TAMPERE UNIVERSITY OF TECHNOLOGY

ACADEMY OF FINLAND

Alessandro Foi

ICIP 2014 – Tutorial T7:

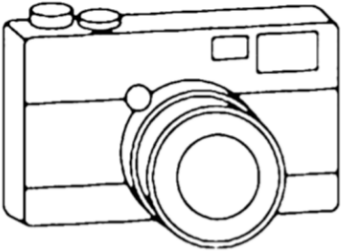
Signal-Dependent Noise and Stabilization of Variance

IEEE International Conference on Image Processing (ICIP 2014), Paris, October 27, 2014

0. A simple experiment

A simple experiment

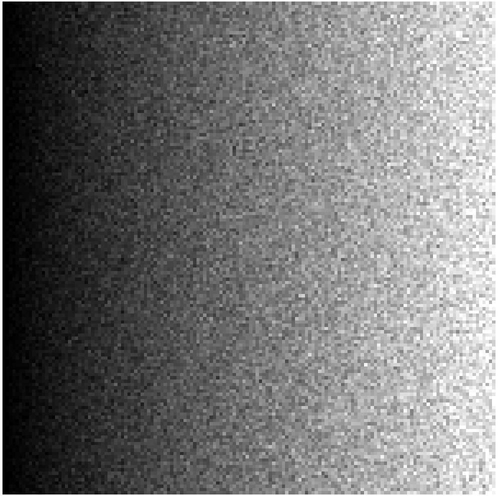
Take photos of a gray scale test ramp



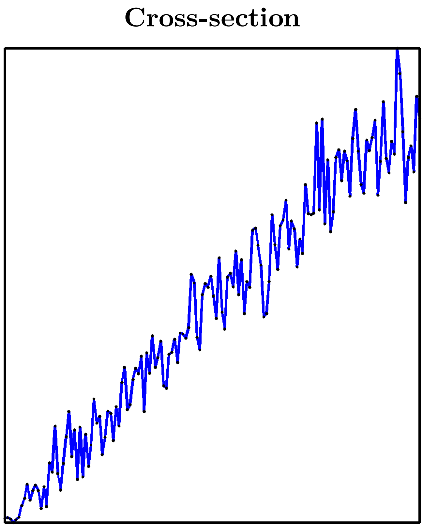
Advice: use a *short* exposure time and *high* ISO value

A simple experiment

Shot #1

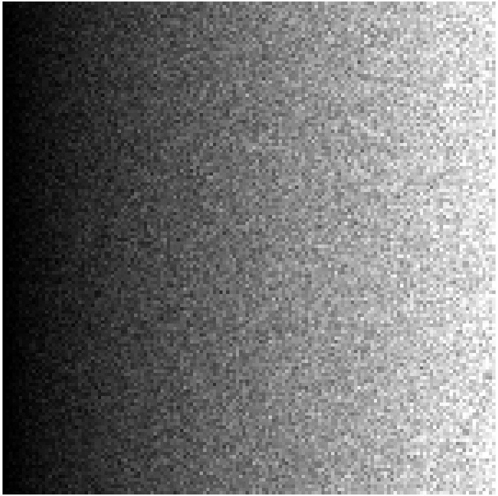


A simple experiment



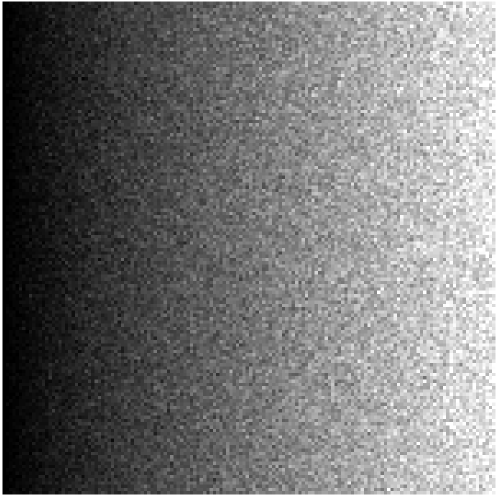
A simple experiment

Shot #2



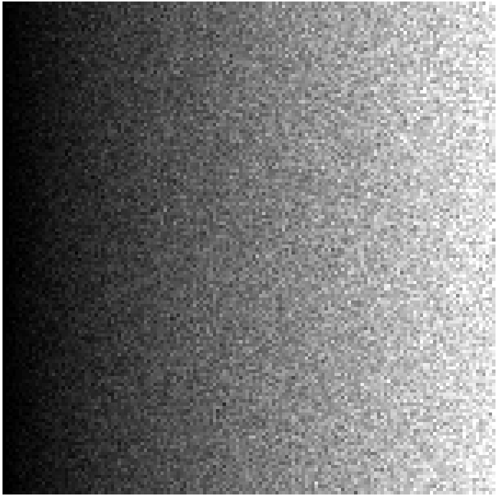
A simple experiment

Shot #3



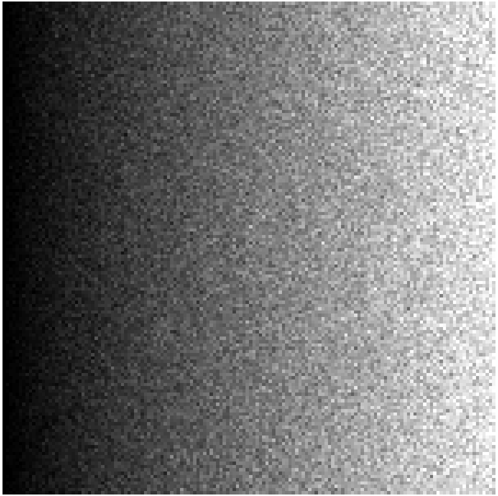
A simple experiment

Shot #4



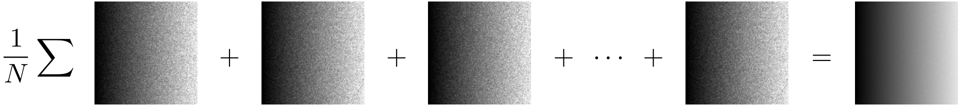
A simple experiment

Shot #5



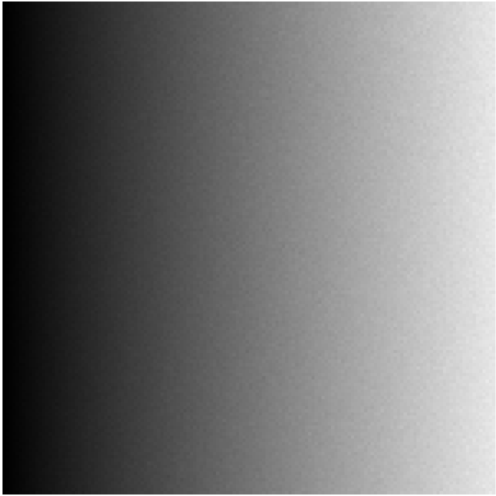
A simple experiment

TAKE MANY MORE SHOTS, AND THEN AVERAGE THEM ALL



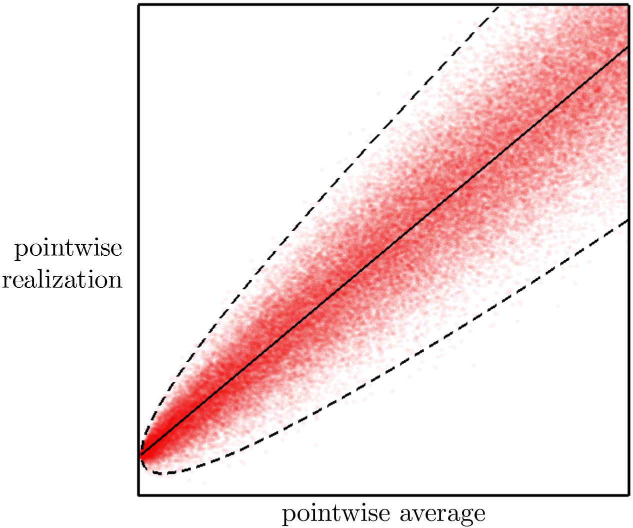
A simple experiment

TAKE MANY MORE SHOTS, AND THEN AVERAGE THEM ALL



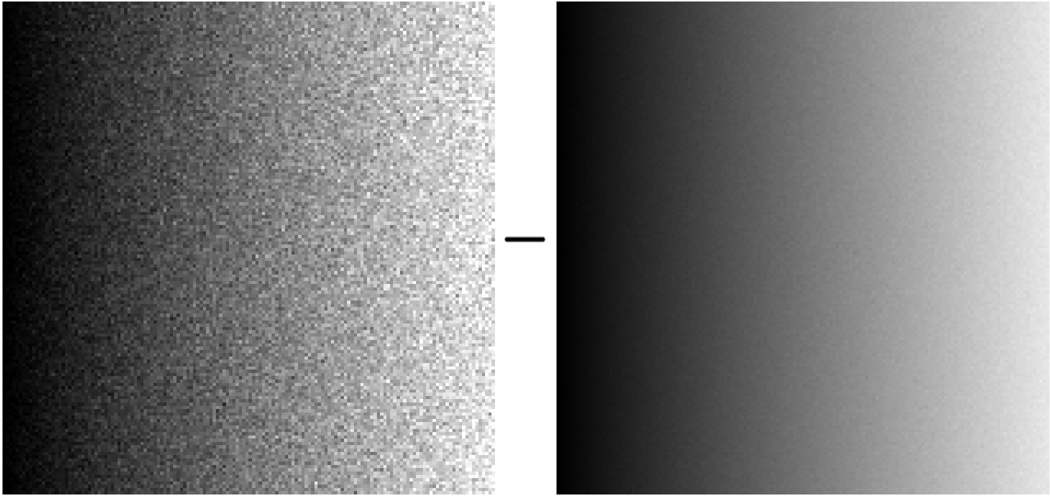
A simple experiment

Scatterplot: average vs realization



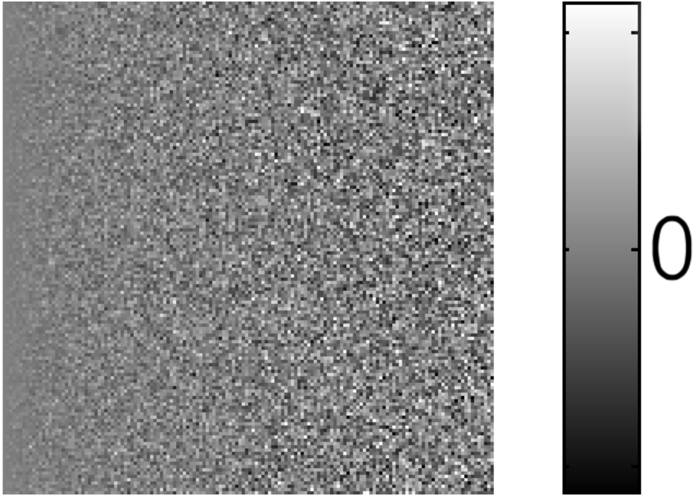
A simple experiment

SUBTRACT THE *AVERAGE OF ALL SHOTS* FROM *ANY OF THE SHOTS*



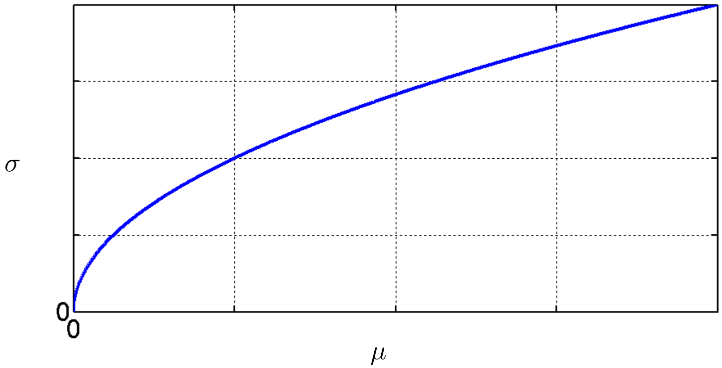
A simple experiment

SUBTRACT THE *AVERAGE OF ALL SHOTS* FROM *ANY OF THE SHOTS*



A simple experiment

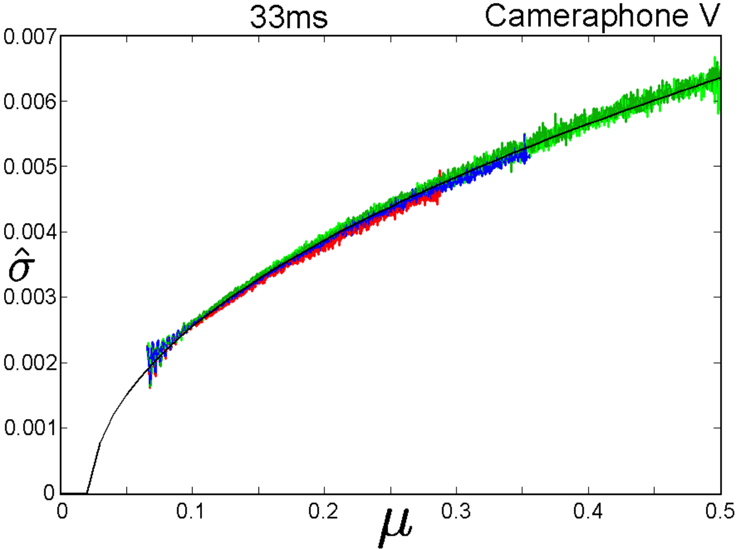
FOR EACH PIXEL, COMPUTE
SAMPLE MEAN AND *SAMPLE STANDARD DEVIATION*
W.R.T. THE VARIOUS SHOTS



NOISE IS STRONGER WHERE THE AVERAGE IMAGE IS BRIGHTER:
STANDARD-DEVIATION IS A FUNCTION OF MEAN

SIGNAL-DEPENDENT NOISE

A simple experiment



analysis of raw data from cameraphone CMOS sensor (F&al.SensJ2007)

1. Rudiments of Noise Modeling

Additive White Gaussian Noise (AWGN) model

$$\boxed{z(x) = y(x) + \sigma\xi(x) \qquad x \in X}$$

$y : X \to Y \subseteq \mathbb{R}$	unknown original image (deterministic)
$\sigma\xi(x)$	i.i.d. zero-mean random error
$z : X \to Z \subseteq \mathbb{R}$	observed noisy image (random)
$x \in X \subseteq \mathbb{Z}$	coordinate in the image domain

$\sigma \in \mathbb{R}^+$	standard deviation of $\sigma\xi(x)$
$\xi(x)$	normal random variable $E\{\xi(x)\} = 0 \quad \mathrm{var}\{\xi(x)\} = 1$

$E\{z(x)\} = y(x)$	expectation
$\mathrm{std}\{z(x)\} = \sigma\,\mathrm{std}\{\xi(x)\} = \sigma$	standard deviation

!! Often z, ξ are used to denote the random variables/processes (when dealing with the model) as well as their realizations (when dealing with the algorithm).

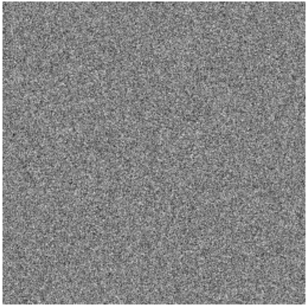
Additive White Gaussian Noise (AWGN) model

z

$=$

y

$+$



$\sigma\xi$

Additive *White* Gaussian Noise (AWGN) model

white:

$$\text{var}\left\{\mathcal{F}\left(\sigma\xi\right)\right\} = \text{constant} \qquad \text{(noise power spectrum is flat)}$$

This nomenclature is perhaps misleading.

What we demand is $\sigma\xi(x)$ to be *independent* and *identically distributed*.

identically distributed:

$$\Pr\left[\sigma\xi(x_1) < c\right] = \Pr\left[\sigma\xi(x_2) < c\right] \qquad \forall c \in \mathbb{R}$$

independent:

$$\Pr\left[\sigma\xi(x_1) < c\right]\Pr\left[\sigma\xi(x_2) < d\right] = \Pr\left[\left(\sigma\xi(x_1) < c\right) \cap \left(\sigma\xi(x_2) < d\right)\right] \qquad \forall c, d \in \mathbb{R}$$

Additive *White* Gaussian Noise (AWGN) model

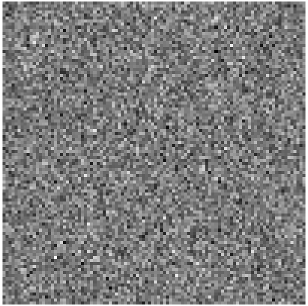
independence implies whiteness:

$$\begin{aligned}\mathcal{F}(\sigma\xi)(\omega) &= \sum_{x\in X} e^{-2\pi i\omega x}\sigma\xi(x)\\ \operatorname{var}\{\mathcal{F}(\sigma\xi)(\omega)\} &= \sum_{x\in X} \left|e^{-2\pi i\omega x}\right|^2 \operatorname{var}\{\sigma\xi(x)\} = \\ &= \sum_{x\in X} \operatorname{var}\{\sigma\xi(x)\} \qquad [\, = \sigma^2 |X| \quad \text{because identically distributed}]\end{aligned}$$

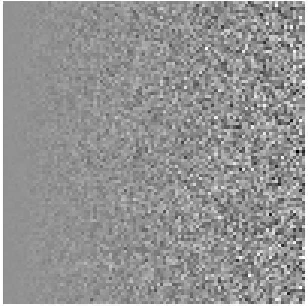
We can have Gaussian white noise that is not i.i.d.!!

How? It suffices to have independent but non identically distributed errors.

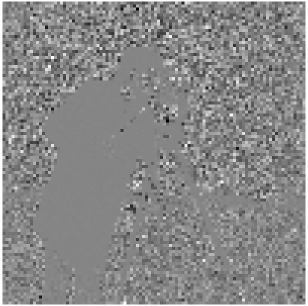
Various examples of Gaussian white noise



i.i.d.



ramp



Cameraman

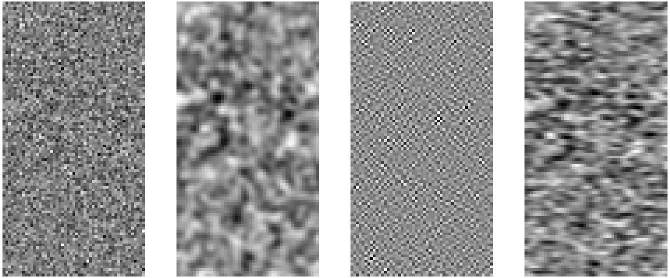
They are all three Gaussian and white, but only the i.i.d. one is what is typically assumed as AWGN.

:-(

Colored noise

Noise is *colored* when the noise power spectrum is markedly not flat.

The band with larger variance determines the "color".



white red blue horizontal

Typically modeled by kernel convolution operator against white noise:

$$\begin{aligned} \mathcal{F}(v \circledast \xi) &= \mathcal{F}(v)\,\mathcal{F}(\xi) \\ \operatorname{var}\{\mathcal{F}(v \circledast \xi)\} &= |\mathcal{F}(v)|^2 \operatorname{var}\{\mathcal{F}(\xi)\} \end{aligned}$$

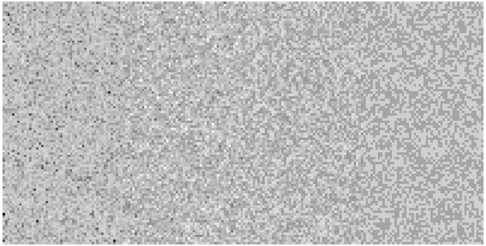
Homoskedasticity vs. Heteroskedasticity

The noise η is **homoskedastic** if different noise samples have same variance:

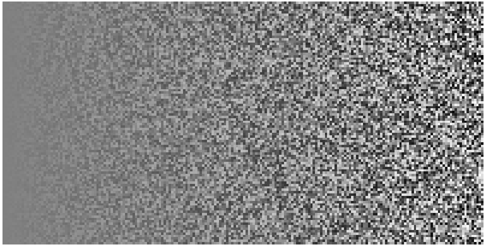
$$\text{var}\{\eta(x')\} = \text{var}\{\eta(x'')\} \quad \forall x', x'' \in X$$

otherwise it is **heteroskedastic** and different noise samples can have different variance:

$$\text{var}\{\eta(x')\} \neq \text{var}\{\eta(x'')\} \qquad \text{for some } \; x', x'' \in X.$$



homoskedastic (but *not* ident.distr.)



heteroskedastic

Standard-deviation map

Let $z(x) = y(x) + \eta(x)$, $x \in X$, with η **heteroskedastic** noise.

Whenever the variance $\mathrm{var}\{\eta\}$ is deterministic, it makes sense to break η into two factors:

$$\eta = \mathrm{std}\{\eta\}\,\xi$$

$\mathrm{std}\{\eta\} : X \to \mathbb{R}^+$ standard-deviation map (deterministic)

$\xi : X \to \mathbb{R}$ homoskedastic noise (random)

$$\mathrm{std}\{\xi\}(x) = 1 \quad \forall x \in X$$

η $\quad = \quad$ $\mathrm{std}\{\eta\}$ $\quad \times \quad$ ξ

Signal-dependent noise

The η noise is *signal-dependent* when the distribution of $\eta(x)$ has some parameter that depends on $y(x)$:

$$\Pr[\eta(x) < c] \quad = \quad F(c, y(x)), \quad \forall x \in X \text{ and } \forall c \in \mathbb{R}$$
$$\text{with } F \text{ functionally independent of } x.$$

The most significant situation arises when the variance of η depends on y, i.e. when the standard-deviation map becomes a function of y:

$$\boxed{z(x) = y(x) + \sigma(y(x))\,\xi(x), \quad x \in X,}$$

$\sigma : \mathbb{R} \to \mathbb{R}^+$	**standard-deviation function or curve** (deterministic),
$\xi(x)$	random variable $E\{\xi(x)\} = 0 \quad \text{var}\{\xi(x)\} = 1$.

Here ξ is homoskedastic noise with unitary variance.
The distribution of $\xi(x)$ may depend on $y(x)$, but what most matters is its variance.

Multiplicative noise

Multiplicative noise is special case of signal-depedent noise where the mean is the direct scaling parameter of the noise distribution.

$$z(x) = y(x) \cdot \eta_{\text{mult}}(x), \quad x \in X,$$

$$\eta_{\text{mult}} \quad \text{i.i.d. noise}, \quad E\{\eta_{\text{mult}}(x)\} = 1, \quad \text{std}\{\eta_{\text{mult}}(x)\} = c.$$

Rewrite in additive signal-dependent form:

$$\begin{aligned} z(x) &= y(x) + y(x)(\eta_{\text{mult}}(x) - 1) = \\ &= y(x) + y(x)\xi'(x) = \\ &= y(x) + \sigma(y(x))\xi(x), \end{aligned}$$

where $\sigma : \mathbb{R} \rightarrow \mathbb{R}^{+}, \ \sigma : y \longmapsto c|y|$

and $\xi(x) = \text{sign}\{y(x)\} c^{-1}\xi'(x) = \text{sign}\{y(x)\} c^{-1}(\eta_{\text{mult}}(x) - 1)$.

We have $E\{\xi(x)\} = 0, \ \text{var}\{\xi(x)\} = 1.$

Poisson distributions

Poisson distributions are discrete integer-valued distributions with non-negative real-valued parameter (mean) $\theta \geq 0$

$$z \sim \mathcal{P}(\theta) \qquad \Pr[z = \zeta | \theta] = e^{-\theta} \frac{\theta^{\zeta}}{\zeta!},\ \zeta \in \mathbb{N}.$$

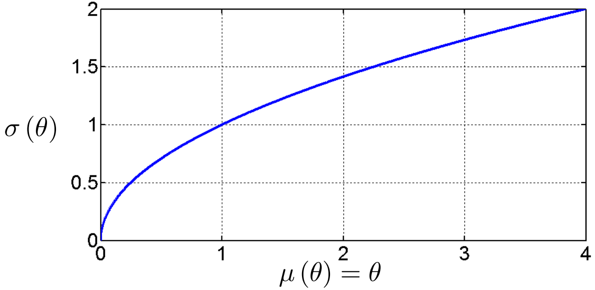
$$\begin{aligned} \mu(\theta) &= E\{z|\theta\} = \theta \\ \sigma^2(\theta) &= \operatorname{var}\{z|\theta\} = \theta = \mu(\theta) \end{aligned}$$

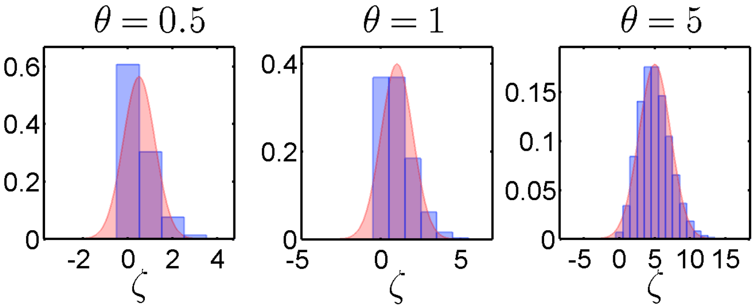
mean and variance coincide and are equal to the parameter θ

Matlab code: z = `poissrnd(theta)`

signal-to-noise ratio (SNR): $\dfrac{\mu(\theta)}{\sigma(\theta)} = \sqrt{\theta} \underset{\theta \to 0}{\longrightarrow} 0 \qquad \dfrac{\mu(\theta)}{\sigma(\theta)} \underset{\theta \to +\infty}{\longrightarrow} +\infty$

Poisson distributions





Discrete Poisson $\mathcal{P}(\theta)$ (blue) and continuous normal approximation $\mathcal{N}(\theta, \theta)$ (red)

Normal approximation of Poisson

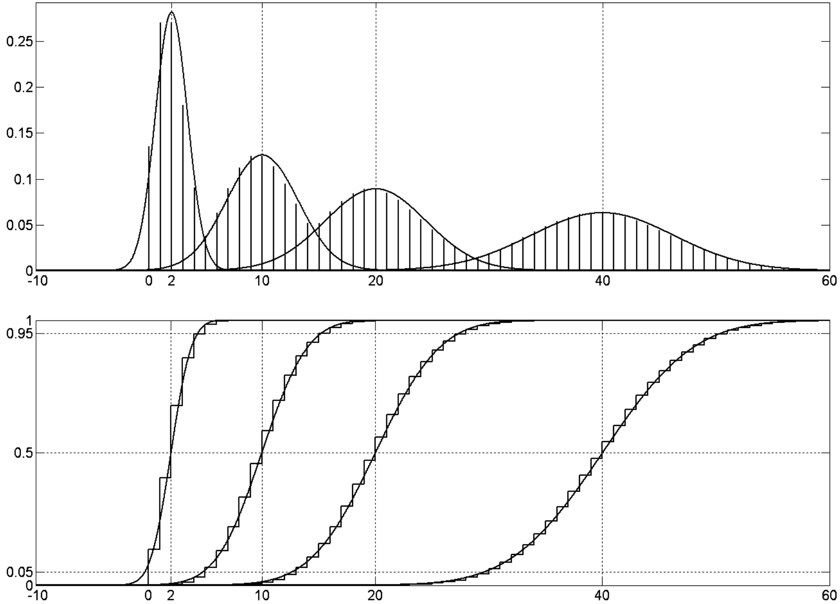
$z \sim \mathcal{P}(\theta)$ means the probability of z $\Pr[z = \zeta|\theta] = e^{-\theta}\frac{\theta^{\zeta}}{\zeta!}$, $\zeta \in \mathbb{N}$

$z \sim \mathcal{N}\left(\mu, \sigma^2\right)$ means the probability density of z is $\wp\left(\zeta|\mu, \sigma^2\right) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\zeta-\mu)^2}{2\sigma^2}}$, $\zeta \in \mathbb{R}$.

$$\mathcal{P}(\theta) \underset{\theta \to +\infty}{\longrightarrow} \mathcal{N}(\theta, \theta)$$

Matlab code: z = `z + sqrt(theta).*randn(size(theta))`

Normal approximation of Poisson



“p.d.f.” (top) and c.d.f. (bottom) for $\mathcal{P}(\theta)$ and $\mathcal{N}(\theta, \theta)$, $\theta = 2, 10, 20, 40$.

Scaled Poisson distributions

Scaled Poisson distributions with scale parameter $\chi > 0$ and mean $\theta \geq 0$

$$z\chi \sim \mathcal{P}(\theta\chi) \qquad \Pr[z = \zeta|\theta] = e^{-\theta\chi}\frac{(\theta\chi)^{\zeta\chi}}{(\zeta\chi)!}, \quad \zeta\chi \in \mathbb{N}, \quad \theta \in [0, +\infty).$$

Discrete taking values that are nonnegative integer multiples of $\frac{1}{\chi}$.

$$\begin{aligned} \mu(\theta) &= E\{z|\theta\} = \theta \\ \sigma^2(\theta) &= \operatorname{var}\{z|\theta\} = \frac{\theta}{\chi} \end{aligned}$$

mean is equal to the parameter θ and coincides with the variance times χ.

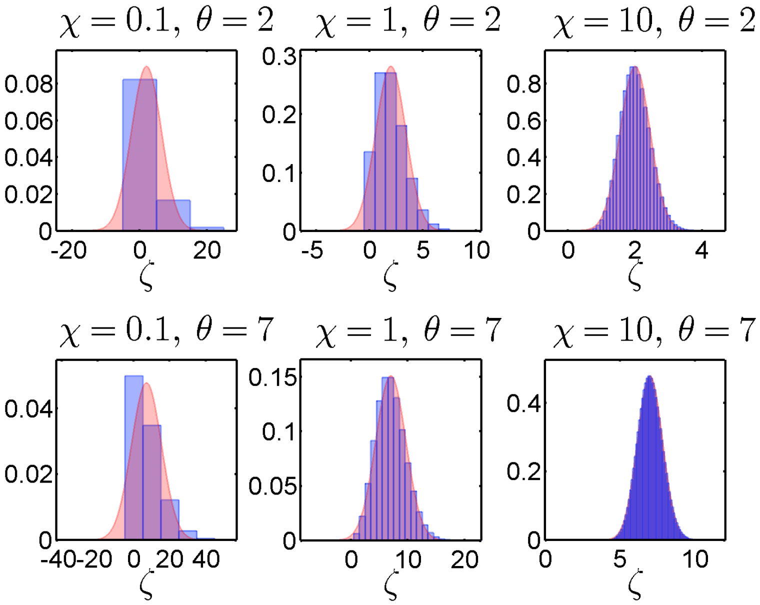
The scale parameter χ controls the relative strength of the noise: SNR $\frac{\mu(\theta)}{\sigma(\theta)} = \sqrt{\chi\theta}$.

Matlab code: z = `poissrnd(chi*theta)/chi`

Normal approximation for large θ: $z \sim \mathcal{N}(\theta, \theta/\chi)$

Matlab code: z = `z + sqrt(theta/chi).*randn(size(theta))`

Scaled Poisson distributions



small χ is detremental when θ varies on a narrow range of values

Poissonian noise

Let $y : X \to Y \subseteq \mathbb{R}^+$	original image (deterministic, possibly unknown)
$\chi > 0$	scaling factor

$$\boxed{z(x)\,\chi \sim \mathcal{P}(\chi y(x)), \qquad \forall x \in X.}$$

$$E\{z(x)\,\chi\} = \chi E\{z(x)\} = \chi y(x) \quad \Longrightarrow \quad E\{z(x)\} = y(x),$$

$$\operatorname{var}\{z(x)\,\chi\} = \chi^2 \operatorname{var}\{z(x)\} = \chi y(x) \quad \Longrightarrow \quad \operatorname{var}\{z(x)\} = \frac{y(x)}{\chi}.$$

This can be rewritten in the usual form as

$$\boxed{z(x) = y(x) + \sqrt{\frac{y(x)}{\chi}}\,\xi(x), \qquad \forall x \in X,}$$

where $E\{\xi(x)\} = 0$ and $\operatorname{var}\{\xi(x)\} = 1$.

The term $\sqrt{\frac{y(x)}{\chi}}\xi(x)$ is the so-called **Poissonian noise**.

Scaled Poisson observations

$\chi = 1000$

Scaled Poisson observations

$\chi = 300$

Scaled Poisson observations

$\chi = 100$

Scaled Poisson observations

$\chi = 50$

Scaled Poisson observations

$\chi = 10$

Scaled Poisson observations

$\chi = 1$

One-parameter families of distributions

A one-parameter family of distributions $\boldsymbol{\mathcal{D}} = \{\mathcal{D}_\theta\}$ is a collection of distributions, each of which is identified bit the value of a univariate parameter $\theta \in \Theta \subseteq \mathbb{R}$.

Let $z \in Z \subseteq \mathbb{R}$ be a random variable distributed according to a one-parameter family of distributions $\boldsymbol{\mathcal{D}} = \{\mathcal{D}_\theta\}$.

For each individual $\theta \in \Theta$: $\mathcal{D}_\theta$ is a distribution, $z|\theta \sim \mathcal{D}_\theta$, $z|\theta \in Z_\theta \subseteq Z$

$\mu(\theta) = E\{z|\theta\}$ conditional expectation of z expressed as function of θ.
$\sigma(\theta) = \operatorname{std}\{z|\theta\}$ conditional standard deviation of z expressed as function of θ.

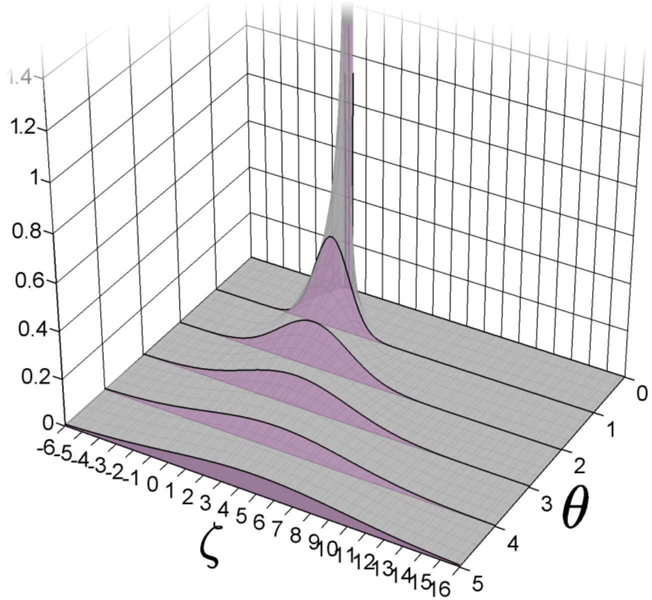
Poisson example:

$\Theta = [0, +\infty) \subset \mathbb{R}$
$\mathcal{D}_\theta$ is one Poisson distribution with parameter $\theta \in \Theta$
$Z_\theta = \{0, 1, 2, \dots\} = \mathbb{N}$
$\mu(\theta) = \theta$
$\sigma(\theta) = \theta$

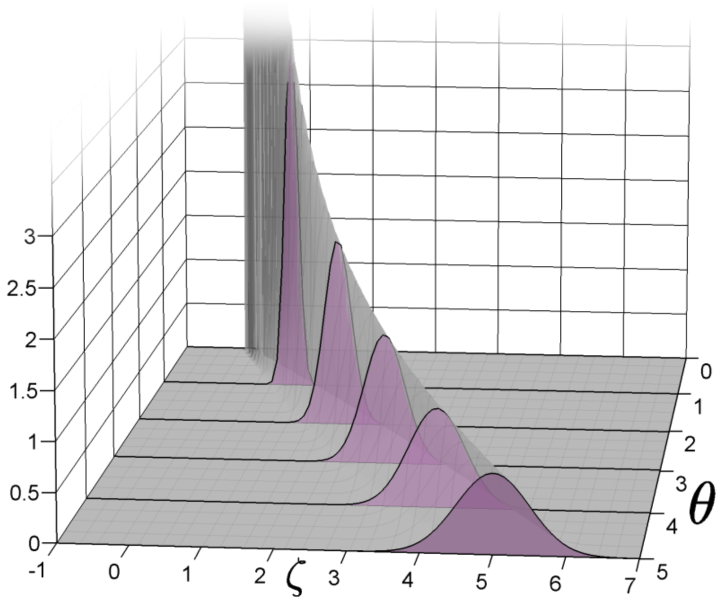
One-parameter families of distributions: examples

$\mathcal{D}_\theta$	$\mu(\theta)$	$\sigma(\theta)$
Poisson		
$\Pr[z=\zeta\vert\theta] = e^{-\theta}\frac{\theta^\zeta}{\zeta!},\ \zeta\in\mathbb{N},\ \theta\in[0,+\infty)$	θ	$\sqrt{\theta}$
Scaled Poisson (scale $\chi>0$)		
$\Pr[z=\zeta\vert\theta] = e^{-\theta\chi}\frac{(\theta\chi)^{\zeta\chi}}{(\zeta\chi)!},\ \zeta\chi\in\mathbb{N},\ \theta\in[0,+\infty)$	θ	$\sqrt{\frac{\theta}{\chi}}$
Binomial (n trials)		
$\Pr[z=\zeta\vert\theta] = \binom{n}{\zeta}\theta^\zeta(1-\theta)^{n-\zeta},\ \zeta\in\mathbb{N},\ \theta\in[0,1]$	$n\theta$	$\sqrt{n\theta(1-\theta)} = \sqrt{\frac{\mu(\theta)(n-\mu(\theta))}{n}}$
Scaled binomial (n trials, scale n)		
$\Pr\left[z=\frac{\zeta}{n}\vert\theta\right] = \binom{n}{\zeta}\theta^\zeta(1-\theta)^{n-\zeta},\ \zeta\in\mathbb{N},\ \theta\in[0,1]$	θ	$\sqrt{\frac{\theta(1-\theta)}{n}}$
Negative binomial (exponent k)		
$\Pr[z=\zeta\vert\theta] = \frac{\Gamma(\zeta+k)}{\zeta!\Gamma(k)}\left(\frac{\theta}{\theta+k}\right)^\zeta\left(\frac{k+\theta}{k}\right)^{-k},\ \zeta\in\mathbb{N},\ \theta\in[0,+\infty)$	θ	$\sqrt{\frac{\theta(\theta+k)}{k}}$
Scaled negative binomial (exponent k, scale $\chi>0$)		
$\Pr\left[z=\frac{\zeta}{\chi}\vert\theta\right] = \frac{\Gamma(\zeta+k)}{\zeta!\Gamma(k)}\left(\frac{\theta}{\theta+k}\right)^\zeta\left(\frac{k+\theta}{k}\right)^{-k},\ \zeta\in\mathbb{N},\ \theta\in[0,+\infty)$	$\frac{\theta}{\chi}$	$\sqrt{\frac{\theta(\theta+k)}{\chi^2 k}} = \sqrt{\frac{\mu(\theta)(\mu(\theta)\chi+k)}{\chi k}}$
Multiplicative normal (scale $\chi>0$)		
$\mathrm{pdf}[z\vert\theta](\zeta) = \frac{\chi}{\theta\sqrt{2\pi}}e^{-\frac{(\zeta-\theta)^2\chi^2}{2\theta^2}}$	θ	$\frac{\theta}{\chi}$
Doubly censored normal with standard-deviation $s(\theta)$		
$\mathrm{pdf}[z\vert\theta](\zeta) = \Phi\left(\frac{-y}{\sigma(y)}\right)\delta_0(\zeta) + \frac{1}{\sigma(y)}\phi\left(\frac{\zeta-y}{\sigma(y)}\right)\chi_{[0,1]} + \left(1-\Phi\left(\frac{1-y}{\sigma(y)}\right)\right)\delta_0(1-\zeta)$		

Multiplicative Gaussian noise $\mathrm{pdf}\,[z|\theta]\,(\zeta) \quad (\chi = 1)$



Multiplicative Gaussian noise $\mathrm{pdf}\,[z|\theta]\,(\zeta) \quad (\chi = 10)$



Poisson-Gaussian noise

Each observed pixel intensity value $z(x)$, $x \in X$, is composed of a scaled Poisson and an additive Gaussian component:

$$z(x) = \alpha p(x) + n(x),$$

where $p(x) \sim \mathcal{P}(y(x))$, $y(x)$ is the unknown noise-free pixel intensity, $\alpha > 0$ is a gain or scaling parameter, and $n(\cdot) \sim \mathcal{N}\left(0, \sigma^2\right)$.

Poisson-Gaussian noise is defined as

$$\eta(x) = z(x) - \alpha y(x).$$

Signal-dependent standard deviation:

$$\text{std}\{z(x)|y(x)\} = \sqrt{\alpha^2 y(x) + \sigma^2}.$$

Rician-distributed data

Let $z \sim \mathcal{R}(\nu, \sigma)$ be the realization of a random variable with Rician p.d.f. with parameters $\nu \geq 0$ and $\sigma > 0$,

$$p(z|\nu,\sigma) = \frac{z}{\sigma^2} e^{-\frac{z^2+\nu^2}{2\sigma^2}} I_0\left(\frac{z\nu}{\sigma^2}\right), \qquad z \geq 0, \tag{1}$$

where I_n denotes the modified Bessel function of order n.

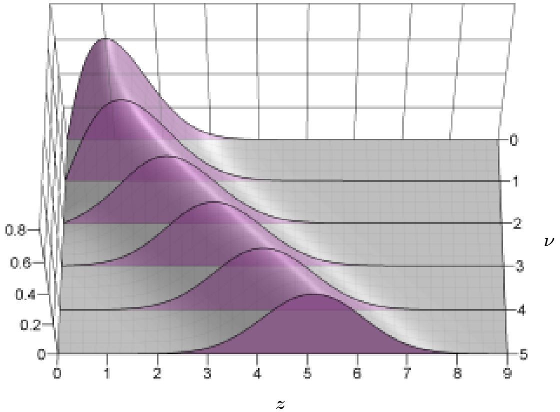
Equivalently, $z = \sqrt{(c_r\nu + \sigma\eta_r)^2 + (c_i\nu + \sigma\eta_i)^2}$,
where c_r and c_i are arbitrary constants such that $0 \leq c_r, c_i \leq 1 = c_r^2 + c_i^2$, and $\eta_r, \eta_i \sim \mathcal{N}(0,1)$.

Observation model for magnitude magnetic resonance (MR) images/volumes:

$z(x) \sim \mathcal{R}(\nu(x), \sigma)$, $\quad x \in X \subset \mathbb{Z}^d$, $d = 2, 3$ (pixel or voxel coordinates).

$\nu : X \to \mathbb{R}^+$ is the unknown original (noise-free) signal

$z : X \to \mathbb{R}^+$ is the raw magnitude MR data.



The one-parameter family of Rician p.d.f.'s $\mathcal{R}(\nu, 1)$ for $\nu \in [0, 5]$.

The parameter σ is assumed as fixed. Thus, z is treated as distributed according to a one-parameter family of Rician distributions, parametrized with respect to ν: $\mathcal{R}(\cdot, \sigma)$.
Assuming $\sigma = 1$ is not a serious restriction: $z \sim \mathcal{R}(\nu, \sigma)$ iff $\lambda z \sim \mathcal{R}(\lambda\nu, \lambda\sigma)$ $\forall \lambda > 0$.
Thanks to this scaling we can carry out all analysis for $\sigma = 1$, and then apply it to other cases $\sigma > 0$ upon simple linear rescaling of data and parameters.
Given $f : \mathbb{R}^+ \to \mathbb{R}$, we have that $\mathrm{var}\{f(z)\,|\nu, \sigma\} = \mathrm{var}\{f_\lambda(w)\,|\lambda\nu, \lambda\sigma\}$, where $z \sim \mathcal{R}(\nu, \sigma)$, $w = \lambda z \sim \mathcal{R}(\lambda\nu, \lambda\sigma)$ and $f_\lambda(w) = f(w/\lambda)$ $\forall w \in \mathbb{R}^+$.

Mean and variance of Rician data

The mean and variance of $z \sim \mathcal{R}(\nu, \sigma)$ are, respectively,

$$\mu = E\{z|\nu,\sigma\} = \sigma\sqrt{\frac{\pi}{2}}L\left(-\frac{\nu^2}{2\sigma^2}\right), \tag{2}$$

$$s^2 = \operatorname{var}\{z|\nu,\sigma\} = 2\sigma^2 + \nu^2 - \frac{\pi\sigma^2}{2}L^2\left(-\frac{\nu^2}{2\sigma^2}\right), \tag{3}$$

where $L(x) = e^{x/2}\left[(1-x)I_0\left(-\frac{x}{2}\right) - xI_1\left(-\frac{x}{2}\right)\right]$.

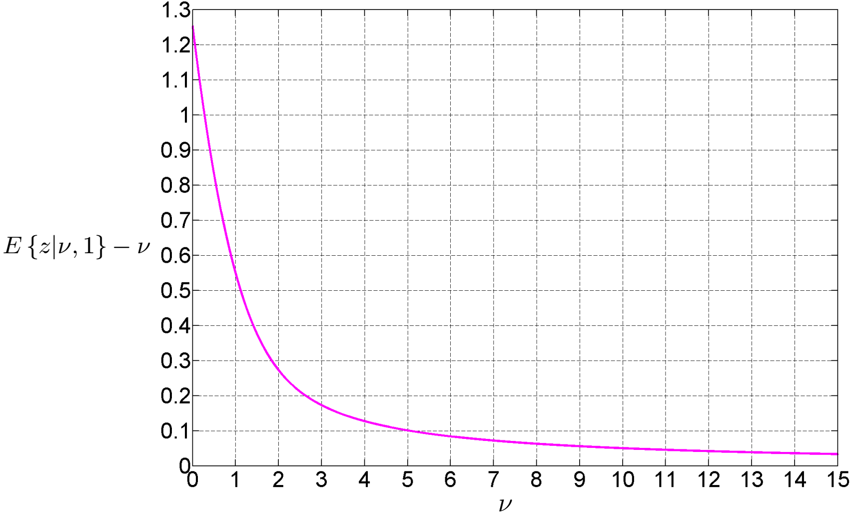
For large values of ν we have

$$E\{z|\nu,\sigma\} \approx \nu + \frac{\sigma^2}{2\nu}, \quad \operatorname{var}\{z|\nu,\sigma\} \approx \sigma^2 - \frac{\sigma^4}{2\nu^2}. \tag{4}$$

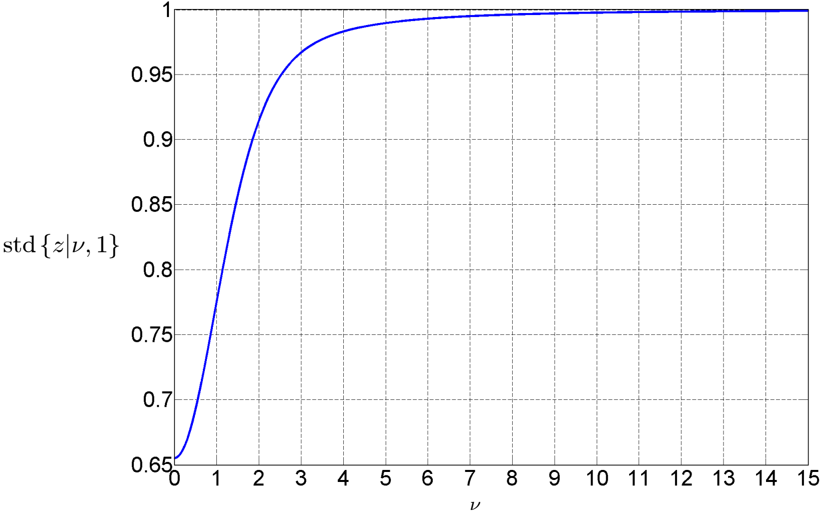
Two crucial issues follow from (2) and (3):
(3) implies that the noise variance is not uniform over the data.
the expectation (2) differs essentially from the parameter of interest, namely ν.

The former problem is addressed by the (forward) variance-stabilizing transformation applied to the data before prior to filtering, wheareas the latter is addressed by the inverse transformation applied upon filtering, which is designed so to directly provide an estimate of ν out of the filtered transformed data.

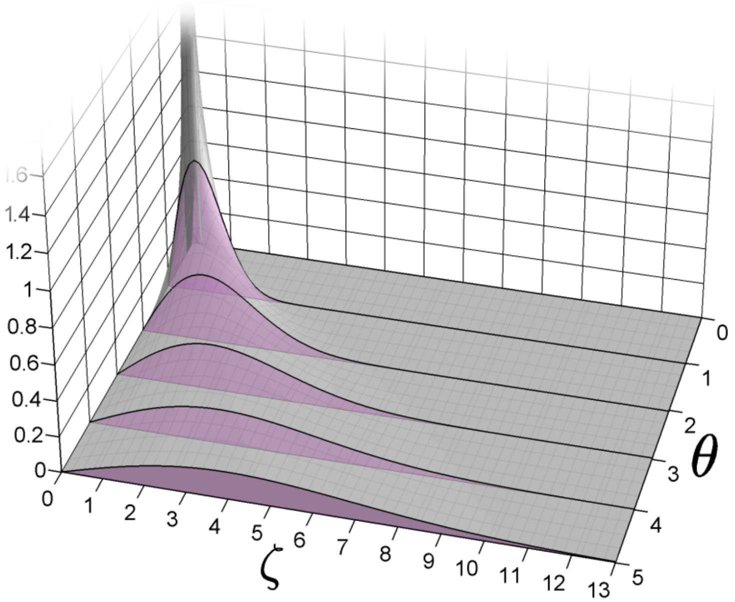
Mean of Rician data



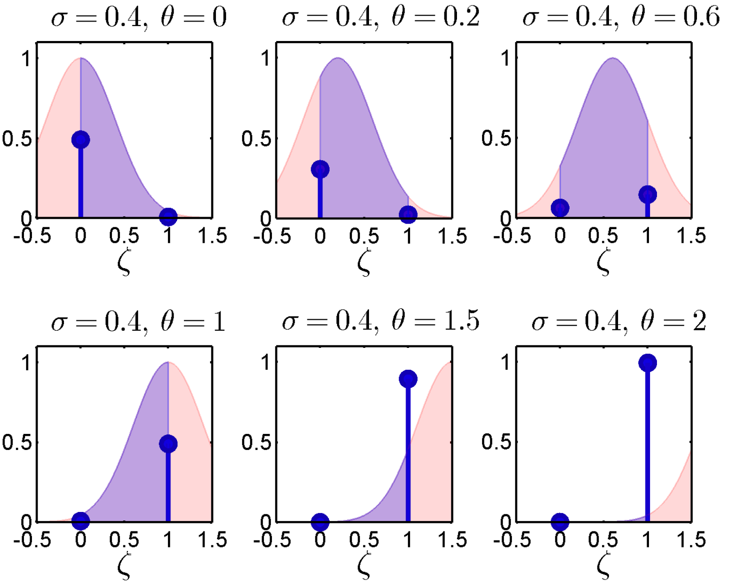
Standard-deviation of Rician data



Rayleigh $\text{pdf}\,[z|\theta]\,(\zeta)$

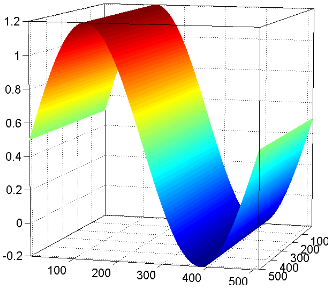


Doubly censored normal (clipping from below and above)

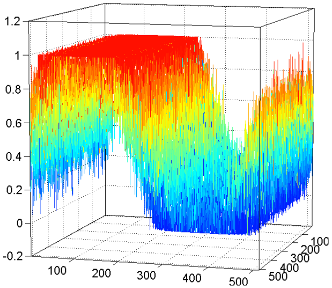


Underlying normal p.d.f. (uncensored) drawn in red

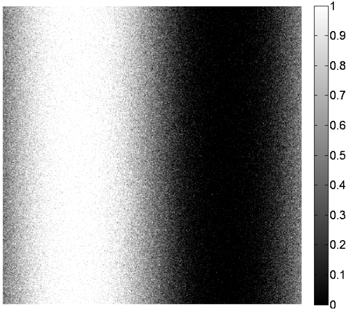
Doubly censored normal as a model for clipped noisy data



original



added AWGN and then clipped



(F&al.TIP2008, F.SigPro2009)

Raw data as clipped signal-dependent observations

$$\tilde{z}(x) = \max\{0, \min\{z(x), 1\}\}, \qquad x \in X \subset \mathbb{Z}^2,$$

$$z(x) = y(x) + \sigma(y(x))\,\xi(x)$$

$y : X \rightarrow Y \subseteq \mathbb{R}$	unknown original image (deterministic)
$\sigma(y(x))\,\xi(x)$	zero-mean random error
$\sigma : \mathbb{R} \rightarrow \mathbb{R}^+$	standard-deviation function (deterministic)
$\xi(x)$	random variable $E\{\xi(x)\} = 0$ $\operatorname{var}\{\xi(x)\} = 1$
$y(x) = E\{z(x)\}$	expectation
$\sigma(y(x)) = \operatorname{std}\{z(x)\}$	standard deviation

Raw data as clipped signal-dependent observations

$$z(x) = y(x) + \sigma(y(x))\,\xi(x)$$

$$\tilde{z}(x) = \max\{0, \min\{z(x), 1\}\}, \qquad x \in X \subset \mathbb{Z}^2,$$

$$\boxed{\tilde{z}(x) = \tilde{y}(x) + \tilde{\sigma}(\tilde{y}(x))\,\tilde{\xi}(x)}$$

$\tilde{y}(x) = E\{\tilde{z}(x)\,\vert\tilde{y}(x)\}$	expectation
$\tilde{\sigma} : [0,1] \to \mathbb{R}^+$	standard-deviation function (of expectation)
$\tilde{\sigma}(\tilde{y}(x)) = \mathrm{std}\{\tilde{z}(x)\,\vert\tilde{y}(x)\}$	standard deviation

Modeling raw-data signal-dependence before clipping

The random error before clipping is composed of two mutually independent parts:

$$\sigma\left(y\left(x\right)\right)\xi\left(x\right)=\eta_{\mathrm{p}}\left(y\left(x\right)\right)+\eta_{\mathrm{g}}\left(x\right)$$

η_{p}	*Poissonian* signal-dependent component	(photonic)
η_{g}	*Gaussian* signal-independent component	(everything else)

$$\begin{aligned}\big(y\left(x\right)+\eta_{\mathrm{p}}\left(y\left(x\right)\right)\big)\chi &\sim \mathcal{P}\left(\chi y\left(x\right)\right), &\quad \chi>0\\ \eta_{\mathrm{g}}\left(x\right) &\sim \mathcal{N}\left(0,b\right), &\quad b>0\end{aligned}$$

$$\sigma^{2}(y(x))=ay(x)+b,\qquad a=\chi^{-1}$$

Variance is an **affine** function of mean.

Higher-order models (e.g., quadratic functions) are also possible and allow to better capture nonlinearities in sensor response.

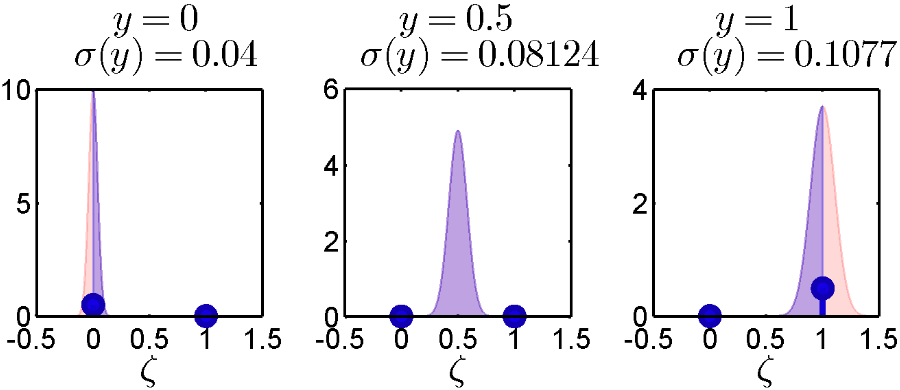
Heteroskedastic normal approximation

$$\tilde{z}(x) = \max\left\{0, \min\left\{z(x), 1\right\}\right\}, \qquad x \in X \subset \mathbb{Z}^2,$$

$$z(x) = y(x) + \sigma(y(x))\,\xi(x)$$

$$\boxed{\sigma\left(y\left(x\right)\right)\xi\left(x\right) = \sqrt{ay(x)+b}\,\xi\left(x\right), \quad \xi\left(x\right) \sim \mathcal{N}\left(0,1\right)}$$

(Generalized) Probability distributions



Before clipping : $\wp_z(\zeta|y) = \frac{1}{\sigma(y)}\phi\left(\frac{\zeta-y}{\sigma(y)}\right)$

After clipping : $\wp_{\tilde{z}}(\zeta|y) = \frac{1}{\sigma(y)}\phi\left(\frac{\zeta-y}{\sigma(y)}\right)\chi_{[0,1]} + \Phi\left(\frac{-y}{\sigma(y)}\right)\delta_0(\zeta) + \left(1-\Phi\left(\frac{1-y}{\sigma(y)}\right)\right)\delta_0(1-\zeta)$

ϕ and Φ are p.d.f. and c.d.f. of $\mathcal{N}(0,1)$

δ_0 is Dirac delta function $\chi_{[0,1]}$ is characteristic (=indicator) function of interval $[0,1]$

Expectations and variances

$$E\left\{\tilde{z}|y\right\} = \tilde{y} = \Phi\left(\tfrac{y}{\sigma(y)}\right) y - \Phi\left(\tfrac{y-1}{\sigma(y)}\right)(y-1) + \sigma(y)\,\phi\left(\tfrac{y}{\sigma(y)}\right) - \sigma(y)\,\phi\left(\tfrac{y-1}{\sigma(y)}\right),$$

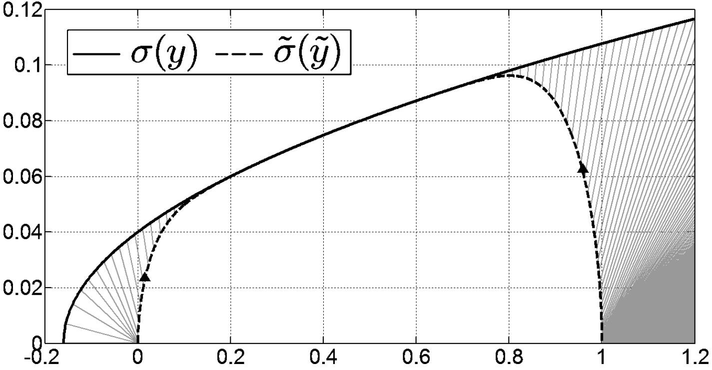
$$\begin{aligned}\operatorname{var}\{\tilde{z}|y\} = \tilde{\sigma}^2(\tilde{y}) = \Phi\left(\tfrac{y}{\sigma(y)}\right)\left(y^2 - 2\tilde{y}y + \sigma^2(y)\right) + \\ + \tilde{y}^2 - \Phi\left(\tfrac{y-1}{\sigma(y)}\right)\left(y^2 - 2\tilde{y}y + 2\tilde{y} + \sigma^2(y) - 1\right) + \\ + \sigma(y)\,\phi\left(\tfrac{y-1}{\sigma(y)}\right)(2\tilde{y} - y - 1) - \sigma(y)\,\phi\left(\tfrac{y}{\sigma(y)}\right)(2\tilde{y} - y)\,.\end{aligned}$$

These equations are “universal”, in the sense that the are valid for any variance function $\sigma^2(y)$, including non-affine ones.

(F.SigPro2009)

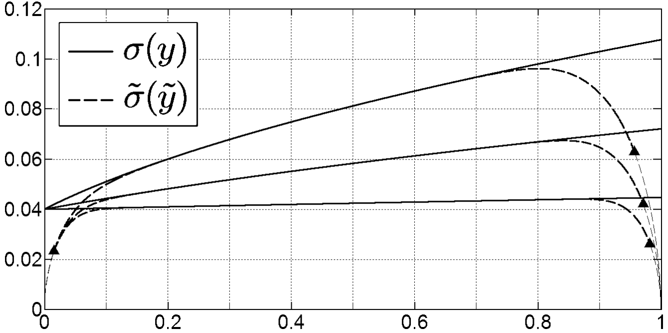
Expectations and variances

$$y = E\{z|y\}, \quad \sigma(y) = \mathrm{std}\{z|y\}, \qquad \tilde{y} = E\{\tilde{z}|y\}, \quad \tilde{\sigma}(\tilde{y}) = \mathrm{std}\{\tilde{z}|y\}.$$

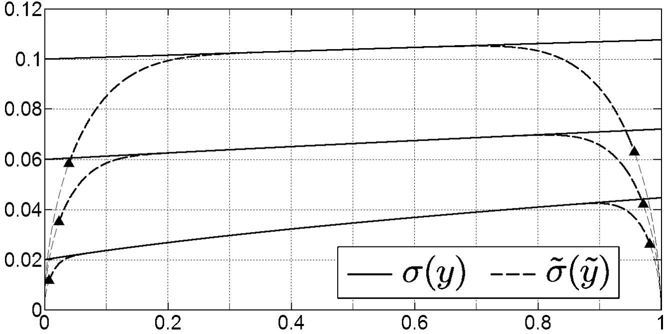


(F.SigPro2009)

Standard-deviation function $\sigma(y) = \sqrt{0.01y + 0.04^2}$ (solid line) and the corresponding standard-deviation curve $\tilde{\sigma}(\tilde{y})$ (dashed line).
The gray segments illustrate the mapping $\sigma(y) \longmapsto \tilde{\sigma}(\tilde{y})$.
The small black triangles ▲ indicate points $(\tilde{y}, \tilde{\sigma}(\tilde{y}))$ which correspond to $y = 0$ and $y = 1$.

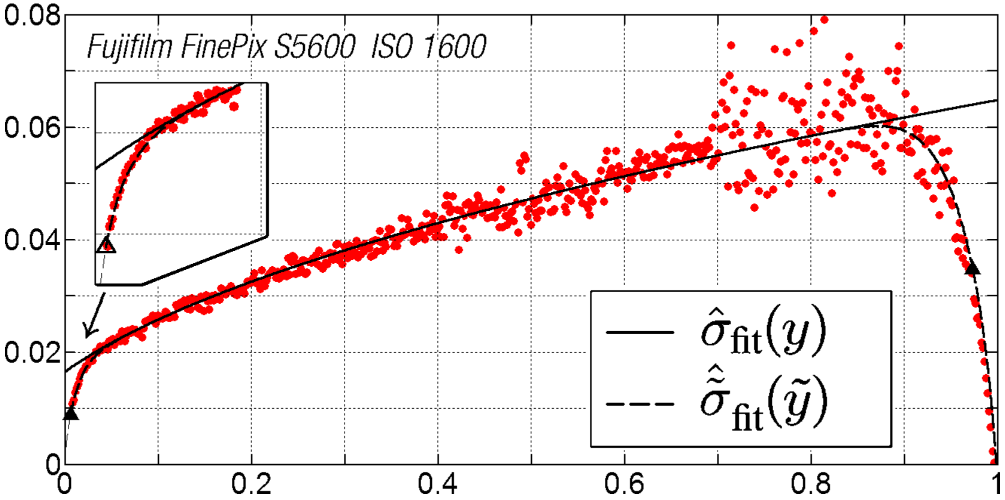


$a = 0.02^2, 0.06^2, 0.10^2$
$b = 0.04^2$



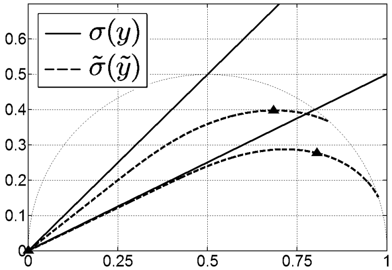
$a = 0.04^2$
$b = 0.02^2, 0.06^2, 0.10^2$

Model does indeed fit the data

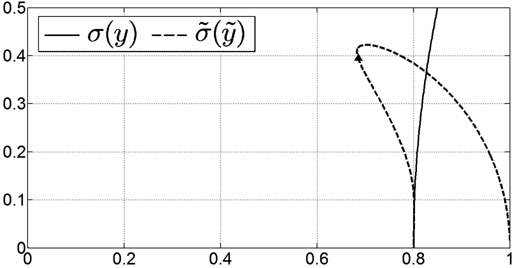


(F.&al.TIP2008)

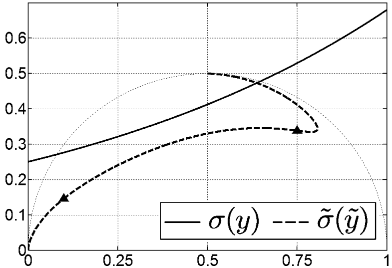
Limit cases and patologies



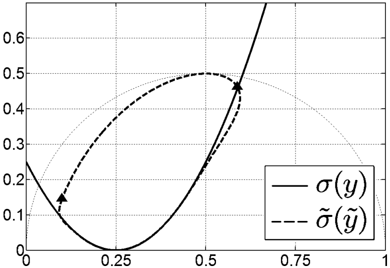
$\sigma(y) = \alpha y, \quad \alpha = 1, \frac{1}{2}$



$\sigma(y) = \sqrt{5y - 4}$



$\sigma(y) = \frac{1}{4}e^y$



$\sigma(y) = 4y^2 - 2y + \frac{1}{4}$

(F.SigPro2009)

2. Variance Stabilization

Motivation

Signal-dependent errors are particularly undesirable because

- basic data analysis and processing methods (such as those studied in earlier courses),
- standard statistical procedures implemented in computing environments (Matlab, R, Mathematica, etc.),
- off-the-shelf algorithms,

are typically designed and implemented for *identically distributed errors.*

Variance stabilization attempts to make the variance of the errors to be the same.

Variance-stabilization problem

Find a function $f : Z \to \mathbb{R}$ such that the transformed variable $f(z)$ has constant standard deviation, say, equal to 1, $\mathrm{std}\{f(z)\,|\theta\} = 1$.

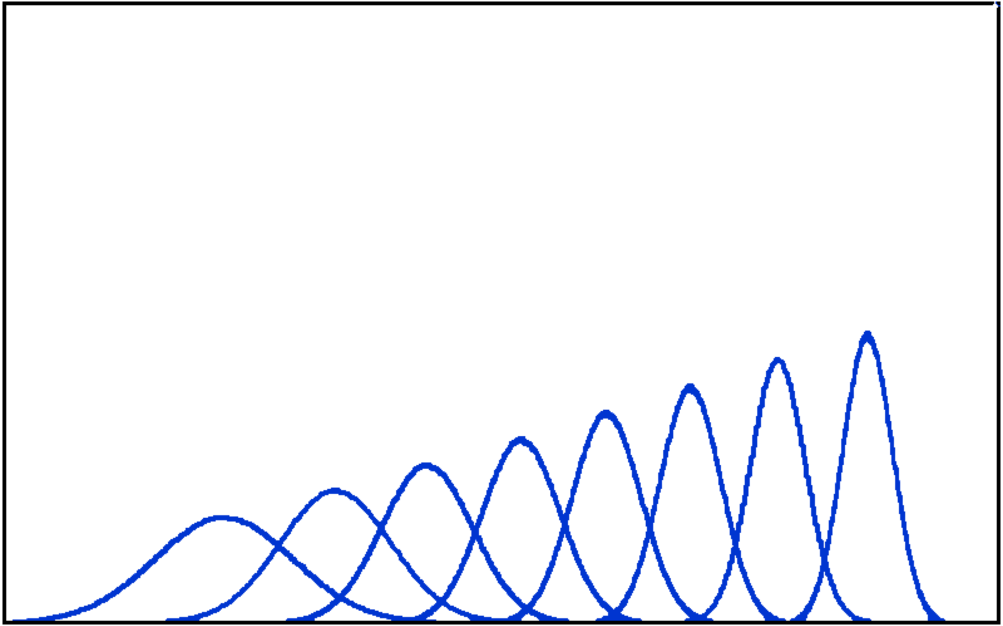
such f is a **variance-stabilizing transformation (VST)**

f should be independent of θ

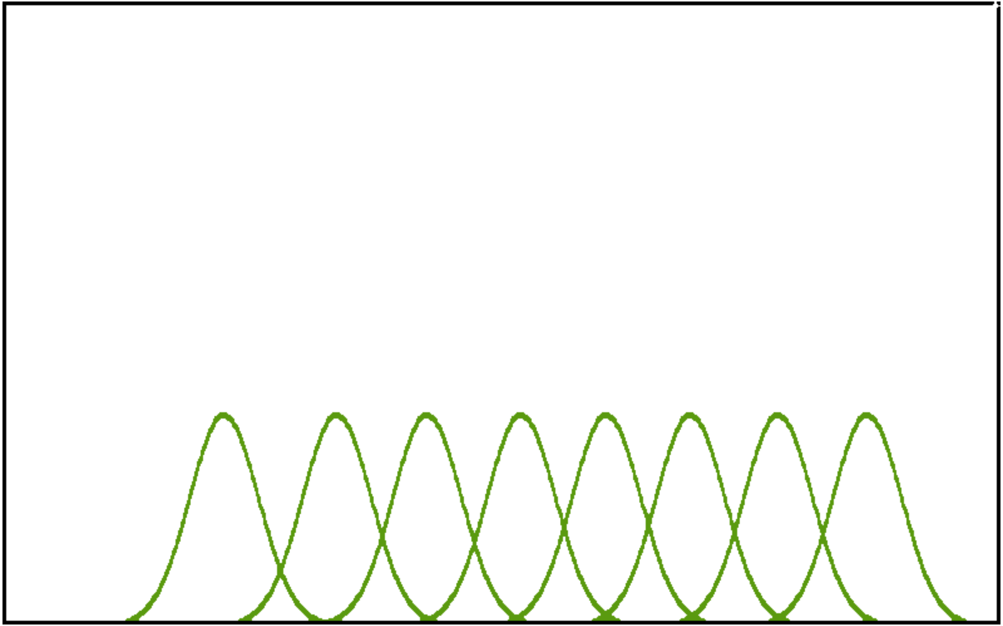
Benefits:

- the (conditional) standard deviation does not depend anymore on the distribution parameter;
- heteroskedastic z turns into a homoskedastic $f(z)$.

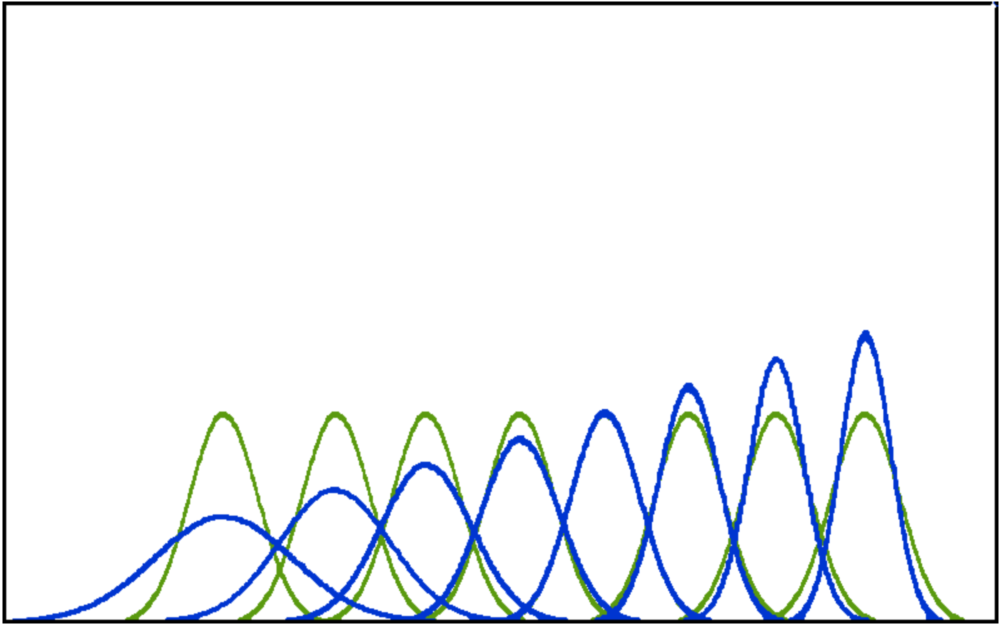
Variance stabilization: heuristics



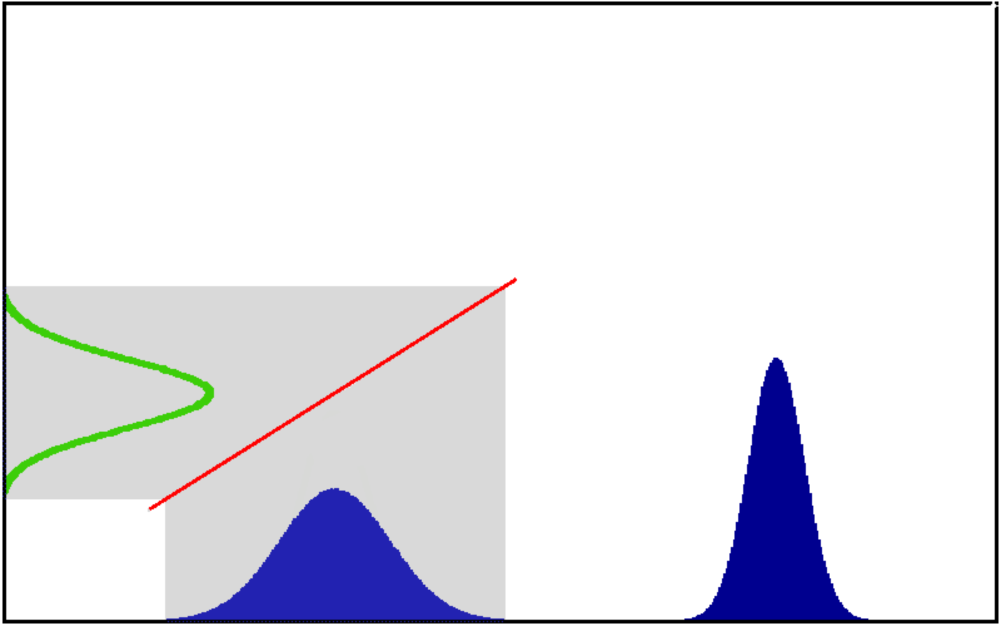
Variance stabilization: heuristics



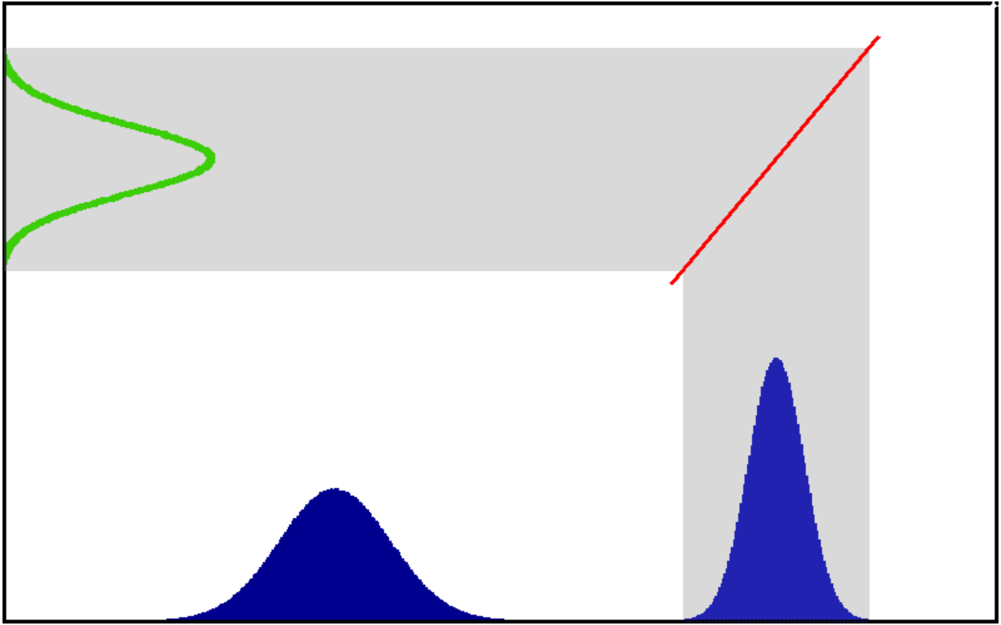
Variance stabilization: heuristics



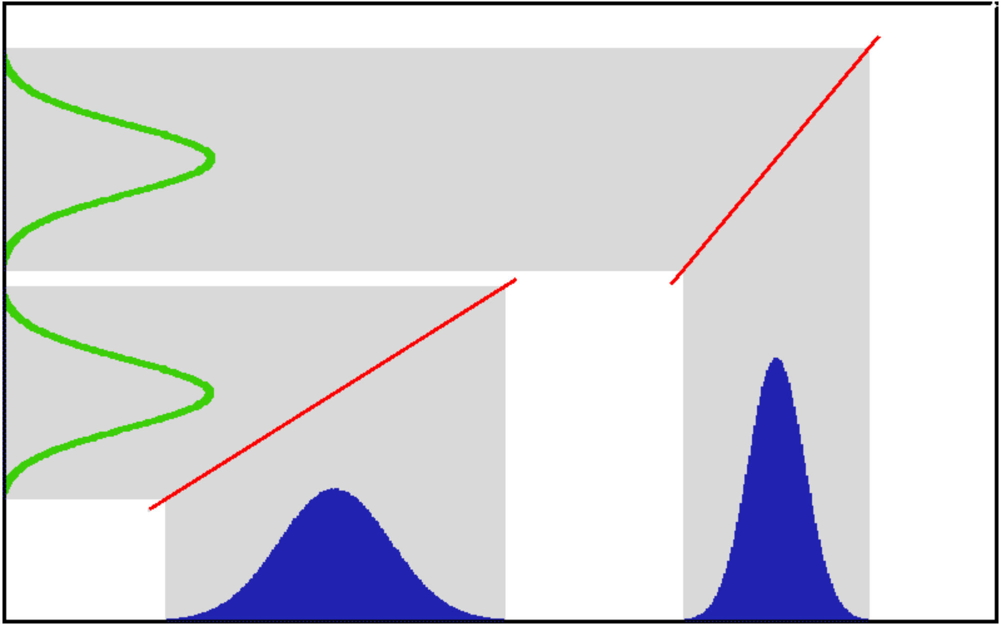
Variance stabilization: heuristics



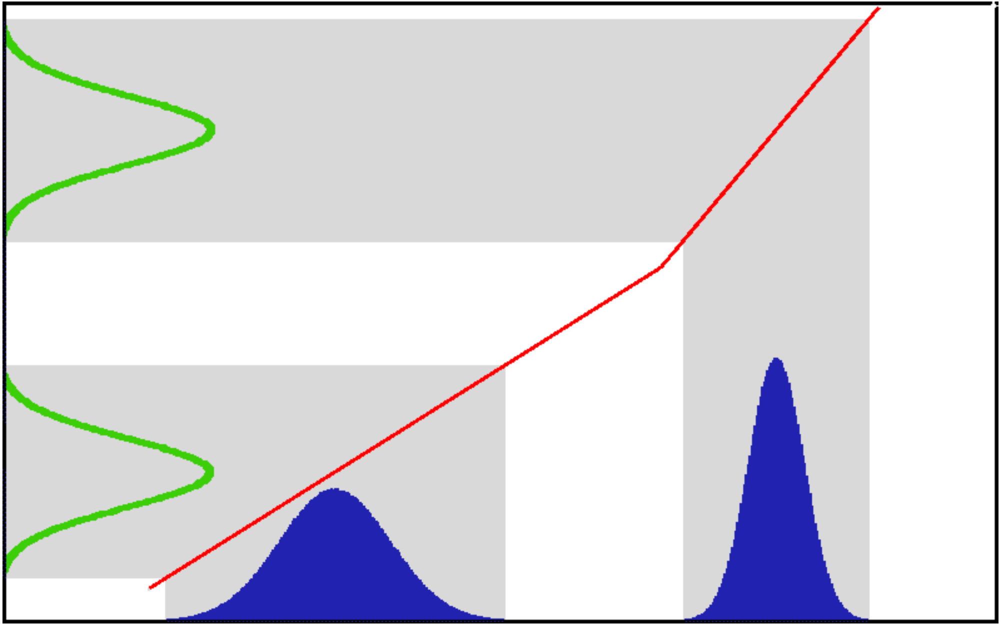
Variance stabilization: heuristics



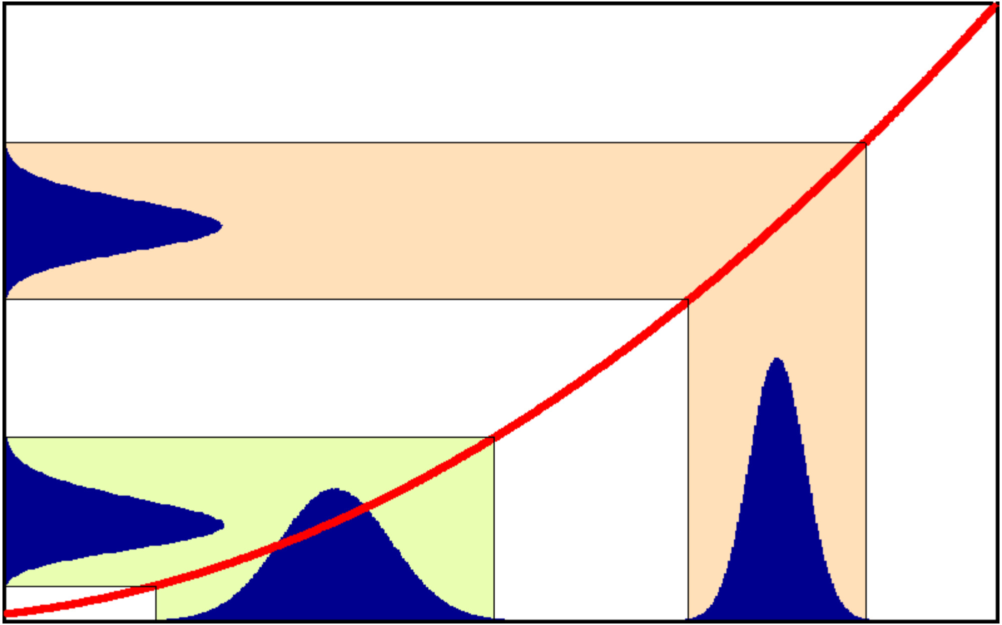
Variance stabilization: heuristics



Variance stabilization: heuristics



Variance stabilization: heuristics



Variance stabilization: heuristics

Classic **heuristic** stabilizer as indefinite integral form

$$f(z) = \int^{z} \frac{1}{\sigma(\theta)} d\mu(\theta) . \tag{5}$$

Idea: consider a local first-order expansion of f at $\mu(\theta)$
(i.e., assume $\sigma(\theta)$ locally constant),

$$f(z) \simeq f(\mu(\theta)) + (z - \mu(\theta)) \frac{\partial f}{\partial z}(\mu(\theta)) ,$$

We have

$$\mathrm{std}\{f(z)|\theta\} \simeq \frac{\partial f}{\partial z}(\mu(\theta)) \sigma(\theta) ,$$

then impose $\mathrm{std}\{f(z)|\theta\} = 1$ and obtain the indefinite integral (5).

Known and used already in the 1930's (e.g., Tippett 1934, Bartlett 1936), often rediscovered in signal processing (e.g., Prucnal&Saleh 1981, Arsenault&Denis 1981, Kasturi et al. 1983, Hirakawa&Parks 2006).

Very rough, but useful as a first guess: nearly all classical stabilizers can be seen as a slight modification of (5).

Exact variance stabilization is typically impossible to achieve

Positive result: multiplicative noise

$$f\left(z\right)=\log\left|z\right|$$

Negative result: Bernoulli

Binary samples $z\in\{0,1\}$ of the Bernoulli distribution with parameter $\theta=E\left\{z|\theta\right\}$ cannot be stabilized to the same constant variance for different values of θ:

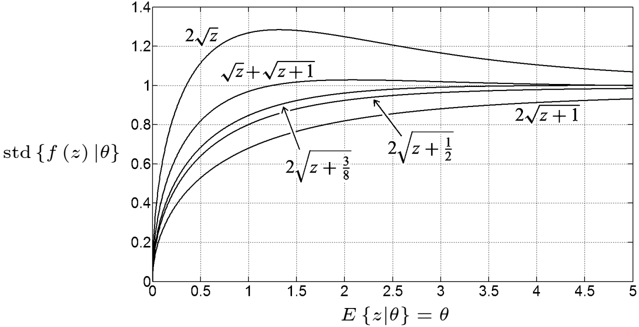
$$E\left\{g\left(z\right)|\theta\right\}=\theta g\left(1\right)+\left(1-\theta\right)g\left(0\right)$$

$$\mathrm{var}\left\{g\left(z\right)|\theta\right\}=E\left\{\left(g\left(z\right)-E\left\{g\left(z\right)|\theta\right\}\right)^{2}|\theta\right\}=\left(g\left(0\right)-g\left(1\right)\right)^{2}\theta\left(1-\theta\right).$$

Exact stabilization is not possible for Poisson, Binomial, and most other families used in applications.

In practice, we deal with either *approximate* or *asymptotic* stabilization.

Classical variance stabilization for Poisson



$$f\left(z\right)=\int^{z}\frac{1}{\sigma(\theta)}d\mu\left(\theta\right)=\int^{z}\frac{1}{\sqrt{\theta}}d\mu\left(\theta\right)=2\sqrt{z}.$$

Bartlett 1936: $2\sqrt{z+\frac{1}{2}}$

Anscombe 1948: $2\sqrt{z+\frac{3}{8}}$ (Anscombe attributes it to A.H.L. Johnson)

Freeman&Tukey 1950: $\sqrt{z}+\sqrt{z+1}$

In the same way stabilizers were derived for the Binomial and Negative Binomial distribution families ("angular" transformations based on the arcsin and hyperbolic arcsin).

Variance stabilization for Poisson and related

Murtagh, Starck, and Bijaoui, 1995: Generalized Anscombe transformation (GAT) for Poisson-Gaussian noise.

GAT is a family of VSTs parametrized by the Poisson gain α and the Gaussian std σ:

$$f_{\alpha,\sigma}(z) = \begin{cases} \frac{2}{\alpha}\sqrt{\alpha z + \frac{3}{8}\alpha^2 + \sigma^2}, & z \geq -\frac{3}{8}\alpha - \frac{\sigma^2}{\alpha} \\ 0, & z < -\frac{3}{8}\alpha - \frac{\sigma^2}{\alpha} \end{cases}.$$

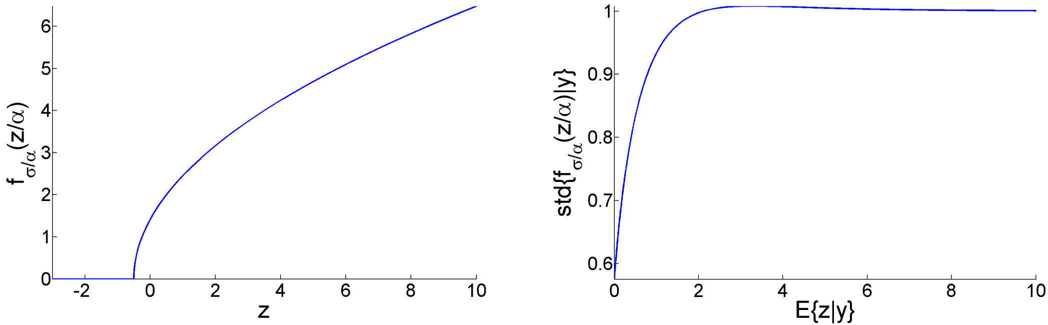
Asymptotically accurate stabilization for large y: $\operatorname{var}\{f_{\alpha,\sigma}(z) \mid y\} = 1 + \mathcal{O}(y^{-2})$
Poor stabilization for small y.

Fryzlewicz, Nason, et al. 2004-2008: wavelet-Fisz transforms that return spectra having approximately constant variance.

Zhang, Fadili, and Starck, 2008: Generalization of Anscombe for filtered (i.e. for linear combinations of) Poisson-Gaussian variates.

All these results enjoy some form of asymptotic optimality, but good stabilization for small θ is never achieved.

Generalized Anscombe transformation



(a) GAT for $\sigma = 0.357$ ($\alpha = 1$)

(b) Stabilized standard deviation. obtained with the GAT in (a)

Variance stabilization: three milestone works

- Curtiss 1943: general asymptotic theorems are proved (and later Bar-Lev&Enis 1990: alternative formulation)
  - gave theoretical support to empirical stabilizers that were already used (and also to others yet to appear).
- Efron 1981: existence of transformations for exact variance stabilization and/or perfect normalization.
  - formalizes sufficient conditions for existence of exact stabilizers ("general transformation families" framework), and provides their analytical expressions.
  - results are nonparametric and nonasymptotic.
  - difficult to use in practice (assumes too much smoothness and invertibility of parametrized mappings).
- Tibshirani 1986: AVAS procedure for regression
  - approximate variance stabilizing transformations are iteratively computed by recursive application of the integral stabilizer (iterative refinement of the stabilizer)
  - developed for data-driven application, hints about potential use for random variables.
  - nonparametric and nonasymptotic.

Exact stabilization for general transformation families (Efron 1981)

Exact stabilization is possible at least for some special classes of distribution families.

General scaled transformation family:

$$z = g^{-1}\left(p\left(\theta\right) + q\left(\theta\right)w\right),$$

where $w \sim \mathcal{N}(0,1)$ and g, p and q are smooth functions.

General transformation family has $q\left(\theta\right) \equiv q$.

Let z follow a general transformation family, $\operatorname{pdf}\left[z|\theta\right]$ be the conditional p.d.f. of z, and $\vartheta\left(\theta\right) = \operatorname{med}\left\{z|\theta\right\}$ be the conditional median of z given θ. The *exact* VST f can be computed as:

$$f\left(z\right) = \int^{z} \frac{\operatorname{pdf}\left[z|\theta\right]\left(\vartheta\right)}{\phi\left(0\right)} d\vartheta \qquad \text{(integration w.r.t. median)},$$

where ϕ is the p.d.f. of the standard normal $\mathcal{N}(0,1)$.

Optimization of VSTs: Motivation

- It is typically impossible to achieve simultaneously good stabilization for all parameter values (see Freeman & Tukey): thus, when a stabilizer appears to be better than another for some values of the parameter, it is likely that for other values it is actually worse. In this sense, there might be no "best stabilizer".

- There is no universal objective criterion for assessing the goodness of a stabilizer. Simply demanding $\mathrm{std}\left\{f\left(z\right)|\theta\right\}$ to be as close as possible to 1 is vague and ambiguous.

- Common stabilizing transformations are often based on coarse asymptotics, developed between the 1930's and 1950's without leveraging any numerical optimization.

(F.2009)

Variance stabilization as a minimization problem

Let

$$e_f(\theta) = \sigma_f(\theta) - c$$

be the local error because of inexact stabilization (where locality is intended by the conditioning on θ) and define a global cost functional as

$$F(f) = \int |e_f(\theta)|\, d\theta. \tag{6}$$

We may formulate the variance stabilization problem as the solution of

$$\operatorname*{argmin}_f F(f) \tag{7}$$

Variance stabilization is exact only when $F(f) = 0$ for some f.

Minimization needs to be constrained to some particular class of functions, such as strictly monotone, Lipschitz, smooth functions, etc.

Variance stabilization as a minimization problem

We have seen that it makes little sense to aim at *exact* variance stabilization simultaneously for *all* parameter values.

We consider a separable *weighted cost functional (stabilization functional)* of the form

$$F(f) = \int_{\Theta} w_{\theta}(\theta)\, w_{e}(e_{f}(\theta))\, d\theta, \tag{8}$$

where the weight functions w_θ and w_e provide different weighting for the different values of θ and different stabilization errors $e_f(\theta)$, respectively.

In particular, we design special weights w_e that *favor approximate stabilization while ignoring very large stabilization errors.*

Stabilization functional

Let $\gamma_{\mathrm{u}},\gamma_{\mathrm{l}} \leq 1$, $r'_{\mathrm{u}},r'_{\mathrm{l}} \geq 0$, $r''_{\mathrm{u}} \geq r'_{\mathrm{u}}$, $r''_{\mathrm{l}} \geq r'_{\mathrm{l}}$, $o_{\mathrm{u}},o_{\mathrm{l}} \geq 1$ be some real constants and χ_{-} be the characteristic (indicator) function of a set - .

We define the weights w_e as

$$w_e\left(e_f\left(\theta\right)\right) = \left|\varphi\left(\overline{e_f}\left(\theta\right)\right)\overline{e_f}\left(\theta\right)\right|,$$

where

$$\begin{aligned}\overline{e_f}\left(\theta\right) &= \overline{\sigma_f}\left(\theta\right) - c = \max\left\{-r''_{\mathrm{l}}, \min\left\{r''_{\mathrm{u}}, e_f\left(\theta\right)\right\}\right\}, \\ \overline{\sigma_f}\left(\theta\right) &= \max\left\{c - r''_{\mathrm{l}}, \min\left\{c + r''_{\mathrm{u}}, \sigma_f\left(\theta\right)\right\}\right\},\end{aligned}$$

and with the function φ given by

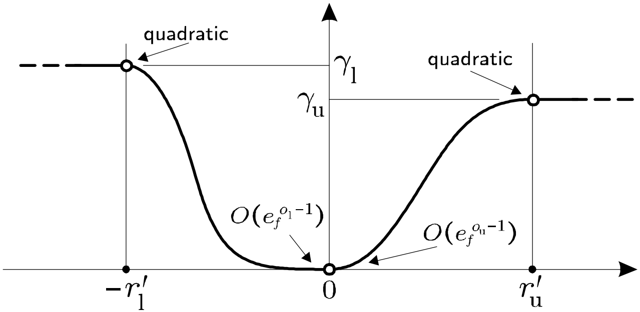
$$\begin{aligned}\varphi\left(e_f\right) &= \gamma_{\mathrm{u}} \cdot \chi_{[0,+\infty)}\left(e_f\right)\left\{\left[1 - \left(\frac{e_f - r'_{\mathrm{u}}}{r'_{\mathrm{u}}}\right)^2\right]^{(o_{\mathrm{u}}-1)} \chi_{(-\infty,r'_{\mathrm{u}})}\left(e_f\right) + \chi_{[r'_{\mathrm{u}},+\infty)}\left(e_f\right)\right\} + \\ &\quad + \gamma_{\mathrm{l}} \cdot \chi_{(-\infty,0)}\left(e_f\right)\left\{\left[1 - \left(\frac{e_f + r'_{\mathrm{l}}}{r'_{\mathrm{l}}}\right)^2\right]^{(o_{\mathrm{l}}-1)} \chi_{\left(-r'_{\mathrm{l}},+\infty\right)}\left(e_f\right) + \chi_{\left(-\infty,-r'_{\mathrm{l}}\right]}\left(e_f\right)\right\}.\end{aligned}$$

Stabilization functional

The clipped argument $\overline{e_f}(\theta)$ cannot distinguish stabilization errors larger than r''_l,r''_u, while the multiplication against the function φ increases the order of the stabilization errors from 1 to o_l,o_u. Note that for a positive (resp. negative) argument, the function φ has a zero of order $o_\mathrm{u}-1$ ($o_\mathrm{l}-1$) at zero and becomes constant (with quadratic-smooth joint) equal to γ_u (γ_l) starting from r'_u (r'_l).

Thus, the cost functional $F(f)$ takes the form

$$F(f) = \int_\Theta w_\theta(\theta) \left|\varphi\left(\overline{e_f}(\theta)\right)\overline{e_f}(\theta)\right| d\theta.$$



The function φ.

Iterative integral algorithm for optimizing f

0. Initialize

$f_0(z) = z$ (identity) or an arbitrary (non-optimal) stabilizer
f monotone increasing

Iterate the following three stages:

1. Compute statistics

$$\vartheta_k(\theta) = \text{med}\{f_k(z)|\theta\} = f_k(\text{med}\{z|\theta\})$$
$$\sigma_k(\theta) = \text{std}\{f_k(z)|\theta\}$$

2. Compute stabilization refinement

$$r_k(z) = \int^z I_k(\theta)\, d[\vartheta_k(\theta)] \qquad \text{(integration with respect to the median)}$$

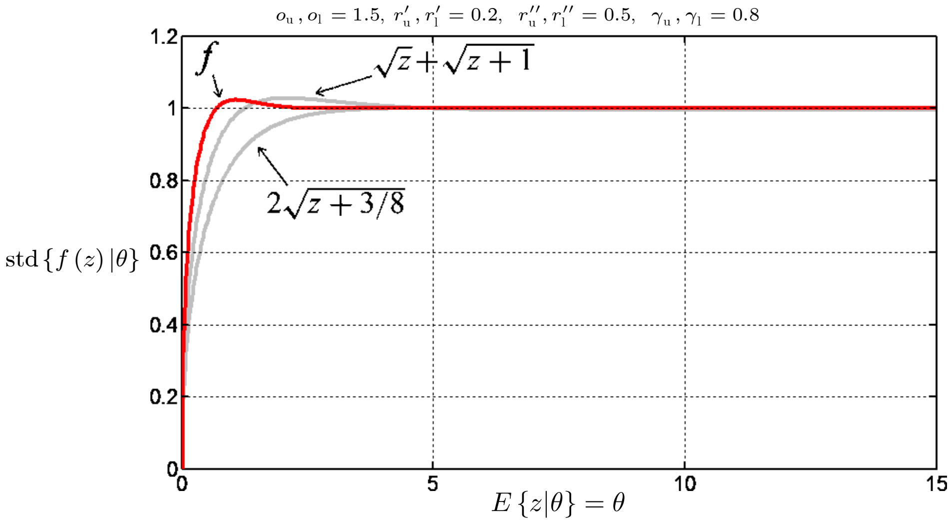
where

$$I_k(\theta) = 1 - \frac{w_\theta(\theta)\,\varphi(\overline{e_k}(\theta))\,\overline{e_k}(\theta)}{\overline{\sigma_k}(\theta)},$$
$$\overline{e_k}(\theta) = \overline{\sigma_k}(\theta) - c = \max\{-r''_{\mathrm{l}}, \min\{r''_{\mathrm{u}}, e_k(\theta)\}\}$$
$$\overline{\sigma_k}(\theta) = \max\{c - r''_{\mathrm{l}}, \min\{c + r''_{\mathrm{u}}, \sigma_{fk}(\theta)\}\},$$

3. Compose

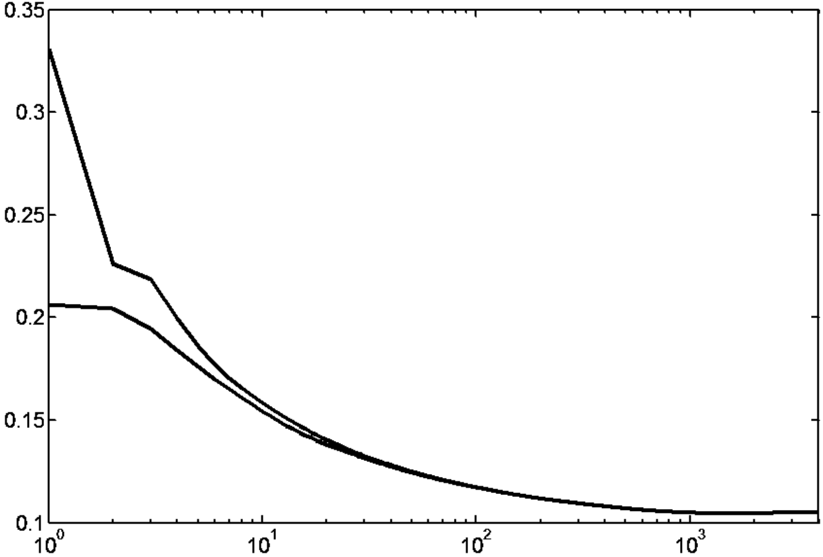
$$f_{k+1}(z) = r_k(f_k(z))$$

Optimization of Poisson stabilizer (iterative integral)



Optimization of Poisson stabilizer (iterative integral)

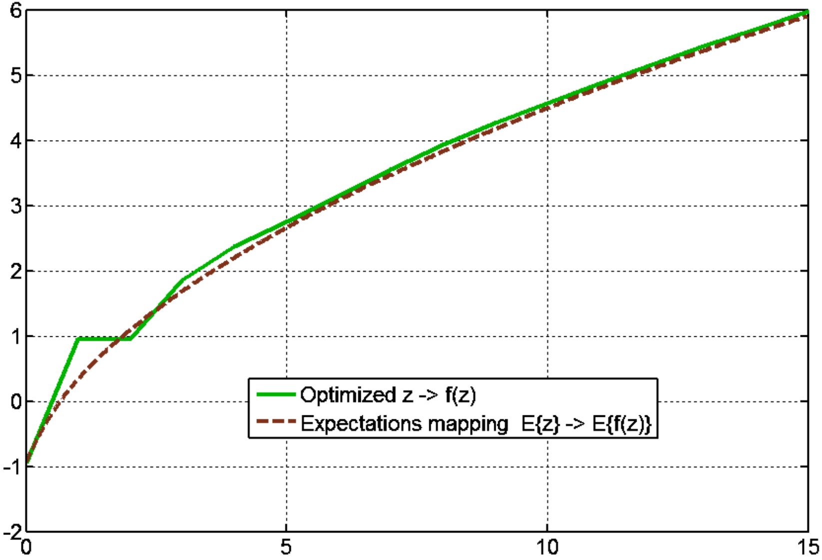
$o_{\mathrm{u}}, o_{\mathrm{l}} = 1.5,\ r'_{\mathrm{u}}, r'_{\mathrm{l}} = 0.2,\ \ r''_{\mathrm{u}}, r''_{\mathrm{l}} = 0.5,\ \ \gamma_{\mathrm{u}}, \gamma_{\mathrm{l}} = 0.8$



stabilization functional $F(f_k)$ vs. iterations (log scale)
$f_0 = \sqrt{z} + \sqrt{z+1}$ (lower) and $f_0 = 2\sqrt{z + 3/8}$ (higher)

Optimization of Poisson stabilizer (iterative integral)[90]

$o_{\mathrm{u}}, o_{\mathrm{l}} = 1.5,\ r'_{\mathrm{u}}, r'_{\mathrm{l}} = 0.2,\ \ r''_{\mathrm{u}}, r''_{\mathrm{l}} = 0.5,\ \ \gamma_{\mathrm{u}}, \gamma_{\mathrm{l}} = 0.8$



variance-stabilizer f and the mapping $E\{z|\theta\} \mapsto E\{f(z)|\theta\}$

stabilization functional $F(f) = 0.1051$

Optimization by iterative integral vs. direct search

Convergence of the iterative integral algorithm, , with monotonically decreasing cost, was verified experimentally, up to the numerical precision of the algorithm, in extensive tests.

However, its limit may differ from the minimizer of the stabilization functional.

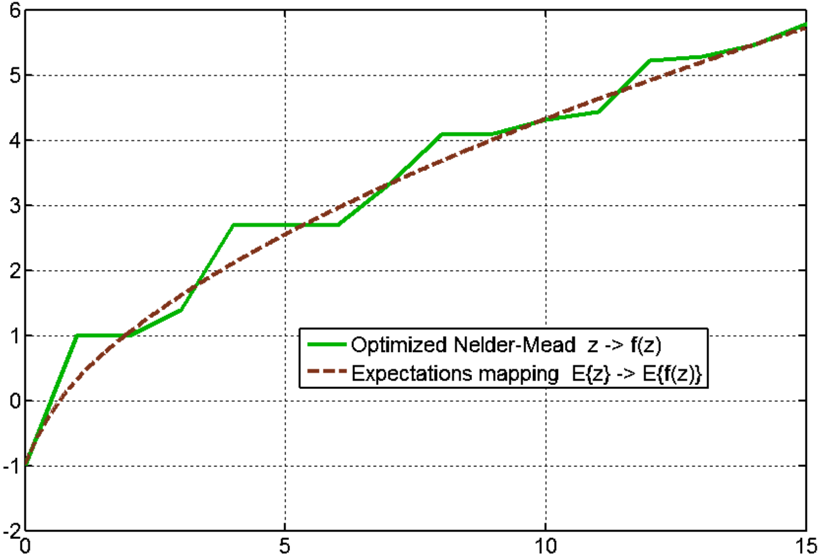
Further drawbacks:
— computational aspects involved in the evalutation of the integrals
— a proof of minimization seems very difficult to achieve (similar situation as for AVAS algorithm)

A practical way to circumvent these issues is to solve the minimization by direct search, which is particularly feasible for discrete distributions.

We use Nelder-Mead downhill simplex algorithm.

Optimization by direct search

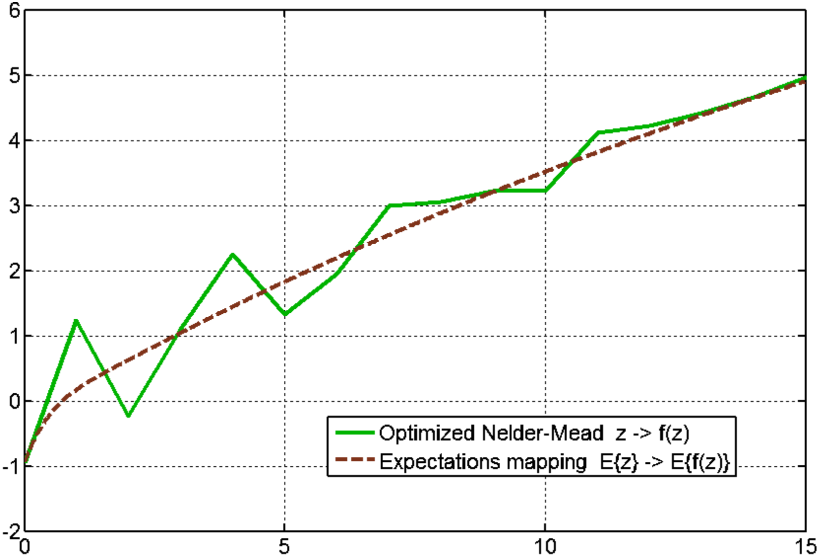
$o_{\rm u}\,, o_{\rm l}\, = 1.5,\ r'_{\rm u}\,, r'_{\rm l} = 0.2,\ \ r''_{\rm u}\,, r''_{\rm l} = 0.5,\ \ \gamma_{\rm u}\,, \gamma_{\rm l} = 0.8$



variance-stabilizer f and the mapping $E\{z|\theta\} \mapsto E\{f(z)|\theta\}$
stabilization functional $F(f) = 0.096$

Optimization by direct search: relaxing monotonicity

$o_u, o_l = 1.5,\ r'_u, r'_l = 0.2,\ \ r''_u, r''_l = 0.5,\ \ \gamma_u, \gamma_l = 0.8$

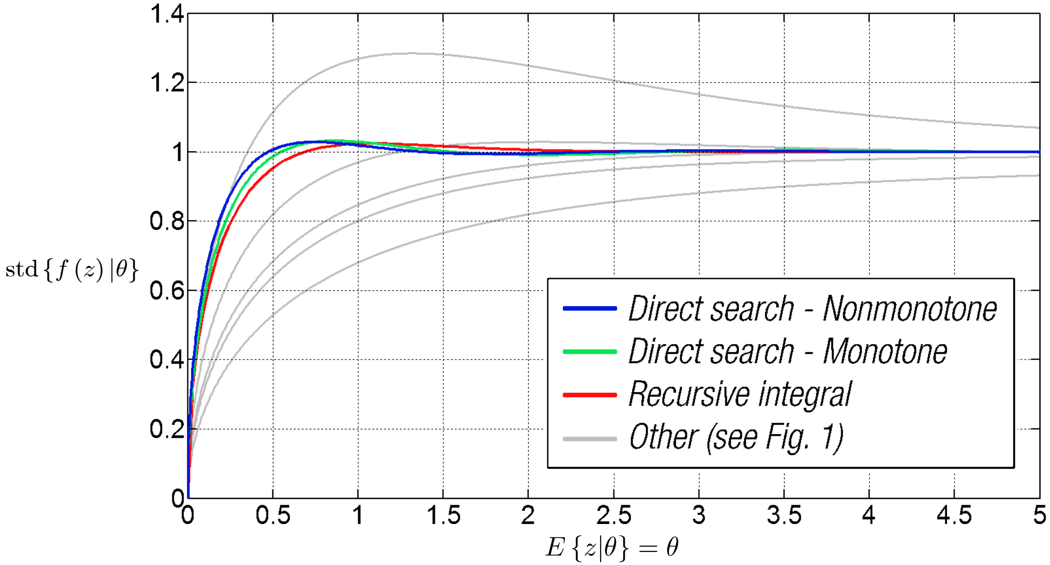


variance-stabilizer f and the mapping $E\{z|\theta\} \mapsto E\{f(z)|\theta\}$

stabilization functional $F(f) = 0.079$

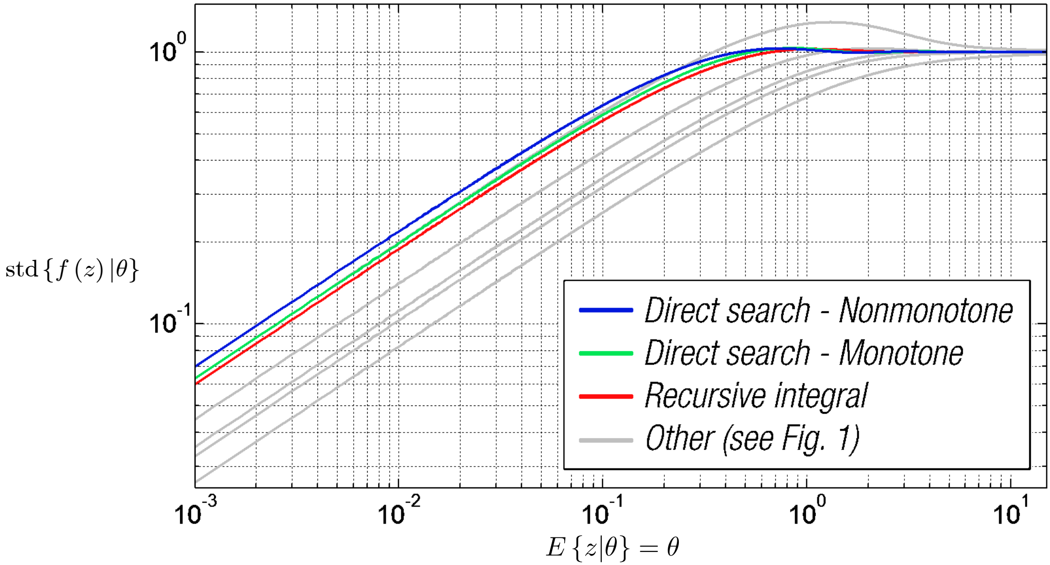
Optimization by direct search

$o_u, o_l = 1.5, \; r'_u, r'_l = 0.2, \; r''_u, r''_l = 0.5, \; \gamma_u, \gamma_l = 0.8$



Optimization by direct search

$o_u, o_l = 1.5, \; r'_u, r'_l = 0.2, \;\; r''_u, r''_l = 0.5, \;\; \gamma_u, \gamma_l = 0.8$



Regularization of VSTs through penalization of the stabilization functional

Introduce penalty terms into the stabilization functional $F(f)$:

$$F(f) = F_{\text{stabil}}(f) + \lambda_{\text{smooth}} \cdot F_{\text{smooth}}(f) + \lambda_{\text{asympt}} \cdot F_{\text{asympt}}(f) + \lambda_{\text{inverse}} \cdot F_{\text{inverse}}(f),$$

where $\lambda_{\text{smooth}}, \lambda_{\text{inverse}}, \lambda_{\text{asympt}} \geq 0$ are penalty parameters.

Constrain direct-search optimization to VSTs f for which the expectations mapping

$$E\{z|\theta\} \mapsto E\{f(z)|\theta\}$$

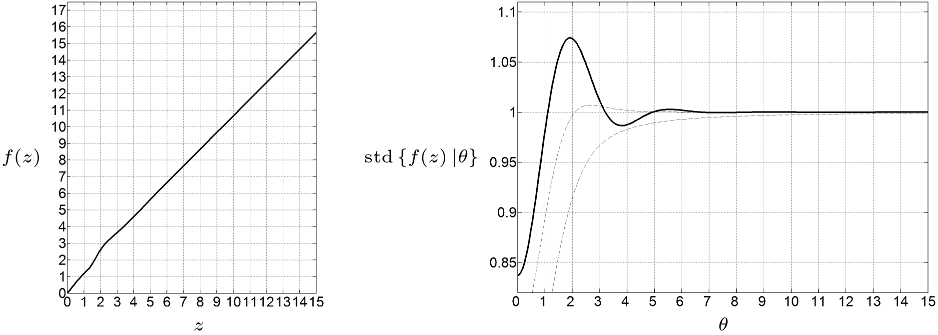
is strictly increasing, so to be able to define the *exact unbiased inverse* $\mathcal{I}_f$

$$\mathcal{I}_f : E\{f(z)|\theta\} \mapsto E\{z|\theta\}.$$

(F.ISBI2011)

Accuracy of stabilization	$F_{\text{stabil}}(f) = \int_{\theta_{\min}}^{\theta_{\max}} (\text{std}\{f(z)\|\theta\} - 1)^2 \, d\theta$
Smoothness of f	$F_{\text{smooth}}(f) = \int_{z_{\min}}^{z_{\max}} (f''(z))^2 \, dz$
Asymptotic	$F_{\text{asympt}}(f) = \int_{z_{\min}}^{z_{\max}} \frac{1}{(z_{\min,\max} - z + \epsilon)^4} (f(z) - f_{\text{asympt}}(z))^2 \, dz$
... other penalties, e.g. higher-order moments of the distribution of $f(z)$	

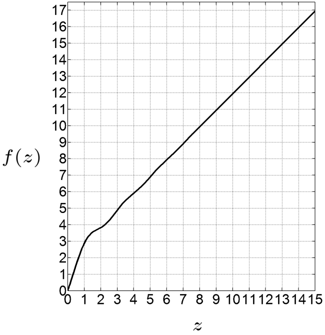
Regularized VSTs for the Rician family of distribution

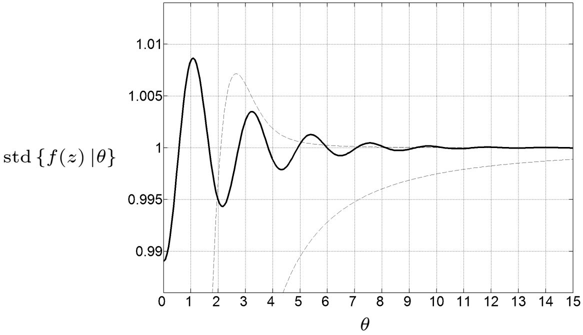


$$\lambda_{\mathrm{asympt}} = 1,\ \lambda_{\mathrm{smooth}} = 10^{-2},\ \lambda_{\mathrm{inverse}} = 10^{-\frac{1}{2}}.$$

(F.ISBI2011)

Regularized VSTs for the Rician family of distribution

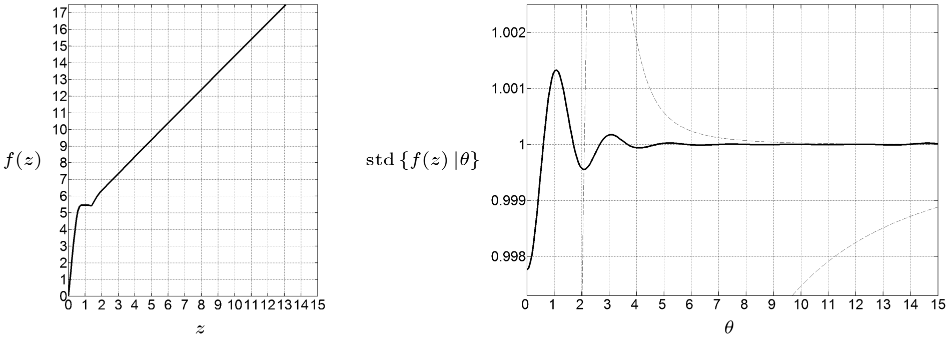




$$\lambda_{\mathrm{asympt}} = 1,\ \lambda_{\mathrm{smooth}} = 10^{-4},\ \lambda_{\mathrm{inverse}} = 0.$$

(F.ISBI2011)

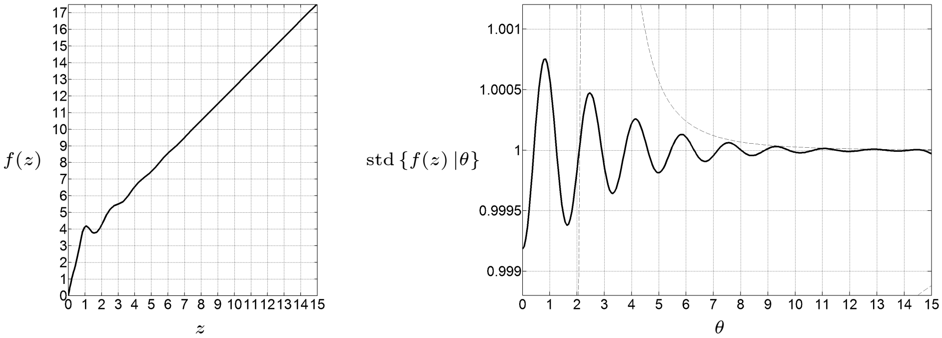
Regularized VSTs for the Rician family of distribution



$$\lambda_{\mathrm{asympt}} = 1,\ \lambda_{\mathrm{smooth}} = 10^{-6},\ \lambda_{\mathrm{inverse}} = 10^{-\frac{5}{2}}.$$

(F.ISBI2011)

Regularized VSTs for the Rician family of distribution



$\lambda_{\mathrm{asympt}} = 1$, $\lambda_{\mathrm{smooth}} = 10^{-8}$, $\lambda_{\mathrm{inverse}} = 0$.

(F.ISBI2011)

Optimization of rational polynomial VST

To effectively regularize the optimization, we can also seek the solution within a specific class of functions.

Poisson-Gaussian VST optimization
Find stabilizer by optimizing the coefficients of polynomials $P(z)$ and $Q(z)$ in

$$f_{1,\sigma}(z) = 2\sqrt{\frac{\sum_{i=0}^{N} p_i z^i}{\sum_{i=0}^{M} q_i z^i}} = 2\sqrt{\frac{P(z)}{Q(z)}}, \tag{9}$$

Constrain polynomials such that the VST necessarily approaches the GAT asymptotically. In this way, the optimized VST always attains good asymptotic stabilization:

$$\frac{P(z)}{Q(z)} - z - \frac{3}{8} - \sigma^2 \to 0 \text{ as } z \to +\infty \tag{10}$$

at a rate of $\mathcal{O}(z^{-1})$. For $N = 3$ we have

$$f_{1,\sigma}(z) = 2\sqrt{\frac{p_3 z^3 + p_2 z^2 + p_1 z + p_0}{p_3 z^2 + \left[p_2 - p_3\left(3/8 + \sigma^2\right)\right] z + 1}}, \tag{11}$$

which depends solely on $\{p_i\}_{i=0}^{3}$.

(MF.TIP2014)

Optimization of rational polynomial VST for Poisson-Gaussian noise

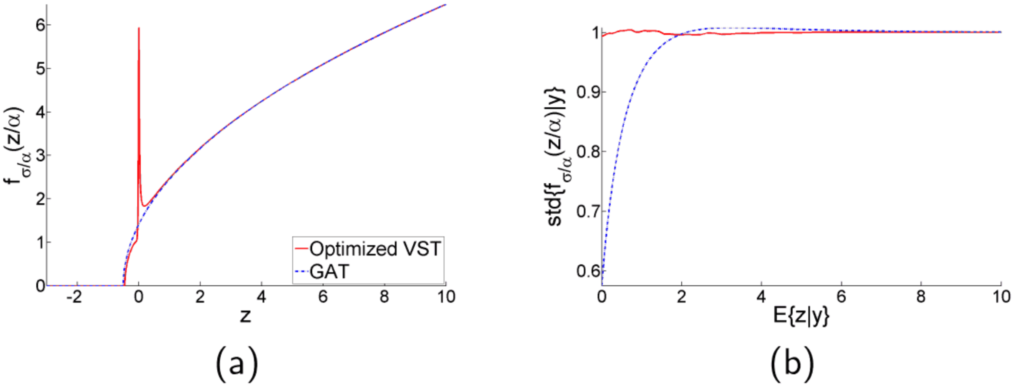


Figure: (a) Optimized rational VST $f_{1,\sigma}(z)$ and the GAT, for $\sigma = 0.357$ ($\alpha = 1$). (b) Stabilized standard deviation obtained with the VSTs in (a).

(MF.TIP2014)

3. Noise parameter estimation

3.1. Scatterplot methods for signal-depedent noise estimation

Goal: estimate the standard-deviation function.

Approach: build a scatterplot (mean, st.dev), fit a curve.

Employ some local or nonlocal low-pass (for mean) and high-pass filtering (for st.dev.); e.g., split image into wavelet approximation and detail coefficients.

Challenge: ignore edges or high-frequency texture

1. Partitioning of the codomain to pair mean and st.dev. estimates (*conditioning*)

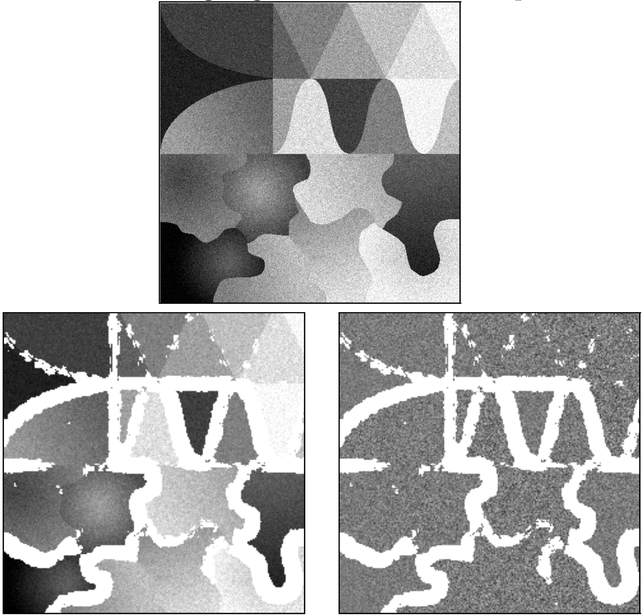
2. Use wavelet approximation coefficients to estimate *conditional* expectations

3. Use wavelet detail coefficients to estimate *conditional* standard-deviation (use MAD)

4. Fit parametric model using nonlinear optimization to maximize posterior likelihood (or any other fitting criterion).

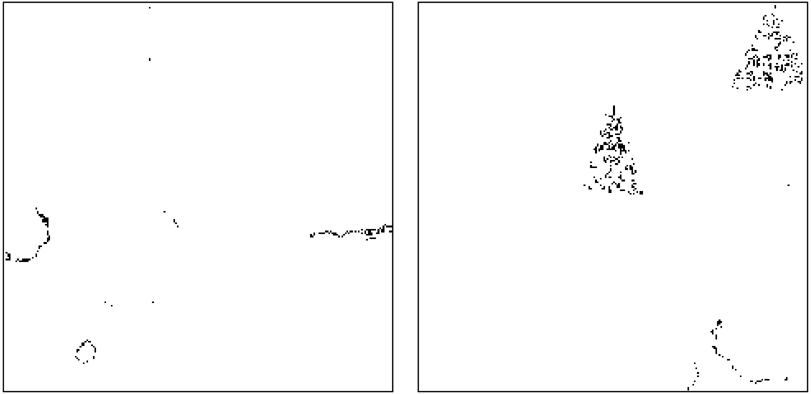
Noise estimation

removal of strong edges and wavelet decomposition



(F.&al.TIP2008)

Noise estimation: codomain partitioning (level sets)

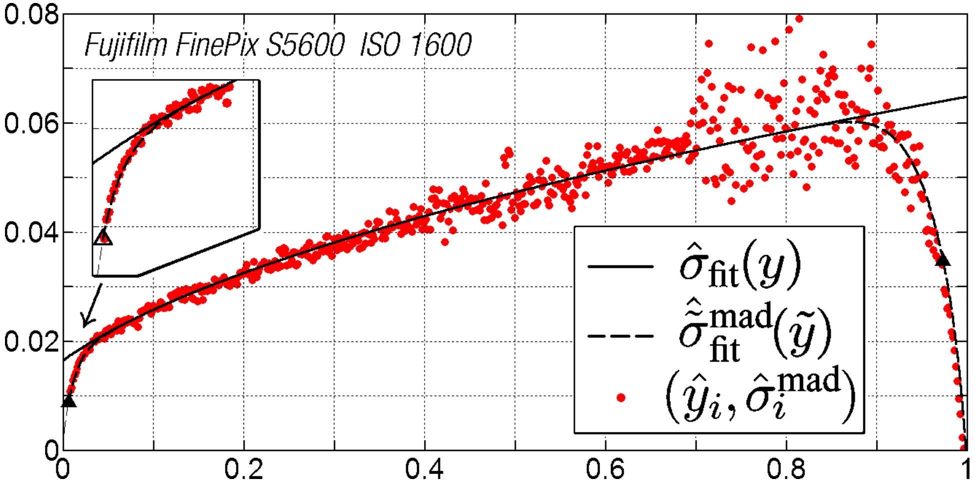


two level sets for different intervals of the codomain partition

(F.&al.SensJ2007,F.&al.TIP2008)

Noise estimation

conditional expectation and conditional std estimation for each level set (red dots)



(F.&al.TIP2008)

Noise estimation: fitting

conditional probability density:

$$\wp\left(\left(\hat{y}_i,\hat{\sigma}_i\right)|\tilde{y}_i=\tilde{y}\right)=\wp\left(\hat{y}_i|\tilde{y}_i=\tilde{y}\right)\wp\left(\hat{\sigma}_i|\tilde{y}_i=\tilde{y}\right)=$$

$$=\frac{1}{2\pi\sqrt{c_i d_i}}\frac{1}{\tilde{\sigma}_{\mathrm{reg}}^2\left(\tilde{y}\right)}e^{-\frac{1}{2\tilde{\sigma}_{\mathrm{reg}}^2\left(\tilde{y}\right)}\left(\frac{\left(\hat{y}_i-\tilde{y}\right)^2}{c_i}+\frac{\left(\hat{\sigma}_i-\tilde{\sigma}_{\mathrm{reg}}\left(\tilde{y}\right)\right)^2}{d_i}\right)}.$$

posterior likelihood:

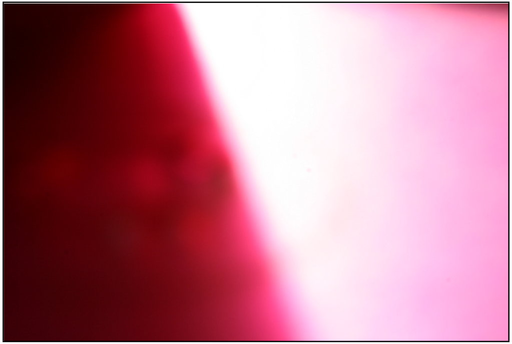
$$\tilde{L}\left(a,b\right)=\prod_{i=1}^{N}\int\wp\left(\left(\hat{y}_i,\hat{\sigma}_i\right)|\tilde{y}_i=\tilde{y}\right)\wp_0\left(y\right)dy$$

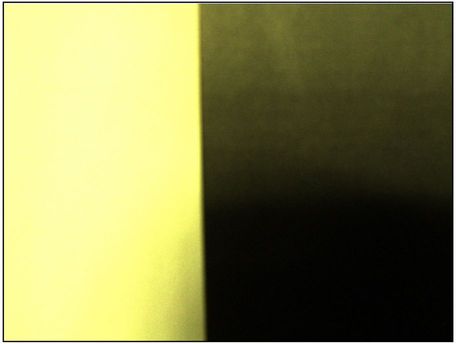
optimization:

$$\begin{aligned}\left(\hat{a},\hat{b}\right) &= \operatorname*{argmax}_{a,b} L\left(a,b\right)=\operatorname*{argmin}_{a,b}-\ln L\left(a,b\right)=\\ &= \operatorname*{argmin}_{a,b}-\sum_{i=1}^{N}\ln\int\wp\left(\left(\hat{y}_i,\hat{\sigma}_i\right)|\tilde{y}_i=\tilde{y}\right)\wp_0\left(y\right)dy.\end{aligned}$$

(F.&al.TIP2008)

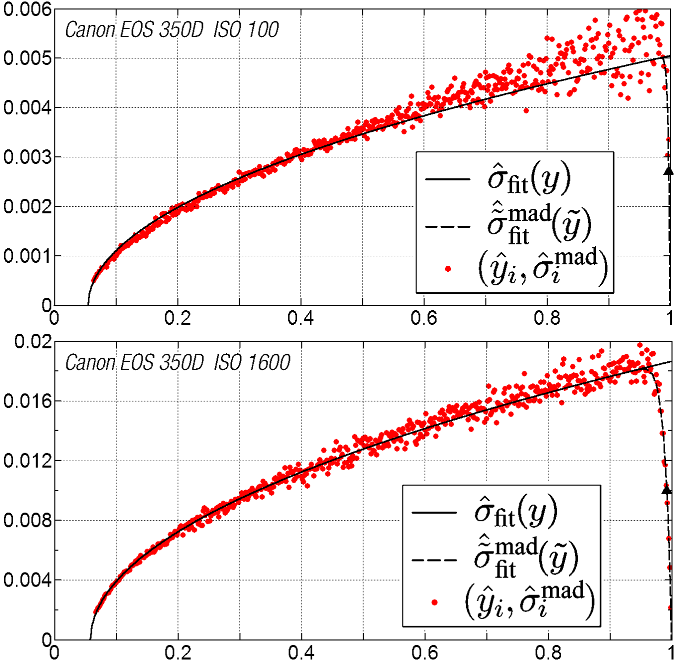
Noise estimation: easy examples





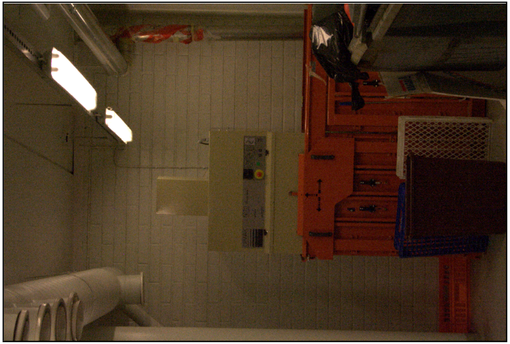
smooth targets with full codomain

Noise estimation: easy examples



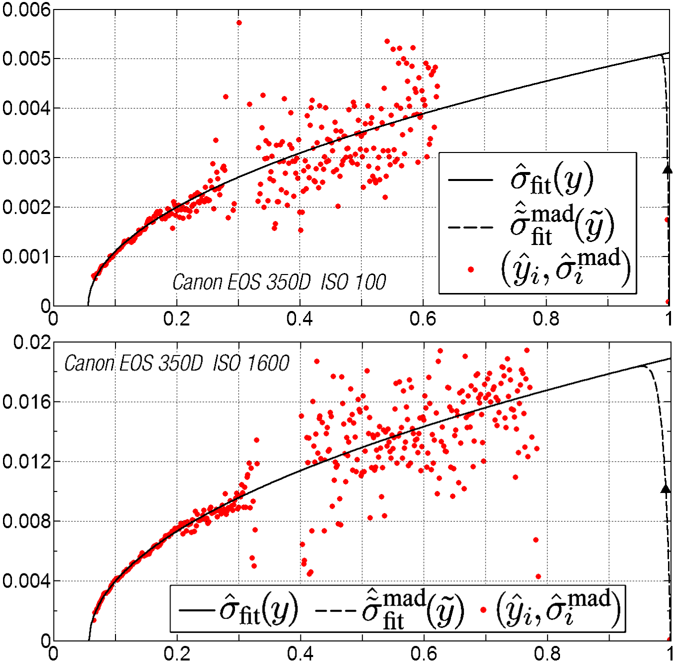
(F.&al.TIP2008)

Importance of a good parametric model



complex targets with incomplete/sparse codomain

Importance of a good parametric model



3.2. Signal-depedent noise estimation using VST

Goal: estimate the standard-deviation function.

Idea: Different standard-deviation functions are typically stabilized by different VSTs: finding a VST that stabilizes the data can be equivalent to finding the standard-deviation function.

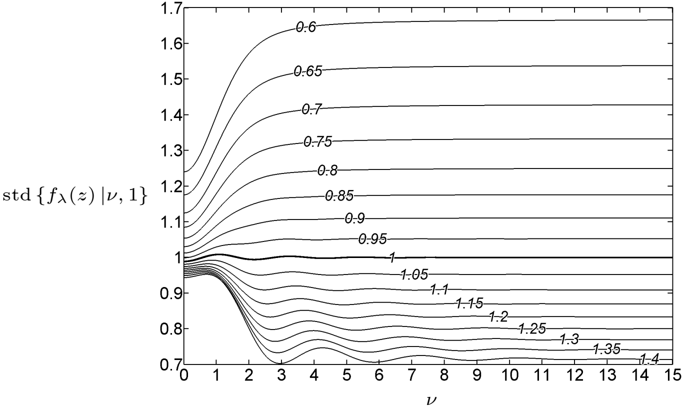
Challenges:

- stabilization is typically inaccurate even when the standard-deviation function is known;
- detecting noise-parameter mismatch

The generic algorithm iterates the following steps:

1. Apply VST $f_{\hat{\sigma}}$ based on current estimate $\hat{\sigma}$ of st.dev. function σ.
2. Assess stabilization of $f_{\hat{\sigma}}(z)$:
   If unable to improve stabilization further, the current $\hat{\sigma}$ is the final estimate;
   else, modify $\hat{\sigma}$ and go to 1.

Rice: Noise-level mismatch



(F.ISBI2011)

Standard deviation of the transformed data std $\{f_\lambda(z)\,|\nu,1\}$, for different values of λ, as indicated by the italic numbers superimposed on the curves. Stabilizer f on page 98.

The stabilizer f_λ is asymptotically affine for large z, with derivative approaching $\frac{1}{\lambda}$. Thus,

$$\operatorname{std}\{f_{\lambda\sigma}(z)\,|\sigma\nu,\sigma\} = \operatorname{std}\{f_\lambda(z)\,|\nu,1\} \underset{\nu\to+\infty}{\longrightarrow} \frac{1}{\lambda}. \qquad (12)$$

In other words, for large ν, *the stabilized standard deviation is approximately equal to the reciprocal of the under- or over-estimation factor.*

Rice: Noise-level estimation

General iterative scheme based on variance stabilization aimed at estimating the value of the σ parameter from a single realization z.

Let $\mathfrak{E}$ denote an estimator of the standard deviation σ of the homoskedastic noise corrupting a signal. Popular examples for estimating σ of AWGN in natural images are the median or mean absolute deviation of the high-pass filtered signal:

$$\mathfrak{E}_{\mathrm{MedianAD}}\{z\} = \mathrm{med}\{|H\{z\}|\}/\Phi^{-1}(3/4),$$

$$\mathfrak{E}_{\mathrm{MeanAD}}\{z\} = \mathrm{mean}\{|H\{z\}|\}\sqrt{\pi/2},$$

where $H\{z\} = z \circledast w_{\mathrm{hi}}$, and w_{hi} is a high-pass convolutional kernel having zero mean and unit L^2-norm,

$$\int w_{\mathrm{hi}} = 0, \qquad \int |w_{\mathrm{hi}}|^2 = 1,$$

such as, e.g., a wavelet function.

(F.ISBI2011)

Rice: Iterative scheme for estimating σ

The proposed scheme is expressed by the following recursive system:

$$\begin{cases} \hat{\sigma}_0 &= \mathfrak{E}\left\{z\right\}, \\ \hat{\sigma}_{k+1} &= \mathfrak{E}\left\{f_{\hat{\sigma}_k}\left(z\right)\right\}\hat{\sigma}_k, \qquad k \geq 0. \end{cases} \tag{13}$$

The idea of this recursion originates from (12). The estimate $\hat{\sigma}_k$ is used to define a variance-stabilizing transformation for z. If the estimated value $\hat{\sigma}_k$ is correct, then the transformation $f_{\hat{\sigma}_k}$ successfully stabilizes the data and when $\mathfrak{E}$ is applied to the stabilized data it should return a value $\mathfrak{E}\left\{f_{\hat{\sigma}_k}\left(z\right)\right\}$ close to 1. If the estimated value $\hat{\sigma}_k$ is not correct (e.g., an under-estimate of σ), then the stabilization is not accurate, being roughly the inverse of the mis-estimation ratio, $\mathfrak{E}\left\{f_{\hat{\sigma}_k}\left(z\right)\right\} \approx \frac{\sigma}{\hat{\sigma}_k}$. Hence, we correct the current estimate $\hat{\sigma}_k$ by multiplying it with $\mathfrak{E}\left\{f_{\hat{\sigma}_k}\left(z\right)\right\}$. Observe that if $\mathfrak{E}\left\{f_{\hat{\sigma}}\left(z\right)\right\} = 1$ for some value $\hat{\sigma}$, then this $\hat{\sigma}$ is a fixed point for (13) and we want the sequence $\hat{\sigma}_k$ to converge to such $\hat{\sigma}$. The system (13) is initialized by the estimator $\mathfrak{E}$ applied on the non-stabilized data z.

Under very general conditions, the iterative scheme (13) is guaranteed to converge with exponential rate to an accurate and stable estimate $\hat{\sigma}$ of the true value σ.

(F.ISBI2011)

Standard deviation contours in Poisson-Gaussian noise

Let $z_{\alpha,\sigma}$ be a Poisson-Gaussian image with (true) parameters α, σ.

Let B be an image block, with $p_B(y)$ being the probability density of y over this block.

Let $\hat{\alpha}, \hat{\sigma}$ be (possibly erroneous) estimates of α, σ.

Consider the VST $f_{\hat{\alpha},\hat{\sigma}}$ (such as GAT or an optimized VST).

Denote the average standard deviation of $f_{\hat{\alpha},\hat{\sigma}}(z_{\alpha,\sigma})$ over B as

$$F_B(\hat{\alpha}, \hat{\sigma}) := \mathfrak{E}_B \{ f_{\hat{\alpha},\hat{\sigma}}(z_{\alpha,\sigma}) \} = \int \mathrm{std} \{ f_{\hat{\alpha},\hat{\sigma}}(z_{\alpha,\sigma}) | y \} \, p_B(y) \, dy.$$

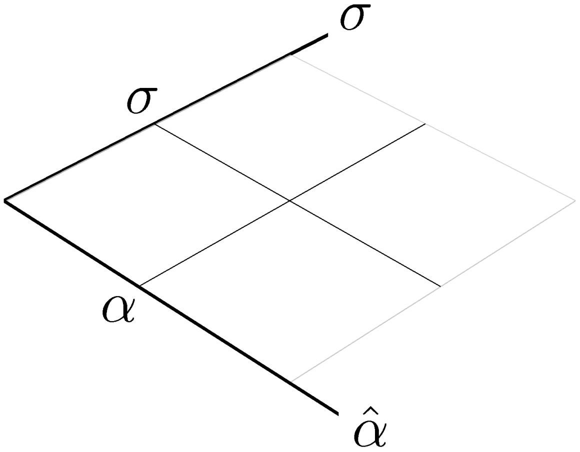
$F_B(\hat{\alpha}, \hat{\sigma})$ is a bivariate function of the parameter estimates $\hat{\alpha}, \hat{\sigma}$.

Under some simplifying assumptions, the unitary standard-deviation contours $F_B(\hat{\alpha}, \hat{\sigma}) = 1$ are smooth curves in a neighbourhood of the true parameter values (α, σ).

We apply the results by devising a VST-based algorithm for estimating α and σ.

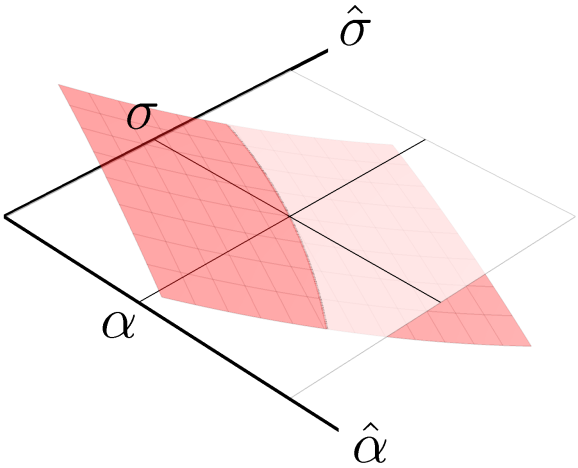
(M.&F.TIP2014)

$(\hat{\alpha}, \hat{\sigma})$ plane and the true parameters (α, σ)



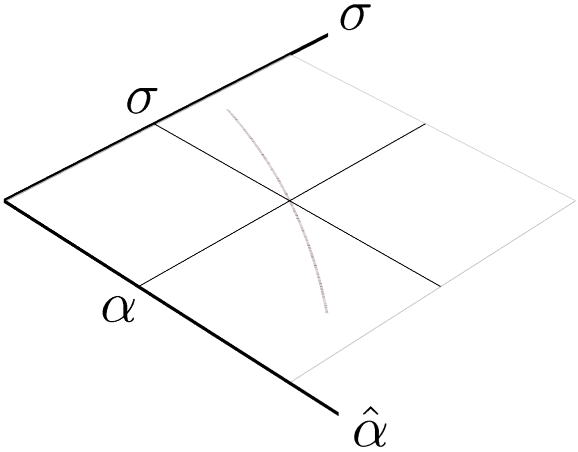
(M.&F.TIP2014)

$(\hat{\alpha}, \hat{\sigma})$ **plane and** $F_B(\hat{\alpha}, \hat{\sigma}) - 1$



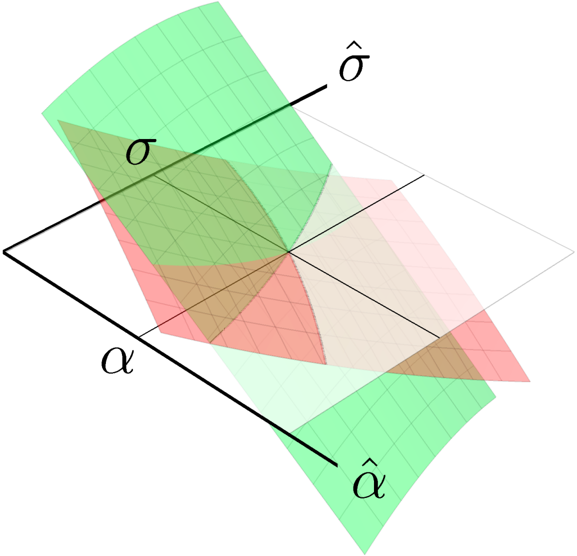
(M.&F.TIP2014)

Unitary contour of $F_B(\hat{\alpha}, \hat{\sigma})$



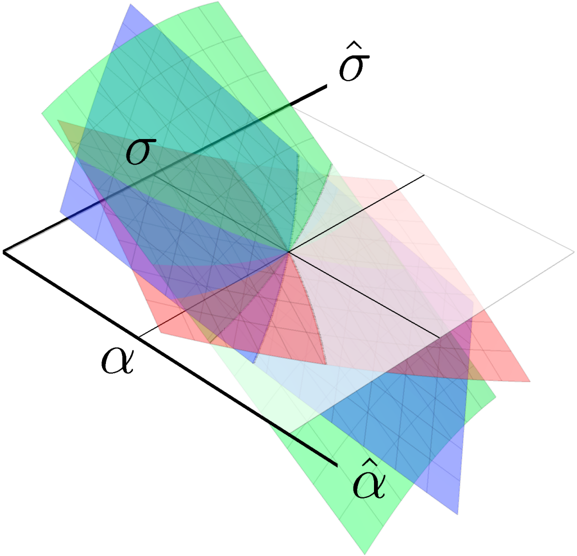
(M.&F.TIP2014)

$F_B(\hat{\alpha}, \hat{\sigma}) - 1$ for different blocks B



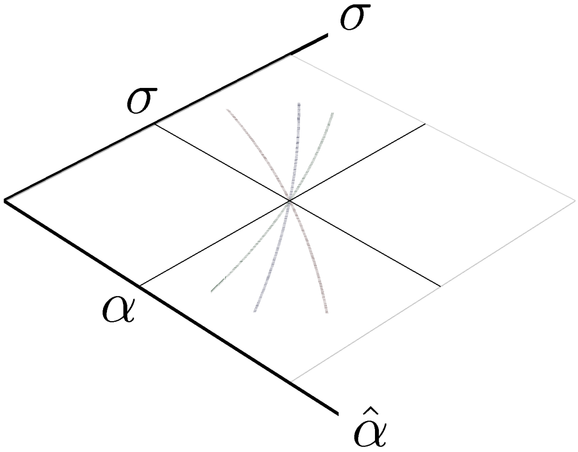
(M.&F.TIP2014)

$F_B(\hat{\alpha}, \hat{\sigma}) - 1$ for different blocks B



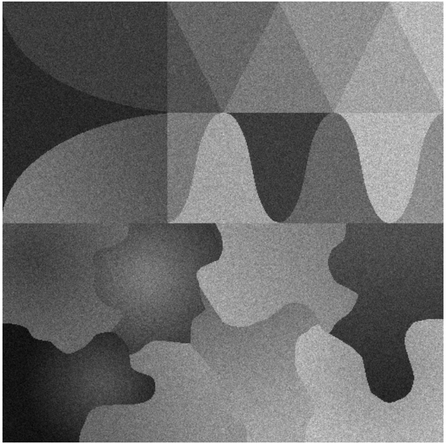
(M.&F.TIP2014)

Intersecting contours $F_B(\hat{\alpha}, \hat{\sigma}) = 1$

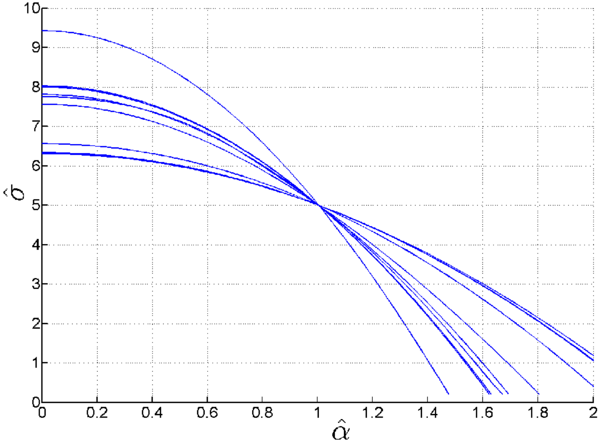


(M.&F.TIP2014)

Standard deviation contours: Example (GAT)



(a) peak 120, $\alpha = 1$, $\sigma = 5$



(b) GAT contours

Ten standard deviation contours $F_B(\hat{\alpha}, \hat{\sigma}) = 1$ computed from ten randomly selected 32×32 blocks B of the 512×512 image (a).

Standard deviation contours: Propositions

- We assume two ideal hypotheses:

  1. We can achieve exact stabilization with the correct noise parameters θ:
  $$\mathrm{std}\left\{f_{\alpha,\sigma}\left(z_{\alpha,\sigma}\right)|y\right\}=1\ \forall y\geq 0. \tag{14}$$

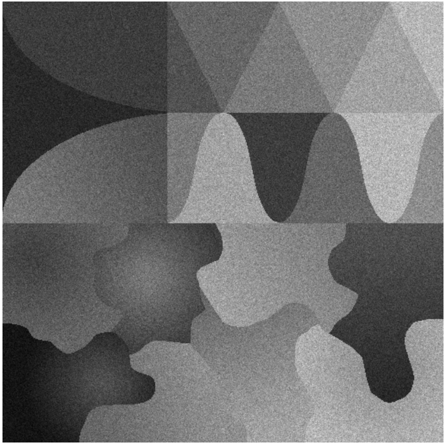
  2. For any VST $f_{\hat{\alpha},\hat{\sigma}}$ and any choice of parameters $(\hat{\alpha},\hat{\sigma})$ and α,σ, the approximation
  $$\mathrm{std}\left\{f_{\hat{\alpha},\hat{\sigma}}\left(z_{\alpha,\sigma}\right)|y\right\}\approx\mathrm{std}\left\{z_{\alpha,\sigma}|y\right\}f'_{\hat{\alpha},\hat{\sigma}}\left(E\left\{z_{\alpha,\sigma}|y\right\}\right) \tag{15}$$
  holds exactly.

Proposition 1. The mean standard deviation of the stabilized image block $f_{\hat{\alpha},\hat{\sigma}}\left(z_{\alpha,\sigma}\right)$ can now be written as
$$\mathfrak{E}_B\left\{f_{\hat{\alpha},\hat{\sigma}}\left(z_{\alpha,\sigma}\right)\right\}=\int\frac{\mathrm{std}\left\{z_{\alpha,\sigma}|y\right\}}{\mathrm{std}\left\{z_{\hat{\alpha},\hat{\sigma}}|y\right\}}p_B\left(y\right)dy. \tag{16}$$

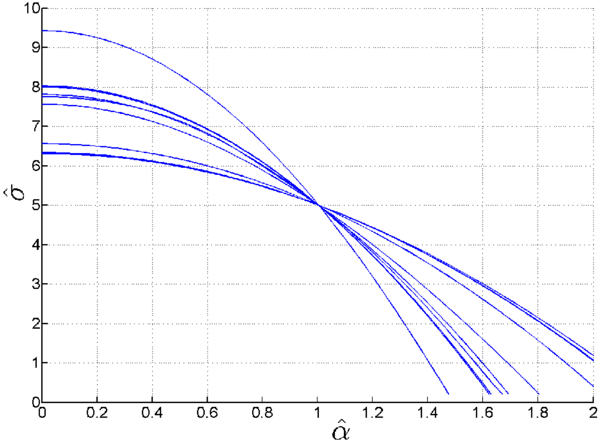
Proposition 2. Given the assumptions in Proposition 1, $F_B\left(\hat{\alpha},\hat{\sigma}\right)$ has a well-behaving (locally smooth and simple) unitary contour near the true parameter values α, σ.

(M.&F.TIP2014)

Standard deviation contours: Example (GAT)



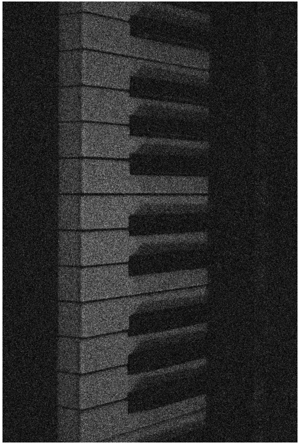
(a) peak 120, $\alpha = 1$, $\sigma = 5$



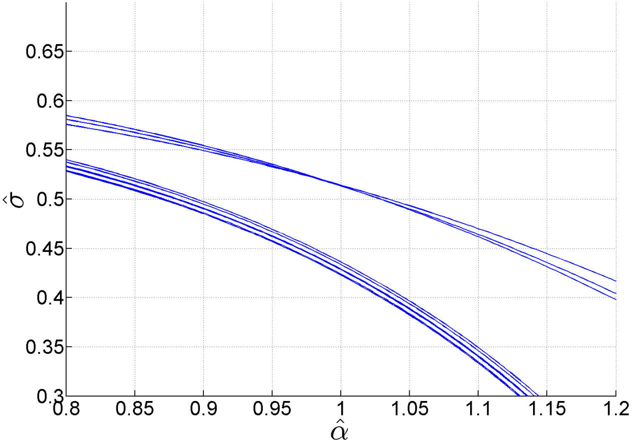
(b) GAT contours

Ten standard deviation contours $F_B(\hat{\alpha}, \hat{\sigma}) = 1$ computed from ten randomly selected 32×32 blocks B of the 512×512 image (a).

Standard deviation contours: Example (GAT)



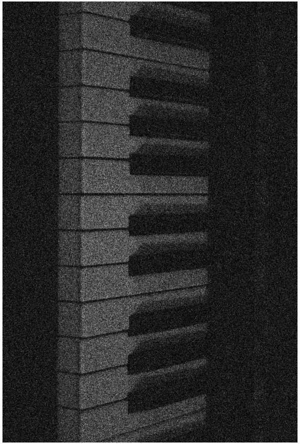
(a) peak 120, $\alpha = 1$, $\sigma = 5$



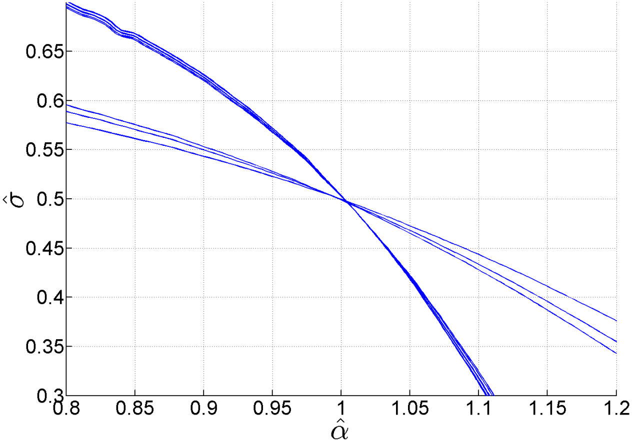
(b) GAT contours

Ten standard deviation contours $F_B(\hat{\alpha}, \hat{\sigma}) = 1$ computed from ten randomly selected 32×32 blocks B of the 1193×795 image (a).

Standard deviation contours: Example (Opt.VST)



(a) peak 120, $\alpha = 1$, $\sigma = 5$



(b) GAT contours

Ten standard deviation contours $F_B(\hat{\alpha}, \hat{\sigma}) = 1$ computed from ten randomly selected 32×32 blocks B of the 1193×795 image (a).

Application to parameter estimation

- The contours $F_B\,(\hat{\alpha}, \hat{\sigma}) = 1$ corresponding to different stabilized blocks B are locally smooth in the $(\hat{\alpha}, \hat{\sigma})$ plane.

- Typically different blocks yield differently oriented curves intersecting each other.

- The intersection has coordinates $(\alpha,\, \sigma)$, i.e. the true parameters.

- A cost functional measuring the lack of stabilization is minimized at the intersection.

(M.&F.TIP2014)

Parameter estimation algorithm

1. Initialize the estimates $\hat{\alpha}$ and $\hat{\sigma}$.
2. Choose M random blocks B_m, $m = 1, \ldots, M$ from the noisy image $z_{\alpha,\sigma}$.
3. Apply a VST $f_{\hat{\alpha},\hat{\sigma}}(z_{\alpha,\sigma})$ to each block.
4. Compute an estimate $F_{B_m}(\hat{\alpha},\hat{\sigma}) = \mathfrak{E}_{B_m}\{f_{\hat{\alpha},\hat{\sigma}}(z_{\alpha,\sigma})\}$ for the standard deviation of each stabilized block, using any AWGN standard deviation estimator $\mathfrak{E}$.
5. Optimize $\hat{\alpha}$ and $\hat{\sigma}$ so to minimize the difference between $F_{B_m}(\hat{\alpha},\hat{\sigma})^2$ and 1 (target variance) over the M blocks.

- We implement the proposed approach in Matlab, using the optimized VSTs (or GAT for comparison), and minimizing the cost functional
$$C(\hat{\alpha},\hat{\sigma}) = \underset{m=1,\ldots,M}{\text{mean}} \left| F_{B_m}(\hat{\alpha},\hat{\sigma})^2 - 1 \right|.$$
- $\mathfrak{E}$ is sample standard deviation of wavelet detail coefficients.
- We estimate $F_{B_m}(\hat{\alpha},\hat{\sigma})$ from $M = 2000$ randomly selected 32×32 image blocks.

(M.&F.TIP2014)

Experiments

Root Histogram-Weighted Normalized MSE (RHWNMSE) : $\sqrt{\int_{\mathbb{R}^+} p(\xi) \frac{\left(\sqrt{\alpha^2\xi+\sigma^2}-\sqrt{\hat{\alpha}^2\xi+\hat{\sigma}^2}\right)^2}{\alpha^2\xi+\sigma^2} d\xi}$

Table: Average RHWNMSE ($\pm$ std) over 10 noise realizations for *Piano* image:

Peak	α	σ	Opt. VST	GAT	Scatterplot
2	0.5	0.2	0.042 ± 0.002	0.286 ± 0.008	$\mathbf{0.024 \pm 0.009}$
2	2.5	0.2	$\mathbf{0.007 \pm 0.005}$	0.676 ± 0.007	0.056 ± 0.016
10	0.5	1.0	$\mathbf{0.006 \pm 0.003}$	0.021 ± 0.002	0.011 ± 0.007
10	2.5	1.0	$\mathbf{0.005 \pm 0.004}$	0.013 ± 0.005	0.016 ± 0.008
30	0.5	3.0	$\mathbf{0.006 \pm 0.003}$	$\mathbf{0.006 \pm 0.003}$	0.016 ± 0.007
30	2.5	3.0	$\mathbf{0.005 \pm 0.003}$	0.008 ± 0.002	0.014 ± 0.006

- Combined with the optimized VSTs, the algorithm yields results that are competitive with the results obtained with scatterplot method (Foi et al., 2008).
- The optimized VSTs plays an important role in the estimation performance for the low-intensity cases.
  - The GAT is inherently unable to accurately stabilize regions with low mean intensity; this violates our assumption that $\text{std}\{f_\theta(z_\theta)|y\} = 1\ \forall y \geq 0$.
  - Optimized VSTs provide highly accurate stabilization also for low intensities.

4. Exact unbiased inversion of VST in denoising

Three steps: stabilization, denoising, and inversion

VSTs are often exploited for the removal of signal-dependent noise through the following three-step procedure:

1. Noise variance is stabilized by applying a VST f to the data; this produces a signal in which the noise can be treated as additive with unitary variance.
2. Noise is removed using a conventional denoising algorithm – denoted by Φ – for additive homoskedastic noise (e.g., additive white Gaussian noise).
3. An inverse transformation is applied to the denoised signal, obtaining the estimate of the signal of interest.

Denoising algorithms attempt to estimate the expectation, thus, $D = \Phi\left(f\left(z\right)\right)$ can be treated as an approximation of $E\{f(z)|\theta\}$.

Exact unbiased inverse

(F.SigPro2009,M.&F.TIP2011)

Since f is necessarily a nonlinear mapping, we may have

$$E\{f(z)|\theta\} \neq f(E\{z|\theta\}),$$

and, thus,

$$f^{-1}(E\{f(z)|\theta\}) \neq E\{z|\theta\},$$

which means that the inverse transformation applied after denoising (Step 3.) should not coincide with the algebraic inverse of f, as this would introduce bias in the estimation of $E\{z|\theta\}$ from the observed z.

The problem of bias in variance-stabilized denoising is solved by the *exact unbiased inverse* that is defined by the mapping

$$\mathcal{I}_f : E\{f(z)|\theta\} \longmapsto E\{z|\theta\} = \mu.$$

This definition assumes that the mapping $E\{z|\theta\} \mapsto E\{f(z)|\theta\}$ is invertible. In particular, we require this mapping to be strictly increasing, or, equivalently, that $E\{f(z)|\theta\}$ is strictly increasing with θ. This condition supplants the traditional requirement of invertibility of f, which instead we may allow to be nonmonotone.

Under modest hypotheses, it can be shown that $\mathcal{I}_f(D)$ is a maximum-likelihood estimate of θ.

Inversion for Poisson stabilized by Anscombe

(M.&F.TIP2011)

Let z be Poisson distributed data.
Applying the Anscombe transform yields $f(z) = 2\sqrt{z + \frac{3}{8}}$.
After filtering of $f(z)$ we obtain $D = \Phi(f(z))$, which we treat as an approximation of $E\{f(z)|\theta\}$.

Algebraic inverse: $\mathcal{I}_A(D) = f^{-1}(D) = \left(\frac{D}{2}\right)^2 - \frac{3}{8}$

Asymptotically unbiased inverse: $\mathcal{I}_B(D) = \left(\frac{D}{2}\right)^2 - \frac{1}{8}$. Typically used in applications.

Exact unbiased inverse: $\mathcal{I}_C : E\{f(z) \mid y\} \longmapsto E\{z \mid y\}$.

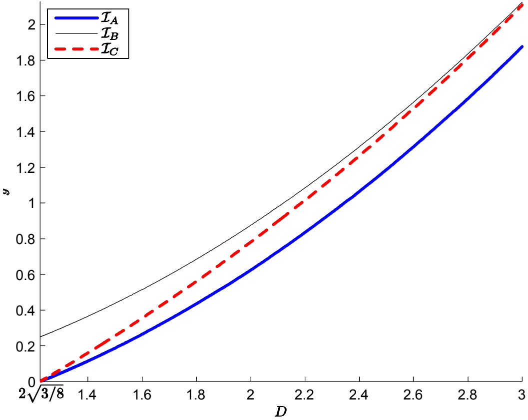
We have discrete Poisson probabilities $P(z \mid y)$, so

$$E\{f(z) \mid y\} = \sum_{z=0}^{+\infty} f(z) P(z \mid y) = 2 \sum_{z=0}^{+\infty} \left(\sqrt{z + \frac{3}{8}} \cdot \frac{y^z e^{-y}}{z!} \right).$$

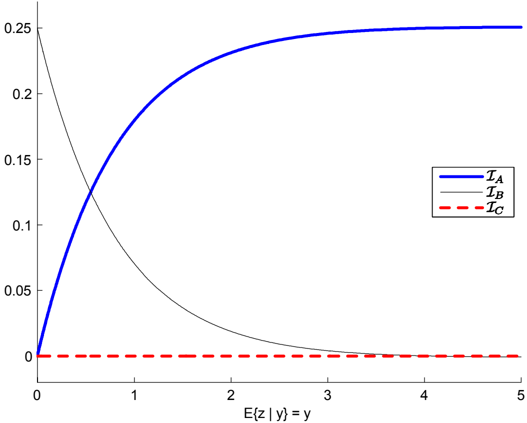
The definition of $\mathcal{I}_C$ is implicit, but we can have a closed form approximation as

$$\mathcal{I}_C(D) \cong \frac{1}{4} D^2 + \frac{1}{4}\sqrt{\frac{3}{2}} D^{-1} - \frac{11}{8} D^{-2} + \frac{5}{8}\sqrt{\frac{3}{2}} D^{-3} - \frac{1}{8}$$

Inversion for Poisson stabilized by Anscombe



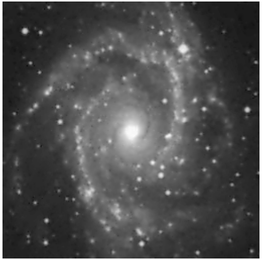
inverse transformations



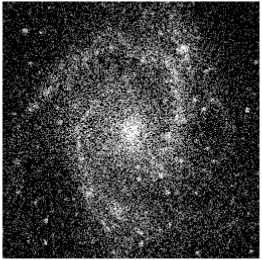
bias

(M.&F.TIP2011)

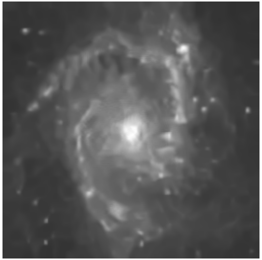
Inversion for Poisson stabilized by Anscombe



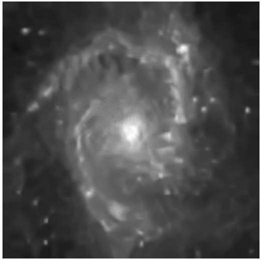
original



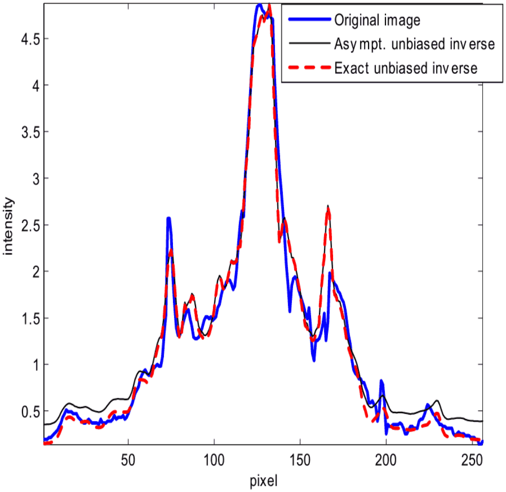
noisy



Ansc.+BM3D+Asy.Unb.Inv.



Ansc.+BM3D+Ex.Unb.Inv.



Cross section

Denoising results: Poisson

AVERAGE NMISE VALUES FOR THE ASYMPTOTICALLY UNBIASED INVERSE AND THE EXACT UNBIASED INVERSE, AND A COMPARISON TO THE RESULTS OBTAINED IN [2] AND [1] WITH ALGORITHMS SPECIFICALLY DESIGNED FOR POISSON NOISE REMOVAL. THE INTENSITY RANGE OF EACH IMAGE IS INDICATED IN BRACKETS.

	Asymptotically unbiased inverse				Exact unbiased inverse			Other algorithms	
	WT [1]	BM3D	SAFIR	BLS-GSM	BM3D	SAFIR	BLS-GSM	PH-HMT [2]	MS-VST [1]
Spots [0.03, 5.02]	2.34	1.7424	1.7495	2.0370	**0.0365**	0.0384	0.1871	0.048	0.069
Galaxy [0, 5]	0.15	0.1026	0.1110	0.1253	**0.0299**	0.0301	0.0385	0.030	0.035
Ridges [0.05, 0.85]	0.83	0.7025	0.7252	0.7694	**0.0128**	0.0173	0.0331	-	0.017
Barbara [0.93, 15.73]	0.26	**0.0881**	0.1178	0.1122	**0.0881**	0.1178	0.1123	0.159	0.17
Cells [0.53, 16.93]	0.095	0.0660	0.0683	0.0718	**0.0649**	0.0671	0.0707	0.082	0.078

(M.&F.TIP2011)

Exact unbiased inverse of Generalized Anscombe Transform for Poisson-Gaussian noise

(M.&F.TIP2013)

Without loss of generality, we can fix $\alpha = 1$ and use scaling for $\alpha \neq 1$.

The EUI of GAT is constructed analogous to the EUI of the Anscombe transformation:

$$\mathcal{I}_\sigma : E\left\{f_\sigma\left(z\right) \mid y, \sigma\right\} \longmapsto E\left\{z \mid y, \sigma\right\}.$$

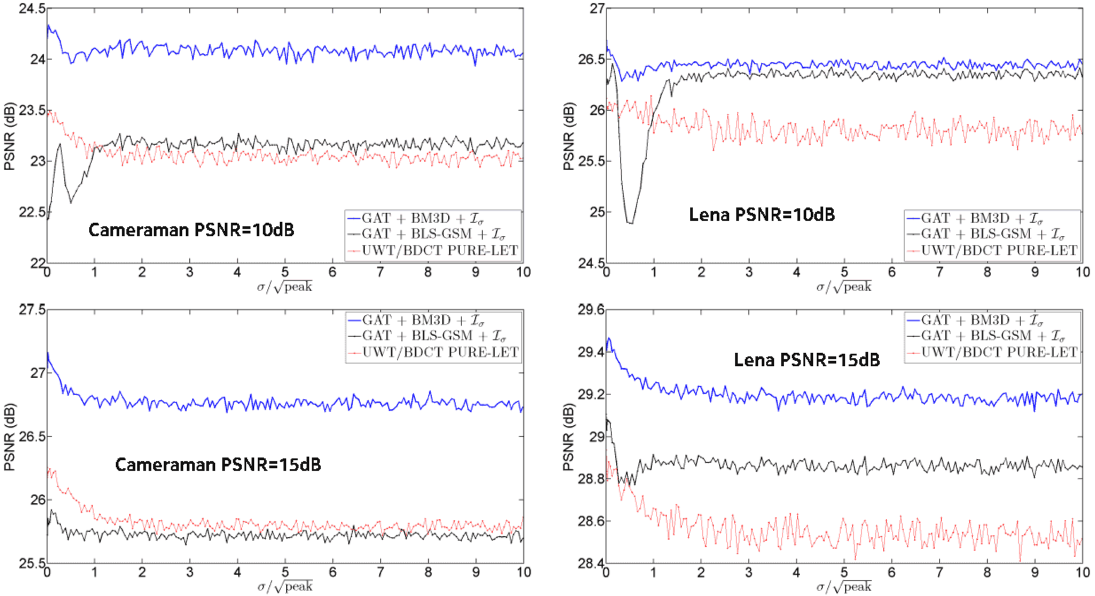
$$E\{f_\sigma\left(z\right) \mid y, \sigma\} = \int_{-\infty}^{+\infty} f_\sigma\left(z\right) p\left(z \mid y, \sigma\right)\, dz =$$

$$= \int_{-\infty}^{+\infty} 2\sqrt{z + \frac{3}{8} + \sigma^2} \sum_{k=0}^{+\infty} \left(\frac{y^k e^{-y}}{k!\sqrt{2\pi\sigma^2}} e^{-\frac{(z-k)^2}{2\sigma^2}} \right) dz.$$

Closed form approximation:

$$\mathcal{I}_\sigma(D) \cong \frac{1}{4}D^2 + \frac{1}{4}\sqrt{\frac{3}{2}}D^{-1} - \frac{11}{8}D^{-2} + \frac{5}{8}\sqrt{\frac{3}{2}}D^{-3} - \frac{1}{8} - \sigma^2.$$

Consistency of GAT+EUI at fixed input PSNR from pure Gaussian to pure Poisson



Volumetric Rician denoising

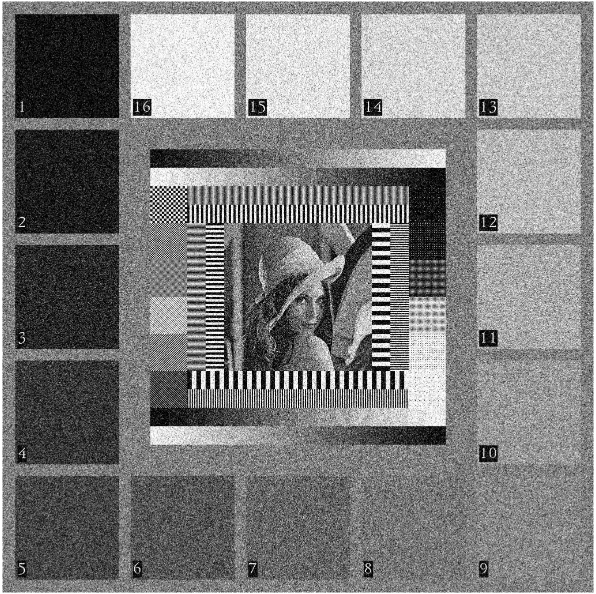
AWGN denoising within VST framework vs. Rician denoising

Noise	Filter	σ									
		1%	3%	5%	7%	9%	11%	13%	15%	17%	19%
Gauss.	*(Noisy data)*	40.00|0.97	30.46|0.81	26.02|0.66	23.10|0.53	20.91|0.43	19.17|0.36	17.72|0.30	16.48|0.25	15.39|0.22	14.42|0.19
	OB-NLM3D$_{\mathcal{N}}$	42.47|**0.99**	37.57|0.97	34.73|0.95	32.82|0.92	31.42|0.90	30.32|0.87	29.40|0.84	28.61|0.82	27.91|0.79	27.28|0.77
	OB-NLM3D-WM$_{\mathcal{N}}$	42.52|**0.99**	37.75|0.97	35.01|0.95	33.13|0.93	31.73|0.90	30.61|0.88	29.68|0.85	28.88|0.83	28.18|0.80	27.55|0.78
	ODCT3D$_{\mathcal{N}}$	43.78|**0.99**	37.53|0.97	34.89|0.95	33.18|0.93	31.91|0.91	30.90|0.89	30.07|0.88	29.35|0.86	28.73|0.85	28.18|0.83
	PRI-NLM3D$_{\mathcal{N}}$	44.04|**0.99**	38.26|**0.98**	35.51|**0.96**	33.67|0.94	32.37|0.92	31.29|0.90	30.40|0.89	29.65|0.87	28.99|0.85	28.40|0.84
	BM4D	**44.09|0.99**	**38.39|0.98**	**35.95|0.96**	**34.38|0.95**	**33.21|0.93**	**32.28|0.92**	**31.50|0.91**	**30.82|0.90**	**30.23|0.88**	**29.70|0.87**
Rician	*(Noisy data)*	40.00|0.97	30.49|0.81	26.09|0.66	23.20|0.53	21.04|0.43	19.32|0.36	17.88|0.30	16.65|0.25	15.57|0.21	14.60|0.18
	OB-NLM3D$_{\mathcal{R}}$	42.41|**0.99**	37.45|0.97	34.54|0.94	32.51|0.91	30.97|0.88	29.71|0.85	28.62|0.81	27.64|0.78	26.74|0.74	25.91|0.70
	VST + OB-NLM3D$_{\mathcal{N}}$	42.48|**0.99**	37.45|0.97	34.40|0.94	32.26|0.91	30.65|0.88	29.34|0.85	28.23|0.81	27.25|0.78	26.37|0.74	25.57|0.71
	OB-NLM3D-WM$_{\mathcal{R}}$	42.44|**0.99**	37.54|0.97	34.66|0.95	32.61|0.92	31.01|0.88	29.69|0.85	28.53|0.81	27.50|0.77	26.57|0.74	25.71|0.70
	VST + OB-NLM3D-WM$_{\mathcal{N}}$	42.53|**0.99**	37.68|0.97	34.75|0.95	32.66|0.92	31.06|0.89	29.77|0.86	28.68|0.83	27.71|0.80	26.84|0.76	26.04|0.73
	ODCT3D$_{\mathcal{R}}$	42.96|**0.99**	37.38|0.97	34.70|0.95	32.90|0.93	31.53|0.90	30.41|0.88	29.48|0.86	28.67|0.84	27.95|0.82	27.30|0.80
	VST + ODCT3D$_{\mathcal{N}}$	43.74|**0.99**	37.51|0.97	34.79|0.95	32.98|0.93	31.59|0.90	30.47|0.88	29.52|0.86	28.71|0.84	27.98|0.82	27.31|0.80
	PRI-NLM3D$_{\mathcal{R}}$	43.97|**0.99**	38.19|**0.98**	35.34|**0.96**	33.37|**0.94**	31.94|0.91	30.74|0.89	29.75|0.87	28.88|0.85	28.10|0.82	27.39|0.80
	VST + PRI-NLM3D$_{\mathcal{N}}$	**44.21|0.99**	38.20|**0.98**	35.34|**0.96**	33.36|**0.94**	31.90|0.91	30.71|0.89	29.71|0.87	28.88|0.85	28.13|0.82	27.46|0.80
	VST + BM4D	44.08|**0.99**	**38.34|0.98**	**35.83|0.96**	**34.17|0.94**	**32.89|0.93**	**31.82|0.91**	**30.90|0.89**	**30.06|0.88**	**29.29|0.86**	**28.57|0.84**

(M.&al.TIP2013)

The Rician denoising quality of a dedicated Rician ($\mathcal{R}$) version of a Gaussian ($\mathcal{N}$) filter can be achieved (and sometimes even surpassed) by the same Gaussian filter endowed by forward and inverse VSTs.

Denoising doubly censored noisy images



(F.SigPro2009)

$\tilde{z}$ PSNR=15.00dB noise parameters $a = 0$, $b = 0.2^2$

Denoising doubly censored noisy images

Denoising heteroskedaskic data using variance-stabilization and conventional denoising algorithm for AWGN.

Main stages:

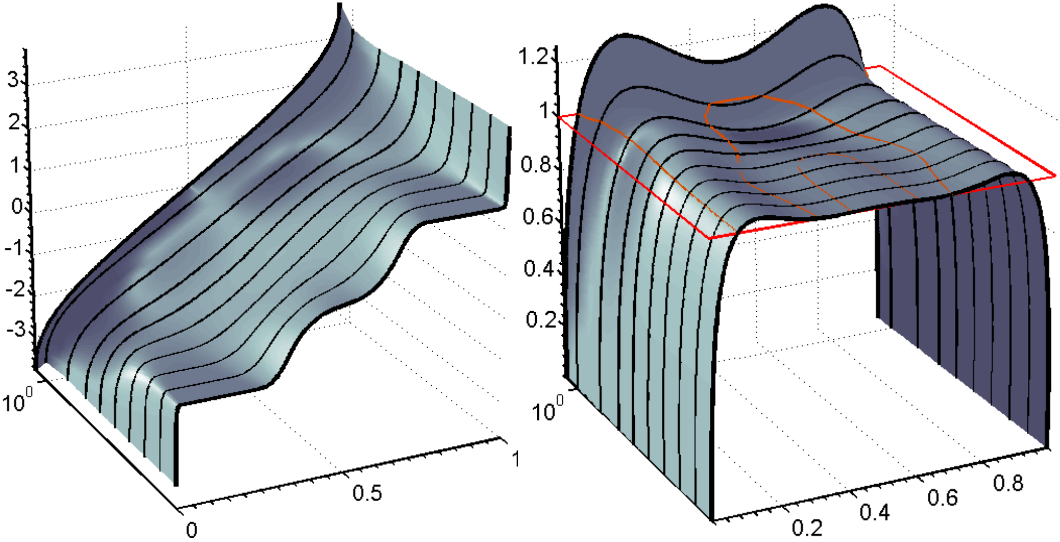
1. variance-stabilization
2. denoising (BM3D public code for AWGN from www.cs.tut.fi/~foi/GCF-BM3D/)
3. inversion of the stabilizer, including EUI and declipping (from $E\{f(\tilde{z})|y\} \mapsto y$)

We compare two alternatives stabilizers:

$$f_0(t) = \int_{t_0}^{t} \frac{c}{\tilde{\sigma}(\tilde{y})} d\tilde{y}, \qquad t, t_0 \in [0,1]$$

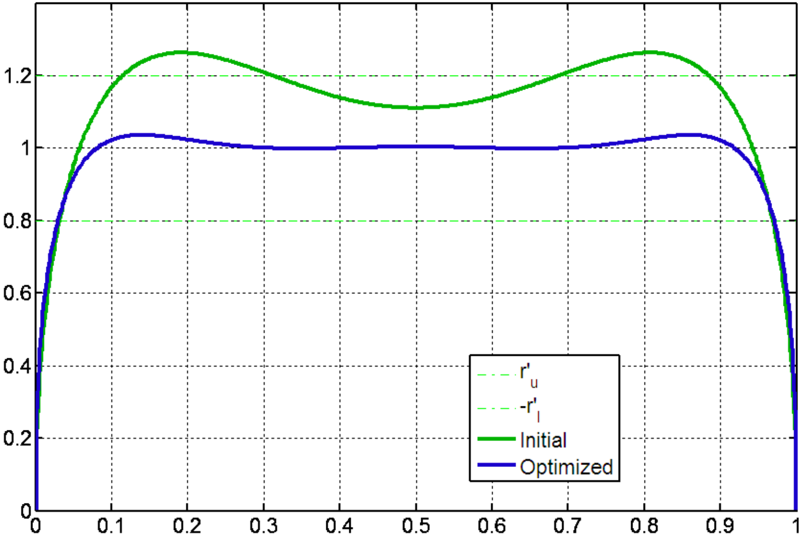
f_{2000} optimization by iterative integral.

Variance stabilization



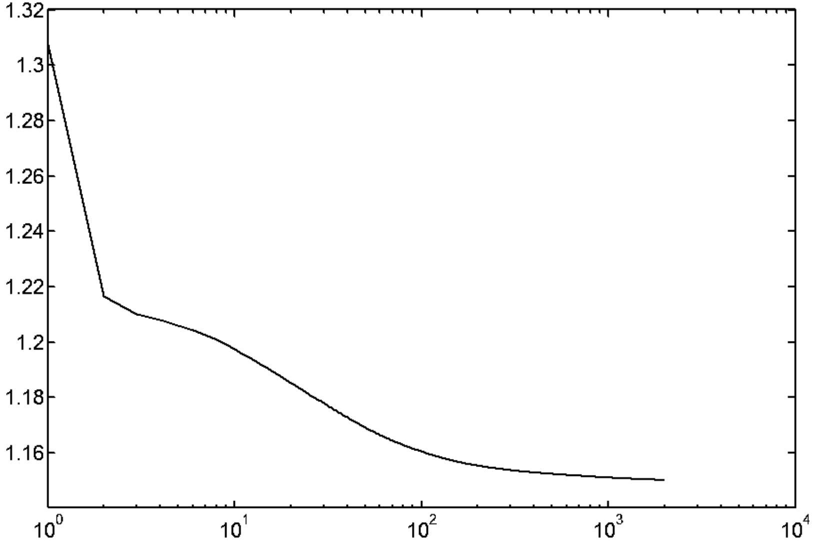
f_k $\qquad$ $\mathrm{std}\left\{f_k\left(\tilde{z}\right)|\tilde{y}\right\}$

Variance stabilization



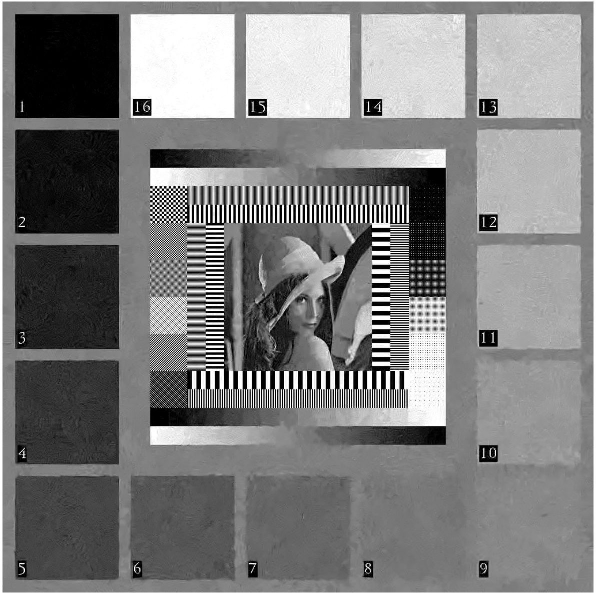
$\mathrm{std}\{f_0(\tilde{z})\,|\tilde{y}\}$ vs $\mathrm{std}\{f_{2000}(\tilde{z})\,|\tilde{y}\}$

Variance stabilization



Convergence of the iterative integral algorithm

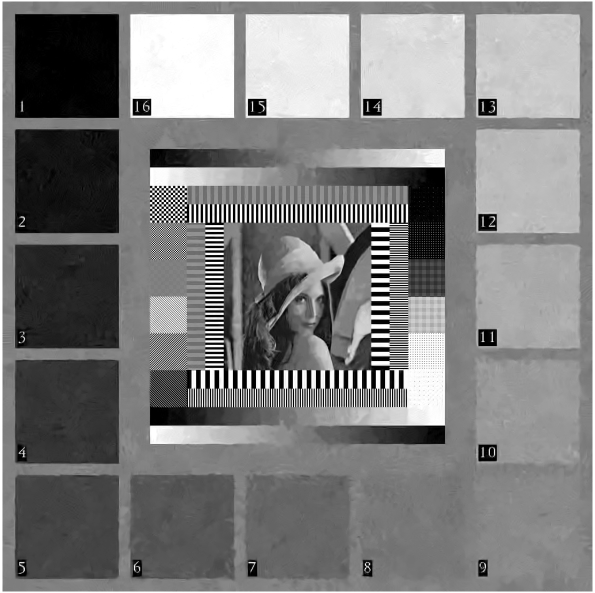
Denoised using f_0 as stabilizer



PSNR=29.37

(F.SigPro2009)

Denoised using f_{2000} as stabilizer



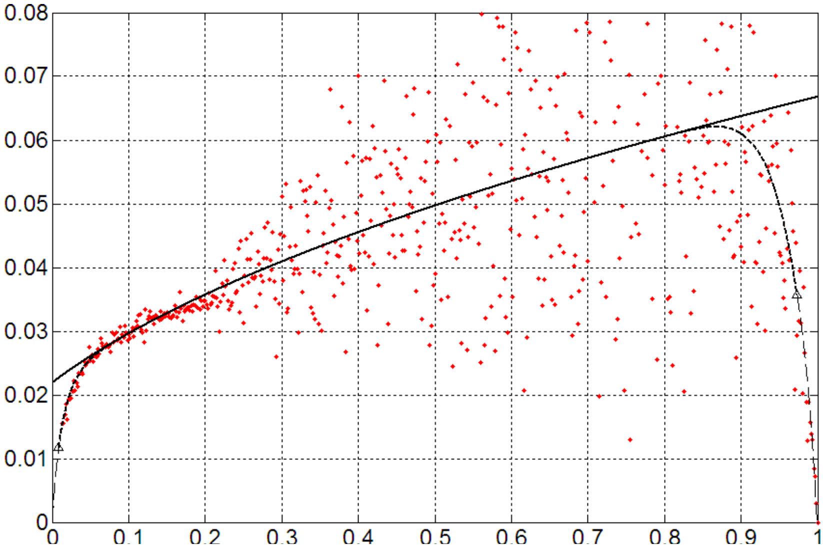
(F.2009)

PSNR=30.67 [1.3dB gain]

Noisy raw-data image $\tilde{z}$

Fujifilm FinePix S9600 (green channel)

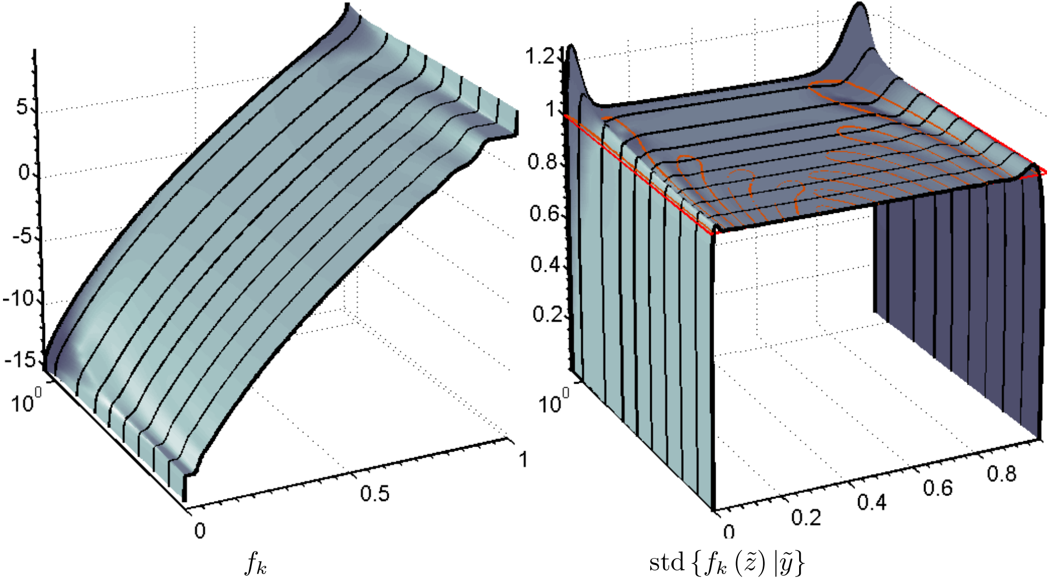
Noise estimation



$a = 0.003978$ $b = 0.0004787$

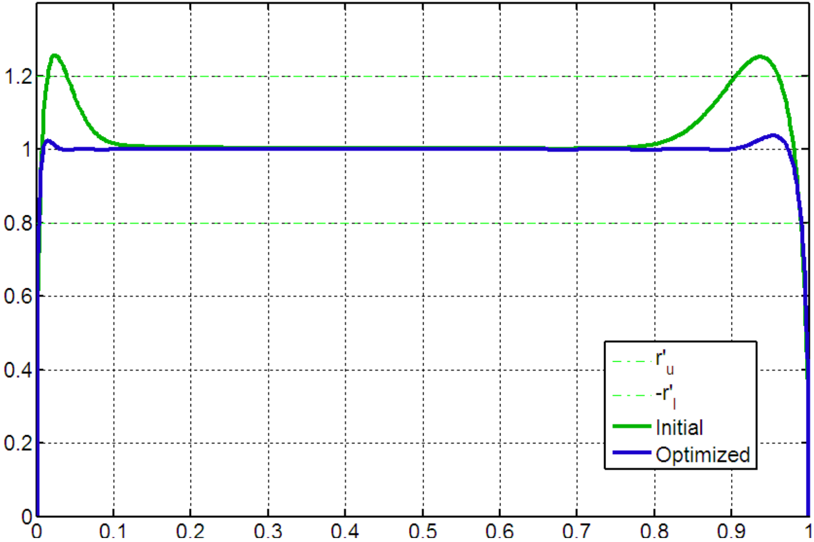
(F.&al.2008)

Variance stabilization



(F.2009)

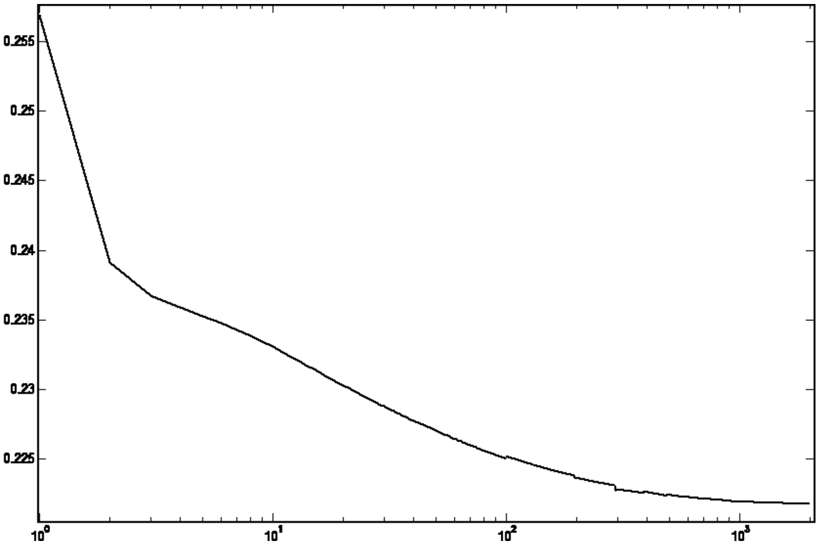
Variance stabilization



$\operatorname{std}\{f_0(\tilde{z})\,|\tilde{y}\}$ vs $\operatorname{std}\{f_{2000}(\tilde{z})\,|\tilde{y}\}$

(F.2009)

Variance stabilization



Convergence of the iterative integral algorithm

Denoised using f_0 as stabilizer

Denoised using f_{2000} as stabilizer

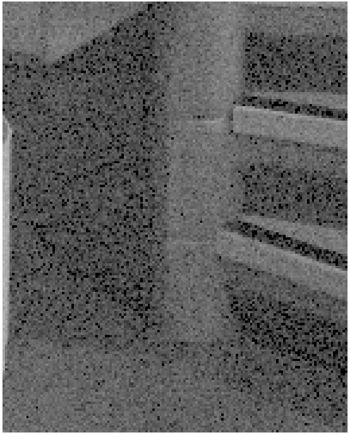
Denoised using f_0 as stabilizer

(gamma-corrected)

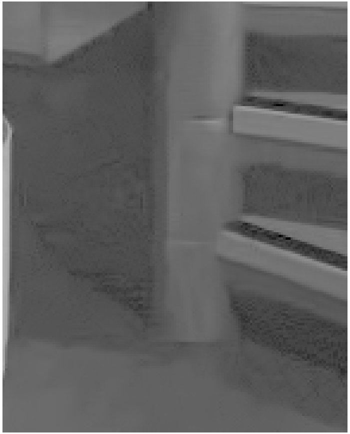
Denoised using f_{2000} as stabilizer

(gamma-corrected)

Comparison of fragments (1/3)



noisy



using f_0

using f_{2000}

(all gamma-corrected)

(F.2009)

Comparison of fragments (2/3)

noisy

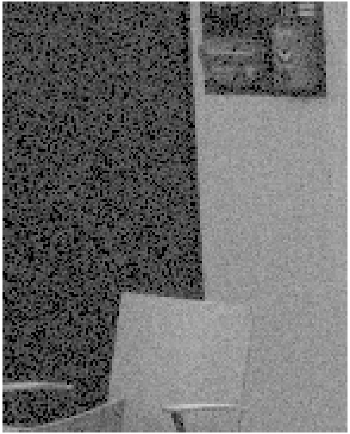
using f_0

using f_{2000}

(all gamma-corrected)

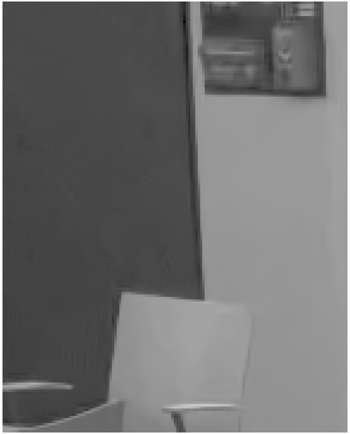
(F.2009)

Comparison of fragments (3/3)



noisy

using f_0



using $\boldsymbol{f_{2000}}$

(all gamma-corrected)

(F.2009)

Thank you!

Hepatica nobilis

Software

Matlab codes for noise estimation, variance stabilization, exact unbiased inversion, and for image, video, and volume filtering can be downloaded from http://www.cs.tut.fi/~foi

References

1. Anscombe, F.J., “The transformation of Poisson, binomial and negative-binomial data”, *Biometrika*, vol. 35, no. 3/4, pp. 246–254, Dec. 1948.
2. Arsenault, H. H., and M. Denis, “Integral expression for transforming signal-dependent noise into signal-independent noise”, *Opt. Lett.*, vol.6, no. 5, pp. 210-212, May 1981.
3. Azzari, L., and A. Foi, “Gaussian-Cauchy mixture modeling for robust signal-dependent noise estimation”, *Proc. 2014 IEEE Int. Conf. Acoustics, Speech, Signal Process. (ICASSP 2014)*, pp. 5357-5361, Florence, Italy, May 2014.
4. Bar-Lev, S.K., and P. Enis, “On the construction of classes of variance stabilizing transformations”, *Statistics & Probability Letters*, vol. 10, pp. 95–100, July 1990.
5. Bartlett, M. S., “The Square Root Transformation in Analysis of Variance,” *J. R. Statist. Soc. Suppl.*, vol.3, no. 1, pp. 68-78, 1936.
6. Cohen, A. C., *Truncated and Censored Samples*, CRC Press, 1991.

7. Conn, A. R., K. Scheinberg, and L. N. Vicente, *Introduction to derivative-free optimization*, MPS-SIAM Series on Optimization, vol. 8, 2009.
8. Coupé, P., P. Yger, S. Prima, P. Hellier, C. Kervrann, and C. Barillot, "An optimized blockwise nonlocal means denoising filter for 3-D magnetic resonance images," *IEEE Transactions on Medical Imaging*, vol. 27, no. 4, pp. 425–441, April 2008.
9. Curtiss, J.H., "On transformations used in the analysis of variance", *The Annals of Mathematical Statistics*, vol. 14, no. 2, pp. 107–122, June 1943.
10. Dabov, K., A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080-2095, August 2007.
11. Efron, B., "Transformation theory: How normal is a family of distributions?", *The Annals of Statistics*, vol. 10, no. 2, pp. 323–339, 1982.
12. Foi, A., "Clipped noisy images: heteroskedastic modeling and practical denoising", *Signal Processing*, vol. 89, no. 12, pp. 2609-2629, December 2009.
13. Foi, A., "Noise estimation and removal in MR imaging: the variance-stabilization approach," *Proc. 2011 IEEE Int. Symp. Biomedical Imaging (ISBI 2011)*, Chicago, IL, USA, 30 March - 2 April 2011.
14. Foi, A., S. Alenius, V. Katkovnik, and K. Egiazarian, "Noise measurement for raw-data of digital imaging sensors by automatic segmentation of non-uniform targets", *IEEE Sensors Journal*, vol. 7, no. 10, pp. 1456-1461, October 2007.
15. Foi, A., M. Trimeche, V. Katkovnik, and K. Egiazarian, "Practical Poissonian-Gaussian noise modeling and fitting for single image raw-data", *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1737-1754, October 2008.
16. Foi, A., "Direct optimization of nonparametric variance-stabilizing transformations," presented at *8èmes Rencontres de Statistiques Mathématiques*, CIRM, Luminy, France, December 2008.
17. Foi, A., "Optimization of variance-stabilizing transformations", preprint, 2009.

18. Foi, A. "Removal of signal-dependent noise: the BM3D filter and optimized variance-stabilizing transformations," presented at *"Patch-based Image Representation, Manifolds and Sparsity" Minisymposium*, INRIA Centre de Rennes Bretagne Atlantique, IRISA, Rennes, France, April 2009. http://videos.rennes.inria.fr/seminaire_Irisa/Vista/

19. Freeman, M. and J. Tukey, "Transformations Related to the Angular and the Square Root," *The Annals of Mathematical Statistics*, vol. 21, no. 4, pp. 607-611, December 1950.

20. Fryzlewicz, P. and G. P. Nason, "A Haar-Fisz algorithm for Poisson Intensity Estimation," *J. Comp. Graph. Stat.*, vol.13, pp. 621-638, 2004.

21. Fryzlewicz, P., "Data-driven wavelet-Fisz methodology for nonparametric function estimation," *Electronic Journal of Statistics*, vol.2, pp. 863-896, 2008.

22. Guan, Y., "Variance stabilizing transformations of Poisson, binomial and negative binomial distributions," *Statistics and Probability Letters*, 2009.

23. Hirakawa, K. and T. W. Parks, "Image denoising using total least squares," *IEEE Trans. Image Process.*, vol.15, no. 9, pp. 2730-2742, September 2006.

24. Jin, X., Z. Xu, and K. Hirakawa, "Noise Parameter Estimation for Poisson Corrupted Images Using Variance Stabilization Transforms", *IEEE Trans. Image Process.*, vol. 23, no. 3, pp. 1329–1339, March 2014.

25. Kasturi, R. and J. F. Walkup and T. F. Krile, "Image restoration by transformation of signal-dependent noise to signal-independent noise," *Applied Optics*, vol.22, no. 22, pp. 3537-3542, November 1983.

26. Kervrann, C., and J. Boulanger, "Local adaptivity to variable smoothness for exemplar-based image denoising and representation", *International Journal of Computer Vision*, vol. 79, no. 1, pp. 45–69, Aug. 2008.

27. Luisier, F., T. Blu, and M. Unser, "Image Denoising in Mixed Poisson-Gaussian Noise", *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 696–708, March 2011.

28. Maggioni, M., V. Katkovnik, K. Egiazarian, and A. Foi, "A Nonlocal Transform-Domain Filter for Volumetric Data Denoising and Reconstruction", *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 119-133, January 2013.

29. Manjón, J. V., P. Coupé, A. Buades, D. L. Collins, and M. Robles, "New methods for MRI denoising based on sparseness and self-similarity," *Medical Image Analysis*, vol. 16, no. 1, pp. 18–27, 2012.

30. Murtagh, F., J.-L. Starck and F. Murtagh, "Image restoration with noise suppression using a multiresolution support", *Astronomy and Astrophysics*, vol. 112, no. 179, 1995.

31. Mäkitalo, M., and A. Foi, "Optimal inversion of the Anscombe transformation in low-count Poisson image denoising", *IEEE Trans. Image Process.*, vol. 20, no. 1, pp. 99–109, January 2011.

32. Mäkitalo, M., and A. Foi, "A closed-form approximation of the exact unbiased inverse of the Anscombe variance-stabilizing transformation", *IEEE Trans. Image Process.*, vol. 20, no. 9, pp. 2697-2698, September 2011.

33. Mäkitalo, M., and A. Foi, "Optimal inversion of the generalized Anscombe transformation for Poisson-Gaussian noise", *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 91–103, January 2013.

34. Mäkitalo, M., and A. Foi, "Noise parameter mismatch in variance stabilization, with an application to Poisson-Gaussian noise estimation", to appear in *IEEE Trans. Image Process.*, 2014.

35. Mäkitalo, M., A. Foi, D. Fevralev, and V. Lukin, "Denoising of single-look SAR images based on variance stabilization and nonlocal filters", *Proc. Int. Conf. Math. Meth. Electromagn. Th., MMET 2010*, Kiev, Ukraine, September 2010.

36. Nason, G. P., *Wavelet Methods in Statistics with R*, Springer, 2008.

37. Nelder, J. A. and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol.7, pp. 308-313, 1965.

38. Portilla, J., V. Strela, M.J. Wainwright, and E.P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain", *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.

39. Prucnal, P. R. and B. E. A. Saleh, "Transformation of image-signal-dependent noise into image signal-independent noise," *Optics Letters*, vol.6, no. 7, July 1981.

40. Pyatykh, S., and J. Hesser, "Image Sensor Noise Parameter Estimation by Variance Stabilization and Normality Assessment," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3990–3998, September 2014.

41. Starck, J. L., F. Murtagh, and A. Bijaoui, *Image Processing and Data Analysis*, Cambridge University Press, Cambridge, 1998.

42. Tippet, L. H. C., "Statistical methods in textile research. Part 2, Uses of the binomial and Poisson distributions," *Shirley Inst. Mem.*, vol.13, pp. 35-72, 1934.

43. Tibshirani, R., "Estimating Transformations for Regression Via Additivity and Variance Stabilization," *Journal of the American Statistical Association*, vol.83, no. 402, pp. 394-405, June 1988.

44. Zhang, B., and M. J. Fadili and J. L. Starck, "Wavelets, Ridgelets and Curvelets for Poisson Noise Removal," *IEEE Trans. Image Process.*, vol.17, no. 7, pp. 1093-1108, July 2008.