

DeCompress: Denoising via Neural Compression

Learn to Compress & Compress to Learn Workshop, 2025

Ali Zafari* Xi Chen* Shirin Jalali

Department of Electrical and Computer Engineering
Rutgers University



Pre-presentation: Beyond DeCompress

- Extended algorithm and theoretical insights:
 - Zero-shot Denoising via Neural Compression*: arXiv:2506.12693
- Happy to chat - stop by our poster ↓



DeCompress: Denoising via Neural Compression

Ali Zafar¹ Xi Chen² Shree N. Jayaram³

¹Department of Electrical and Computer Engineering, Rutgers University
²IBM Research



Denoising Problem Setup

Signal of interest $\mathbf{x} = [x_1, x_2, \dots, x_N]^T \in \mathbb{R}^N$. Observations $\mathbf{y} = [y_1, y_2, \dots, y_N]^T$. The observations are conditionally independent given \mathbf{x} , and follow the same noise distribution:

$$y_i = x_i + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2)$$

A denoising method aims to estimate $\hat{\mathbf{x}}$ given the noisy observations \mathbf{y} .

Compression-based Denoising

Background: Recovered signals are more compressible than noisy ones.

- Denoise (D): estimate $\hat{\mathbf{x}}$ given noisy observations \mathbf{y} .
- Compress (C): compress \mathbf{x} into \mathbf{z} using a compressor \mathcal{C} .

Theoretical foundation: Let $\mathcal{Q} = \mathcal{N}(\mathbf{0}, \Sigma)$ denote the joint distribution of interest with an associated natural logarithm of a fixed size \mathcal{Q} . A loss compression pipe for \mathcal{Q} is defined by an encoder-decoder pair $(\mathcal{E}, \mathcal{D})$, $\mathcal{E}: \mathcal{Q} \rightarrow \mathcal{Z}$ and $\mathcal{D}: \mathcal{Z} \rightarrow \mathcal{Q}$, where $\mathcal{Z} = \{z_1, z_2, \dots, z_M\}$ is a finite set of M codewords. The set of reconstruction errors for a codeword z_i is defined as:

$$\mathcal{E}_i = \{x_i - z_i, i = 1, \dots, M\} \subset \mathbb{R}^N$$

One expects \mathcal{E}_i to be more compressible than their noisy counterparts, for values of their elements.

- The more likely codewords have a larger signal-to-noise ratio, yielding better denoising.

1. **RECOVER** $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{x}))$

2. **DENOISE** $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$

Denoising by prepping for noisy observations with the learned codewords.

3. **POST-PROCESS** $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$ (See Fig. 1 and please refer to Fig. 1 for more details).

4. **RECOVER** $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$

Alternatively, to avoid the non-linear and non-invertibility of the log from a compressed signal, one can use a linear compressor \mathcal{C} and a linear decoder \mathcal{D} .

5. **RECOVER** $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$

See our arXiv preprint (2024) for further theoretical results.

Theorem 1 (MSE)

Let \mathcal{D} denote the output of the compression-based denoising (D) given noisy observations \mathbf{y} . Then:

$$\text{MSE}(\mathcal{D}) \leq \text{MSE}(\mathcal{C}) + \frac{\sigma^2}{\lambda}$$

with equality when $\lambda = \sigma^2$.

Constant Squared Signal

Under a family of compression codes such that, when used with the denoiser defined in (1), the variance of codewords

$$\text{Var}(\mathcal{E}_i) = \lambda \mathbf{I}, \quad \lambda > 0$$

with probability of error $\lambda \rightarrow 0$ as $M \rightarrow \infty$ and $\mathcal{C} = \text{DCT}$ or $\mathcal{C} = \text{DFT}$.

Denoising via Neural Compression

RECOVER $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{x}))$

DENOISE $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$

POST-PROCESS $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$

RECOVER $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$

1. **RECOVER** $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{x}))$

2. **DENOISE** $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$

3. **POST-PROCESS** $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$

4. **RECOVER** $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$

5. **RECOVER** $\hat{\mathbf{x}} = \mathcal{D}(\mathcal{C}(\mathbf{y}))$

Experimental Results

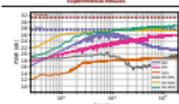


Fig. 1. Denoising via Neural Compression. Left: Denoising via Neural Compression (Denoise) and Denoising via Neural Compression (Denoise+Post-Process). Right: Denoising via Neural Compression (Denoise) and Denoising via Neural Compression (Denoise+Post-Process).



Method	SNR	MSE
DCT	0	0.0000
	10	0.0000
	20	0.0000
	30	0.0000
DFT	0	0.0000
	10	0.0000
	20	0.0000
	30	0.0000
DCT+Denoise	0	0.0000
	10	0.0000
	20	0.0000
	30	0.0000
DFT+Denoise	0	0.0000
	10	0.0000
	20	0.0000
	30	0.0000
DCT+Denoise+Post-Process	0	0.0000
	10	0.0000
	20	0.0000
	30	0.0000
DFT+Denoise+Post-Process	0	0.0000
	10	0.0000
	20	0.0000
	30	0.0000

REFERENCES

- David Lipp, Doreen, The denoising complex. Department of Statistics, Harvard University (2024).
- Thomas Blahut and Erik Dubrofski. The exponential distribution of rate-compressed source vectors. IEEE Transactions on Information Theory (2023).
- Ali Zafar, Xi Chen, and Shree N. Jayaram. Denoising via neural compression. arXiv preprint, 2023.
- Ali Zafar, Xi Chen, and Shree N. Jayaram. Denoising via neural compression: theoretical and algorithmic frameworks. arXiv preprint, 2024.

Outline

Denoising Problem

Compression-based denoising

Denoising via Neural Compression

Final Remarks

Denoising Problem

- Signal of interest: $\mathbf{x} = (x_1, \dots, x_n) \in \mathbb{R}_+^n$
- Observations: $\mathbf{y} = (y_1, \dots, y_n)$
 - conditionally independent given \mathbf{x}

$$\mathbf{y} \sim \prod_{i=1}^n p(y_i | x_i).$$

Denosing Problem

- Signal of interest: $\mathbf{x} = (x_1, \dots, x_n) \in \mathbb{R}_+^n$
- Observations: $\mathbf{y} = (y_1, \dots, y_n)$
 - conditionally independent given \mathbf{x}

$$\mathbf{y} \sim \prod_{i=1}^n p(y_i | x_i).$$

Denosing problem: Estimate \mathbf{x} from noisy observation \mathbf{y} .

- Key ingredient: Using the source structure

Denoising: An old problem, new challenges

- Denoising has a long history with a wide range of approaches
- Mature field under standard setting: Gaussian noise in natural images is well-understood
- Emerging imaging applications introduce **non-traditional sources** and **non-standard noise models**, requiring fresh solutions

NLM [Buades et al. 2005], BM3D [Dabov et al. 2007], DnCNN [Zhang et al. 2016], Restormer [Zamir et al. 2022], DIP [Ulyanov et al. 2017], Deep Decoder [Heckel et al. 2018], Noise2Void [Krull et al. 2019], Noise2Score [Kim et al. 2021], ...

Compression-based Denoising

- Clean signal \mathbf{x} is **structured**, and *compressible*
- Compression-based denoising idea:

Lossy compression of noisy signal at a distortion level adjusted based on noise power \equiv denoising

Compression-based Denoising

- Clean signal \mathbf{x} is **structured**, and *compressible*
- Compression-based denoising idea:
Lossy compression of noisy signal at a distortion level adjusted based on noise power \equiv denoising
- Key theoretical works on compression-based denoising:
 - Minimum Kolmogorov Complexity Estimator [Donoho, 2002]
 - Universal denoising of **discrete-valued** sources [Weissman et al., 2005]: universal lossy compression followed by suitable post-processing

Compression-based Denoising

- Clean signal \mathbf{x} is *structured*, and *compressible*
- Compression-based denoising idea:
Lossy compression of noisy signal at a distortion level adjusted based on noise power \equiv denoising
- Key theoretical works on compression-based denoising:
 - Minimum Kolmogorov Complexity Estimator [Donoho, 2002]
 - Universal denoising of *discrete-valued* sources [Weissman et al., 2005]: universal lossy compression followed by suitable post-processing
- Despite strong theoretical foundations, existing compression-based denoising methods i) often yield sub-optimal results in practice, 2) require access to pre-defined compression codes.

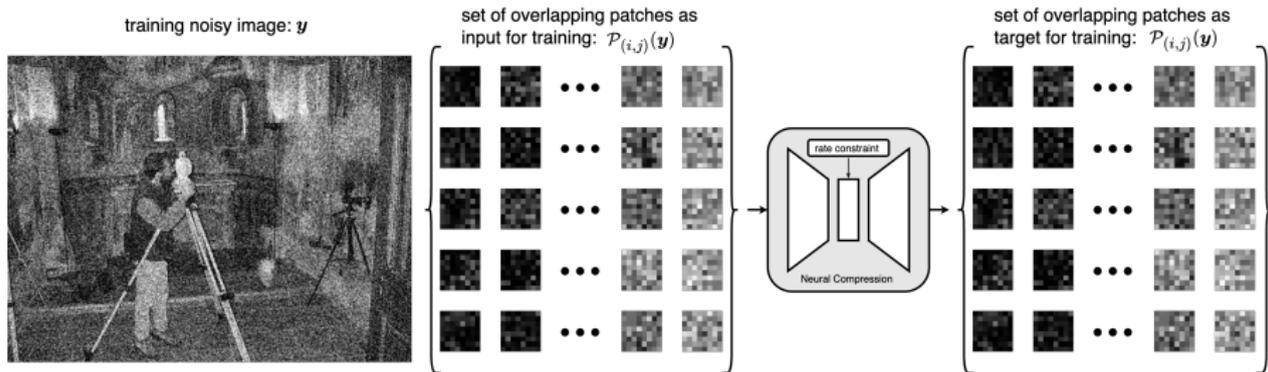
Our contribution

- Inspired by the recent success of neural compression methods, we introduce **DeCompress**:
 - a novel compression-based denoising method
- Key features of **DeCompress**:
 - **Unsupervised**: does not require clean-noisy image pairs for training
 - **Data-efficient**: Can operate with access to limited training data (only a single noisy image)
 - **High-performance**: Achieves state-of-the-art performance among zero-shot denoisers (for both Gaussian and Poisson noise)

DeCompress: Training phase

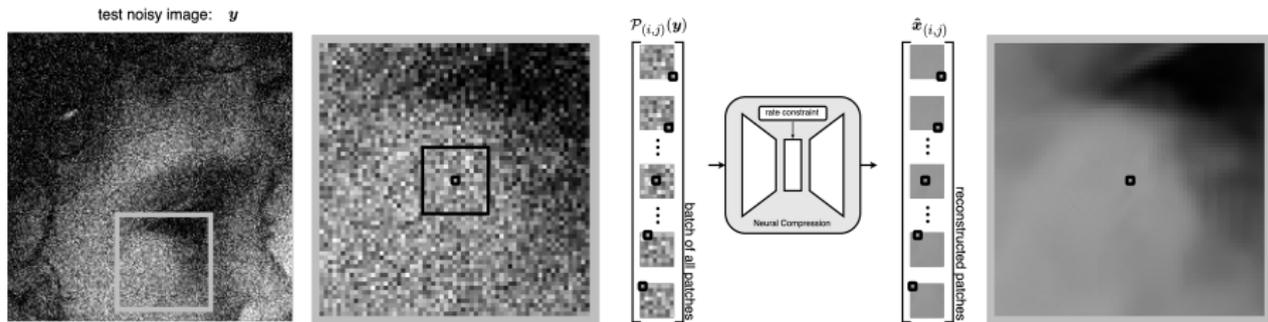
Step 1. Overlapping patches (e.g. 8×8) are extracted from a noisy image.

Step 2. A neural compression network is trained on extracted noisy patches.



DeCompress: Denoising phase

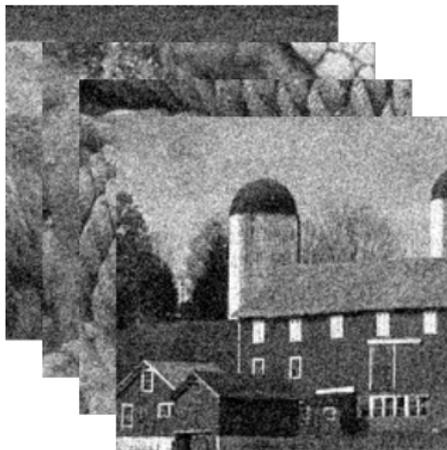
1. Extract overlapping patches from test image (`torch.nn.Unfold`).
2. Compress the batch of patches using the trained neural compressor (in parallel).
3. Combine and average all patches to reconstruct the denoised image (`torch.nn.Fold`).



DeCompress variants

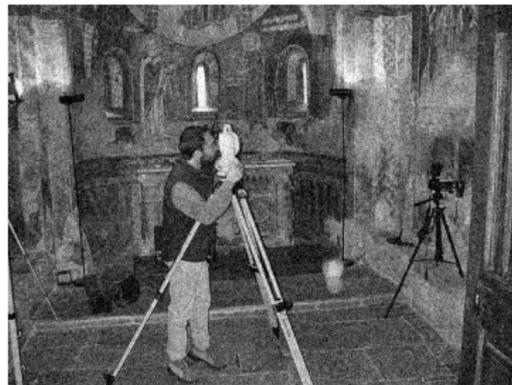
Two neural compression networks are trained for denoising without access to the ground-truth.

DeCompress (BSD400)



A set of 400 grayscale images of size 180×180 .

DeCompress (single)



An image from *Imagenet* validation set.
(ILSVRC2012_val_00000059.png)

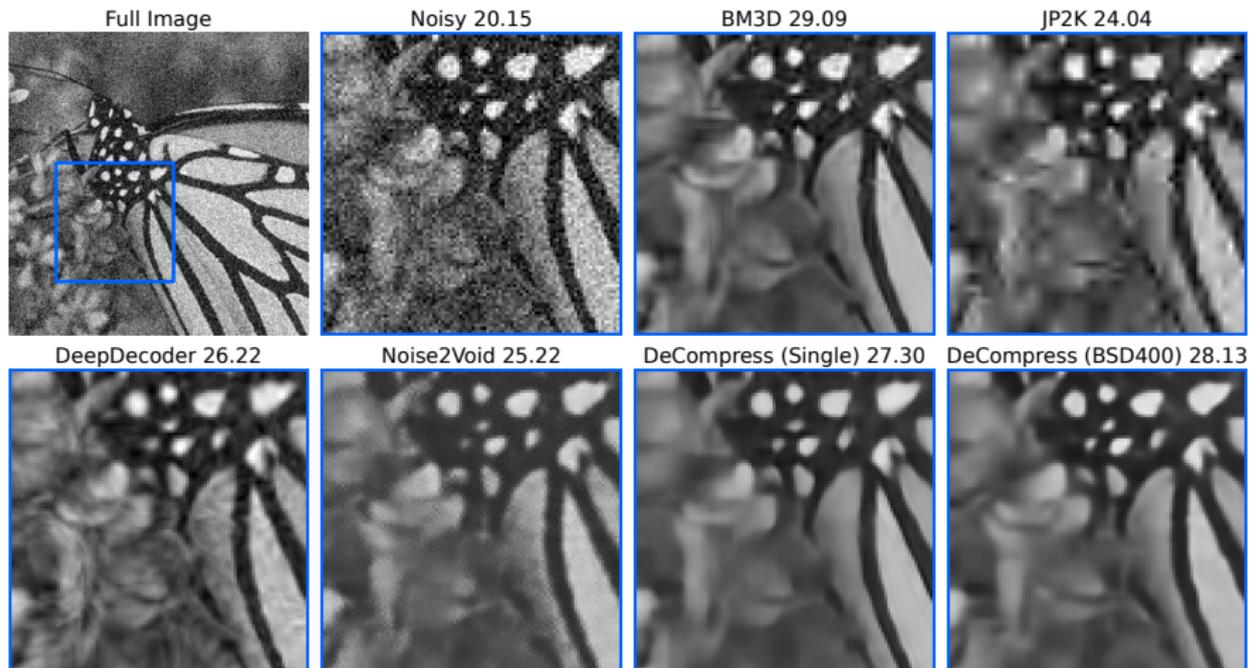
Denoising performance comparison

Denoising performance on *Set11* images.



	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$
noisy image	24.61	20.18	14.16
BM3D	32.20	29.76	26.51
JPEG2K	27.39	24.91	22.07
Deep Decoder	27.49	25.74	23.34
Noise2Void	26.88	24.77	19.88
DeCompress (single)	29.67	27.24	24.41
DeCompress (BSD400)	30.15	27.64	24.90

DeCompress: Sample visual comparison



Key takeaways & future directions

- Neural-compression-based denoising provides an effective denoising solution, especially in applications with little or no training data.
- Opens a new research direction: Exploring application of neural compression methods in solving extensive inverse problems.



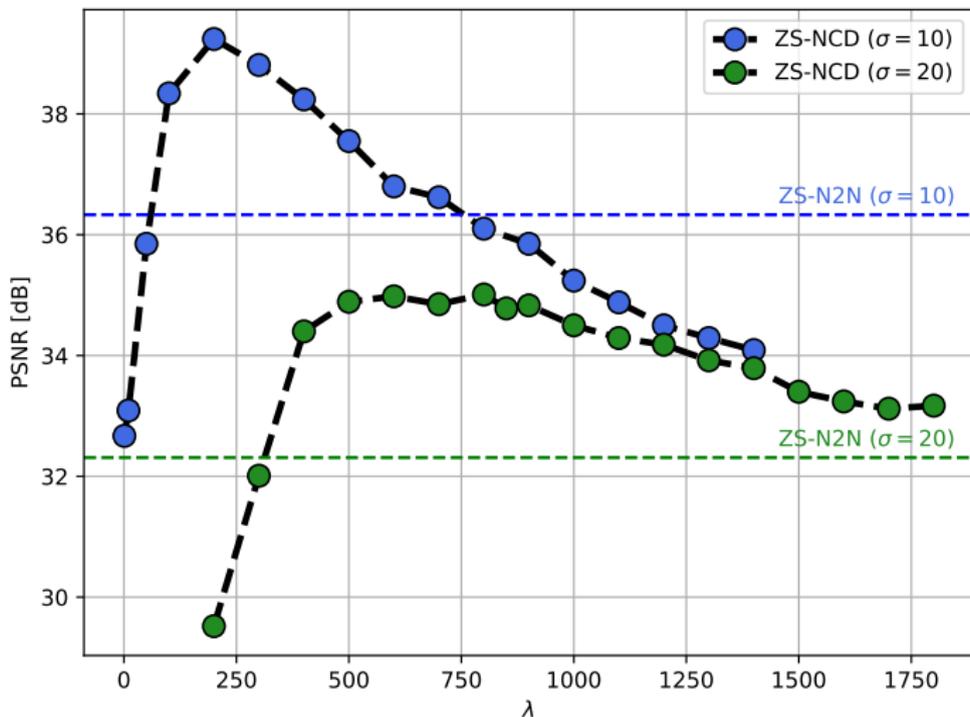
*Zero-shot Denoising via
Neural Compression:*
arXiv:2506.12693

References

-  Donoho, David Leigh [2002]. *The kolmogorov sampler*. Department of Statistics, Stanford University.
-  Weissman, Tsachy and Ordentlich, Erik [2005]. “The empirical distribution of rate-constrained source codes”. In: *IEEE transactions on information theory*.

A set of extra slides on zero-shot denoising via neural compression.

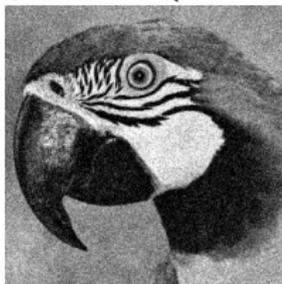
Effect of λ on denoising performance



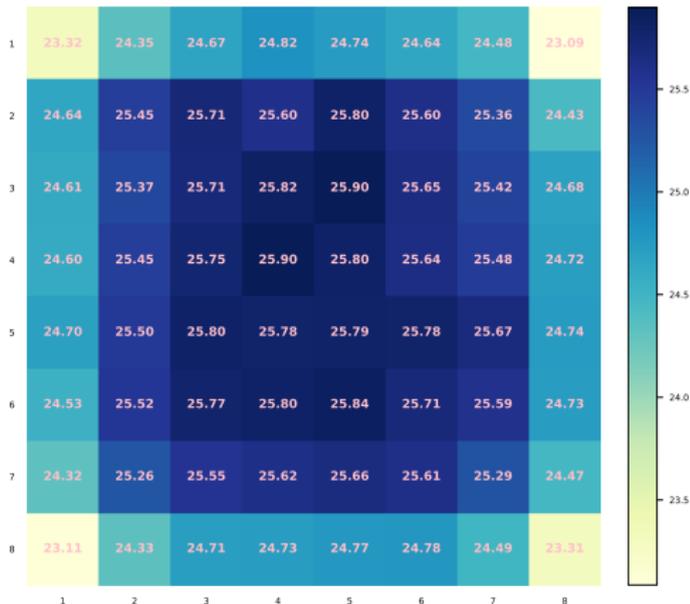
Effect of averaging

- Patch size: 8×8 .
- PSNR achieved with only keeping single compressed pixel (no averaging).

Noisy Parrot (20.15 dB)



DeCompress (28.15 dB)



Zero-shot Denoising via Neural Compression

Let $\mathcal{Q} \subset \mathbb{R}^n$ denote the signal class of interest, such as vectorized natural images of a fixed size. A **lossy compression code** for \mathcal{Q} is defined by an encoder-decoder pair (f, g) , $f : \mathcal{Q} \rightarrow \{1, \dots, 2^R\}$, and $g : \{1, \dots, 2^R\} \rightarrow \mathbb{R}^n$ characterized by: i) **Rate** R , and ii) **Distortion** δ :

$$\delta = \sup_{\mathbf{x} \in \mathcal{Q}} \frac{1}{n} \|\mathbf{x} - g(f(\mathbf{x}))\|_2^2.$$

The set of reconstructions forms the **codebook**:

$$\mathcal{C} = \{g(i) : i = 1, \dots, 2^R\} \subset \mathbb{R}^n.$$

Zero-shot Denoising via Neural Compression - Cont'd

We propose compression-based denoising as a **structured maximum likelihood (ML)** estimation. Given \mathbf{y} and a lossy compression code (f, g) for \mathcal{Q} , the compression-based ML denoiser solves

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{c} \in \mathcal{C}} \mathcal{L}(\mathbf{c}; \mathbf{y}),$$

where $\mathcal{L}(\mathbf{c}; \mathbf{y}) := -\sum_{i=1}^n \log p(y_i | c_i)$.

- Clean signals are more compressible than their noisy counterparts, by virtue of their structure.
- The most likely codeword from a clean-signal codebook naturally performs denoising.

Zero-shot Denoising via Neural Compression - Cont'd

1. **AWGN.** $\mathbf{y} = \mathbf{x} + \mathbf{z}$, where $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \sigma_z^2 I_n)$,

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{c} \in \mathcal{C}} \|\mathbf{y} - \mathbf{c}\|_2^2. \quad (1)$$

Denoising by projecting the noisy observation onto the nearest codeword.

2. **Poisson noise.** $y_i \sim \text{Poisson}(\alpha x_i)$ (low-light and photon-limited imaging scenarios)

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{c} \in \mathcal{C}} \sum_{i=1}^n (\alpha c_i - y_i \log c_i). \quad (2)$$

Theorem (AWGN)

Assume that $\mathbf{x} \in \mathcal{Q}$ and let (f, g) denote a lossy compression for \mathcal{Q} that operates at rate R and distortion δ . Consider $\mathbf{y} = \mathbf{x} + \mathbf{z}$, where $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \sigma_z^2 I_n)$. Let $\hat{\mathbf{x}}$ denote the output of the compression-based denoiser defined by (f, g) as in (1). Then,

$$\frac{1}{\sqrt{n}} \|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq \sqrt{\delta} + 2\sigma_z \sqrt{\frac{(2 \ln 2)R}{n}} (1 + 2\sqrt{\eta}),$$

with a probability larger than $1 - 2^{-\eta R+2}$.

Corollary (Sparse Signals)

Let \mathcal{Q}_n denote the set of k -sparse vectors in \mathbb{R}^n satisfying $\|\mathbf{x}\|_2 \leq 1$. Fix a parameter $\eta \in (0, 1)$, and suppose $\mathbf{y} = \mathbf{x} + \mathbf{z}$ where $\mathbf{z} \sim \mathcal{N}(0, \sigma_z^2 I_n)$. Then, there exists a family of compression codes such that, when used with the denoiser defined in (1), the estimate $\hat{\mathbf{x}}$ satisfies

$$\frac{1}{n\sigma_z^2} \|\hat{\mathbf{x}} - \mathbf{x}\|_2^2 \leq C \cdot \frac{k \log_2 n}{n} + \gamma_n,$$

with probability at least $1 - \frac{4}{(kn^{3k/2})^\eta}$. Here, $\gamma_n = o(1)$ and $C = 4 \ln 2 (1 + 2\sqrt{\eta})^2$.

Theorem (Poisson)

Assume that $\mathbf{x} \in \mathcal{Q}$ and let (f, g) denote a lossy compression for \mathcal{Q} that operates at rate R and distortion δ . Assume that for any $\mathbf{x} \in \mathcal{Q}$, $x_i \in (x_{\min}, x_{\max})$, where $0 < x_{\min} < x_{\max} < 1$. Assume that y_1, \dots, y_n are independent with $y_i \sim \text{Poisson}(\alpha x_i)$. Let $\hat{\mathbf{x}}$ denote the solution of (2). Let $C_1 = x_{\max}^5 / (x_{\min}^2)$ and $C_2 = \frac{x_{\max}^2}{x_{\min}^3} \beta \sqrt{\left(\frac{4}{\ln 2}\right)} (\sqrt{1 + \eta} + \sqrt{\eta})$. Then, with a probability larger than $1 - 2^{-\eta R + 2}$,

$$\frac{1}{n} \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 \leq C_1 \delta + C_2 \sqrt{\frac{R}{n\alpha}}. \quad (3)$$