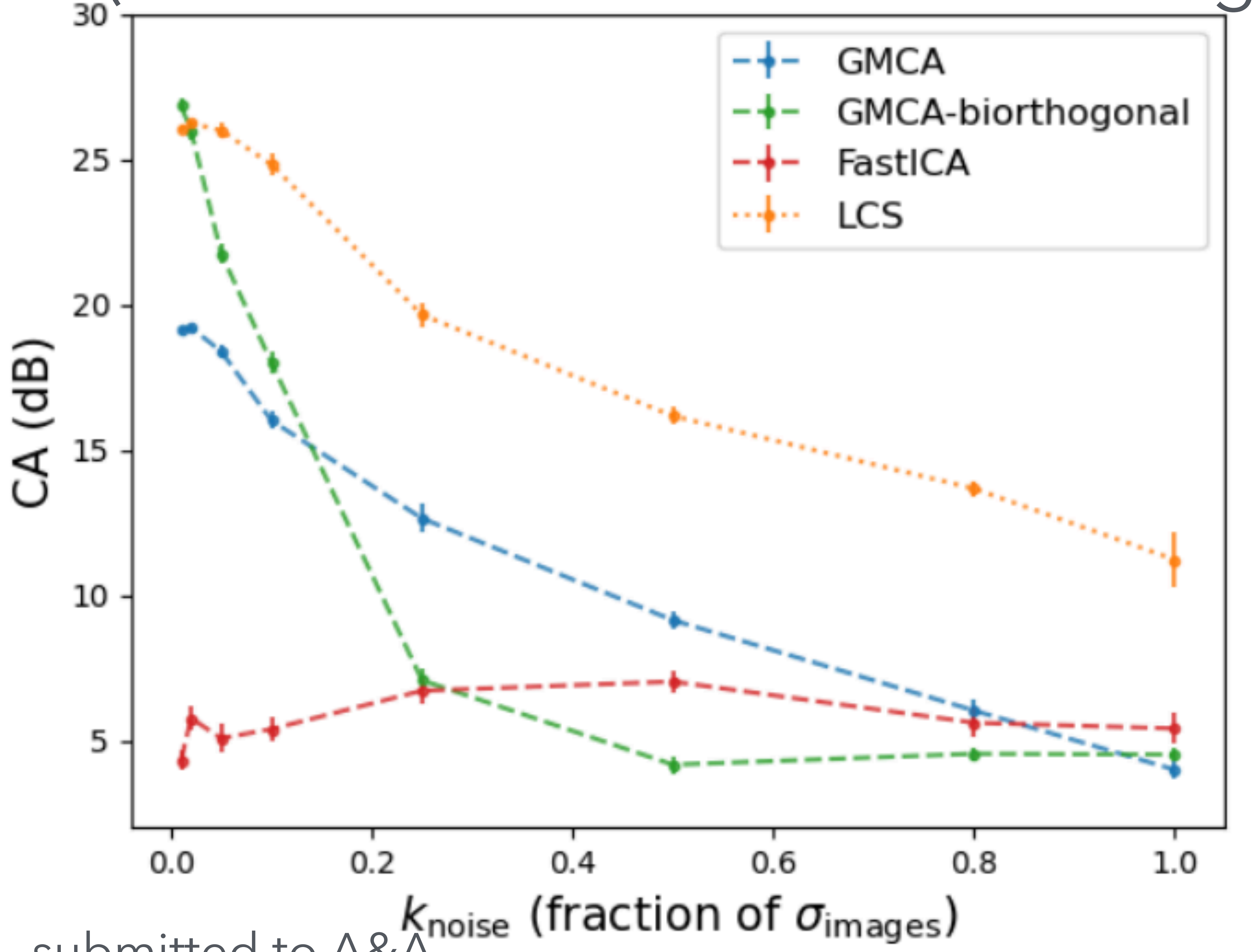
Learnlet Component Separator (LCS): results



Toy model with multiple realizations of A (6 channels) and N (gaussian white) with evolving σ

(with common learnlets from ImageNet)



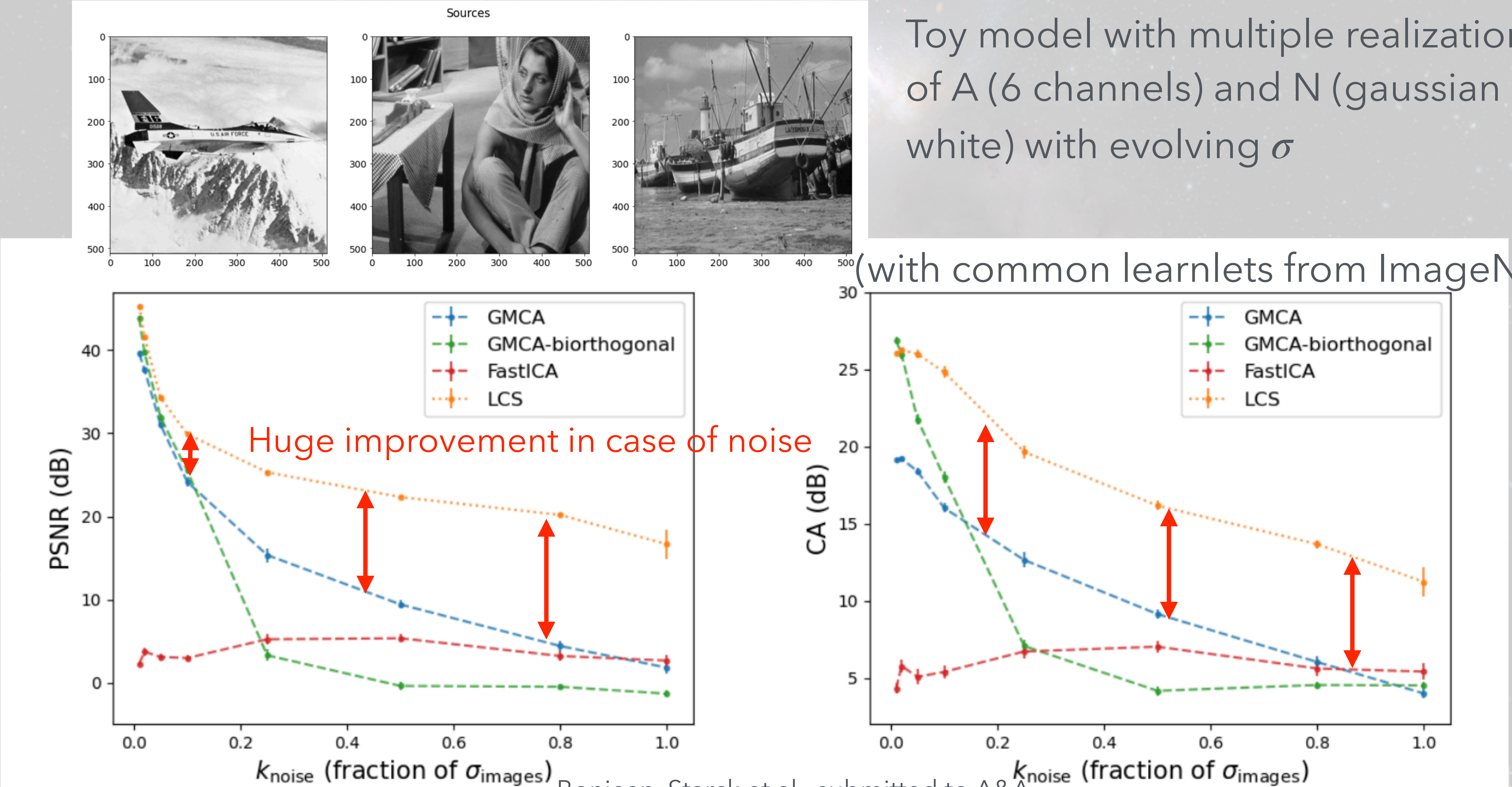


Bonjean, Starck et al., submitted to A&A

Learnlet Component Separator (LCS): results

Toy model with multiple realizations of A (6 channels) and N (gaussian white) with evolving σ

(with common learnlets from ImageNet)



Bonjean, Starck et al., submitted to A&A

Learnlet Component Separator (LCS): results

Learnlets trained for each components i:

Illustration

Learnlet Component Separator (LCS): results

Learnlets trained for each components i:

DTD texture dataset (Cimpoi et al, 2014), 120 images per 47 classes, here focused on 2 classes:

banded

dotted

Illustration

Learnlet Component Separator (LCS): results

Learnlets trained for each components i:

DTD texture dataset (Cimpoi et al, 2014), 120 images per 47 classes, here focused on 2 classes:

banded

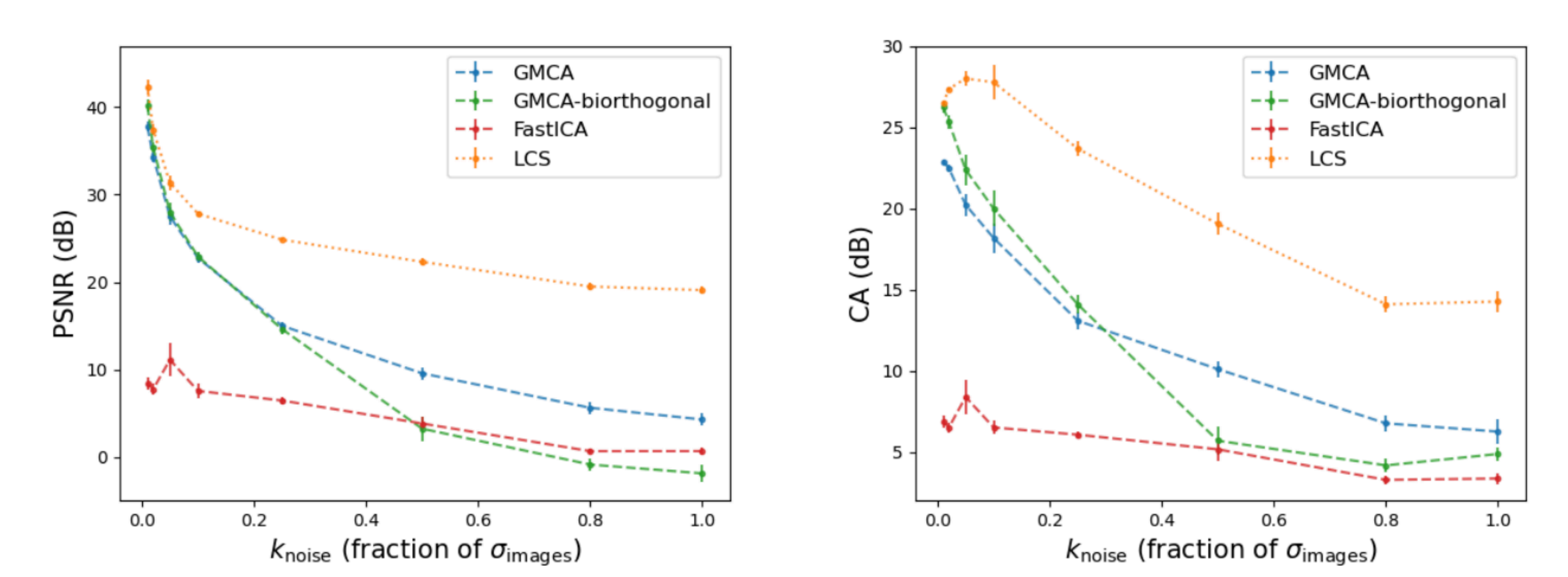
dotted

→ One training per class: $\mathscr{L}_{\text{banded}}$ and $\mathscr{L}_{\text{dotted}}$ with 119 images (transfer learning from ImageNet)

Learnlet Component Separator (LCS): results

Learnlets trained for each components i:

Different realizations of A and N for different σ with the 120th images of the classes



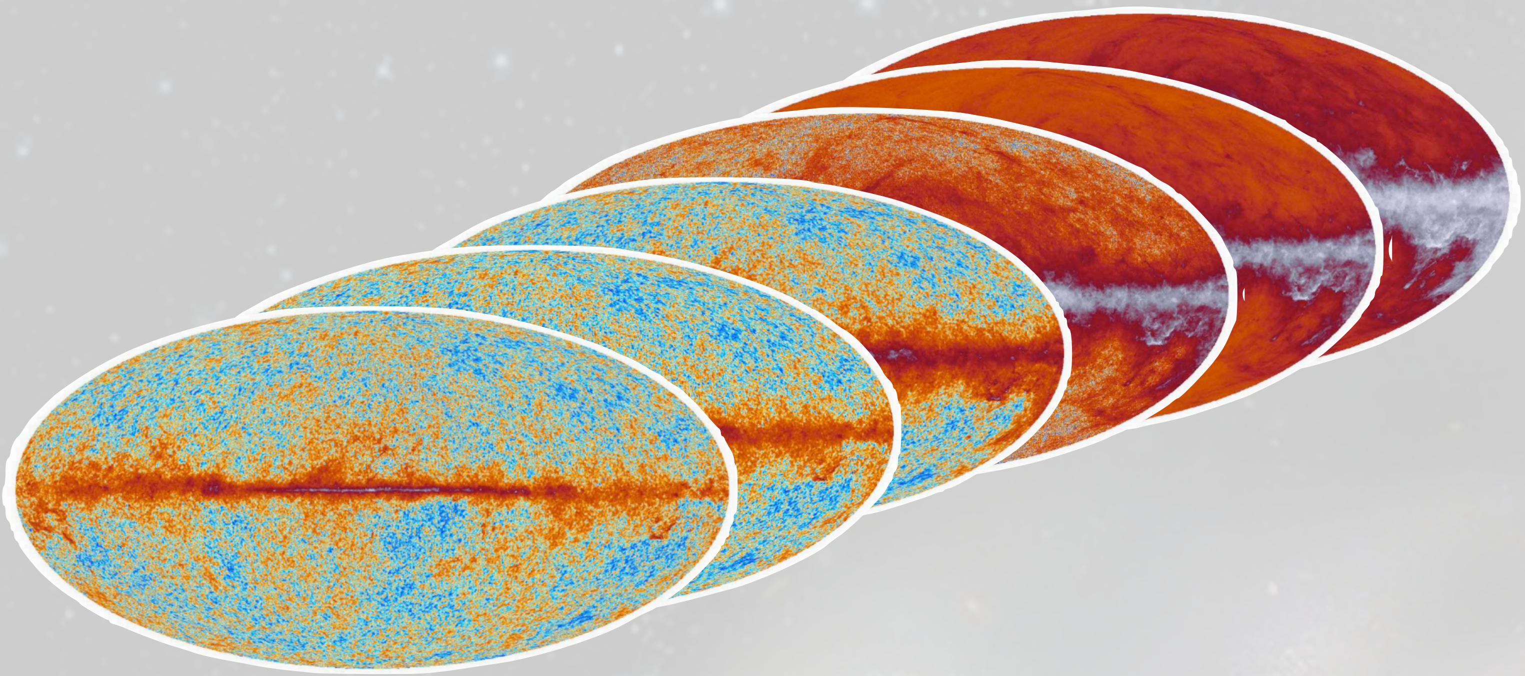
Illustration

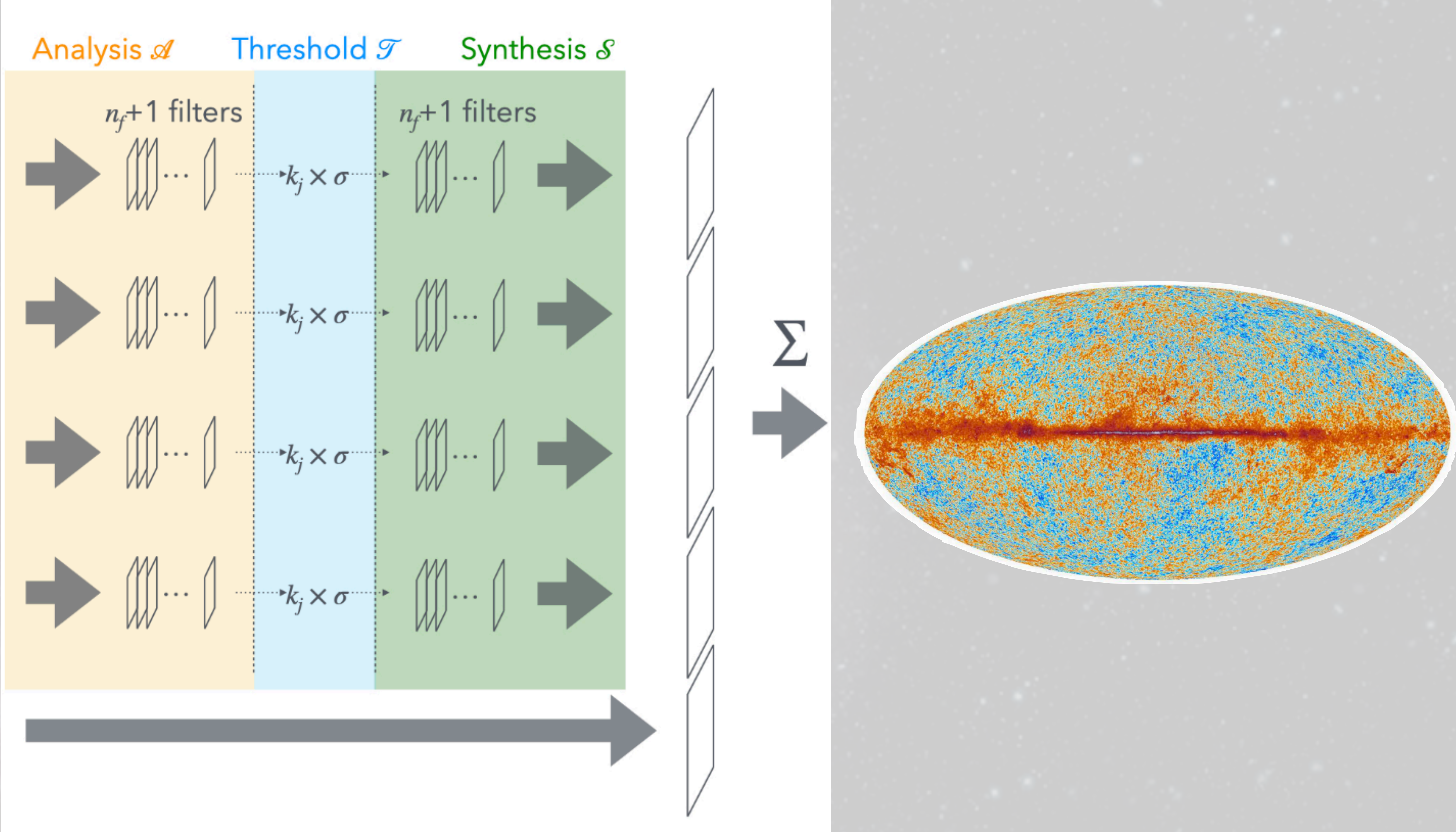
LCS on the sphere: application to CMB data

Illustration

LCS on the sphere: application to CMB data

Starlets transform computed on the sphere with alms at each iteration (or not?)





Illustration

LCS on the sphere: application to CMB data

WebSky numerical simulations (Stein et al., 2020) of the millimetre sky in HEALPIX at Nside=512 in 5 Planck HFI Frequencies:

3 components: CMB, SZ, CIB ⚠️ Invalid linear model!

Illustration

LCS on the sphere: application to CMB data

WebSky numerical simulations (Stein et al., 2020) of the millimetre sky in HEALPIX at Nside=512 in 5 Planck HFI Frequencies:

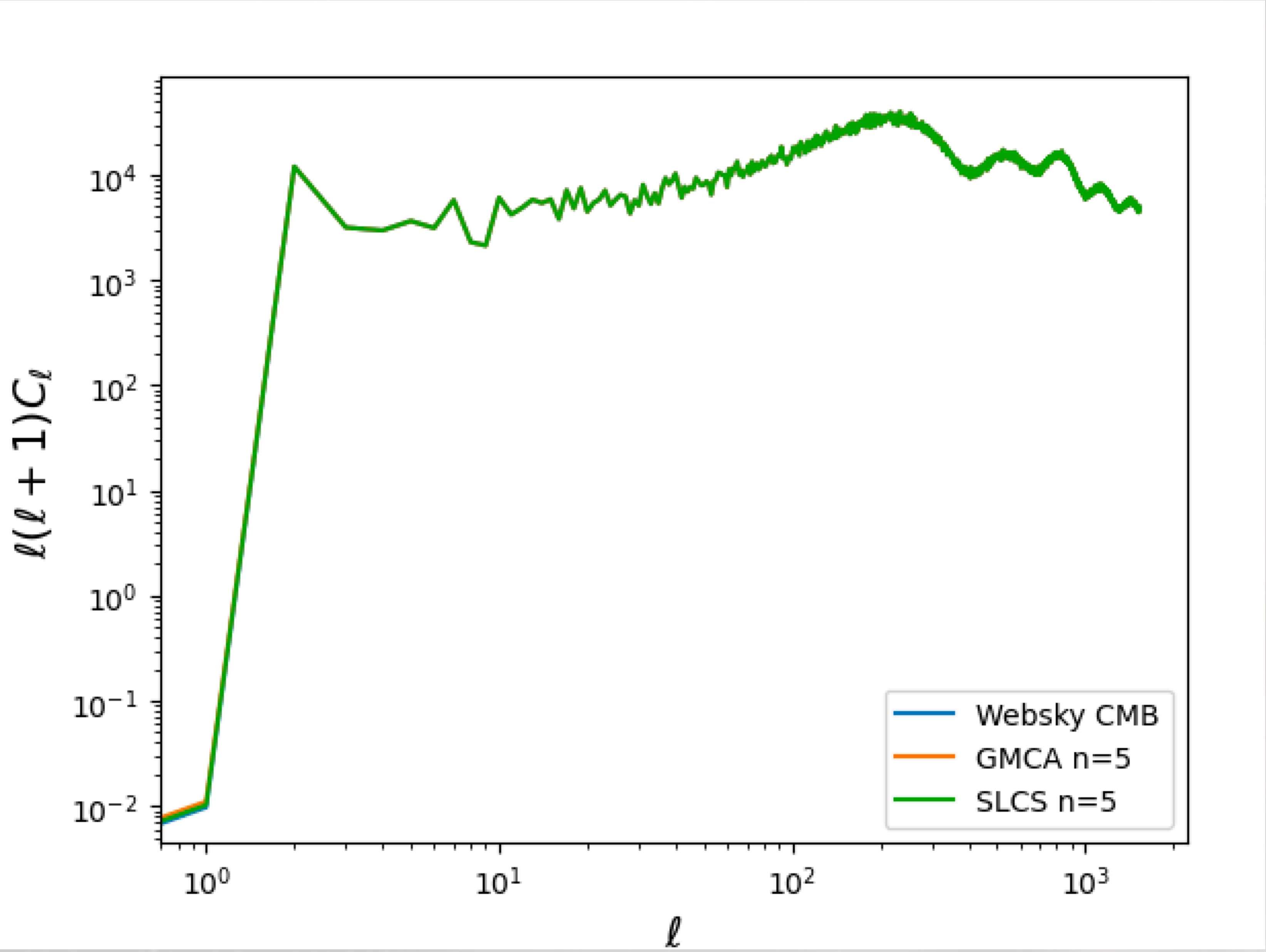
3 components: CMB, SZ, CIB ⚠️ Invalid linear model!

One training per class: $\mathscr{L}_{\mathrm{CMB}}$, $\mathscr{L}_{\mathrm{SZ}}$ and $\mathscr{L}_{\mathrm{CIB}}$ (transfer learning from ImageNet)

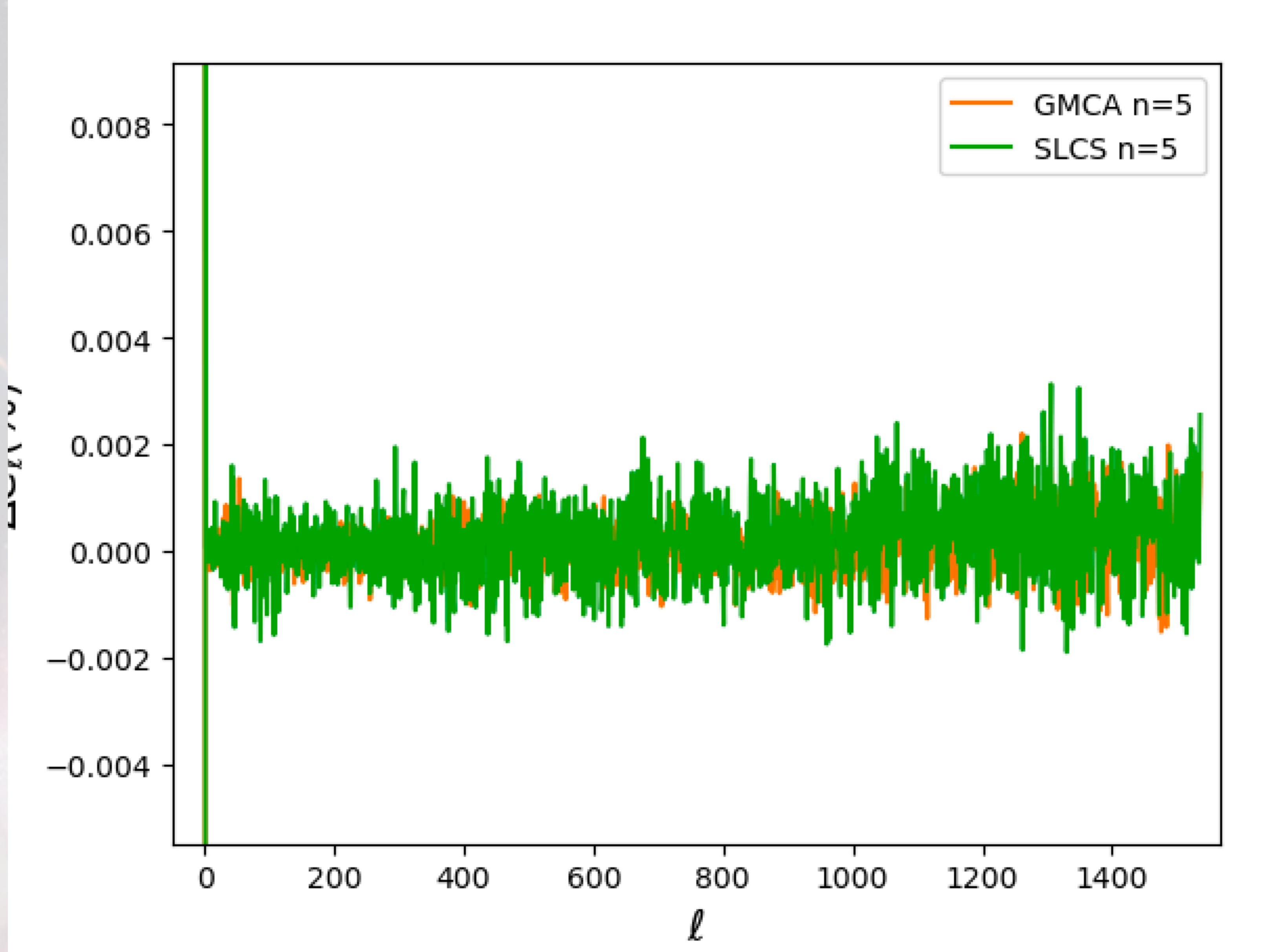
Illustration

LCS on the sphere: application to CMB data

Results on CMB (« easy » component):



Cl

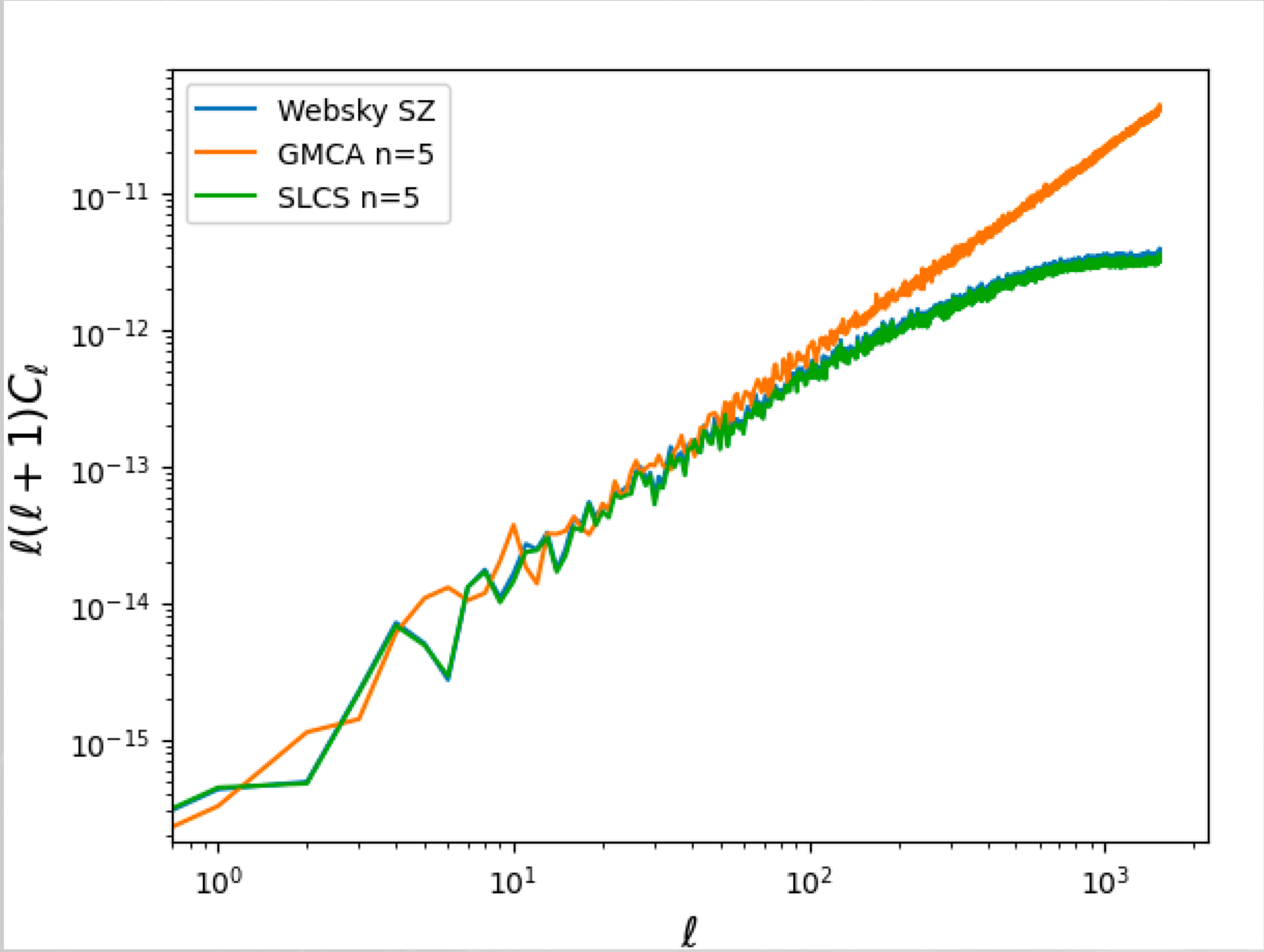


Residuals

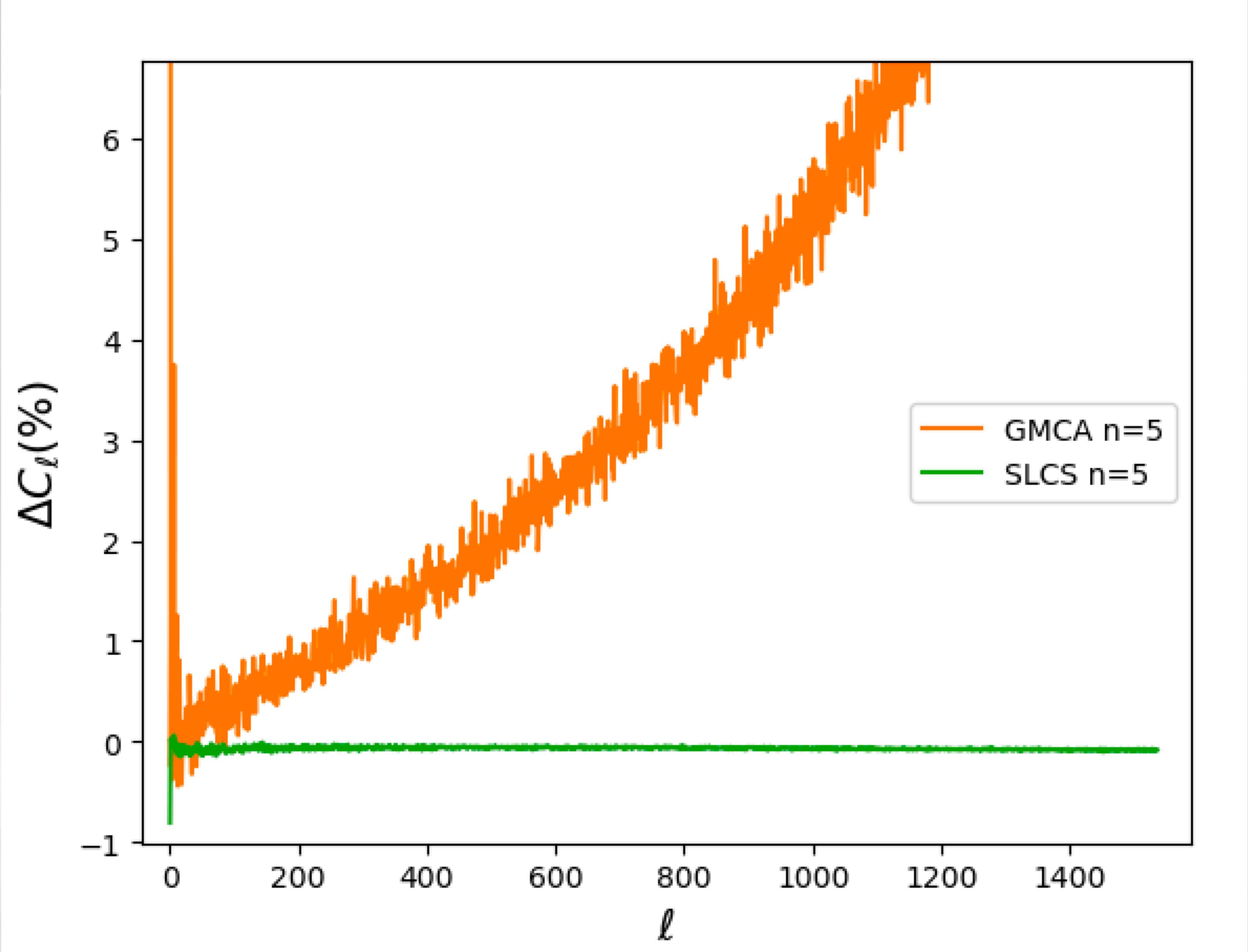
Illustration

LCS on the sphere: application to CMB data

Results on SZ (**challenge**):



Cl



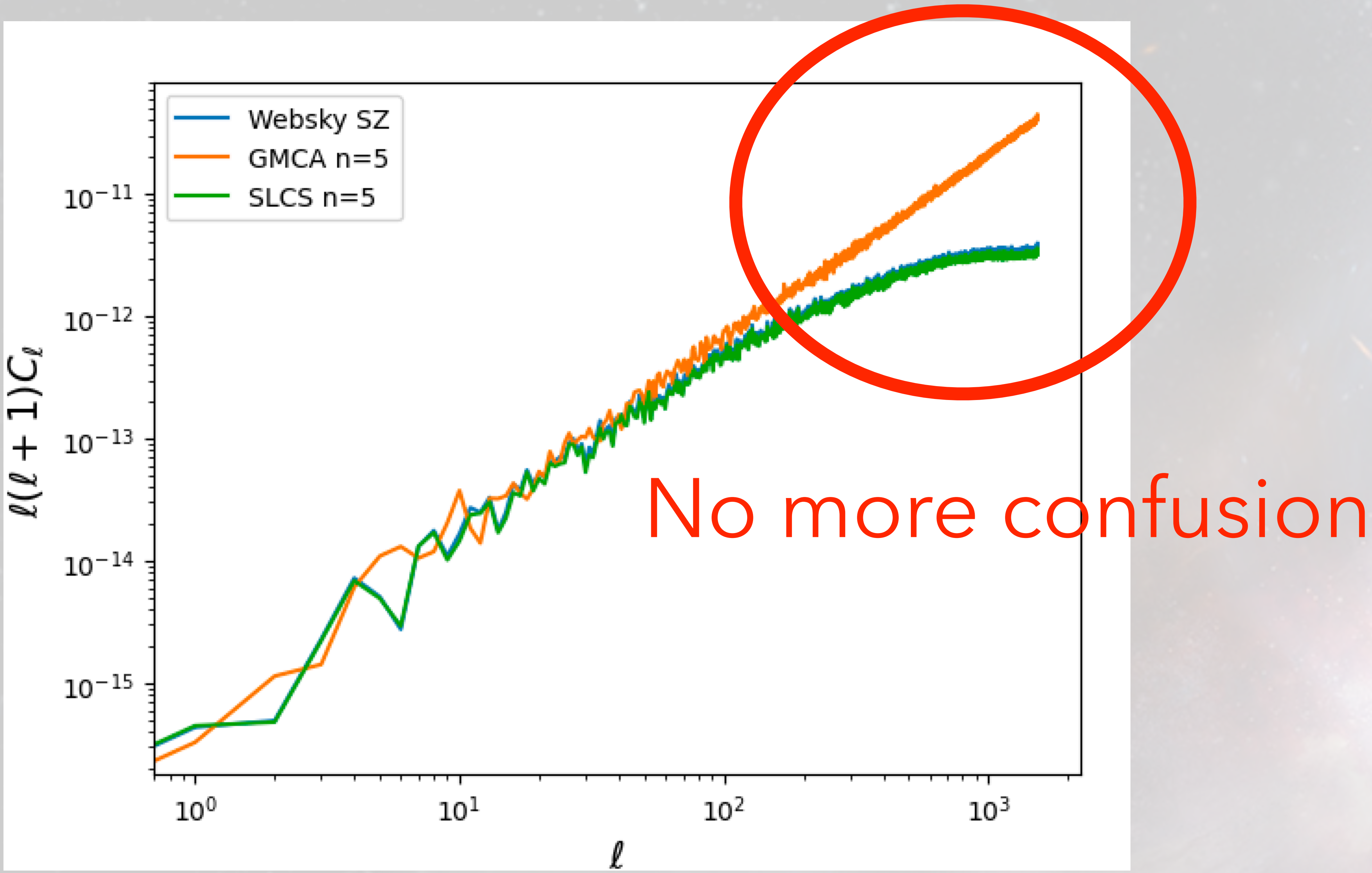
Residuals

3 components: CMB, SZ, CIB ⚠️ Invalid model!

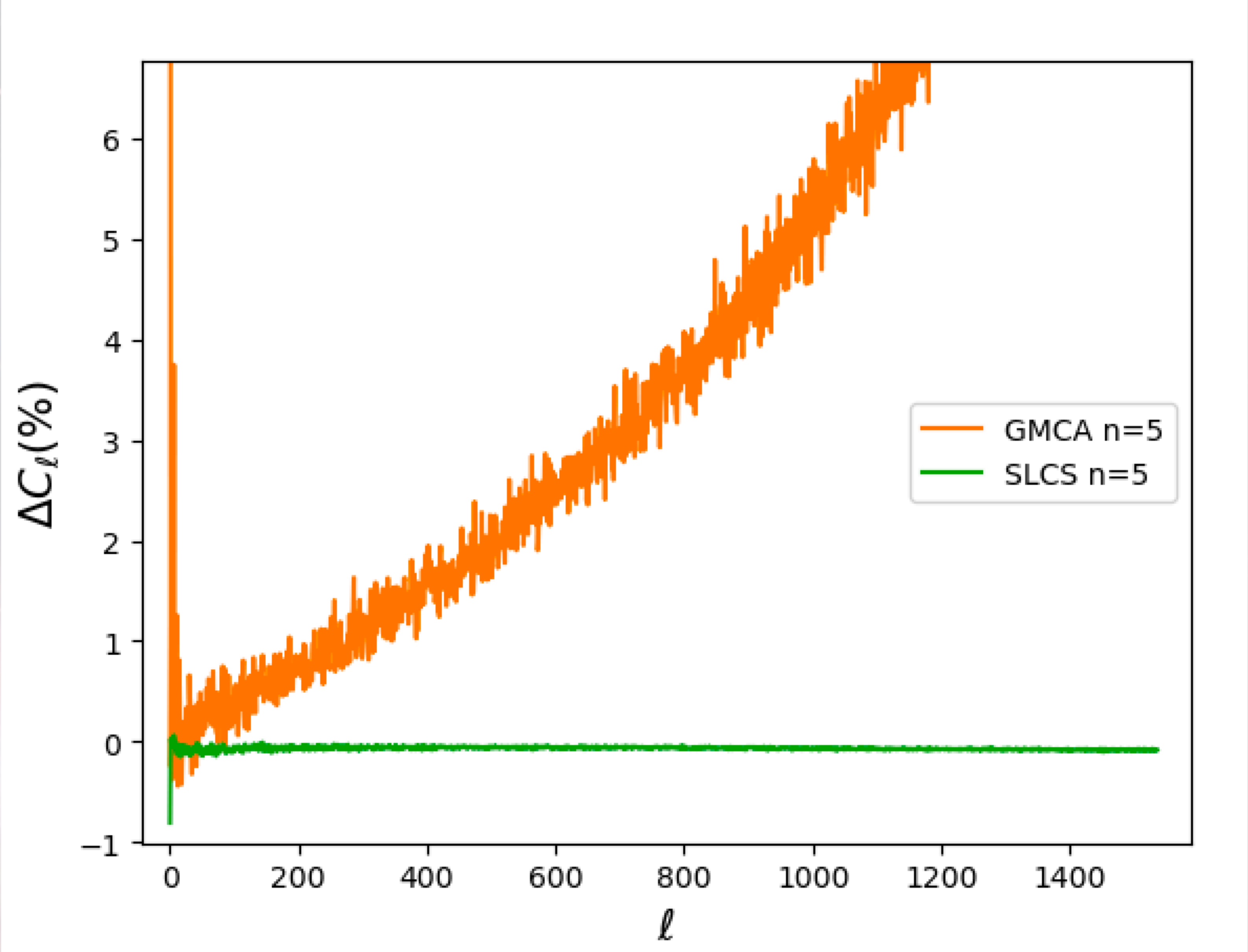
Illustration

LCS on the sphere: application to CMB data

Results on SZ (**challenge**):



Cl



Residuals

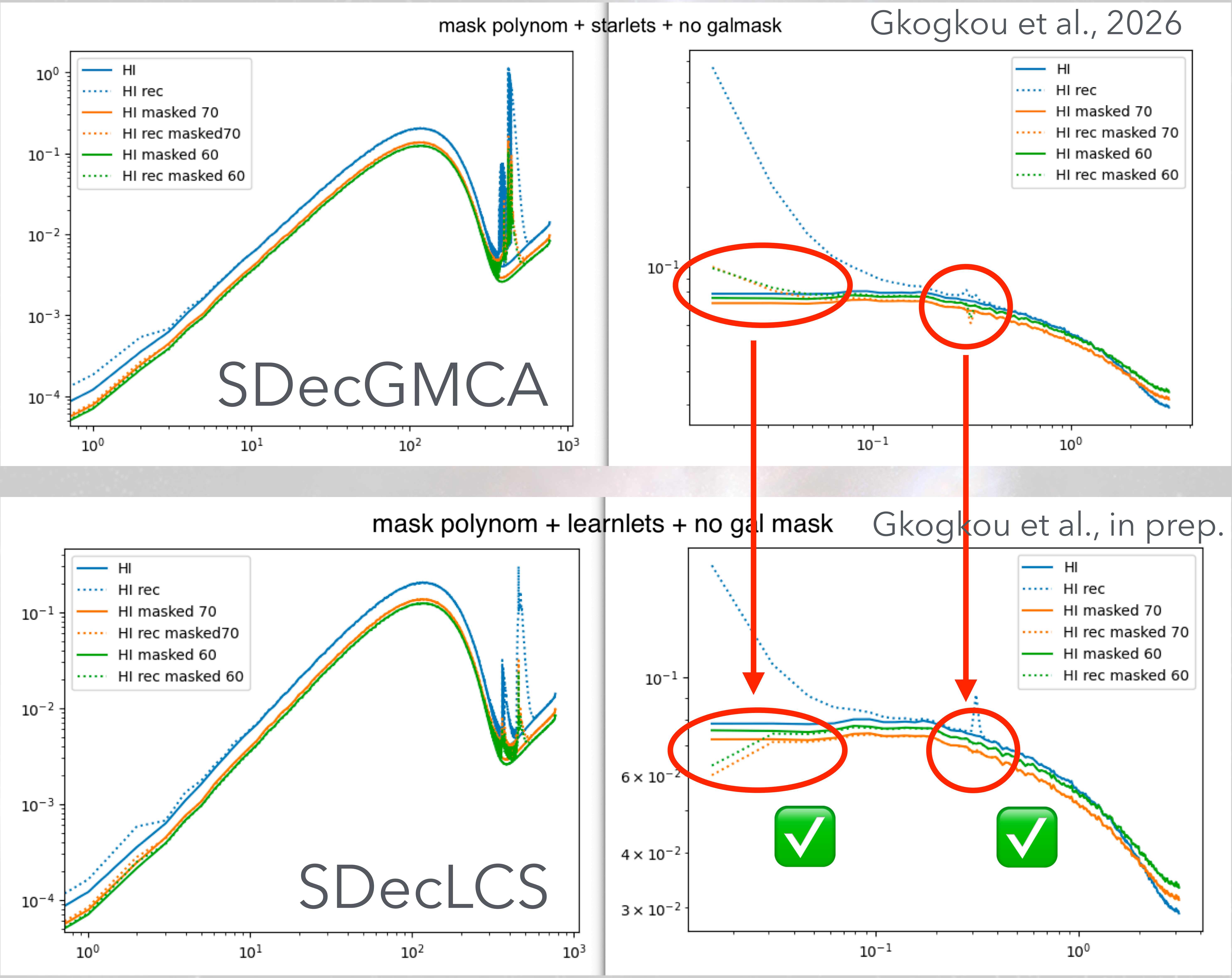
3 components: CMB, SZ, CIB ⚠️ Invalid model!

More results to come: with noise, beam, and foregrounds

Adaptation on LCS on the sphere, with deconvolution, application to radio data:

HI recovery:

See Sia's talk tomorrow!



Summary

- New BSS algorithm based on learnlets: LCS (combining expressivity of deep learning and mathematical properties of wavelets)
- Outperforms state-of-the-art BSS algorithms
- To the sphere
- Combined with deconvolution (**Sia's talk**)
- Promising for SKA (HI extraction) and SO, Litebird (CMB, SZ, dust)

Next steps

- Error estimation?
- Include beams, foregrounds and deconvolution in CMB data

Illustration