Signal Estimation with Wavelets

Problem:

$$y(n) = x(n) + w(n)$$

observation, signal, noise

Given y, estimate x.

"denoise" y

Classical Approach:

Linear estimation (e.g., Wiener filter)

Wavelet-Based Approach:

Wavelet transform + Decision Theory

$\Rightarrow$ near-optimal <u>nonlinear</u> estimation

Linear Estimation & Wiener Filtering

$$y(n) = x(n) + w(n), \quad n = 0, \ldots, N-1$$

View signal $x(n)$ as a random process:

Do does x have to be Gaussian or not?

$$E[x(n)] = 0$$

$$E[x(n)\,x(n+m)] = R(m,n)$$

Minimum MSE Estimation:

$$\hat{x}(n) = E[x(n) \mid y(0), \ldots, y(N-1)]$$

in general, a complicated nonlinear function that is difficult (or impossible) to compute

Linear Estimation:

$$\hat{x}(n) = \sum_{k=0}^{N-1} h(n,k)\, y(k)$$

$$\underline{\hat{x}} = \underline{H}\ \underline{y}$$

Wiener Filter:

Find $\underline{H}$ so that

$$MSE = E\left[(\underline{x} - \underline{H}\underline{y})^T(\underline{x} - \underline{H}\underline{y})\right]$$

is minimized.

$$0 = \frac{\partial MSE}{\partial \underline{H}} = E\left[\frac{\partial}{\partial \underline{H}}(\underline{x} - \underline{H}\underline{y})^T(\underline{x} - \underline{H}\underline{y})\right]$$

$$= E\left[(\underline{x} - \underline{H}\underline{y})\,\underline{y}^T\right]$$

Wiener Khinchen equations

$$\Rightarrow \quad \underline{H}\,\underbrace{E\left[\underline{y}\underline{y}^T\right]}_{\underline{R}_{yy}} = \underbrace{E\left[\underline{x}\underline{y}^T\right]}_{\underline{R}_{xy}}$$

$$\underline{H} = \underline{R}_{xy}\underline{R}_{yy}^{-1}$$

$$\underline{R}_{xy} = E\left[\underline{x}(\underline{x}+\underline{w})^T\right] = E\left[\underline{x}\underline{x}^T\right] = \underline{R}_{xx}$$

$$\underline{R}_{yy} = E\left[(\underline{x}+\underline{w})(\underline{x}+\underline{w})^T\right] = \underline{R}_{xx} + \underline{R}_{ww}$$

$$\Rightarrow \quad \boxed{\underline{H} = \underline{R}_{xx}\left(\underline{R}_{xx} + \underline{R}_{ww}\right)^{-1}}$$

$R_{ww} = \sigma^2 \underline{I}$ for white noise

Wiener Filtering & the FFT

Now suppose that $\underline{x}$ is a circular stationary process. That is, $\underline{R}_{xx}$ is circulant. In this case we know that $\underline{R}_{xx}$ is diagonalized by the DFT.

If
$$\underline{U} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & \cdots & & 1 \\ 1 & e^{-j\frac{2\pi}{N}} & e^{-j\frac{2\pi}{N}\cdot 2} & \cdots & e^{-j\frac{2\pi}{N}(N-1)} \\ 1 & e^{-j\frac{2\pi}{N}\cdot 2} & e^{-j\frac{2\pi}{N}\cdot 4} & \cdots & e^{-j\frac{2\pi}{N}\cdot 2(N-1)} \\ \vdots & \vdots & \vdots & & \vdots \end{bmatrix}$$

then
$$\underline{U}^H \underline{x} = \text{DFT of } \underline{x}$$

and
$$\underline{U}\, \underline{R}_{xx} \underline{U}^H = \underline{U}\, E[\underline{x}\, \underline{x}^T]\, \underline{U}^H$$

$$= D = \begin{bmatrix} \beta_0^2 & & 0 \\ & \ddots & \\ 0 & & \beta_{N-1}^2 \end{bmatrix}$$

diagonal

circular convolution
$\Updownarrow$
multiply in DFT domain

Now consider the Wiener filter in this case.

$$\underline{\hat{x}} = \underbrace{\underline{R}_{xx}\left(\underline{R}_{xx} + \sigma^2 \underline{I}\right)^{-1}}_{\underline{H}} \underline{y}$$

$$= \underline{U}\underline{D}\underline{U}^H \left(\underline{U}\underline{D}\underline{U}^H + \sigma^2 \underline{U}\underline{I}\underline{U}^H\right)^{-1} \underline{y}$$

$$= \underline{U}\underline{D}\,\underline{U}^H\underline{U}\left(\underline{D} + \sigma^2\underline{I}\right)^{-1}\underline{U}^H\underline{y}$$

$$= \underline{U}\,\underline{D}\left(\underline{D} + \sigma^2\underline{I}\right)^{-1} \cdot \underbrace{\underline{U}^H\underline{y}}_{\text{DFT of } \underline{y}}$$

$$\Longrightarrow$$

$$\underbrace{\underline{U}^H\underline{\hat{x}}}_{\text{DFT of } \underline{\hat{x}}} = \underbrace{\underline{D}\left(\underline{D} + \sigma^2\underline{I}\right)^{-1}}_{\text{frequency domain representation of Wiener filter}} \cdot \underbrace{\underline{U}^H\underline{y}}_{\text{DFT of } \underline{y}}$$

Key point:

$$D(D+\sigma^2 I)^{-1} = \begin{bmatrix} \frac{\beta_0^2}{\beta_0^2+\sigma^2} & & 0 \\ & \ddots & \\ 0 & & \frac{\beta_{N-1}^2}{\beta_{N-1}^2+\sigma^2} \end{bmatrix}$$

diagonal!

Let $\hat{\theta}_m$ be the m-th DFT coeff of $\hat{\underline{x}}$ and w_m denote the m-th DFT coeff of the data $\underline{y}$.

$$\hat{\theta}_m = \underbrace{\frac{\beta_m^2}{\beta_m^2+\sigma^2}}_{\substack{\text{attenuation factor} \\ \text{between 0 and 1}}} \cdot w_m$$

$$\hat{\underline{x}} = \sum_{n=0}^{N-1} \hat{\theta}_m \cdot \underline{u}_m$$

m-th DFT basis vector

$$= \sum_{n=0}^{N-1} \frac{\beta_m^2}{\beta_m^2+\sigma^2} \langle \underline{y}, \underline{u}_m \rangle \cdot \underline{u}_m$$

inner product

$\underline{u}_m^H \cdot \underline{y}$

Summary:

1. Transform data $\underline{y}$ into a convenient basis (DFT)

$$\underline{y} \mapsto \underline{U}^H \underline{y} = \underline{w}$$

2. Process individual coefficients in transform domain

$$\underline{w} \mapsto \hat{\underline{\theta}} = \underline{D}(\underline{D} + \sigma^2 \underline{I})^{-1} \underline{w}$$

$$\Rightarrow \hat{\theta}_m = \underbrace{\frac{\beta_m^2}{\beta_m + \sigma^2}}_{\alpha_m} \cdot w_n$$

3. Transform back to original domain

$$\hat{\underline{\theta}} \mapsto \underline{U}\hat{\underline{\theta}} = \hat{\underline{x}}$$

☆ DFT and Wiener filter work well <u>if</u> $\underline{R}_{xx}$ is known and circulant

(and if a few $\beta_n^2 \gg \sigma^2$, but most $\beta_m^2 \ll \sigma^2$; i.e., sparseness)

most signal enery packed into few DFT bands

<u>Special Case</u>: Projection estimator.

Suppose $\beta_0^2, \beta_1^2, \beta_2^2, \cdots, \beta_L^2 \gg \sigma^2$

$$\beta_{L+1}^2, \ldots, \beta_{N-1}^2 = 0$$

i.e., lowpass signal.

Then

$$\alpha_m = \begin{cases} 1, & m = 0, \ldots, L \\ 0, & \text{otherwise} \end{cases}$$

bandpass (lowpass) filter

<u>Keep</u> $\underbrace{\text{in-band}}_{\text{span}(\underline{u}_0, \ldots, \underline{u}_L)}$ data component

<u>Kill</u> $\underbrace{\text{out-of-band}}_{\text{span}(\underline{u}_{L+1}, \ldots, \underline{u}_{N-1})}$ component

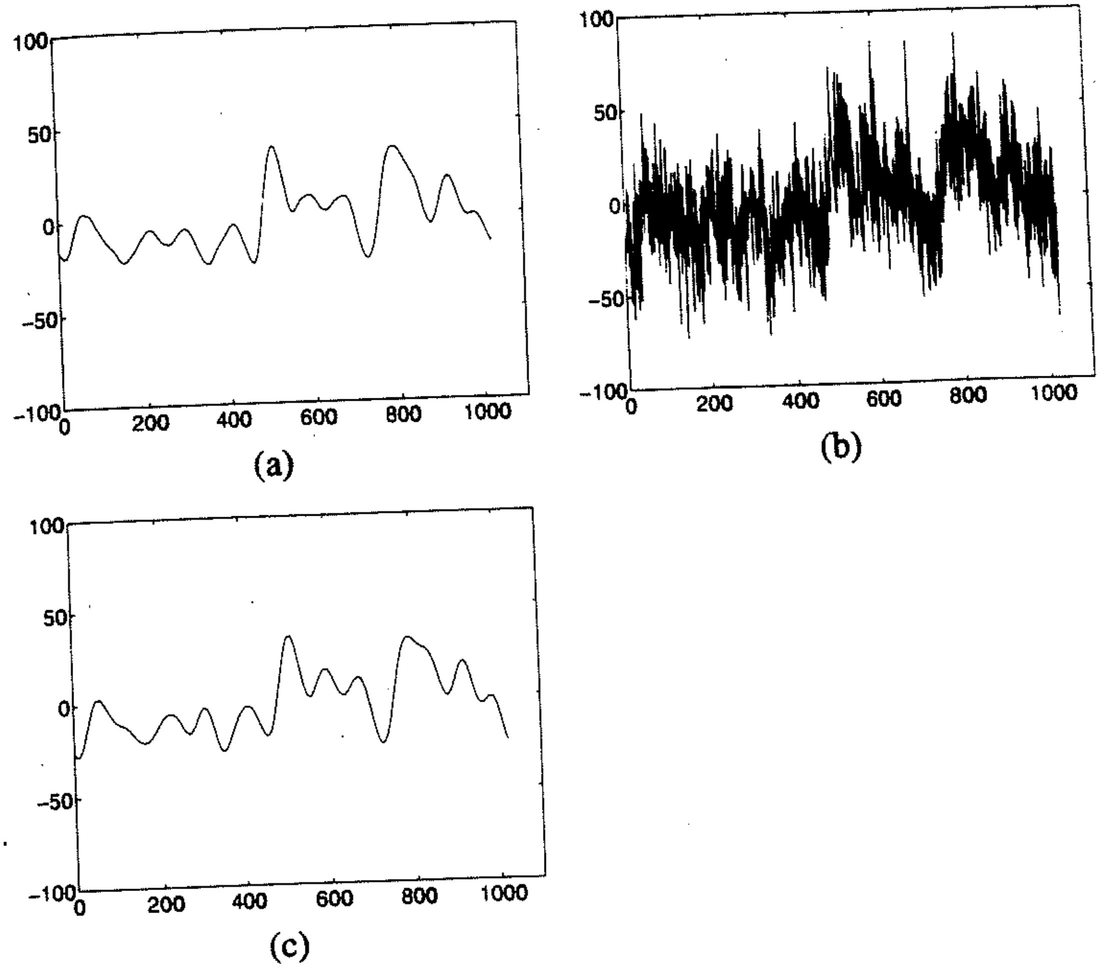


FIGURE 10.1 (a): Realization of a Gaussian process Y. (b): Noisy signal obtained by adding a Gaussian white noise (SNR $= -0.48$ db). (c): Wiener estimation $\tilde{Y}$ (SNR $=$ 15.2 db).

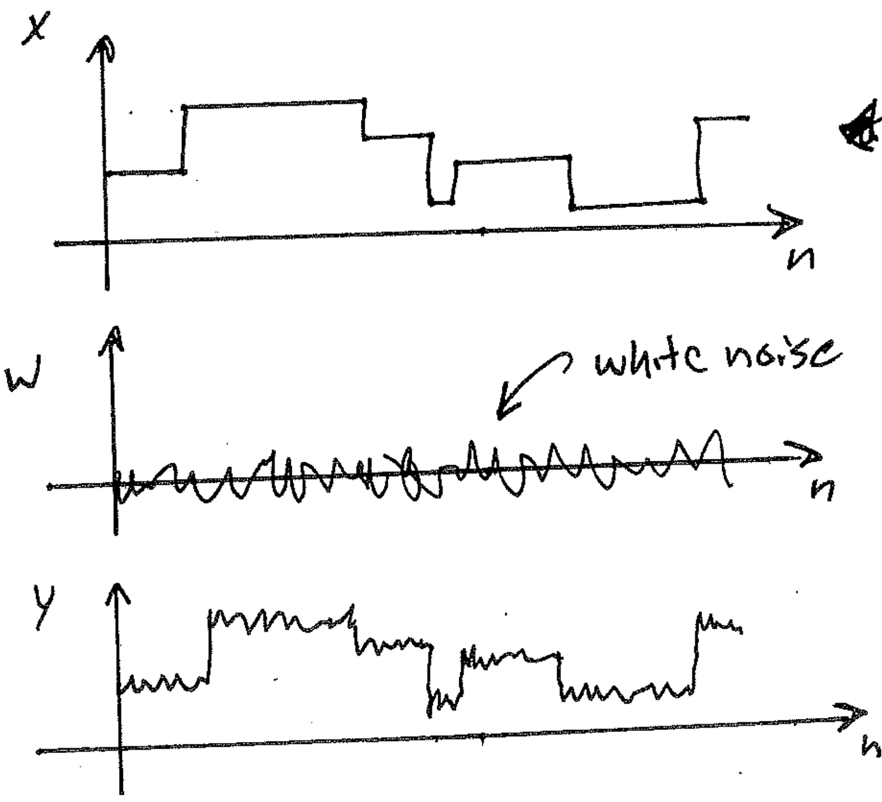
The signal Y was obtained by circularly convolving a sequence of Gaussian white noise with a lowpass filter, resulting in a lowpass, Gaussian, circular stationary process. In this case, the optimal Wiener filter is a simple frequency domain filtering method.

Piecewise Smooth Signals

Frequency domain filtering works well if the signal is well-matched to the DFT basis (e.g., smooth everywhere). The DFT basis is not well-matched to piecewise smooth functions.

Let's consider an extreme case:

Piecewise constant signals in noise



what basis would you use for this signal?

Haar

Nonlinear estimators based on the Haar DWT (instead of the DFT) can dramatically outperform linear estimators by not averaging across discontinuities. To get a feeling for this, suppose we know where the breakpoints are in advance and compute the minimum MSE fit of a constant to each piece of the signal. In this case, the estimation error can be bounded as follows.

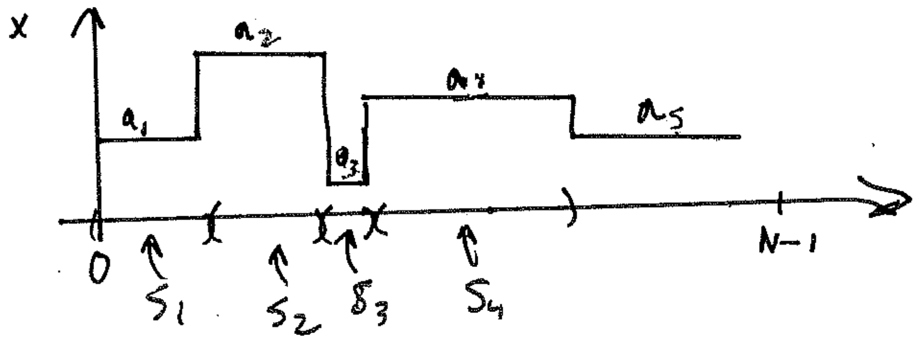
Proposition: If $\underline{x}$ consists of K constant pieces, and if we know where the breakpoints are located, then the signal estimate obtained by simply fitting a constant to each piece of the noisy signal (in a least squares sense) produces an estimation error

$$\mathrm{MSE} = \frac{E\left[\|\underline{x} - \hat{\underline{x}}\|^2\right]}{N} = \frac{\overset{\text{Const.}}{\cancel{K}}\sigma^2}{N} \sim O(N^{-1})$$

proof: Let S_ℓ, $\ell = 1, \dots, K$

denote the K intervals on which

$\underline{x}$ is known to be constant.



The best (least squares) fit on each segment is simply

$$\hat{a}_\ell = \frac{1}{|S_\ell|} \sum_{n \in S_\ell} y(n)$$

length of S_ℓ

Average over corresponding set of noisy observations

$$E[\hat{a}_\ell] = a_\ell \quad \text{(unbiased)}$$

$$var(\hat{a}_\ell) = \frac{\sigma^2}{|S_\ell|} =$$

$E[(a_\ell - \hat{a}_\ell)^2] = E[a_\ell^2 - 2a_\ell \hat{a}_\ell + \hat{a}_\ell^2] = a_\ell^2 - 2a_\ell^2 + E[\hat{a}_\ell^2] = -a_\ell^2$

Total MSE:

$$E[\|\underline{x} - \hat{\underline{x}}\|^2] = E[\|\underline{a} - \hat{\underline{a}}\|^2]$$

$$= \sum E(a_\ell - \hat{a}_\ell)^2 = \sum var(\hat{a}_\ell)$$

$$= \frac{\sigma^2}{N} \sum_{\ell=1}^{K} \overbrace{\frac{N}{|S_\ell|}}^{1/P_\ell} = \frac{\sigma^2}{N} \cdot \frac{1}{P_\ell}$$

P_ℓ is const (indep of N because as N↑, $|S_\ell|$↑)

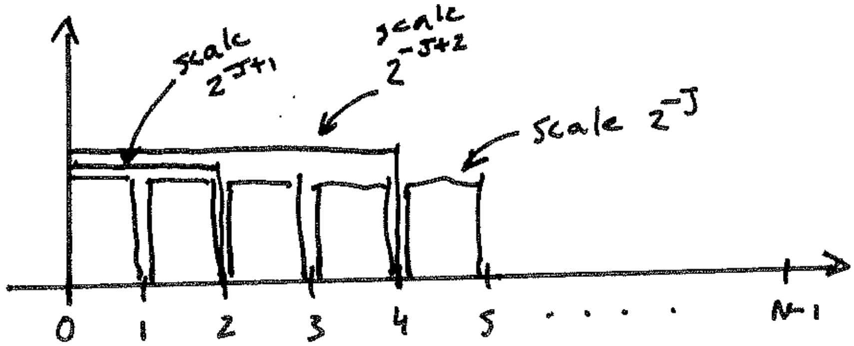
What if we don't know where the breakpoints are? How well can we do?

Linear Wavelet Estimation:

Suppose we consider estimators based on the Haar wavelet transform.

In particular, let's consider estimators based on Haar wavelets with a limited upper resolution (less than or equal to the resolution of our samples).

Ex. Suppose N, our signal length, is a power of two. Then use a Haar approximation at scale 2^{-j}, $j \le \log N$.



The idea is that we will estimate $\underline{x}$ by projecting $\underline{y}$ onto a Haar approx subspace of a certain fixed resolution.

This projection is simply a linear filtering process.

How well will this linear Haar-based filtering approach work?

Certainly it seems like the Haar approximations are ideally suited to piecewise constant signals.

Convergence Rate for Linear Estimators

Consider a piecewise constant signal consisting of K pieces:

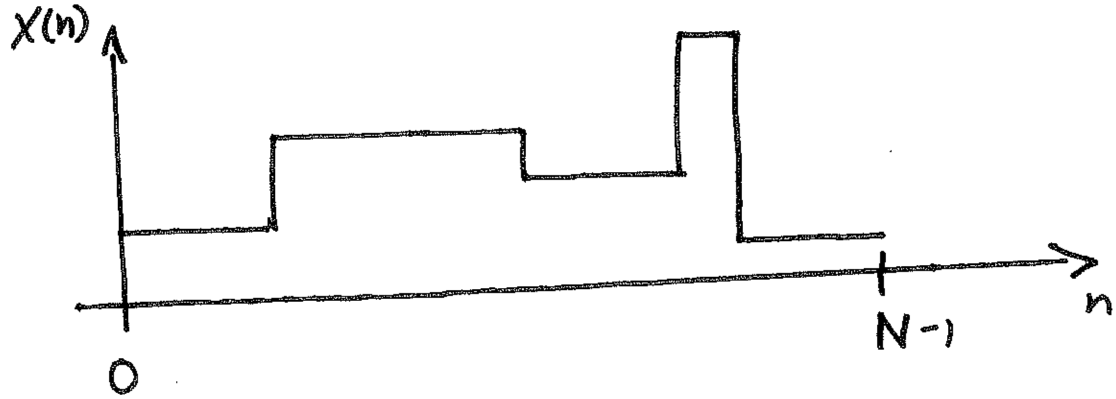
$$x(n) = \sum_{k=1}^{K} a_k I_{S_k}(n) \quad ; \quad n = 0, \ldots, N-1$$

where $S_1 = [0, t_1]$, $S_2 = [t_1, t_1 + t_2]$,

$S_K = [t_1 + \cdots + t_{K-1}, t_1 + \cdots + t_K]$

with $\sum_{i=1}^{K} t_i = N$

K=5



Assume that $K \ll N$, but the positions of the breakpoints are unknown.

Suppose we observe x in additive GWN:

$$y(n) = x(n) + w(n)$$

$$w(n) \sim N(0, \sigma^2).$$

We want to estimate x from y.

Haar basis with fixed resolution,

Let us consider piecewise constant estimators with equal-width pieces:

$$\hat{x}_M(n) = \sum_{l=1}^{M} \hat{\theta}_l \, I_{[(l-1)\frac{N}{M},\, l(\frac{N}{M})]}(n)$$

width of each piece $= \frac{N}{M}$

total # of pieces $= M$.

Note that the breakpoint positions of the estimator $\hat{x}_M$ are <u>fixed</u>.

Therefore, the optimal (in MSE sense) estimate of the coefficients $\hat{b}_\ell$ is simply

$$\hat{b}_\ell = \frac{M}{N} \sum_{n=(\ell-1)\frac{N}{M}}^{\ell(\frac{N}{M}-1)} y(n)$$

(the average over each ~~width~~ interval)

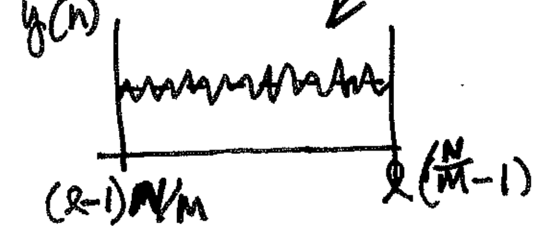
In each interval, one of two things is true:

1. the signal x is constant over the entire interval
2. the signal x contains one or more breakpoints in the interval

Let's compute the error in both cases.

Case 1:

take average and approximate as constant

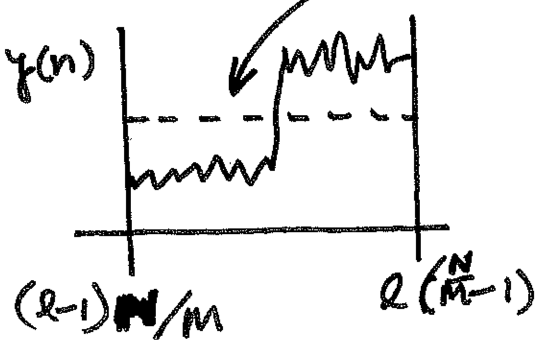


$$err \sim \sigma^2$$

error due to noise alone

case 2:

take average; estimate as constant



$$err \sim \underbrace{C \cdot \frac{N}{M}}_{\text{error due to approx.}} + \sigma^2$$

error due to noise

C = const depending on x

($\propto$ size of jump or diff betw. constant regions)

- Since there are at most K intervals containing a breakpoint of the original signal x,

$$\boxed{\text{total error} \sim K\left(c\frac{N}{M} + \sigma^2\right) + (M-K)\sigma^2}$$

feel edge

don't feel edge

$$= Kc\frac{N}{M} + M\sigma^2$$

- $\Rightarrow$

$$\frac{E\left[\|x-\hat{x}\|^2\right]}{N} \sim \frac{K\cdot c}{M} + \frac{M}{N}\sigma^2$$

min w.r.t. M

$$\frac{-Kc}{M^2} + \frac{\sigma^2}{N} = 0$$

$$\frac{Kc}{M^2} = \frac{\sigma^2}{N} \Rightarrow NcK = \sigma^2 M^2$$

$$\Rightarrow \quad M \sim \sqrt{N} \qquad M \sim \sqrt{N}$$

$$\sim \frac{Kc}{\sqrt{N}} + \frac{\sigma^2}{\sqrt{N}} \sim O(1/\sqrt{N})$$

$$\Rightarrow \quad \frac{E\left[\|x-\hat{x}\|^2\right]}{N} \sim O\left(N^{-\frac{1}{2}}\right)$$

- Compare to $O(N^{-1})$ for estimator that <u>knows</u> where true breakpoints are located.

Summary:

If we have a piecewise constant signal in noise, then we have established the following:

1. if we know the breakpoints, then our estimator has an MSE $\sim O(N^{-1})$

2. if we don't know the breakpoints and use a fixed resolution Haar projection (linear) estimator, we get MSE $\sim O(N^{-\frac{1}{2}})$

Next time:

We will show that the MSE of a Haar thresholding estimator (denoising, nonlinear) behaves like

$$MSE \sim O\left(\frac{\log^2 N}{N}\right)$$