Sparsity-Assisted Signal Smoothing (Revisited)

Ivan Selesnick

Electrical and Computer Engineering
Tandon School of Engineering
New York University
Brooklyn, New York

March 2017

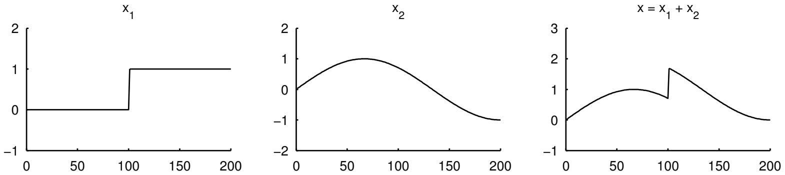
NYU TANDON SCHOOL OF ENGINEERING

Sparsity-Assisted Signal Smoothing (SASS)

We model the signal to be estimated as

$$x = x_1 + x_2$$

$$x,\ x_1,\ x_2 \in \mathbb{R}^N$$



I. W. Selesnick. Sparsity-assisted signal smoothing. In R. Balan et al., editors, *Excursions in Harmonic Analysis, Volume 4*, pages 149–176. Birkhäuser Basel, 2015.

Signal Model

We model the signal to be estimated as

$$x = x_1 + x_2, \qquad x,\ x_1,\ x_2 \in \mathbb{R}^N,$$

where

- Dx_1 is *sparse* where D is differentiation of order K,
  i.e., a suitable regularizer for x_1 is

$$\|Dx_1\|_1$$

- x_2 is a *low-frequency* signal,
  i.e., for a zero-phase low-pass filter H,

$$x_2 = H(x_2)$$

$$x_2 \approx H(x_2 + w)$$

  where w is white Gaussian noise.

Signal Estimation

Goal: Estimate unknown signal x from noisy signal y.

Signal in additive white Gaussian noise (AWGN):

$$\begin{aligned} y &= x + w \\ y &= x_1 + x_2 + w \\ y - x_1 &= x_2 + w \end{aligned}$$

Hence,

$$x_2 \approx H(x_2 + w) \implies x_2 \approx H(y - x_1).$$

If x_1 were known, then we could estimate x_2 by low-pass filtering $y - x_1$.

Thus, we estimate x_2 as

$$\hat{x}_2 = H(y - \hat{x}_1).$$

Signal Estimation

We estimate x as

$$\hat{x} = \hat{x}_1 + \hat{x}_2$$
$$\hat{x} = \hat{x}_1 + H(y - \hat{x}_1)$$
$$\hat{x} = (I - H)\hat{x}_1 + Hy$$
$$\hat{x} = G\hat{x}_1 + Hy$$

where G is a zero-phase high-pass filter

$$G = I - H.$$

Signal Estimation

We assume G is a high-pass filter that admits the factorization

$$G = RD$$

where D is differentiation of order K.

G could be a Butterworth or Chebshev-II filter. Then

$$\hat{x} = RD\hat{x}_1 + Hy$$

i.e.,

$$\hat{x} = R\hat{u} + Hy$$

where

$$\hat{u} := D\hat{x}_1$$

is *sparse*.

SASS Optimization Problem

Given

$$y = x + w,$$

to estimate x as

$$\hat{x} = R\hat{u} + Hy$$

where u is sparse, we use sparse-regularized least squares

$$\hat{u} = \arg\min_{u} \left\{ \frac{1}{2} \|y - (Ru + Hy)\|_2^2 + \lambda \|u\|_1 \right\}$$

where

$$\|x\|_2^2 := \sum_n x(n)^2, \qquad \|x\|_1 := \sum_n |x(n)|, \qquad \lambda > 0.$$

SASS Optimization Problem

Problem

$$\hat{u} = \arg\min_{u} \left\{ \frac{1}{2} \|y - (Ru + Hy)\|_2^2 + \lambda \|u\|_1 \right\}$$

i.e.,

$$\hat{u} = \arg\min_{u} \left\{ \frac{1}{2} \|(I - H)y - Ru\|_2^2 + \lambda \|u\|_1 \right\}$$

can be solved via forward-backward splitting (FBS), ISTA, FISTA, etc.

P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In H. H. Bauschke et al., editors, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212. Springer-Verlag, 2011.

Filters as Matrices

A banded Toeplitz matrix P

$$P = \begin{bmatrix} p_2 & p_1 & p_0 & & & \\ & p_2 & p_1 & p_0 & & \\ & & \ddots & & \ddots & \\ & & & p_2 & p_1 & p_0 \end{bmatrix}$$

represents an LTI system.

Convolution:

$$[Px]_n = (p * x)(n)$$

Transfer function:

$$P(z) = \sum_n p_n z^{-n}$$

Frequency response:

$$P(\mathrm{e}^{\mathrm{j}\omega}) = \sum_n p_n \, \mathrm{e}^{-\mathrm{j}n\omega}$$

Filters as Matrices

Consider cost function

$$J(x) = \|Q(y - x)\|_2^2 + \alpha\|Px\|_2^2, \qquad \alpha > 0$$

where P and Q are banded Toeplitz matrices as above.

The function J is minimized by

$$x = Hy$$

where

$$H := (Q^{\mathsf{T}}Q + \alpha P^{\mathsf{T}}P)^{-1}Q^{\mathsf{T}}Q,$$

i.e.,

$$\boxed{H := A^{-1}Q^{\mathsf{T}}Q}$$

where

$$A := Q^{\mathsf{T}}Q + \alpha P^{\mathsf{T}}P.$$

A is banded.

Filters as Matrices

Matrix H represents an LTI system with transfer function

$$H(z) = \frac{Q(z)Q(1/z)}{Q(z)Q(1/z) + \alpha P(z)P(1/z)}$$

and frequency response

$$H(\mathrm{e}^{\mathrm{j}\omega}) = \frac{|Q(\mathrm{e}^{\mathrm{j}\omega})|^2}{|Q(\mathrm{e}^{\mathrm{j}\omega})|^2 + \alpha |P(\mathrm{e}^{\mathrm{j}\omega})|^2}$$

Note that $H(\mathrm{e}^{\mathrm{j}\omega})$ is zero-phase (i.e., real-valued).

Butterworth Low-pass Filter

Some classical filters have transfer functions of the form

$$H(z) = \frac{Q(z)Q(1/z)}{Q(z)Q(1/z) + \alpha P(z)P(1/z)}.$$

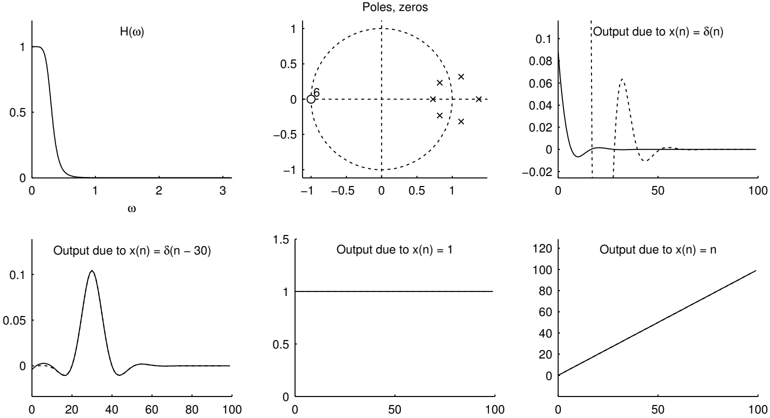
For example, the Butterworth low-pass filter has

$$P(z) = (1 - z^{-1})^d$$
$$Q(z) = (1 + z^{-1})^d.$$

When $d = 3$,

$$P = \begin{bmatrix} -1 & 3 & -3 & 1 & & & \\ & -1 & 3 & -3 & 1 & & \\ & & \ddots & & & \ddots & \\ & & & -1 & 3 & -3 & 1 \end{bmatrix}, \; Q = \begin{bmatrix} 1 & 3 & 3 & 1 & & & \\ & 1 & 3 & 3 & 1 & & \\ & & \ddots & & & \ddots & \\ & & & 1 & 3 & 3 & 1 \end{bmatrix}.$$

Butterworth Low-pass Filter



Zero-phase Butterworth filter for finite-length data. (The dashed line is the old formulation which has end-point transients.)

High-pass Filter

If H is a zero-phase low-pass filter, then $G = I - H$ is a zero-phase high-pass filter

$$G(z) = 1 - H(z) = \frac{\alpha P(z) P(1/z)}{Q(z) Q(1/z) + \alpha P(z) P(1/z)}.$$

and

$$\begin{aligned} G &= I - H \\ &= I - A^{-1} Q^{\mathsf{T}} Q \\ &= A^{-1} (A - Q^{\mathsf{T}} Q) \\ &= \alpha A^{-1} P^{\mathsf{T}} P \end{aligned}$$

where we used $A = Q^{\mathsf{T}} Q + \alpha P^{\mathsf{T}} P$ from above.

Factorization

Let D be the K-order difference

$$D(z) = (1 - z^{-1})^K$$

When $K = 2$,

$$D = \begin{bmatrix} 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & & \ddots & \\ & & & 1 & -2 & 1 \end{bmatrix}$$

If $1 \leqslant K \leqslant d$, then

$$\begin{aligned} P(z) &= (1 - z^{-1})^d \\ &= (1 - z^{-1})^{d-K} (1 - z^{-1})^K \\ &= P_1(z)\, D(z) \\ P &= P_1 D \end{aligned}$$

Filters as Matrices

Summarizing:

$$
\begin{aligned}
A &= Q^\mathsf{T} Q + \alpha P^\mathsf{T} P \\
H &= A^{-1} Q \\
P &= P_1 D \\
G &= I - H \\
&= \alpha A^{-1} P^\mathsf{T} P \\
&= \alpha A^{-1} P^\mathsf{T} P_1 D \\
&= R D
\end{aligned}
$$

where

$$
R = \alpha A^{-1} P^\mathsf{T} P_1
$$

SASS Optimization Problem

The SASS cost function

$$J(u) = \frac{1}{2}\|(I - H)y - Ru\|_2^2 + \lambda\|u\|_1$$

where

$$I - H = RD$$

can then be written

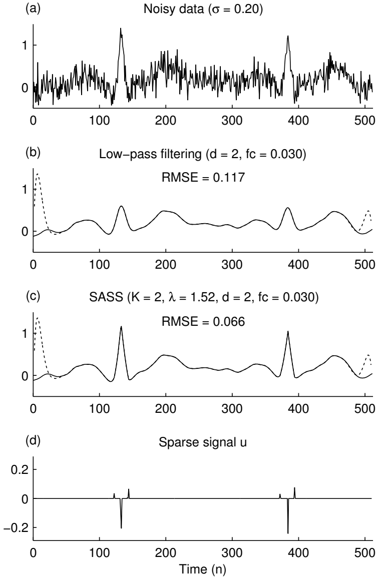
$$J(u) = \frac{1}{2}\|\alpha A^{-1}P^{\mathsf{T}}Py - \alpha A^{-1}P^{\mathsf{T}}P_1 u\|_2^2 + \lambda\|u\|_1$$

or

$$J(u) = \frac{1}{2}\|\alpha A^{-1}P^{\mathsf{T}}(Py - P_1)u\|_2^2 + \lambda\|u\|_1$$

which can be solved using proximal algorithms.

Example


(a)
Noisy data (σ = 0.20)
1
0
0
100
200
300
400
500
(b)
Low−pass filtering (d = 2, fc = 0.030)
RMSE = 0.117
1
0
0
100
200
300
400
500
(c)
SASS (K = 2, λ = 1.52, d = 2, fc = 0.030)
RMSE = 0.066
1
0
0
100
200
300
400
500
(d)
Sparse signal u
0.2
0
−0.2
0
100
200
300
400
500
Time (n)

Conclusion

Numerous signals can be modeled as the sum of two component signals:
(1) signal with a sparse K-order derivative.
(2) low-frequency signal and

LTI filters over-smooth discontinuities (e.g., 'corners' of a signal).

Sparsity-assisted signal smoothing (SASS) combines and unifies LTI low-pass filtering and generalized total-variation denoising.

The SASS algorithm formulates the denoising problem as a sparse deconvolution problem which can be solved via proximal algorithms.

For the formulation and efficient implementation of SASS, we formulate zero-phase recursive filtering of finite-length input signals in terms of banded Toeplitz matrices.