Mathematical Foundations of Data Sciences

Gabriel Peyré
CNRS & DMA
École Normale Supérieure
gabriel.peyre@ens.fr
https://mathematical-tours.github.io
www.numerical-tours.com

September 11, 2024

Chapter 4

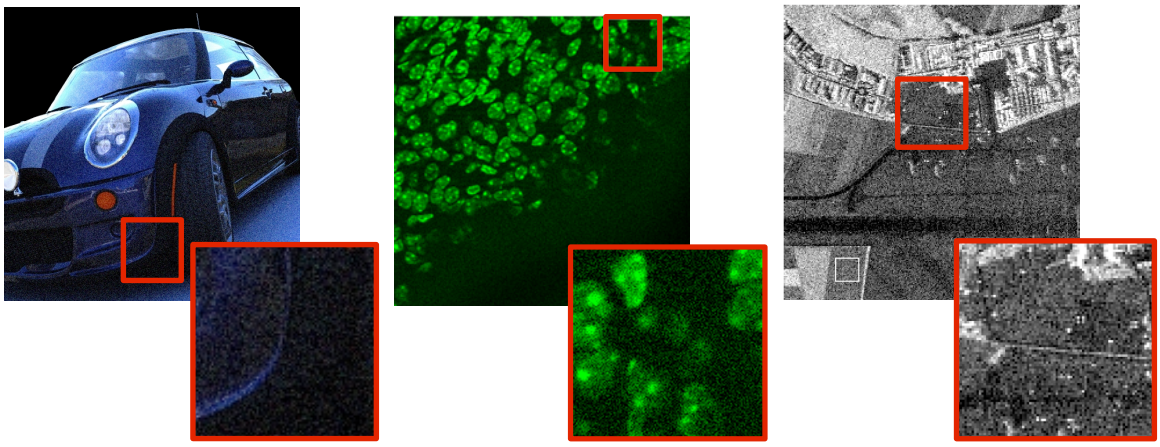
Denoising

Together with compression, denoising is the most important processing application, that is pervasive in almost any signal or image processing pipeline. Indeed, data acquisition always comes with some kind of noise, so modeling this noise and removing it efficiently is crucial.

4.1 Noise Modeling

4.1.1 Noise in Images

Image acquisition devices always produce some noise. Figure 4.1 shows images produced by different hardware, where the regularity of the underlying signal and the statistics of the noise is very different.



Digital camera Confocal imaging SAR imaging

Figure 4.1: Example of noise in different imaging device.

One should thus model both the acquisition process and the statistics of the noise to fit the imaging process. Then one should also model the regularity and geometry of the clean signal to choose a basis adapted to its representation. This chapter describes how thresholding methods can be used to perform denoising in some specific situations where the noise statistics are close to being Gaussian and the mixing operator is a sum or can be approximated by a sum.

Since noise perturbs discrete measurements acquired by some hardware, in the following, we consider only finite dimensional signal $f \in \mathbb{C}^N$.

4.1.2 Image Formation

Figure 4.2 shows an idealized view of the image formation process, that mixes a clean image f_0 with a noise w to obtain noisy observations $Y := f_0 \oplus w$, where $\oplus$ might for instance be a sum or a multiplication. Note that Y is thus here modeled as a random vector.

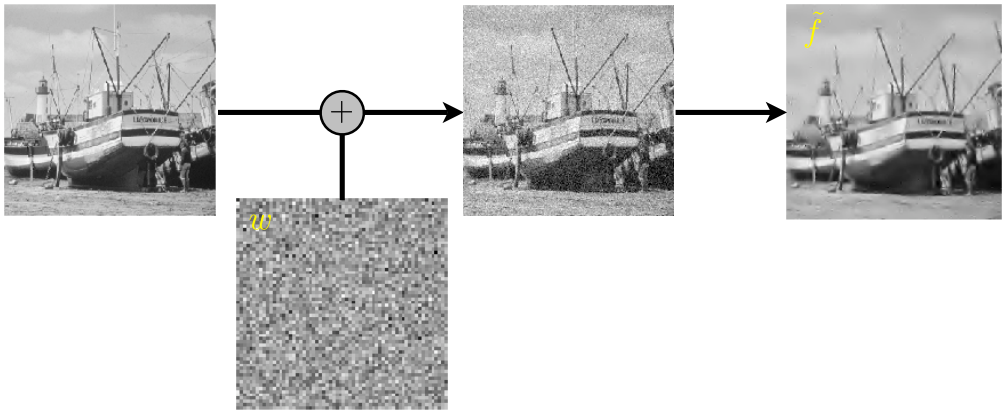


Figure 4.2: Image formation with noise modeling and denoising pipepline.

Statistical modeling considers w as a random vector with known distribution, while numerical computations are usually done on a single realization of this random vector, still denoted as w.

Additive Noise. The simplest model for such image formation consists of assuming that it is an additive perturbation of a clean signal f_0

$$Y := f_0 + w$$

where w is the noise residual. Statistical noise modeling assumes that w is a random vector, and in practice one only observes a realization of this vector. This modeling thus implies that the image f to be processed is also a random vector. Figure 4.3 and 4.4 show examples of noise addition to a clean signal and a clean image.

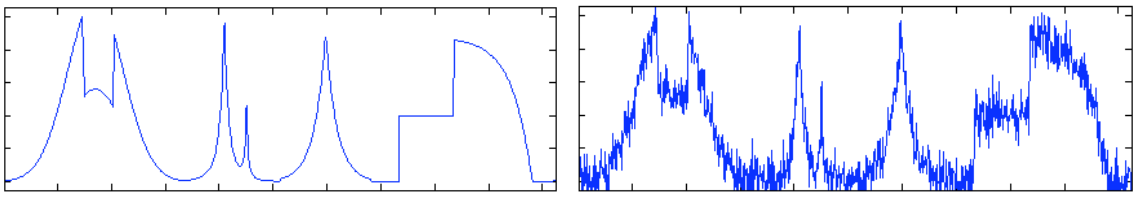


Figure 4.3: 1-D additive noise example.

The simplest noise model assumes that each entry w_n of the noise is a Gaussian random variable of variance σ^2, and that the w_n are independent, i.e. $w \sim \mathcal{N}(0, \mathrm{Id}_N)$. This is the white noise model.

Depending on the image acquisition device, one should consider different noise distributions, such as for instance uniform noise $w_n \in [-a, a]$ or Impulse noise

$$\mathbb{P}(w_n = u) \propto e^{-|u/\sigma|^\alpha} \quad \text{where} \quad \alpha < 2$$

In many situations, the noise perturbation is not additive, and for instance its intensity might depend on the intensity of the signal. This is the case with Poisson and multiplicative noises considered in Section 4.4.

f_0 $\qquad$ $y \sim Y = f_0 + w$

Figure 4.4: 2-D additive noise example.

4.1.3 Denoiser

A denoiser (also called estimator) is an estimation $\tilde{f} = \mathcal{D}(Y)$ (we use both notations in the following) of f_0 computed from the observation Y alone so that $\mathcal{D}$ is a deterministic function. It is thus also a random vector that depends on the noise w. Since Y is a random vector of mean f_0, the numerical denoising process corresponds to the estimation of the mean of a random vector from a single realization. Figure 4.5 shows an example of denoising.

The quality of a denoiser is measured using the average mean square risk $\mathcal{D}_w(\|f_0 - \tilde{f}\|^2)$. where $\mathcal{D}_w$ is the expectation (averaging) with respect to the noise w. Since f_0 is unknown, this corresponds to a theoretical measure of performance, that is bounded using a mathematical analysis. In the numerical experiments, one observes a single realization $y \sim Y = f_0 + w$, and the performance is estimated from this single denoising using the SNR $\mathrm{SNR}(\mathcal{D}(y), f_0)$, where

$$\mathrm{SNR}(f, g) := -20 \log_{10}(\|f - g\|/\|g\|).$$

The SNR is expressed in "decibels", denoted dB. This measure of performance requires the knowledge of the clean signal f_0, and should thus only be considered as an experimentation tool, that might not be available in a real-life denoising scenario where clean data are not available. Furthermore, the use of an ℓ^2 measure of performance is questionable, and one should also observe the result to judge of the visual quality of the denoising.

4.2 Linear Denoising using Filtering

4.2.1 Translation Invariant Estimators

A linear estimator $\mathcal{D}(Y) = \tilde{f}$ of f_0 depends linearly on Y, so that $\mathcal{D}(f+g) = \mathcal{D}(f) + \mathcal{D}(g)$. A translation invariant estimator commutes with translation, so that $\mathcal{D}(f_\tau) = \mathcal{D}(f)_\tau$, where $f_\tau(t) = f(t - \tau)$ (we assume periodic boundary conditions, say in 1-D or 2-D). Such a denoiser can always be written as a filtering

$$\mathcal{D}(f) = f \star h$$

where $h \in \mathbb{R}^N$ is a (low pass) filter. In practice, since the noise has zero mean, it satisfies

$$\sum_n h_n = \hat{h}_0 = 1$$

where $\hat{h}$ is the discrete Fourier transform.

f_0 y $\mathcal{D}(y)$

Figure 4.5: Left: clean image, center: noisy image, right: denoised image.

Figure 4.6 shows an example of denoising using a low pass filter.

The filtering strength is usually controlled the width s of h. A typical example is the (discretized) Gaussian filter

$$\forall\ -N/2 < i \leqslant N/2, \quad h_{s,i} = \frac{1}{Z_s} \exp\left(-\frac{i^2}{2s^2}\right) \tag{4.1}$$

where Z_s ensures that $\sum_i h_{s,i} = 1$ (low pass). Figure 4.6 shows the effect of Gaussian filtering over the spacial and Fourier domains.

Figure 4.7 shows the effect of low pass filtering on a signal and an image with an increasing filter width s. Linear filtering introduces a blur and are thus only efficient to denoise smooth signals and image. For signals and images with discontinuities, this blur deteriorates the signal. Removing a large amount of noise necessitates to also smooth significantly edges and singularities.

4.2.2 Optimal Filter Selection and Bias-Variance Tradeoff

The selection of an optimal filter is a difficult task. Its choice depends both on the regularity of the (unknown) data f_0 and the noise level σ. A simpler option is to optimize the filter width s among a parametric family of filters, such as for instance the Gaussian filters defined in (4.1).

The denoising error can be decomposed as

$$\|\tilde{f} - f_0\| \leqslant \|h_s \star f_0 - f_0\| + \|h_s \star w\|$$

The filter width s should be optimized to perform a tradeoff between removing enough noise ($\|h_s \star w\|$ decreases with s) and not smoothing too much the singularities (($\|h_s \star f_0 - f_0\|$ increases with s).

Figure (4.8) shows the oracle SNR performance, defined in (**??**).

Figure 4.9 and 4.10 show the results of denoising using the optimal filter width $s^\star$ that minimizes the SNR for a given noisy observation.

These optimal filtering appear quite noisy, and the optimal SNR choice is usually quite conservative. Increasing the filter width introduces a strong blurring that deteriorates the SNR, although it might look visually more pleasant.

4.2.3 Oracle denoiser

In order to select h, we will here "cheat" by assuming we have access to the ground trust f_0. Recall we consider the noise model $Y = f_0 + w$, where w is a Gaussian white noise, i.e., $w_k \sim \mathcal{N}(0, \sigma^2)$ and the w_k

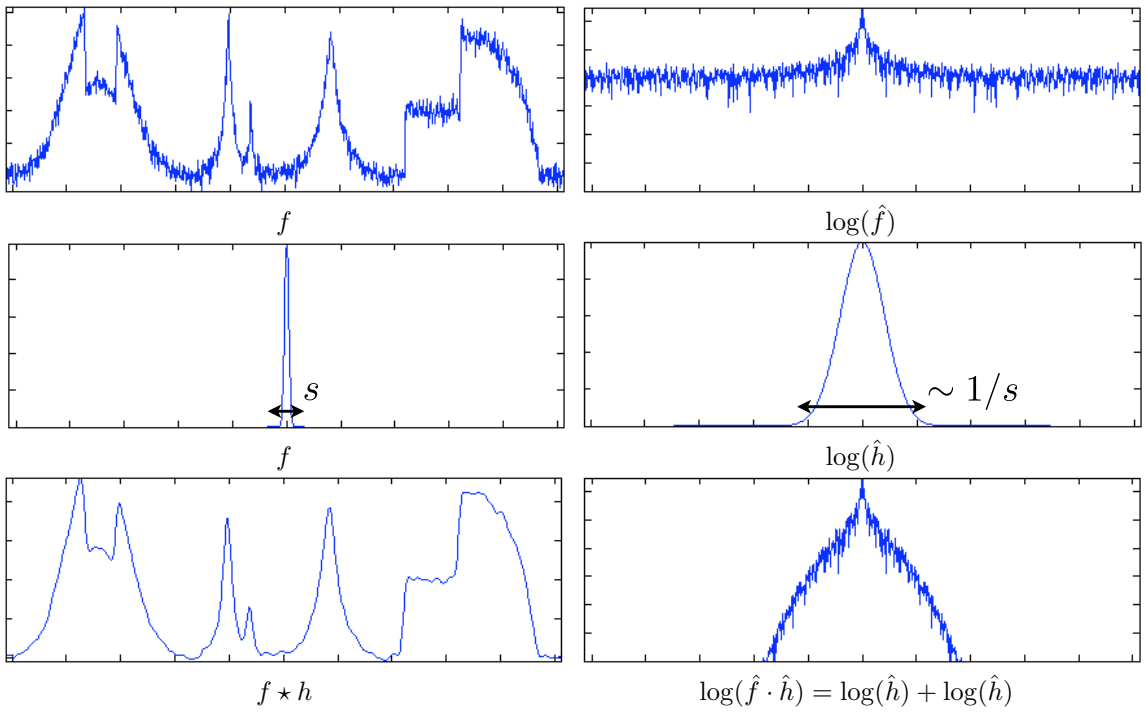


Figure 4.6: Denoising by filtering over the spacial (left) and Fourier (right) domains.

are i.i.d. Here f_0 is the (a priori unknown) ground trust. We consider an orthogonal basis $(\psi_k)_k$, which is typically the Fourier basis. We consider a linear denoiser which is diagonal in this basis,

$$\tilde{f} \coloneqq \sum_k \lambda_k \langle Y, \psi_k \rangle \psi_k. \tag{4.2}$$

For the Fourier basis, this means $\tilde{f} = Y \star h$ where $\tilde{h} = \lambda$. Note that $\tilde{f}$ is a random vector. Here we will "cheat" and assume access to f_0 to select λ, but we we still impose that $\tilde{f}$ is a linear function of Y of the above form, so the problem is non-trivial and still quite informative (we cannot just use $\tilde{f} = f_0$).

Denoting $c_k \coloneqq \langle f_0, \psi_k \rangle$ (for the Fourier basis, $c = \hat{f}_0$), the expected risk of this denoiser is

$$\mathbb{E}_w \|\tilde{f} - f_0\|^2 = \sum_k \mathbb{E}_w \left| \langle \tilde{f} - f_0, \psi_k \rangle \right|^2 = \sum_k \mathbb{E}_w \left| \lambda_k \left(c_k + \langle w, \psi_k \rangle \right) - c_k \right|^2 .$$

One then uses the fact that since $(\psi_k)_k$ is an orthonormal basis, $\langle w, \psi_k \rangle$ is also a Gaussian white noise of variance σ^2. By expanding the square and using $\mathbb{E}(\langle w, \psi_k \rangle) = 0$, we obtain

$$\mathbb{E}_w \|\tilde{f} - f_0\|^2 = \sum_k |c_k|^2 |\lambda_k - 1|^2 + |\lambda_k|^2 \sigma^2.$$

In practice, we do not know c_k. But if we assume we know it (in practice, one can make some rough assumptions about it), then the best possible denoising minimizes $\mathbb{E}_w \left(\|\tilde{f} - f_0\|^2 \right)$ with respect to λ, so that it solves for each k

$$\min_{\lambda_k} \left(|c_k|^2 |\lambda_k - 1|^2 + |\lambda_k|^2 \sigma^2 \right).$$

The solution thus satisfies

$$|c_k|^2 (\lambda_k - 1) + \lambda_k \sigma^2 = 0,$$

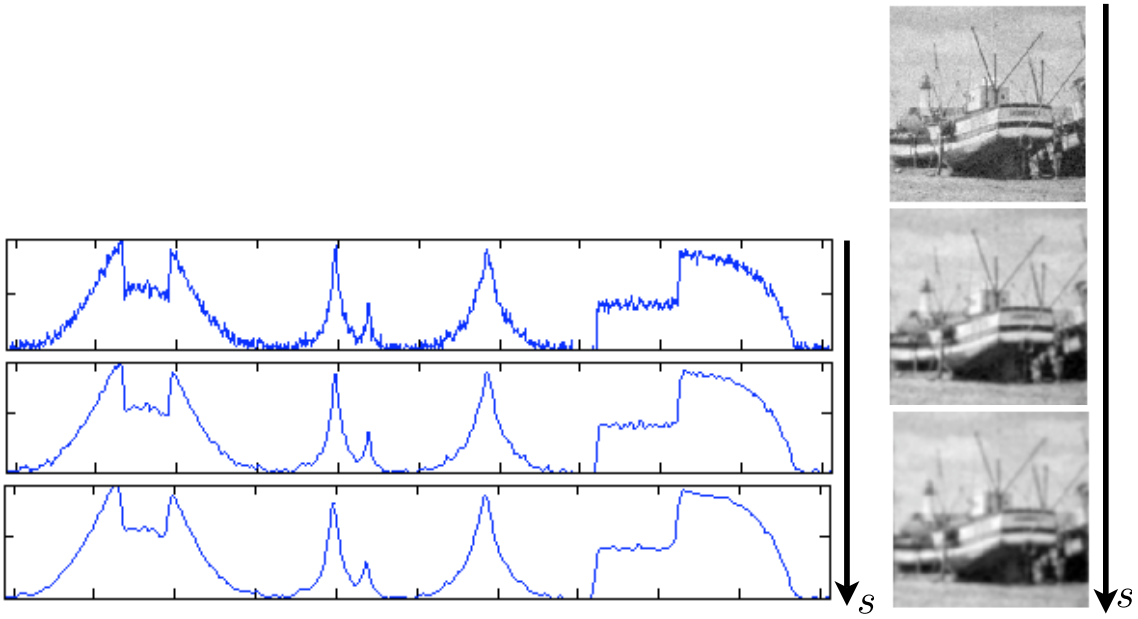


Figure 4.7: Denoising using a filter of increasing width s.

i.e.,

$$\lambda_k = \frac{|c_k|^2}{|c_k|^2 + \sigma^2} = \frac{1}{1 + \sigma^2/|c_k|^2}.$$

If $\sigma = 0$, one has $\lambda = 1$ (identity mapping). Assuming $|c_k|$ is decaying, as σ increases, the coefficients are more and more concentrated near 0.

4.2.4 Wiener Filter

Here we will "cheat" a bit less and assume also a random model for the signal f_0 which we assume to know. If one has a random model both for the noise $w \sim W$ and for the signal $f_0 \sim F$, one can derives an optimal filters in average over both the noise and the signal realizations. One further assumes that W and F are independent. We assume $\mathcal{D}$ has the form (4.2).

The optimal h thus minimizes

$$\mathbb{E}_{w,F} \|\mathcal{D}(F + w) - F\|^2$$

Since W and F are independent, the same computation as above caries over so that

$$\mathbb{E}_{w,F} \|\mathcal{D}(F + w) - F\|^2 \sum_k \alpha_k |\lambda_k - 1|^2 + |\lambda_k|^2 \sigma^2 \quad \text{where} \quad \alpha_k := \mathbb{E}_F |\langle F, \psi_k \rangle|^2$$

The optimal filter thus has coefficients

$$\lambda_k = \frac{1}{1 + \sigma^2/\alpha_k}. \tag{4.3}$$

With respect to the oracle filter of the previous section, the coefficient $|c_k|^2$ is replaced by the so-called power spectrum of the distribution $\alpha_k > 0$. This Filter is known as the Wiener filter.

When $(\psi_k)_k$ is the discrete Fourier basis, then this reads

$$\alpha_k = \mathbb{E}_F |\hat{F}_k|^2 = \mathbb{E}_F \mathcal{F}(F \star F(-\cdot))_k = \mathcal{F}(\mathbb{E}(F \star F(-\cdot))) = \hat{\eta}(F)$$

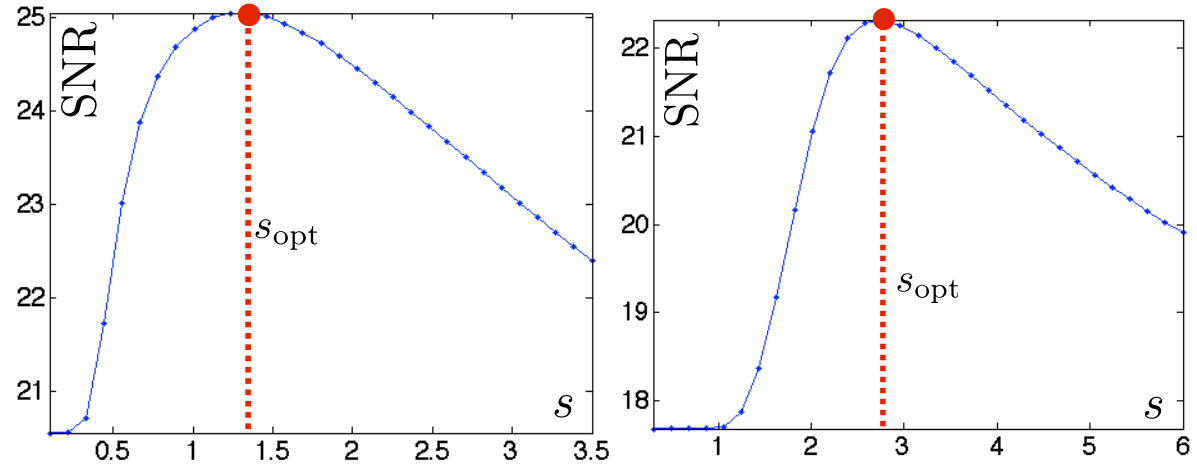


Figure 4.8: Curves of SNR as a function of the filtering width in 1-D (left) and 2-D (right).

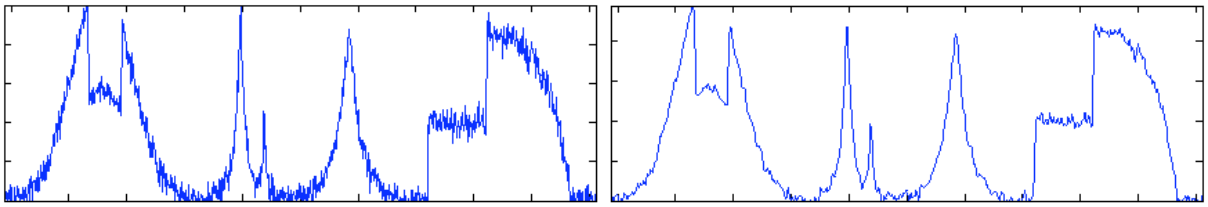


Figure 4.9: Noisy image (left) and denoising (right) using the optimal filter width.

where $\mathcal{F}(g) = \hat{g}$ is the Fourier transform where $F(-\cdot) = (F_{-k})$ is the reversed signal, which has Fourier transform $\hat{F}^*$ (complex conjugate). Here $\eta(F) \in \mathbb{R}^N$ is the auto-correlation of the random vector F,

$$\eta(F)_i := \mathbb{E}(F \star F(-\cdot))_i = \mathbb{E}(\sum_k F_i F_{i+j}).$$

It measures the average correlation between pairs of coordinates separated by j.

We have assumed here that $\mathcal{D}$ is diagonal in ψ_m, having form (4.2). Alternatively, one can prove that if F is stationary, i.e. its laws is translation invariance, i.e. the law of $F(\cdot - \tau)$ is equal to the law of F, then the Wiener filter (4.3) for the Fourier basis is optimal among all possible linear denoiser.

In practice, one can try to approximate α_k by $|\hat{y}_k|^2$ where y is a single realization, but unfortunately, one can be shown that this estimation is very bad, especially for large frequencies k, which typically needs to be truncated. This is called "empirical Wiener".

4.2.5 Denoising and Linear Approximation

In order to study linear (and also non-linear, see the section below) denoising without assuming a random signal model, one should use approximation theory as studied in Chapter **??**. We thus consider an ortho-basis $\mathcal{B} = (\psi_m)_m$ of $\mathbb{R}^N$, and consider a simple denoising obtained by keeping only the M first term elements of the approximation of the noisy observation in $\mathcal{B}$

$$\mathcal{D}(f) = \tilde{f} \stackrel{\text{def.}}{=} \sum_{m=1}^{M} \langle f, \psi_m \rangle \psi_m. \tag{4.4}$$

Figure 4.10: Noisy image (left) and denoising (right) using the optimal filter width.

This is a linear projection on the space spanned by $(\psi_m)_{m=1}^M$. This denoising scheme is thus parameterized by some integer $M > 0$, increasing M increases the denoising strength. For instance, when $\mathcal{B}$ is the discrete Fourier basis, this corresponds to an ideal low-pass filter against a (discretized) Dirichlet kernel. This corresponds to a special case of the linear denoiser considered in (4.2) where the coefficients λ_m are binary $\lambda_m \in \{0, 1\}$. This simplification is still quite expressive (in particular, it can be shown that in the setup considered in (3), this type of filter is still almost optimal) while not necessitating a strong knowledge on the ground trust f_0 (in particular, we do not assume it is drawn from a random distribution F).

Note also that we do here all the proof in finite dimension N, but all this construction and proof does not depend on N, so it works in infinite dimension (see below for more comments on this).

Theorem 3. *We assume that $f_0 \in \mathbb{R}^N$ has a linear approximation error decay that satisfies*

$$\forall\, M, \quad \|f_0 - f_{0,M}^{lin}\|^2 \leqslant CM^{-2\beta} \quad \textit{where} \quad f_{0,M}^{lin} \overset{\text{def.}}{=} \sum_{m=1}^{M} \langle f_0,\, \psi_m\rangle \psi_m$$

for some constant C. Then the linear denoising error using (4.4) satisfies

$$\mathcal{D}(\|f_0 - \tilde{f}\|^2) \leqslant 2C^{\frac{1}{2\beta+1}} \sigma^{2-\frac{1}{\beta+1/2}},$$

when choosing

$$M = C^{\frac{1}{2\beta+1}} \sigma^{-\frac{2}{2\beta+1}}. \tag{4.5}$$

Proof. One has, thanks to the ortho-normality of $(\psi_m)_m$

$$\begin{aligned}
\mathcal{D}(\|f_0 - \tilde{f}\|^2) &= \mathcal{D}(\sum_m \langle f_0 - \tilde{f},\, \psi_m\rangle^2) = \mathcal{D}(\sum_{m=1}^{M} \langle f_0 - f,\, \psi_m\rangle^2 + \sum_{m>M} \langle f_0,\, \psi_m\rangle^2) \\
&= \mathcal{D}\left(\sum_{m=1}^{M} \langle w,\, \psi_m\rangle^2\right) + \sum_{m>M} \langle f_0,\, \psi_m\rangle^2 = M\sigma^2 + \|f_0 - f_{0,M}^{\text{lin}}\|^2 \\
&\leqslant M\sigma^2 + CM^{-2\beta}.
\end{aligned}$$

Here we use the fundamental fact that $(\langle w,\, \psi_m\rangle)_m$ is also $\mathcal{N}(0, \sigma^2 \mathrm{Id}_N)$. Choosing M such that $M\sigma^2 = CM^{-2\beta}$, i.e. $M = C^{\frac{1}{2\beta+1}} \sigma^{-\frac{2}{2\beta+1}}$ leads to

$$\mathcal{D}(\|f_0 - \tilde{f}\|^2) = 2CM^{-2\beta} = 2CC^{-\frac{2\beta}{2\beta+1}} \sigma^{\frac{4\beta}{2\beta+1}} = 2C^{\frac{1}{2\beta+1}} \sigma^{2-\frac{1}{\beta+1/2}}.$$

□

There are several important remark regarding this simple but important result:

- Thanks to the decay of the linear approximation error, the denoising error $\mathcal{D}(\|f_0 - \tilde{f}\|^2)$ is bounded *independently* of the sampling size N, although the input noise level $\mathcal{D}(\|w\|^2) = N\sigma^2$ growth with N.
- If the signal is well approximated linearly, i.e. if β is large, then the denoising error decays fast when the noise level σ drops to zero. The upper bound approaches the optimal rate σ^2 by taking β large enough.
- This theory is finite dimensional, i.e. this computation makes only sense when introducing some discretization step N. This is natural because random noise vectors of finite energy are necessarily finite dimensional. For the choice (4.5) to be realizable, one should however have $M \leqslant N$, i.e. $N \geqslant C^{\frac{1}{2\beta+1}} \sigma^{-\frac{2}{2\beta+1}}$. Thus N should increase when the noise diminish for the denoising effect to kick-in.
- Section **??** bounds the linear approximation error for infinite dimensional signal and image model. This theory can be applied provided that the discretization error is smaller than the denoising error, i.e. once again, one should use N large enough.

A typical setup where this denoising theorem can be applied is for the Sobolev signal and image model detailed in Section **??**. In the discrete setting, where the sampling size N is intended to grow (specially if σ diminishes), one can similarly consider a "Sobolev-like" model, and similarely as for Proposition **??**, this model implies a decay of the linear approximation error.

Proposition 13. *Assuming that*

$$\sum_{m=1}^{N} m^{2\alpha} |\langle f_0, \psi_m \rangle|^2 \leqslant C \tag{4.6}$$

then

$$\forall\, M, \quad \|f_0 - f_{0,M}^{lin}\|^2 \leqslant C M^{-2\alpha}$$

Proof.

$$C \geqslant \sum_{m=1}^{N} m^{2\alpha} |\langle f_0, \psi_m \rangle|^2 \geqslant \sum_{m>M} m^{2\alpha} |\langle f_0, \psi_m \rangle|^2 \geqslant M^{2\alpha} \sum_{m>M} |\langle f_0, \psi_m \rangle|^2 \geqslant M^{2\alpha} \|f_0 - f_{0,M}^{\text{lin}}\|^2.$$

□

If ψ_m is the discrete Fourier basis defined in (2.8), then this discrete Sobolev model (4.6) is equivalent to the continuous Sobolev model of Section **??**, up to a discretization error which tends to 0 as N increase. Choosing N large enough shows that smooth signals and image are thus efficiently denoised by a simple linear projection on the first M element of the Fourier basis.

4.3 Non-linear Denoising using Thresholding

4.3.1 Hard Thresholding

We consider an orthogonal basis $\{\psi_m\}_m$ of $\mathbb{C}^N$, for instance a discrete wavelet basis. The noisy coefficients satisfy

$$\langle f, \psi_m \rangle = \langle f_0, \psi_m \rangle + \langle w, \psi_m \rangle. \tag{4.7}$$

Since a Gaussian white noise is invariant under an orthogonal transformation, $\langle w, \psi_m \rangle$ is also a Gaussian white noise of variance σ^2. If the basis $\{\psi_m\}_m$ is efficient to represent f_0, then most of the coefficients $\langle f_0, \psi_m \rangle$ are close to zero, and one observes a large set of small noisy coefficients, as shown on Figure 4.11. This idea of using thresholding estimator for denoising was first systematically explored by Donoho and Jonhstone [1].

A thresholding estimator removes these small amplitude coefficients using a non-linear hard thresholding

$$\tilde{f} = \sum_{|\langle f, \psi_m \rangle| > T} \langle f, \psi_m \rangle \psi_m = \sum_m S_T(\langle f, \psi_m \rangle) \psi_m.$$

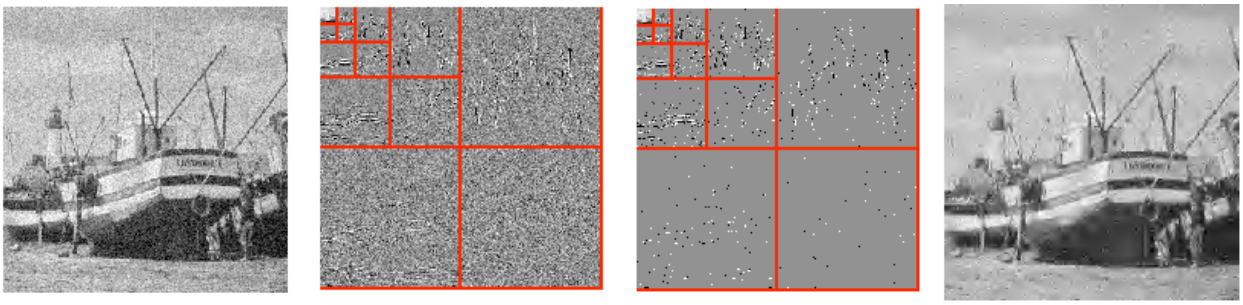


Figure 4.11: Denoising using thresholding of wavelet coefficients.

where S_T is defined in (**??**). This corresponds to the computation of the best M-term approximation $\tilde{f} = f_M$ of the noisy function f. Figure 4.11 shows that if T is well chose, this non-linear estimator is able to remove most of the noise while maintaining sharp features, which was not the case with linear filtering estimatiors.

4.3.2 Soft Thresholding

We recall that the hard thresholding operator is defined as

$$S_T(x) = S_T^0(x) = \begin{cases} x & \text{if } |x| > T, \\ 0 & \text{if } |x| \leqslant T. \end{cases} \tag{4.8}$$

This thresholding performs a binary decision that might introduces artifacts. A less aggressive nonlinearity is the soft thresholding

$$S_T^1(x) = \max(1 - T/|x|, 0)x. \tag{4.9}$$

Figure 4.12 shows the 1-D curves of these 1-D non-linear mapping.

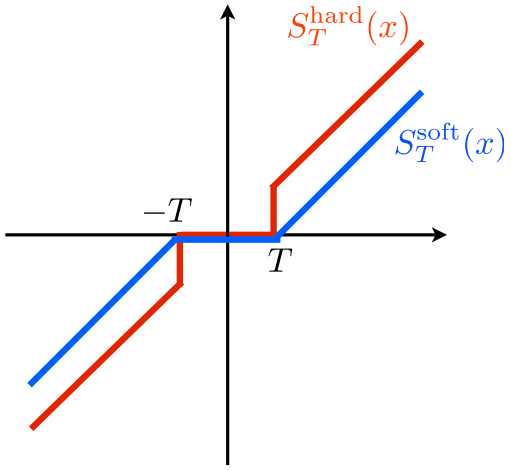


Figure 4.12: Hard and soft thresholding functions.

For $q = 0$ and $q = 1$, these thresholding defines two different estimators

$$\tilde{f}^q = \sum_m S_T^q(\langle f, \psi_m \rangle)\psi_m \tag{4.10}$$

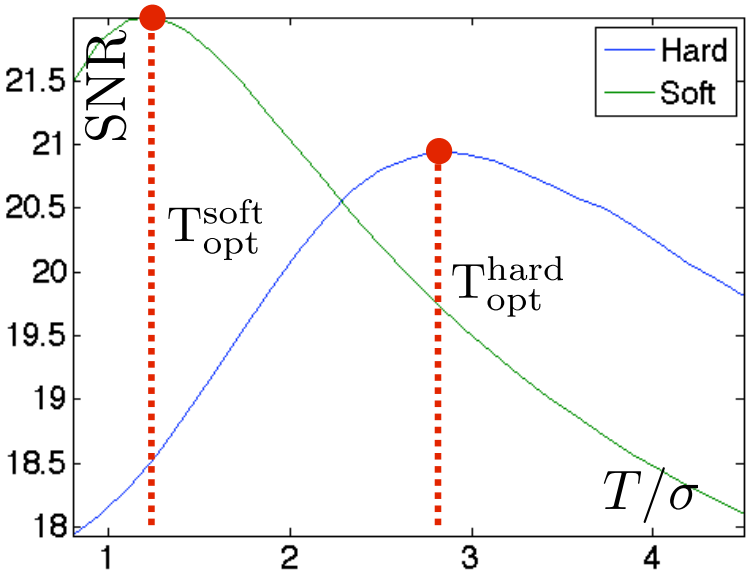


Figure 4.13: Curves of SNR with respect to T/σ for hard and soft thresholding.

Coarse scale management. The soft thresholded S^1_T introduces a bias since it diminishes the value of large coefficients. For wavelet transforms, it tends to introduces unwanted low-frequencies artifacts by modifying coarse scale coefficients. If the coarse scale is 2^{j_0}, one thus prefers not to threshold the coarse approximation coefficients and use, for instance in 1-D,

$$\tilde{f}^1 = \sum_{0 \leqslant n < 2^{-j_0}} \langle f, \varphi_{j_0,n} \rangle \varphi_{j_0,n} + \sum_{j=j_0}^{0} \sum_{0 \leqslant n < 2^{-j}} S^1_T(\langle f, \psi_{j_0,n} \rangle) \psi_{j_0,n}.$$

Empirical choice of the threshold. Figure 4.13 shows the evolution of the SNR with respect to the threshold T for these two estimators, for a natural image f_0. For the hard thresholding, the best result is obtained around $T \approx 3\sigma$, while for the soft thresholding, the optimal choice is around $T \approx 3\sigma/2$. These results also shows that numerically, for thresholding in orthogonal bases, soft thresholding is slightly superior than hard thresholding on natural signals and images.

Although these are experimental conclusions, these results are robust across various natural signals and images, and should be considered as good default parameters.

4.3.3 Minimax Optimality of Thresholding

Sparse coefficients estimation. To analyze the performance of the estimator, and gives an estimate for the value of T, we first assumes that the coefficients

$$a_{0,m} = \langle f_0, \psi_m \rangle \in \mathbb{R}^N$$

are sparse, meaning that most of the $a_{0,m}$ are zero, so that its ℓ^0 norm

$$\|a_0\|_0 = \#\{m \,;\, a_{0,m} \neq 0\}$$

is small. As shown in (4.7), noisy coefficients

$$\langle f, \psi_m \rangle = a_m = a_{0,m} + z_m$$

20.9dB 21.8dB

Figure 4.14: Comparison of hard (left) and soft (right) thresholding.

are perturbed with an additive Gaussian white noise of variance σ^2. Figure 4.15 shows an example of such a noisy sparse signal.

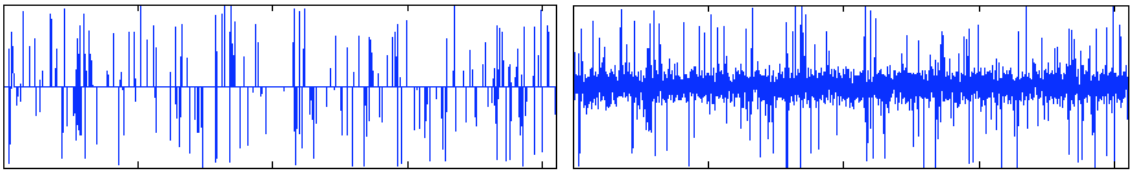


Figure 4.15: Left: sparse signal a, right: noisy signal.

Universal threshold value. If

$$\min_{m:a_{0,m}\neq 0} |a_{0,m}|$$

is large enough, then $\|f_0 - \tilde{f}\| = \|a_0 - S_T(a)\|$ is minimum for

$$T \approx \tau_N = \max_{0\leqslant m<N} |z_m|.$$

τ_N is a random variable that depends on N. One can show that its mean is $\sigma\sqrt{2\log(N)}$, and that as N increases, its variance tends to zero and τ_N is highly concentrated close to its mean. Figure 4.16 shows that this is indeed the case numerically.

Asymptotic optimality. Donoho and Jonhstone [1]Â have shown that the universal threshold $T = \sigma\sqrt{2\log(N)}$ is a good theoretical choice for the denoising of signals that are well approximated non-linearly in $\{\psi_m\}_m$. The obtain denoising error decay rate with σ can also be shown to be in some sense optimal.

Theorem 4. *We assume that $f_0 \in \mathbb{R}^N$ has a non-linear approximation error decay that satisfies*

$$\forall\, M, \quad \|f_0 - f_{0,M}^{nlin}\|^2 \leqslant CM^{-2\beta} \quad \textit{where} \quad f_{0,M}^{nlin} \stackrel{\text{def.}}{=} \sum_{r=1}^{M} \langle f_0,\, \psi_{m_r}\rangle \psi_{m_r}$$

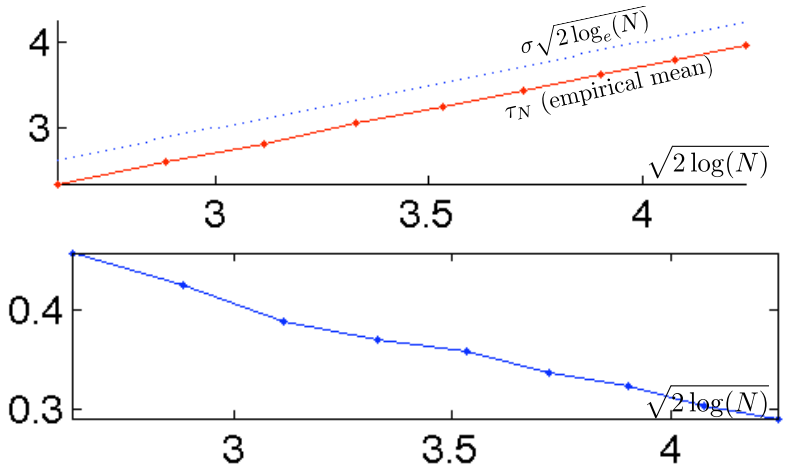


Figure 4.16: Empirical estimation of the mean of Z_n (top) and standard deviation of Z_n (bottom)

for some constant C, where here $(\langle f_0, \psi_{m_r}\rangle)_r$ are the coefficient sorted by decaying magnitude. Then the non-linear denoising error using (4.4) satisfies

$$\mathcal{D}(\|f_0 - \tilde{f}^q\|^2) \leqslant C' \ln(N) \sigma^{2-\frac{1}{\beta+1/2}},$$

for some constant C', when choosing $T = \sqrt{2\ln(N)}$, where $\tilde{f}^q$ is defined in (4.10) for $q \in \{0, 1\}$.

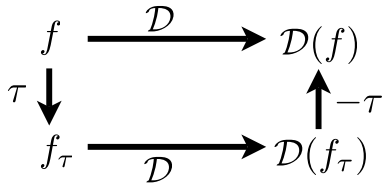
This universal threshold choice $T = \sqrt{2\ln(N)}$ is however very conservative since it is guaranteed to remove almost all the noise. In practice, as shown in Figure 4.14, better results are obtained on natural signals and images by using $T \approx 3\sigma$ and $T \approx 3\sigma/2$ for hard and soft thresholdings.

4.3.4 Translation Invariant Thresholding Estimators

Translation invariance. Let $f \mapsto \tilde{f} = \mathcal{D}(f)$ by a denoising method, and $f_\tau(x) = f(x-\tau)$ be a translated signal or image for $\tau \in \mathbb{R}^d$, ($d = 1$ or $d = 2$). The denoising is said to be translation invariant at precision Δ if

$$\forall\, \tau \in \Delta, \quad \mathcal{D}(f) = \mathcal{D}(f_\tau)_{-\tau}$$

where Δ is a lattice of $\mathbb{R}^d$. The denser Δ is, the more translation invariant the method is. This corresponds to the fact that $\mathcal{D}$ computes with the translation operator.



Imposing translation invariance for a fine enough set Δ is a natural constraint, since intuitively the denoising results should not depend on the location of features in the signal or image. Otherwise, some locations might be favored by the denoising process, which might result in visually unpleasant denoising artifacts.

For denoising by thresholding

$$\mathcal{D}(f) = \sum_m S_T(\langle f, \psi_m\rangle)\psi_m.$$

then translation invariance is equivalent to asking that the basis $\{\psi_m\}_m$ is translation invariant at precision Δ,

$$\forall\, m,\ \forall\, \tau \in \Delta,\ \exists m,\ \exists \lambda \in \mathbb{C}, \qquad (\psi_{m'})_\tau = \lambda \psi_m$$

where $|\lambda| = 1$.

The Fourier basis is fully translation invariant for $\Delta = \mathbb{R}^d$ over $[0,1]^d$ with periodic boundary conditions and the discrete Fourier basis is translation invariant for all interger translations $\Delta = \{0, \ldots, N_0 - 1\}^d$ where $N = N_0$ is the number of points in 1-D, and $N = N_0 \times N_0$ is the number of pixels in 2-D.

Unfortunately, an orthogonal wavelet basis

$$\{\psi_m = \psi_{j,n}\}_{j,n}$$

is not translation invariant both in the continuous setting or in the discrete setting. For instance, in 1-D,

$$(\psi_{j',n'})_\tau \notin \{\psi_{j,n}\} \quad \text{for} \quad \tau = 2^j/2.$$

Cycle spinning. A simple way to turn a denoiser Δ into a translation invariant denoiser is to average the result of translated images

$$\mathcal{D}_{\text{inv}}(f) = \frac{1}{|\Delta|} \sum_{\tau \in \Delta} \mathcal{D}(f_\tau)_{-\tau}. \tag{4.11}$$

One easily check that

$$\forall\, \tau \in \Delta, \quad \mathcal{D}_{\text{inv}}(f) = \mathcal{D}_{\text{inv}}(f_\tau)_{-\tau}$$

To obtain a translation invariance up to the pixel precision for a data of N samples, one should use a set of $|\Delta| = N$ translation vectors. To obtain a pixel precision invariance for wavelets, this will result in $O(N^2)$ operations.

Figure 4.17 shows the result of applying cycle spinning to an orthogonal hard thresholding denoising using wavelets, where we have used the following translation of the continuous wavelet basis $\Delta = \{0, 1/N, 2/N, 3/N\}^2$, which corresponds to discrete translation by $\{0,1,2,3\}^2$ on the discretized image. The complexity of the denoising scheme is thus 16 wavelet transforms. The translation invariance brings a very large SNR improvement, and significantly reduces the oscillating artifacts of orthogonal thresholding. This is because this artifacts pop-out at random locations when τ changes, so that the averaging process reduces significantly these artifacts.

Figure 4.18 shows that translation invariant hard thresholding does a slightly better job than translation invariant soft thresholding. The situation is thus reversed with respect to thresholding in an orthogonal wavelet basis.

Translation invariant wavelet frame. An equivalent way to define a translation invariant denoiser is to replace the orthogonal basis $\mathcal{B} = \{\psi_m\}$ by a redundant family of translated vectors

$$\mathcal{B}_{\text{inv}} = \{(\psi_m)_\tau\}_{m, \tau \in \Delta}. \tag{4.12}$$

One should be careful about the fact that $\mathcal{B}_{\text{inv}}$ is not any more an orthogonal basis, but it still enjoy a conservation of energy formula

$$\|f\|^2 = \frac{1}{|\Delta|} \sum_{m,\tau \in \Delta} |\langle f, (\psi_m)_\tau \rangle|^2 \quad \text{and} \quad f = \frac{1}{|\Delta|} \sum_{m,\tau \in \Delta} \langle f, (\psi_m)_\tau \rangle (\psi_m)_\tau.$$

This kind of redundant family are called tight frames.

One can then define a translation invariant thresholding denoising

$$\mathcal{D}_{\text{inv}}(f) = \frac{1}{|\Delta|} \sum_{m,\tau \in \Delta} S_T(\langle f, (\psi_m)_\tau \rangle)(\psi_m)_\tau. \tag{4.13}$$

21.8dB 23.4dB

Figure 4.17: Comparison of wavelet orthogonal soft thresholding (left) and translation invariant wavelet hard thresholding (right).

This denoising is the same as the cycle spinning denoising defined in (4.11).

The frame $\mathcal{B}_{\text{inv}}$ might contain up to $|\Delta||\mathcal{B}|$ basis element. For a discrete basis of signal with N samples, and a translation lattice of $|\Delta| = N$ vectors, it corresponds to up to N^2 elements in $\mathcal{B}_{\text{inv}}$. Hopefully, for a hierarchical basis such as a discrete orthogonal wavelet basis, one might have

$$(\psi_m)_\tau = (\psi_{m'})_{\tau'} \quad \text{for} \quad m \neq m' \quad \text{and} \quad \tau \neq \tau',$$

so that the number of elements in $\mathcal{B}_{\text{inv}}$ might be much smaller than N^2. For instance, for an orthogonal wavelet basis, one has

$$(\psi_{j,n})_{k2^j} = \psi_{j,n+k},$$

so that the number of basis elements is $|\mathcal{B}_{\text{inv}}| = N \log_2(N)$ for a 2-D basis, and $3N \log_2(N)$ for a 2-D basis. The fast translation invariant wavelet transform, also called "a trou" wavelet transform, computes all the inner products $\langle f, (\psi_m)_\tau \rangle$ in $O(N \log_2(N))$ operations. Implementing formula (4.13) is thus much faster than applying the cycle spinning (4.11) equivalent formulation.

Translation invariant wavelet coefficients are usually grouped by scales in $\log_2(N)$ (for $d = 1$) or by scales and orientations $3 \log_2(N)$ (for $d = 2$) sets of coefficients. For instance, for a 2-D translation invariant transform, one consider

$$\forall\, n \in \{0, \ldots, 2^j N_0 - 1\}^2,\ \forall\, k \in \{0, \ldots, 2^{-j}\}^2, \quad d_j^\omega[2^{-j}n + k] = \langle f, (\psi_{j,n})_{k2^j} \rangle$$

where $\omega \in \{V, H, D\}$ is the orientation. Each set d_j^ω has N coefficients and is a band-pass filtered version of the original image f, as shown on Figure 4.19.

Figure 4.20 shows how these set of coefficients are hard thresholded by the translation invariant estimator.

4.3.5 Exotic Thresholdings

It is possible to devise many thresholding nonlinearities that interpolate between the hard and soft thresholder. We present here two examples, but many more exist in the literature. Depending on the statistical distribution of the wavelet coefficients of the coefficients of f in the basis, these thresholders might produce slightly better results.

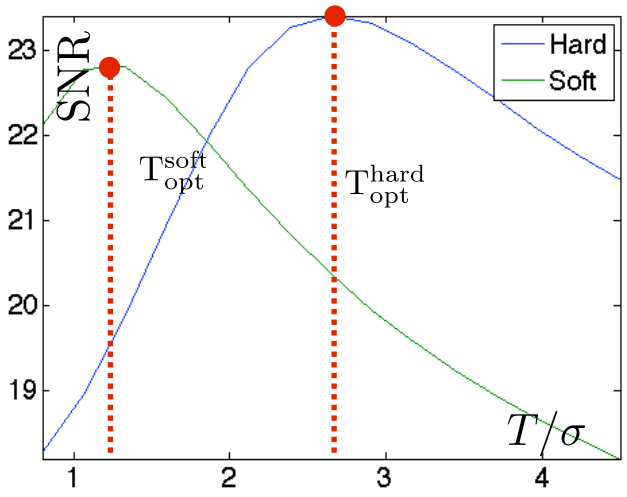


Figure 4.18: Curve of SNR with respect to T/σ for translation invariant thresholding.

Semi-soft thresholding. One can define a family of intermediate thresholder that depends on a parameter $\mu > 1$

$$S_T^\theta(x) = g_{\frac{1}{1-\theta}}(x) \quad \text{where} \quad g_\mu(x) = \begin{cases} 0 & \text{if} \quad |x| < T \\ x & \text{if} \quad |x| > \mu T \\ \text{sign}(x)\frac{|x|-T}{\mu-1} & \text{otherwise.} \end{cases}$$

One thus recovers the hard thresholding as S_T^0 and the soft thresholding as S_T^1. Figure 4.21 display an example of such a non-linearity.

Figure 4.22 shows that a well chosen value of μ might actually improves over both hard and soft thresholders. The improvement is however hardly noticeable visually.

Stein thresholding. The Stein thresholding is defined using a quadratic attenuation of large coefficients

$$S_T^{\text{Stein}}(x) = \max\left(1 - \frac{T^2}{|x|^2}, 0\right) x.$$

This should be compared with the linear attenuation of the soft thresholding

$$S_T^1(x) = \max\left(1 - \frac{T}{|x|}, 0\right) x.$$

The advantage of the Stein thresholder with respect to the soft thresholding is that

$$|S_T^{\text{Stein}}(x) - x| \to 0 \quad \text{whereas} \quad |S_T^1(x) - x| \to T,$$

where $x \to \pm\infty$. This means that Stein thresholding does not suffer from the bias of soft thresholding.

For translation invariant thresholding, Stein and hard thresholding perform similarly on natural images.

4.3.6 Block Thresholding

The non-linear thresholding method presented in the previous section are diagonal estimators, since they operate a coefficient-by-coefficient attenuation

$$\tilde{f} = \sum_m A_T^q(\langle f, \psi_m \rangle)\langle f, \psi_m \rangle \psi_m$$

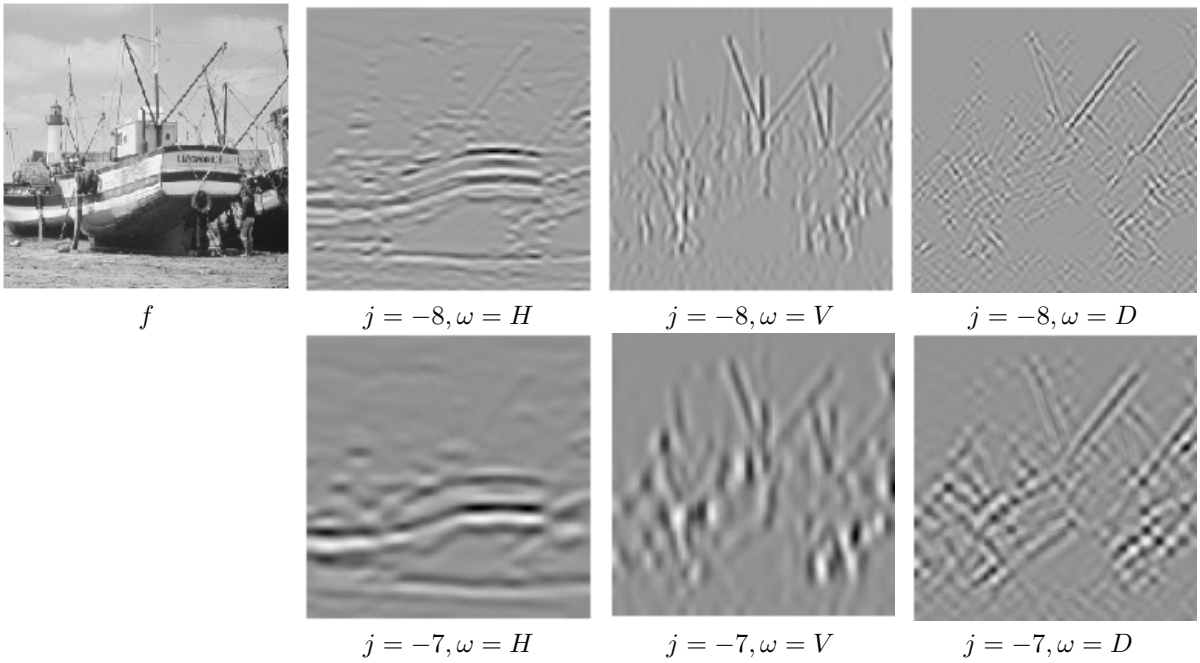


Figure 4.19: Translation invariant wavelet coefficients.

where

$$A_T^q(x) = \begin{cases} \max(1-x^2/T^2,0) & \text{for} \quad q=\text{Stein} \\ \max(1-|x|/T,0) & \text{for} \quad q=1 \text{ (soft)} \\ 1_{|x|>T} & \text{for} \quad q=0 \text{ (hard)} \end{cases}$$

Block thresholding takes advantage of the statistical dependancy of wavelet coefficients, by computing the attenuation factor on block of coefficients. This is especially efficient for natural images, where edges and geometric features create clusters of high magnitude coefficients. Block decisions also help to remove artifacts due to isolated noisy large coefficients in regular areas.

The set of coefficients is divided into disjoint blocks, and for instance for 2-D wavelet coefficients

$$\{(j,n,\omega)\}_{j,n,\omega} = \bigcup_k B_k,$$

where each B_k is a square of $s \times s$ coefficients, where the block size s is a parameter of the method. Figure 4.24 shows an example of such a block.

The block energy is defined as

$$B_k = \frac{1}{s^2} \sum_{m \in B_k} |\langle f,\, \psi_m \rangle|^2,$$

and the block thresholding

$$\tilde{f} = \sum_m S_T^{\text{block},q}(\langle f,\, \psi_m \rangle)\psi_m$$

makes use of the same attenuation for all coefficients within a block

$$\forall\, m \in B_k, \quad S_T^{\text{block},q}(\langle f,\, \psi_m \rangle) = A_T^q(E_k)\langle f,\, \psi_m \rangle.$$

for $q \in \{0, 1, \text{stein}\}$. Figure 4.24 shows the effect of this block attenuation, and the corresponding denoising result.

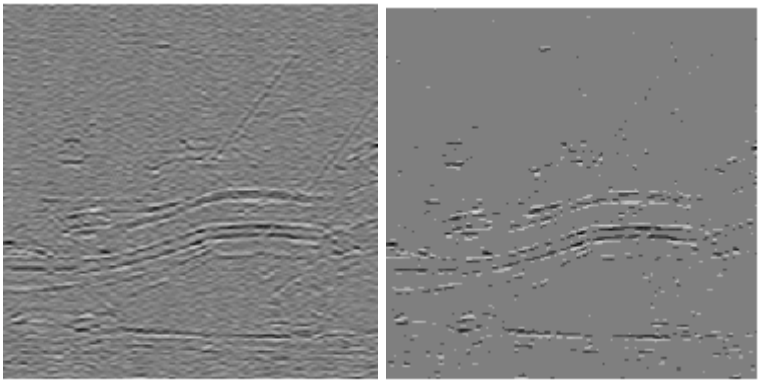


Figure 4.20: Left: translation invariant wavelet coefficients, for $j = -8, \omega = H$, right: tresholded coefficients.

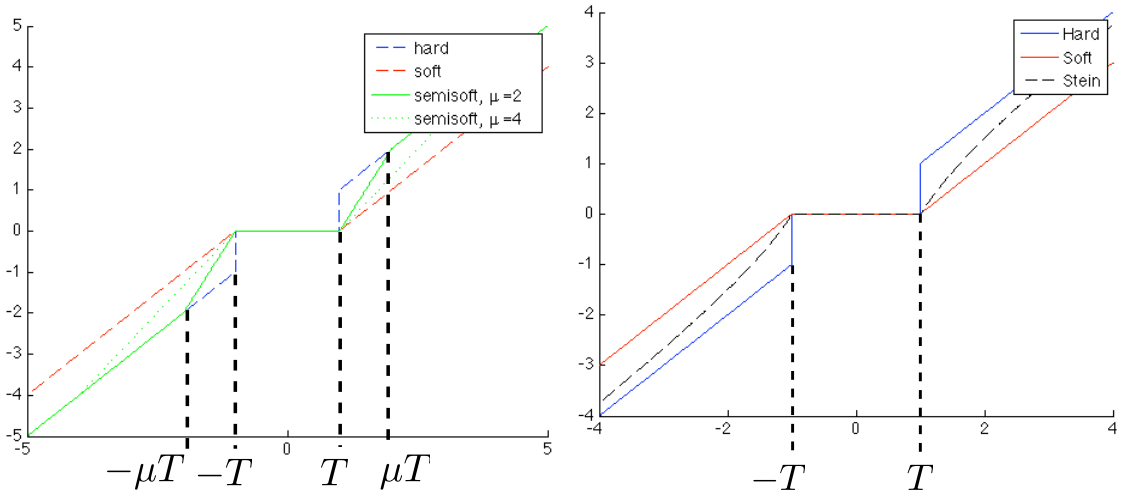


Figure 4.21: Left: semi-soft thresholder, right: Stein thresholder.

Figure 4.25, left, compares the three block thresholding obtained for $q \in \{0, 1, \text{stein}\}$. Numerically, on natural images, Stein block thresholding gives the best results. Figure 4.25, right, compares the block size for the Stein block thresholder. Numerically, for a broad range of images, a value of $s = 4$ works well.

Figure 4.26 shows a visual comparison of the denoising results. Block stein thresholding of orthogonal wavelet coefficients gives a result nearly as good as a translation invariant wavelet hard thresholding, with a faster algorithm. The block thresholding strategy can also be applied to wavelet coefficients in translation invariant tight frame, which produces the best results among all denoisers detailed in this book.

Code **??** implement this block thresholding.

One should be aware that more advanced denoisers use complicated statistical models that improves over the methods proposed in this book, see for instance [4].

4.4 Data-dependant Noises

For many imaging devices, the variance of the noise that perturbs $f_{0,n}$ depends on the value of $f_{0,n}$. This is a major departure from the additive noise formation model considered so far. We present here two popular examples of such non-additive models.

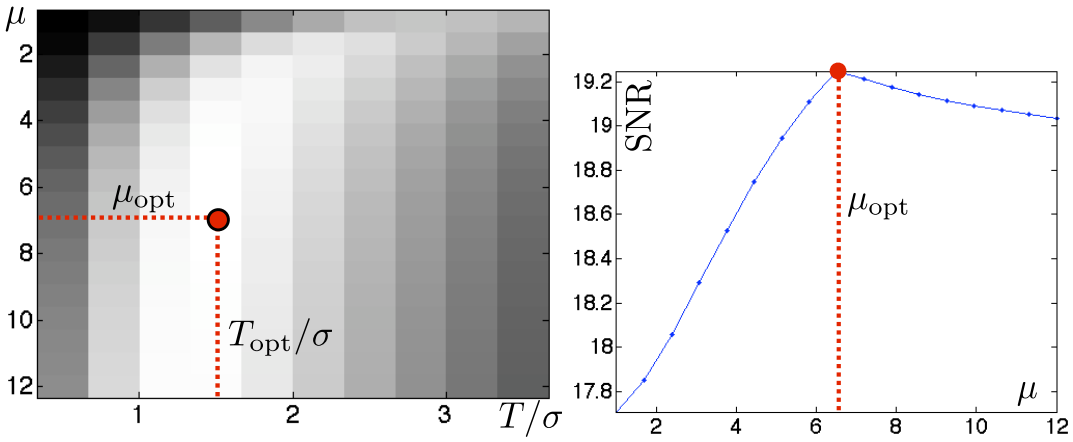


Figure 4.22: Left: image of SNR with respect to the parameters μ and T/σ, right: curve of SNR with respect to μ using the best T/σ for each μ.

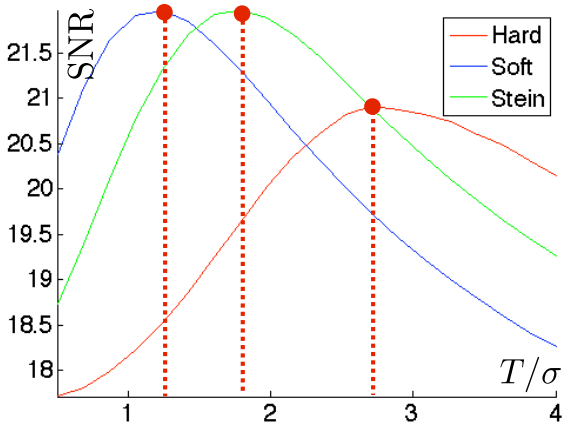


Figure 4.23: SNR curves with respect to T/σ for Stein threhsolding.

4.4.1 Poisson Noise

Many imaging devices sample an image through a photons counting operation. This is for instance the case in digital camera, confocal microscopy, TEP and SPECT tomography.

Poisson model. The uncertainty of the measurements for a quantized unknown image $f_{0,n} \in \mathbb{N}$ is then modeled using a Poisson noise distribution

$$f_n \sim \mathcal{P}(\lambda) \quad \text{where} \quad \lambda = f_{0,n} \in \mathbb{N},$$

and where the Poisson distribution $\mathcal{P}(\lambda)$ is defined as

$$\mathbb{P}(f_n = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

and thus varies from pixel to pixel. Figure 4.27 shows examples of Poisson distributions.

One has

$$\mathcal{D}(f_n) = \lambda = f_{0,n} \quad \text{and} \quad \text{Var}(f_n) = \lambda = f_{0,n}$$

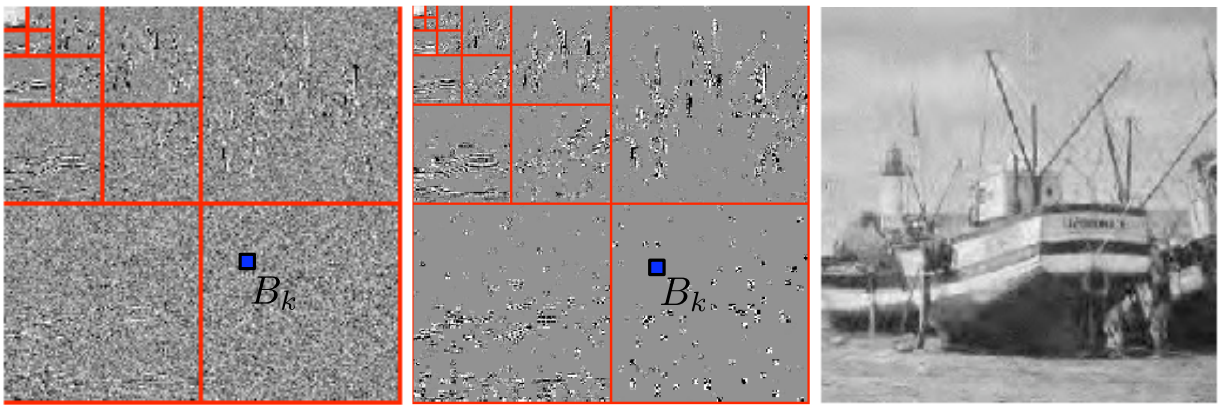


Figure 4.24: Left: wavelet coefficients, center: block thresholded coefficients, right: denoised image.

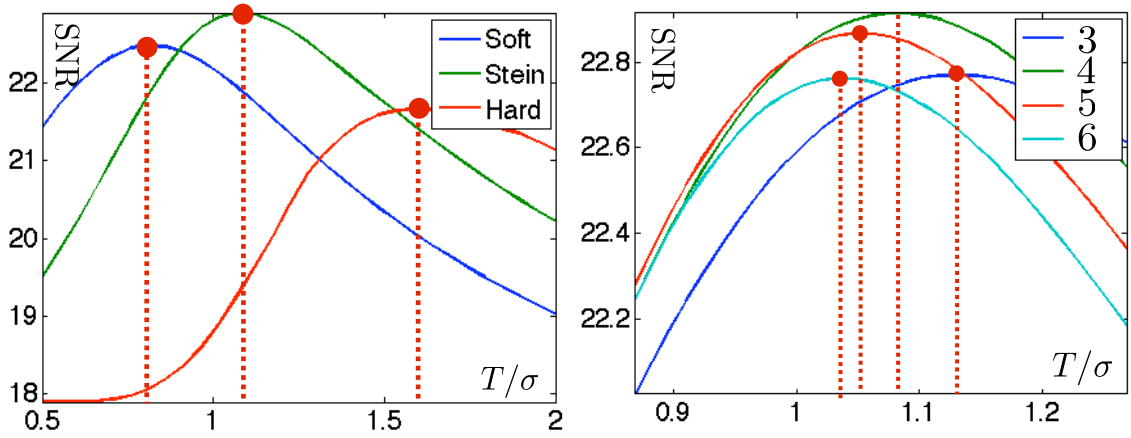


Figure 4.25: Curve of SNR with respect to T/σ (left) and comparison of SNR for different block size (right).

so that the denoising corresponds to estimating the mean of a random vector from a single observation, but the variance now depends on the pixel intensity. This shows that the noise level increase with the intensity of the pixel (more photons are coming to the sensor) but the relative variation $(f_n - f_{0,n})/f_{0,n}$ tends to zero in expectation when $f_{0,n}$ increases.

Figure 4.28 shows examples of a clean image f_0 quantized using different values of $\lambda_{\max}$ and perturbed with the Poisson noise model.

Variance stabilization. Applying thresholding estimator

$$\mathcal{D}(f) = \sum_m S_T^q(\langle f, \psi_m \rangle)\psi_m$$

to f might give poor results since the noise level fluctuates from point to point, and thus a single threshold T might not be able to capture these variations. A simple way to improve the thresholding results is to first apply a variance stabilization non-linearity $\varphi : \mathbb{R} \to \mathbb{R}$ to the image, so that $\varphi(f)$ is as close as possible to an additive Gaussian white noise model

$$\varphi(f) \approx \varphi(f_0) + w \tag{4.14}$$

where $w_n \sim \mathcal{N}(0, \sigma)$ is a Gaussian white noise of fixed variance σ^2.

SNR=23.4dB 22.8dB 23.8dB

Figure 4.26: Left: translation invariant wavelet hard thresholding, center: block orthogonal Stein thresholding, right: block translation invariant Stein thresholding.

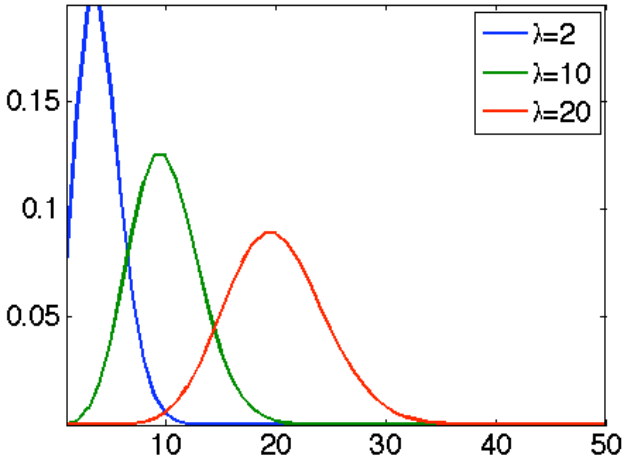


Figure 4.27: Poisson distributions for various λ.

Perfect stabilization is impossible, so that (4.14) only approximately holds for a limited intensity range of $f_{0,n}$. Two popular variation stabilization functions for Poisson noise are the Anscombe mapping

$$\varphi(x) = 2\sqrt{x + 3/8}$$

and the mapping of Freeman and Tukey

$$\varphi(x) = \sqrt{x+1} + \sqrt{x}.$$

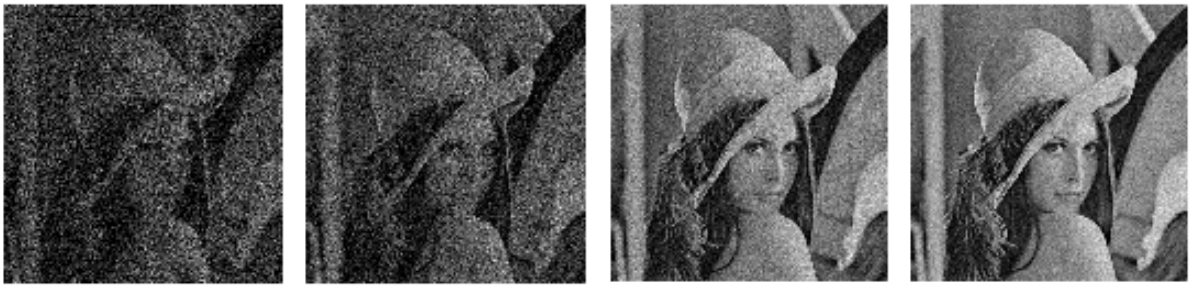
Figure 4.29 shows the effect of these variance stabilizations on the variance of $\varphi(f)$.

A variance stabilized denoiser is defined as

$$\Delta^{\mathrm{stab},q}(f) = \varphi^{-1}(\sum_m S_T^q(\langle \varphi(f), \psi_m \rangle)\psi_m)$$

where φ^{-1} is the inverse mapping of φ.

Figure 4.30 shows that for moderate intensity range, variance stabilization improves over non-stabilized denoising.



$\lambda_{\max} = 5$ $\lambda_{\max} = 50$ $\lambda_{\max} = 50$ $\lambda_{\max} = 100$

Figure 4.28: Noisy image with Poisson noise model, for various $\lambda_{\max} = \max_n f_{0,n}$.

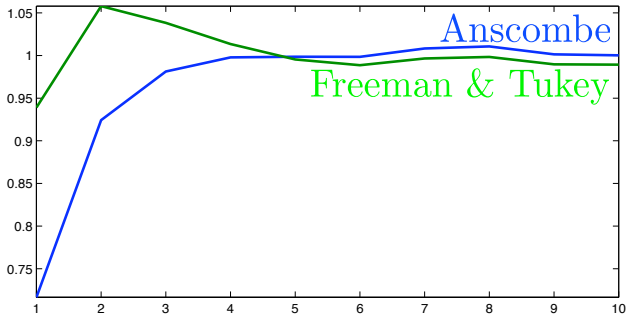


Figure 4.29: Comparison of variariance stabilization: display of $\mathrm{Var}(\varphi(f_n))$ as a function of $f_{0,n}$.

4.4.2 Multiplicative Noise

Multiplicative image formation. A multiplicative noise model assumes that

$$f_n = f_{0,n} w_n$$

where w is a realization of a random vector with $\mathcal{D}(w) = 1$. Once again, the noise level depends on the pixel value

$$\mathcal{D}(f_n) = f_{0,n} \quad \text{and} \quad \mathrm{Var}(f_n) = f_{0,n}^2 \sigma^2 \quad \text{where} \quad \sigma^2 = \mathrm{Var}(w).$$

Such a mutiplicative noise is a good model for SAR satellite imaging, where f is obtained by averaging S images

$$\forall\, 0 \leqslant s < K, \quad f_n^{(s)} = f_{0,n} w_n^{(s)} + r_n^{(s)}$$

where $r^{(s)}$ is a Gaussian white noise, and $w_n^{(s)}$ is distributed according to a one-sided exponential distribution

$$\mathcal{P}(w_n^{(s)} = x) \propto e^{-x}\, \mathbb{I}_{x>0}.$$

For K large enough, averaging the images cancels the additive noise and one obtains

$$f_n = \frac{1}{K} \sum_{s=1}^{K} f_n^{(s)} \approx f_{0,n} w_n$$

where w is distributed according to a Gamma distribution

$$w \sim \Gamma(\sigma = K^{-\frac{1}{2}}, \mu = 1) \quad \text{where} \quad \mathbb{P}(w = x) \propto x^{K-1} e^{-Kx},$$

Figure 4.30: Left: noisy image, center: denoising without variance stabilization, right: denoising after variance stabilization.

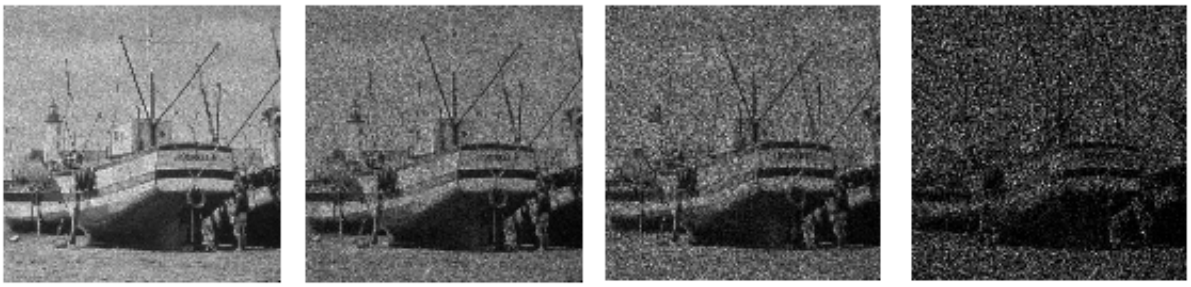


Figure 4.31: Noisy images with multiplicative noise, with varying σ.

One should note that increasing the value of K reduces the overall noise level.

Figure **??** shows an example of such image formation for a varying number $K = 1/\sigma^2$ of averaged images.

A simple variance stabilization transform is

$$\varphi(x) = \log(x) - c$$

where

$$c = \mathrm{E}(\log(w)) = \psi(K) - \log(K) \quad \text{where} \quad \psi(x) = \Gamma'(x)/\Gamma(x)$$

and where Γ is the Gamma function that generalizes the factorial function to non-integer. One thus has

$$\varphi(f)_n = \varphi(f_0)_n + z_n,$$

where $z_n = \log(w) - c$ is a zero-mean additive noise.

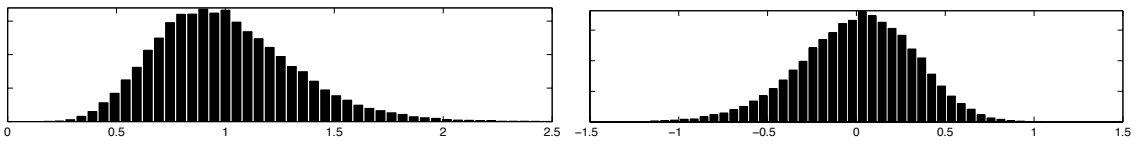


Figure 4.32: Histogram of multiplicative noise before (left) and after (right) stabilization.

Figure 4.32 shows the effect of this variance stabilization on the repartition of w and z.

Figure 4.33 shows that for moderate noise level σ, variance stabilization improves over non-stabilized denoising.

Figure 4.33: Left: noisy image, center: denoising without variance stabilization, right: denoising after variance stabilization.

Bibliography

[1] D. Donoho and I. Johnstone. Ideal spatial adaptation via wavelet shrinkage. *Biometrika*, 81:425–455, Dec 1994.

[2] Stephane Mallat. *A wavelet tour of signal processing: the sparse way.* Academic press, 2008.

[3] Gabriel Peyré. *L'algèbre discrète de la transformée de Fourier.* Ellipses, 2004.

[4] J. Portilla, V. Strela, M.J. Wainwright, and Simoncelli E.P. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Trans. Image Proc.*, 12(11):1338–1351, November 2003.