

IEEE Signal Processing Magazine

Lecture Note

Accepted Article

ℓ_2 and ℓ_1 Trend Filtering: A Kalman Filter Approach^a

ARMAN KHEIRATI ROONIZI

ℓ_2 AND ℓ_1 trend filtering are two of the most popular denoising algorithms that are widely used in science, engineering, and statistical signal and image processing applications. They are typically treated as separate entities, with the former as a linear time invariant (LTI) filter which is commonly used for smoothing the noisy data and detrending the time-series signals while the latter is a nonlinear filtering method suited for the estimation of piecewise-polynomial signals (e.g., piecewise-constant, piecewise-linear, piecewise-quadratic and etc.) observed in additive white Gaussian noise. In this article, we propose a Kalman filtering approach to design and implement ℓ_2 and ℓ_1 trend filtering with the aim of teaching these two approaches and explaining their differences and similarities. Hopefully the framework presented in this article will provide a straightforward and unifying platform for understanding the basis of these two approaches. In addition, the material may be useful in lecture courses in signal and image processing, or indeed, it could be useful to introduce our colleagues in signal processing to the application of Kalman filtering in the design of ℓ_2 and ℓ_1 trend filtering.

^aThis is the author's version of the manuscript accepted for publication in IEEE Signal Processing Magazine, Volume 38, Issue 6, August 2021. The paper can be found in its final format at: IEEE Signal Processing Magazine website doi: 10.1109/MSP.2021.3102900

20xx IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Relevance

Trend filtering is the process of smoothing a time series by filtering out the noise. Assuming that a time series y_k , $k = 1, \dots, L$, consists of an underlying slowly time varying trend x_k and a more rapidly varying random component v_k (i.e., $y_k = x_k + v_k$), the goal of trend filtering is to reconstruct the trend component x_k or, equivalently, estimate and remove the random component $v_k = y_k - x_k$ from the time series y_k . It can be considered as an optimization problem with the objective of maximizing the smoothness of x_k and minimizing the residual error between the actual and smooth series. The process of estimating x_k is known as “filtering” or “smoothing” in some contexts. The term “filtering” is used in analogy with trend filtering. It is actually a batch method for estimating the trend component taking all measurements (past, present and future samples) into account, so implemented non-causally in the literature.

Trend filtering has a broad range of applications in science, engineering, statistical signal and image processing, financial time series analysis, geophysics, and biological and medical sciences. Among many trend filtering methods (e.g., moving average filtering, band-pass filtering, smoothing splines, Gaussian process regression, and many other popular methods), ℓ_2 trend filtering and its variation ℓ_1 trend filtering provide significant advances for estimating signals that exhibit varying degrees of smoothness across the input domain [1, 2]. In the following, a brief discussion on these two methods is given.

ℓ_2 trend filtering

The ℓ_2 trend filtering also known as quadratic variation (QV) regularization is a linear time invariant (LTI) filter which is commonly used for smoothing the noisy data and detrending the time-series signals. There are a lot of connections in the literature to ℓ_2 trend filtering. Some of them are QV regularization and smoothness priors; the ill-posed inverse problems and problems of statistical Tikhonov regularization; the problem of estimating a smooth trend embedded in white noise addressed by Whittaker and Bohlmann’s work who studied the application of regularized method for time series smoothing; Hodrick-Prescott (H-P) filtering also known as Hodrick-Prescott decomposition to remove the cyclical component from time-series data and estimate the trend signal (see [3, 4, 5] and the references therein). ℓ_2 trend filtering is

defined as the problem of minimizing the ℓ_2 -norm of the signal derivative subject to a data fidelity constraint [see equation (2a)]. It is an effective smoothing filter method for recovering the smooth polynomial (slowly time varying) signals. Figure 1 illustrates an example of ℓ_2 trend filtering. The noise-free signal (slowly time varying signal), the Gaussian white noise and noisy signal (the mixture of the signal and Gaussian white noise) with SNR = 5 dB are shown in Figures 1(a)-1(c). The time scale for changes in the noise and the signal are different, so the signal and the noise does not overlap in frequency domain (i.e., the signal is a low-frequency component and noise is a high-frequency component). It can be seen from their spectral representation that are shown at the bottom of each figure. The signal obtained using ℓ_2 trend filtering (with different orders, $n = 1, 2, 3$) are shown in Figures 1(d)-1(f). For quantitative comparison, the result of NSR which is a classical ratio between the power of the reconstruction error and the power of the signal (i.e., $\text{NSR} = \sqrt{\sum_k (x_k - \hat{x}_k)^2 / \sum_k x_k^2}$, where x_k and \hat{x}_k are respectively the original and estimated signal [3]) is also reported. The results show that ℓ_2 trend filtering is able to effectively recover the desired signal. Note that the original signal is highly smooth. Therefore, the ℓ_2 trend estimate becomes close to the original signal as the order of the method increases. The reason is that ℓ_2 trend filtering has a strong connection to LTI filtering and smoothing. Specially, it is a smoothing method defined by a problem of optimization with a smoothing constraint, in which the weight controlling the constraint is directly related to cut-off frequencies and the smoothing order is directly controlled with the order of derivatives [3, 4, 5]. Therefore, it is able to effectively separate the slow time varying signal and the rapidly time varying noise. However, in some signal processing applications, the signals are more complex. For instance, consider a situation in which a discrete event phenomenon is observed in the presence of a smooth (polynomial) signal (i.e., the signal derivative is discontinuous). Such signals are known as sparse-derivative (step or piecewise-polynomial) signal. In such cases, the time scale for changes in the noise and the signal are similar, so the signal and the noise overlap in frequency domain considerably. As a result, for such signals, ℓ_2 trend filtering has some limitations that make it inefficient to reconstruct the signal of interest. In other word, ℓ_2 trend filtering is unable to separate signal and noises that overlap in frequency domain. As an example, a piecewise-constant signal and its noisy signal are depicted in Figure 2(a) and 2(d), respectively. Such noisy step-like time series is ob-

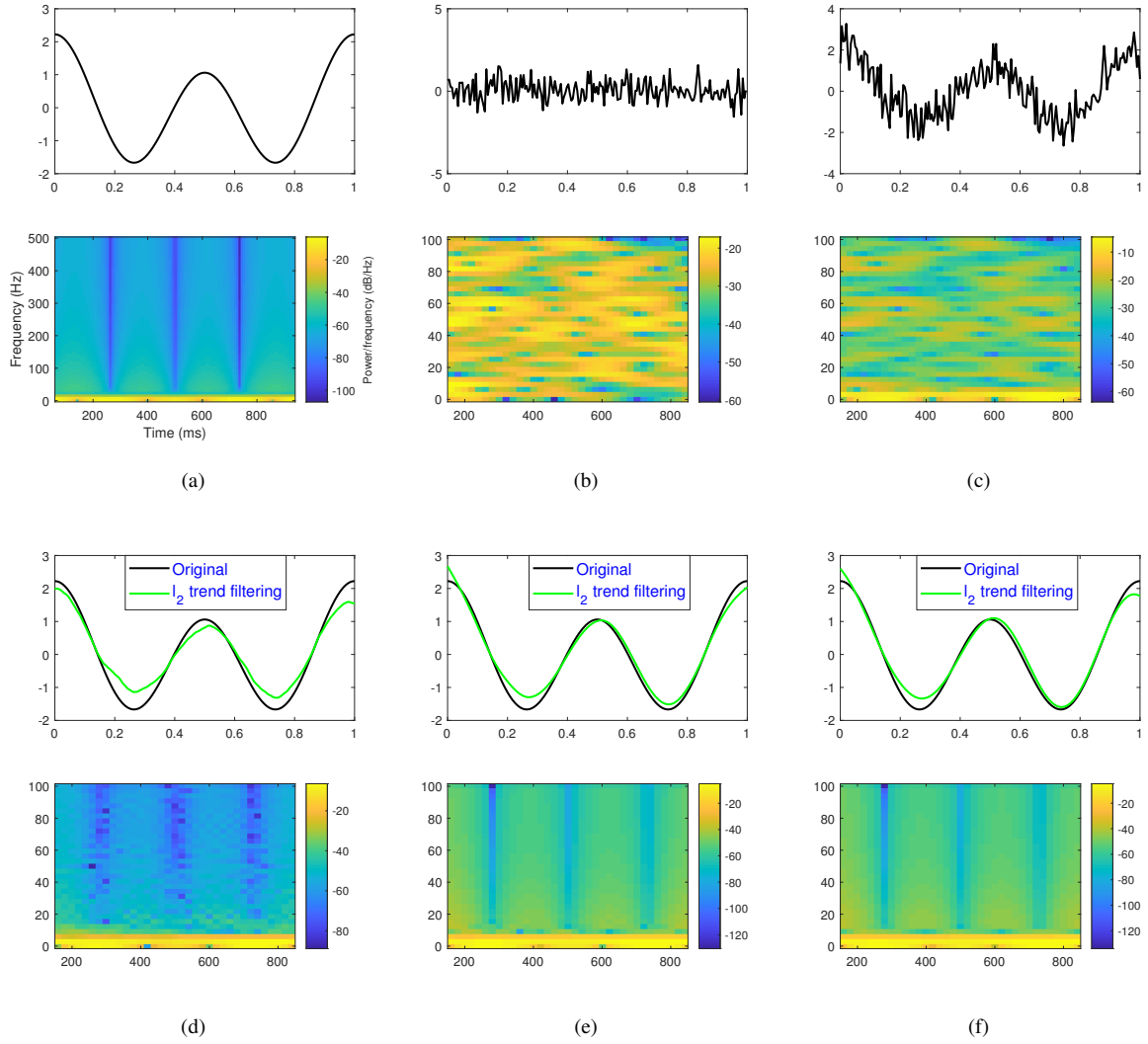


Figure 1: Example 1. ℓ_2 trend filtering. (a) Noise-free signal. (b) Additive white Gaussian noise (c) Noisy signal with SNR = 5 dB (d) Output of the first order ℓ_2 trend filtering (NSR = 0.26) (e) Output of the second order ℓ_2 trend filtering (NSR = 0.15) (f) Output of the third order ℓ_2 trend filtering (NSR = 0.13).

served in molecular machines which are corrupted by molecular noise [6]. We need to remove the noise and reconstruct the desired time series. The signal obtained using the first order ℓ_2 trend filtering is shown in Figure 2(e). The estimated signal does not have sharp discontinuity. The reason is that the signal and noise overlap in frequency domain (see their spectral representation). Another reason is that ℓ_2 -norm is sensitive to the large values of signal derivative [the signal derivative is depicted in Figure 2(b)]. Therefore, there is a need for another solution that is less sensitive to the abrupt changes of the signal derivative. ℓ_1 -norm is less sensitive to the large values than ℓ_2 -norm. So, it is more effective to estimate the signal by minimizing the ℓ_1 -norm of the signal derivative instead [see equation (2b)]. The method is known as ℓ_1 trend filtering. The

details of the method is provided in the following subsections. The output of ℓ_1 trend filtering is illustrated in Figure 2(f). As can be seen it is approximately a step function.

ℓ_1 trend filtering

The ℓ_1 trend filtering better known as total variation (TV) regularization is a nonlinear filtering method which produces trend estimates that have a sparse derivative, and therefore is well suited for the estimation of sparse derivative (i.e., piecewise-polynomial) signals observed in additive white Gaussian noise. It was firstly introduced by Rudin et al. in the field of signal/image processing as a variation on ℓ_2 trend filtering (or QV regularization) which substitutes a sum

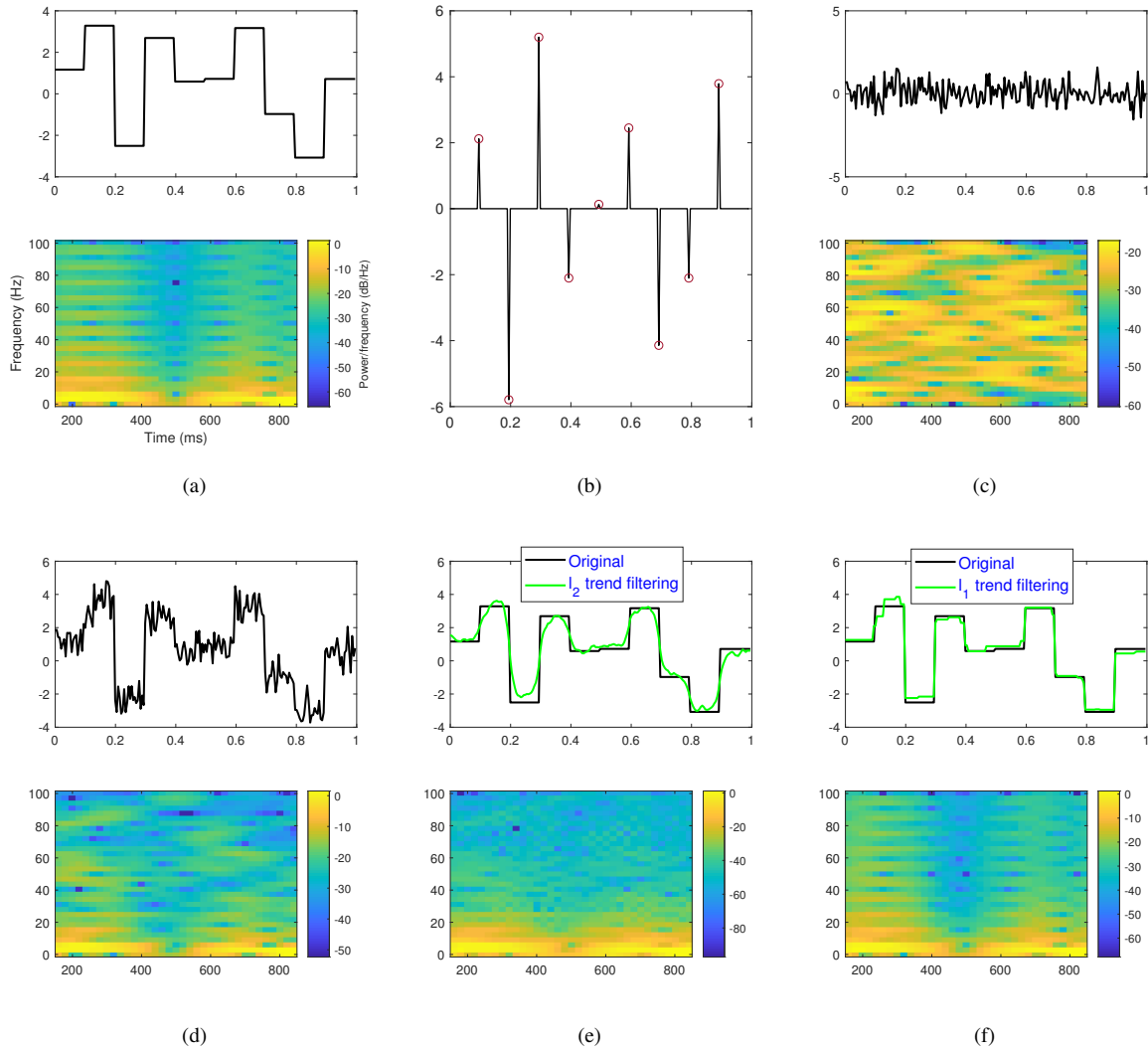


Figure 2: Example 2. Piecewise constant signal smoothing. (a) Noise-free signal. (b) the first derivative (c) Additive white Gaussian noise (d) Noisy signal with SNR = 10 dB (e) Output of the first order ℓ_2 trend filtering (NSR = 0.29) (f) Output of the first order ℓ_1 trend filtering (NSR = 0.12).

of absolute values (i.e., an ℓ_1 -norm) for the sum of squares used in ℓ_2 trend filtering to penalize variations in the estimated trend [7] and later popularized by Kim et al. as a variation of H-P filtering in applied mathematics literature [2]. The term TV (ℓ_1 -norm) was used as a regularization approach in a wide range of applications in image processing, including image restoration, denoising, deconvolution, interpolation and compressed sensing, etc. The ℓ_1 trend filtering is similar to ℓ_2 trend filtering but produces a piecewise-polynomial estimate of the trend makes it suitable for analyzing the time series with an underlying piecewise-polynomial trend. The ℓ_1 trend filtering is challenging due to the non-differentiability of the ℓ_1 -norm while ℓ_2 trend filtering is not since the ℓ_2 -norm is differentiable.

In this lecture note, we present a unified framework based on Kalman filter (KF) and Kalman smoother (KS) to implement these two approaches. The proposed framework provides a unifying platform for understanding the basis of these two approaches and explaining their differences and similarities. It is shown that the ℓ_2 trend filtering is a special case of the ℓ_1 trend filtering. As mentioned before, the ℓ_2 and ℓ_1 trend filtering approaches are implemented non-causally in the literature. Based on the proposed framework, it is possible to causally implement these trend filters: while KS can be used to implement the traditional non-causal trend filters, the KF can be used to convert them to a causal filter design scheme.

Prerequisites

This lecture note requires basic knowledge of system theory, state-space model, estimation theory and Kalman filter.

Problem Statement

The problem is to estimate an unknown signal x_k from the observation y_k in the model:

$$y_k = x_k + v_k, \quad k = 1, \dots, L, \quad (1)$$

where v_k is the unwanted additive noise signal, which we can use the different ℓ_2 trend filtering (QV regularization) or ℓ_1 trend filtering (TV regularization) for rejecting the noise and preserving the smoothness of the estimated x_k , by solving the following optimization problems [3, 7]

$$\ell_2 \text{ trend filtering: } \operatorname{argmin}_{x_k} \sum_{j=1}^L [y_j - x_j]^2 + \lambda \sum_{j=1}^L [\nabla^n x_j]^2 \quad (2a)$$

$$\ell_1 \text{ trend filtering: } \operatorname{argmin}_{x_k} \sum_{j=1}^L [y_j - x_j]^2 + \lambda \sum_{j=1}^L |\nabla^n x_j|, \quad (2b)$$

for all $k = 1, \dots, L$, where the regularization parameter λ controls the degree of smoothing, $\nabla x_j = x_j - x_{j-1}$ is the first order difference and $\nabla^n x_j = \nabla(\nabla^{n-1})x_j$ is the n -th order difference which is precisely defined by

$$\nabla^n x_j = \sum_{i=0}^n (-1)^i \binom{n}{i} x_{j-i}.$$

For instance, the first and second order difference are defined as

$$\begin{cases} \nabla x_j \triangleq x_j - x_{j-1} \\ \nabla^2 x_j \triangleq x_j - 2x_{j-1} + x_{j-2} \end{cases}. \quad (3)$$

Consequently the first and second order ℓ_2 trend filtering are

respectively defined as

$$\begin{cases} \operatorname{argmin}_{x_k} \sum_{j=1}^L [y_j - x_j]^2 + \lambda \sum_{j=1}^L [x_j - x_{j-1}]^2 \\ \operatorname{argmin}_{x_k} \sum_{j=1}^L [y_j - x_j]^2 + \lambda \sum_{j=1}^L [x_j - 2x_{j-1} + x_{j-2}]^2 \end{cases}, \quad (4)$$

while in ℓ_1 trend filtering, they are defined as

$$\begin{cases} \operatorname{argmin}_{x_k} \sum_{j=1}^L [y_j - x_j]^2 + \lambda \sum_{j=1}^L |x_j - x_{j-1}| \\ \operatorname{argmin}_{x_k} \sum_{j=1}^L [y_j - x_j]^2 + \lambda \sum_{j=1}^L |x_j - 2x_{j-1} + x_{j-2}| \end{cases}. \quad (5)$$

In the derivation that follows, we consider the problem of signal smoothing using ℓ_2 and ℓ_1 trend filtering. We derive a KF model for implementing these two approaches and pointing out their differences and similarities.

Solution

In this section, we derive a KF model for ℓ_2 trend filtering and ℓ_1 trend filtering, respectively. First, we derive a dynamic model in state-space form to represent each approach and combine it with the Kalman filtering framework to estimate the desired signal.

0.1 ℓ_2 trend filtering

The unconstrained optimization problem (2a) can be represented in the form of the optimal Wiener smoothing filter. See [4, section 4] for more details. The linear state-space model corresponding to it can be represented as [4, equation (16)]

$$\begin{cases} \nabla^n x_k = w_k \\ y_k = x_k + v_k \end{cases}, \quad (6)$$

where v_k is known as the observation noise, w_k is an additive zero-mean random term and known as the process (model) noise. The optimal value of λ in (2a) is related to the process noise power σ_w^2 and the observation noise power σ_v^2 (see [4, equation (20)] and [8, equation (30)]):

$$\lambda = \frac{\sigma_v^2}{\sigma_w^2} \quad (7)$$

0.2 ℓ_1 trend filtering

In other word, the effect of process noise power and observation noise power are hidden in the regularization parameter. Therefore, constrained optimization parameter, λ , plays the role of balancing the optimization between the two noise terms power. In other word, λ can be chosen based on the ratio of variance of the two noises. As mentioned before, the ℓ_2 trend filtering is a low-pass smoothing filter and the regularization parameter is related to the cutoff frequency [3, equation (23)]:

$$\lambda = \frac{1}{(2 \sin \frac{\omega_c}{2})^{2n}}. \quad (8)$$

So, for extracting a signal within a predetermined frequency band, the regularization parameter can be chosen based on the value of cutoff frequency (8). It is more convenient to express (6) as

$$\begin{cases} x_k = \sum_{i=1}^n \alpha_i x_{k-i} + w_k \\ y_k = x_k + v_k \end{cases}, \quad (9)$$

where $\alpha_i = (-1)^{i+1} \binom{n}{i}$. In order to estimate the smooth trend signal x_k using KF structure, we need to consider the canonical observable representation of (9). A standard dynamical model of (9) that can be used in the framework of KF is as follows:

$$\begin{cases} \mathbf{x}_k = \mathbf{A}_n \mathbf{x}_{k-1} + \mathbf{b}_n w_k \\ y_k = \mathbf{h}_n \mathbf{x}_k + v_k \end{cases}, \quad (10)$$

where

$$\mathbf{x}_k = \begin{pmatrix} x_k \\ x_{k-1} \\ \vdots \\ x_{k-n+1} \end{pmatrix}, \mathbf{A}_n = \begin{pmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_{n-1} & \alpha_n \\ 1 & 0 & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & 1 & 0 & 0 \\ 0 & \dots & 0 & 1 & 0 \end{pmatrix}^T$$

$$\mathbf{b}_n = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \mathbf{h}_n = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (11)$$

For instance, for the first and second order ℓ_2 trend filtering, the matrix \mathbf{A}_n , the vectors \mathbf{x}_k , \mathbf{b}_n and \mathbf{h}_n are respectively

defined as

1-order ℓ_2 trend filtering: $\mathbf{x}_k = x_k$, $\mathbf{A}_1 = 1$, $\mathbf{b}_1 = 1$, $\mathbf{h}_1 = 1$,

2-order ℓ_2 trend filtering: $\mathbf{x}_k = \begin{pmatrix} x_k \\ x_{k-1} \end{pmatrix}$, $\mathbf{A}_2 = \begin{pmatrix} 2 & -1 \\ 1 & 0 \end{pmatrix}$,

$\mathbf{b}_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $\mathbf{h}_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}^T$.

0.2 ℓ_1 trend filtering

In ℓ_1 trend filtering (or TV regularization), the unknown piecewise-polynomial signal x is estimated by solving (2b) which formulates the sparsity based denoising as the problem of minimizing the ℓ_1 norm of the derivative of x subject to the data fidelity constraint. Unfortunately, the second regularization term is non-differentiable. That is why the optimization problem (2b) is known as a difficult minimization problem with no explicit solution. One way to deal with is to replace it with a sequence of simpler ones. This procedure is known as majorization-minimization (MM) method (also known as bound optimization or surrogate optimization method). An overview of MM algorithms in signal processing, machine learning and communications is presented in [9]. As an application of MM approach, it can be used to convert the optimization problem (2b) to a simpler one. To this purpose, the MM approach proposes the following majorizer for the $|x_k|$ [9, Example 6]:

$$|x_k| \leq \frac{1}{2} \frac{x_k^2}{|\hat{x}_k^{(m)}|} + \frac{1}{2} |\hat{x}_k^{(m)}|, \quad (12)$$

where $\hat{x}_k^{(m)}$ denotes the estimated signal after m iterations (with an initialization, e.g., $\hat{x}_k^{(0)} = y_k$). We leave the derivation of the majorizer in (12) as an exercise to the reader. The idea of using the MM approach to ℓ_1 trend filtering (TV regularization) was first proposed by Figueiredo et al. [10]. With it, a majorizer for the ℓ_1 norm is defined as

$$\sum_{j=1}^L \frac{1}{2} |\nabla^n x_j| := \sum_{j=1}^L \frac{1}{2} \frac{[\nabla^n x_j]^2}{|\nabla^n \hat{x}_j^{(m)}|} + C, \quad (13)$$

where

$$C \triangleq \frac{1}{2} \sum_{j=1}^L |\nabla^n \hat{x}_j^{(m)}|$$

and does not depend on x_j . Therefore, in order to solve (2b), one can solve the following iterative optimization problem,

instead:

$$\operatorname{argmin}_{x_k} \frac{1}{2} \sum_{j=1}^L [y_j - x_j]^2 + \lambda \sum_{j=1}^L \frac{1}{2} \frac{[\nabla^n x_j]^2}{|\nabla^n \hat{x}_j^{(m)}|} + C. \quad (14)$$

Note that the second regularization term is now differentiable and $\hat{x}_k^{(m)}$ is considered as a constant with respect to x_k . In the next section, we derive a KF model for ℓ_1 trend filtering. (14) can be expressed as

$$\operatorname{argmin}_{x_k} \frac{1}{2} \sum_{j=1}^L [y_j - x_j]^2 + \lambda \sum_{j=1}^L \frac{1}{2} \left[\frac{\nabla^n x_j}{\sqrt{|\nabla^n \hat{x}_j^{(m)}|}} \right]^2 + C. \quad (15)$$

As shown in the appendix, the linear state-space model corresponding to (15) can be represented as

$$\begin{cases} \frac{\nabla^n x_k}{\sqrt{|\nabla^n \hat{x}_k^{(m)}|}} = w_k \\ y_k = x_k + v_k \end{cases}, \quad (16)$$

which can also be rewritten as

$$\begin{cases} \nabla^n x_k = \phi_k^{(m)} w_k \\ y_k = x_k + v_k \end{cases} \quad (17)$$

where $\phi_k^{(m)} = \sqrt{|\nabla^n \hat{x}_k^{(m)}|}$. We express (17) as

$$\begin{cases} x_k = \sum_{i=1}^n \alpha_i x_{k-i} + \phi_k^{(m)} w_k \\ y_k = x_k + v_k, \end{cases} \quad (18)$$

where α_i is defined before. (18) can be rewritten in the following state space dynamical model:

$$\begin{cases} \mathbf{x}_k = \mathbf{A}_n \mathbf{x}_{k-1} + \mathbf{b}_{n,k} w_k \\ y_k = \mathbf{h}_n \mathbf{x}_k + v_k \end{cases}, \quad (19)$$

where \mathbf{A}_n , \mathbf{x}_k and \mathbf{h}_n are defined before and $\mathbf{b}_{n,k} = [\phi_k^{(m)}, 0, \dots, 0]^T$. For instance, for the first and second order ℓ_1 trend filtering, the matrix \mathbf{A}_n , the vectors \mathbf{x}_k , \mathbf{b}_n and \mathbf{h}_n

are respectively defined as

$$\text{First order } \ell_1 \text{ trend filtering: } \mathbf{x}_k = x_k, \mathbf{A}_1 = 1, \mathbf{b}_{1,k} = \phi_k^{(m)}, \mathbf{h}_1 = 1,$$

$$\text{Second order } \ell_1 \text{ trend filtering: } \mathbf{x}_k = \begin{pmatrix} x_k \\ x_{k-1} \end{pmatrix}, \mathbf{A}_2 = \begin{pmatrix} 2 & -1 \\ 1 & 0 \end{pmatrix}, \mathbf{b}_{2,k} = \begin{pmatrix} \phi_k^{(m)} \\ 0 \end{pmatrix}, \mathbf{h}_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}^T.$$

0.3 Kalman filter implementation

The state space models derived in the previous sections can be coupled with the Kalman filtering framework in order to estimate the desired signal. The KF for (10) and (19) are given as follows [11]:

Time Update:

$$\begin{cases} \hat{\mathbf{x}}_{k|k-1} = \mathbf{A}_n \hat{\mathbf{x}}_{k-1|k-1} \\ \mathbf{P}_{k|k-1} = \mathbf{A}_n \mathbf{P}_{k-1|k-1} \mathbf{A}_n^T + \mathbf{b}_{n,k} q_k \mathbf{b}_{n,k}^T \end{cases} \quad (20)$$

Measurement update:

$$\begin{cases} \hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k [y_k - \mathbf{h}_n \hat{\mathbf{x}}_{k|k-1}] \\ \mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{h}_n^T (\mathbf{h}_n \mathbf{P}_{k|k-1} \mathbf{h}_n^T + r_k)^{-1}, \\ \mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{h}_n \mathbf{P}_{k|k-1} \end{cases} \quad (21)$$

where $q_k \triangleq \mathbb{E}\{w_k^2\}$, $r_k \triangleq \mathbb{E}\{v_k^2\}$, $\hat{\mathbf{x}}_{k|k-1} \triangleq \mathbb{E}\{\mathbf{x}_k | y_{k-1}, \dots, y_1\}$ is the *a priori* estimate of the state vector \mathbf{x}_k in the k -th stage using the observation y_1 to y_{k-1} , and $\hat{\mathbf{x}}_{k|k} \triangleq \mathbb{E}\{\mathbf{x}_k | y_k, \dots, y_1\}$ is the *a posteriori* estimate of the state vector after using the k -th observation y_k . The matrices $\mathbf{P}_{k|k-1} \triangleq \mathbb{E}\{(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})^T\}$ and $\mathbf{P}_{k|k} \triangleq \mathbb{E}\{(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k})(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k})^T\}$ are also defined as the *prior* and *posterior* state covariance matrices, while \mathbf{K}_k is the Kalman gain.

A KS is usually employed after KF to produce a smoother results. It consists of a forward KF stage followed by a backward recursive smoothing stage. Comparing the KF equations for ℓ_1 and ℓ_2 trend filtering, the only difference is that the system noise gain matrix in ℓ_1 trend filtering is time variant ($\mathbf{b}_{n,k} = [\phi_k^{(m)}, 0, \dots, 0]^T$) while it is constant for ℓ_2 trend filtering ($\mathbf{b}_{n,k} = [1, 0, \dots, 0]^T$). It is worth to mention that against the KF implementation of ℓ_2 trend filtering that we run

KF only one time over all k , in ℓ_1 trend filtering, one should run KF over all k , for a given m ; and then this has to run multiple times for $m = 1, 2, 3, \dots$. Finally, ℓ_2 trend filtering is a special case of ℓ_1 trend filtering when we set $\phi_k^{(0)}$ to 1 (i.e., the initial estimation, $\hat{x}_k^{(0)}$, is chosen such that $\nabla^n \hat{x}_k^{(0)} = 1$). In other words, the initial estimation is chosen as a polynomial of order n .

In the context of estimation theory, it is proved that for stationary processes, the KF converges to the optimal Wiener filter in steady state. It is also shown in appendix that the optimization problems (2b) and its particular case (2a) converge to the optimal Wiener smoothing filter. The time of convergence depends on the covariances of the process and measurement noises, namely q_k and r_k [or to their ratio, as seen in equations (22) and (32)]. Therefore, in the Kalman filtering framework, we set q_k and r_k such that

$$\lambda = \frac{\sigma_v^2}{\sigma_w^2} = \frac{r_k}{q_k}. \quad (22)$$

Illustrative Example

The motivation behind ℓ_1 trend filtering was discussed in the previous sections. It was shown that ℓ_2 trend filtering is a special case of ℓ_1 trend filtering. An example of piecewise-constant signal smoothing was given in Section 2 and the first order ℓ_1 trend filtering was used to reconstruct the desired signal. We saw that the first order ℓ_1 trend filtering can be used to effectively separate the piecewise-constant signal and the random noise even if they overlap in the frequency domain. The key point is that for fitting a piecewise-polynomial of degree $n - 1$ to the data, a weighted ℓ_1 -norm of the n -th order difference of the signal can be used as the penalty term in ℓ_1 trend filtering formula. In this section, we mainly focus on ℓ_1 trend filtering as ℓ_2 trend filtering is a special case of it. Let us consider again the piecewise-constant signal given in Example 2. In Figure 2, we saw the result of ℓ_2 trend filtering using KS (trend smoothing in the Kalman filtering). The result of applying the first order ℓ_1 trend filtering implemented by KF and KS to the noisy piecewise-constant signal is shown in Figure 3. The KF is a causal algorithm and can be used for real time applications. The KS algorithm preserves the edges better than KF. This is because the KS algorithm uses information brought by ‘‘future’’ observations, therefore, provides better estimates of the current states than KF. However, it can

only be used for offline applications.

Now, we consider the signals that have more complex shape. We give an example from electrocardiogram (ECG) signal smoothing. ECG is a time-varying signal which provides some information of the cardiac health status. It consists of different complex waveforms, classically labeled as P-wave (linked to atrial depolarization), QRS complex (corresponds to ventricular depolarization) and T-wave (corresponds to ventricular repolarization). In the following, we use ℓ_2 and ℓ_1 trend filtering for ECG signal smoothing. We use the MIT-BIH Atrial Fibrillation Database [12]. As an example, Figure 4(a) shows a specific case record 08378m from this database. Its noisy data is plotted in Figure 4(b). The noisy data is the result of contaminating the original signal by a random noise with SNR = 10. The ECG smoothing for this specific case using ℓ_2 and ℓ_1 trend filtering are reported in panels 4(c) and 4(d), respectively. For both ℓ_2 and ℓ_1 trend filtering algorithm, we use $n = 3$, which corresponds to modeling the ECG components segment as having a sparse order-3 derivative (i.e., approximately piecewise polynomial with polynomial segments of order 2). As can be seen, ℓ_2 trend filtering leads to the distortion of QRS complexes, while ℓ_1 trend filtering does not, as highlighted using arrows in panels 4(c) and 4(d). It is possible to solve the problem of ℓ_2 trend filtering by decreasing the regularization parameter (i.e., increasing the cutoff frequency). In that case, it follows the shape of QRS complexes but follows the noises in low-frequency components as highlighted using arrows in Figure 5.

What we have learned

Based on this article, using KF and KS, readers could implement the ℓ_2 and ℓ_1 trend filtering. One of the major advantages of the proposed framework is that it brings ℓ_2 and ℓ_1 trend filters together in a unified way. Therefore, it can be used for explaining their differences and similarities.

Appendix

Considering the fact that the n -th order difference of x_k can be expressed as the convolution of $\theta_{n,k}$ with the signal

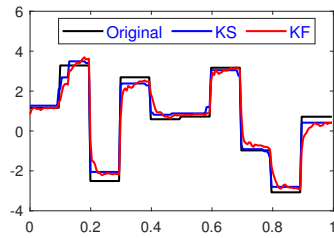


Figure 3: The piecewise-constant (sparse first order derivative) signal smoothing using ℓ_1 trend filtering implemented by KF and KS procedure.

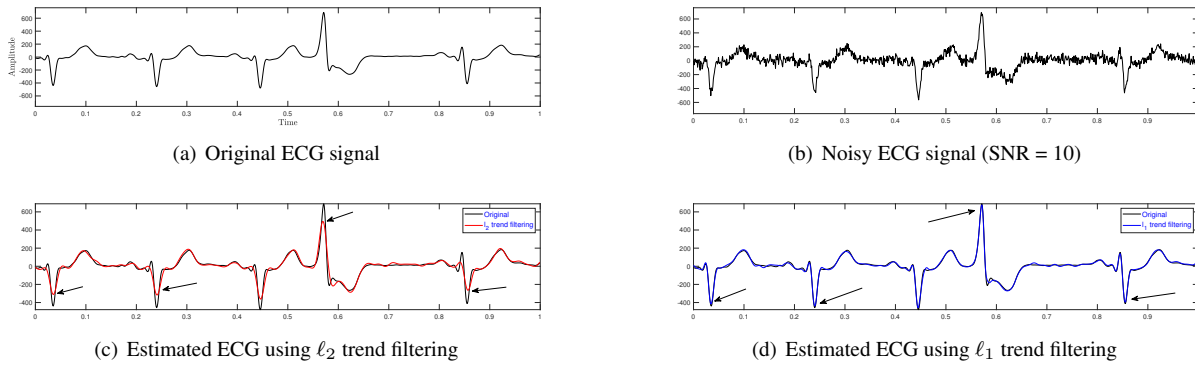


Figure 4: ECG signal smoothing for record 08378m from the MIT-BIH Atrial Fibrillation Database (afdb) using the third order ℓ_2 and ℓ_1 trend filtering (a) Original ECG signal (b) Noise ECG signal (c) Estimated ECG using ℓ_2 trend filtering with $\lambda = 4000$ (NSR = 0.3) (d) Estimated ECG using ℓ_1 trend filtering (NSR = 0.11).

x_k , where $\theta_{n,k}$ is

$$\theta_{n,k} = \sum_{i=0}^n (-1)^i \binom{n}{i} \delta_{k-i} \quad (23)$$

and δ_k is the Dirac delta function, (15) is expressed as:

$$\operatorname{argmin}_{x_k} \frac{1}{2} \sum_{j=1}^L [y_j - x_j]^2 + \lambda \sum_{j=1}^L \frac{1}{2} \left[\frac{\theta_{n,j} * x_j}{\phi_j^{(m)}} \right]^2 + C, \quad (24)$$

where $*$ denotes the convolution operator. In the following, for simplifying the notation, we will drop the superscript (m) and we will simply refer to $\phi^{(m)}$ as ϕ . The optimal solution of (24) is found by setting its derivative with respect to x_k to zero. Doing it and considering that $\partial/\partial x_k (\theta_{n,k} * x_k)^2 = 2\theta_{n,-k} * \theta_{n,k} * x_k$ [4, Lemma 2], the optimal solution using

ℓ_1 trend filtering is obtained as

$$\phi_k^2 x_k + \lambda \theta_{n,-k} * \theta_{n,k} * x_k = \phi_k^2 y_k. \quad (25)$$

On the other hand, the state-space model (17) can be expressed as

$$\begin{cases} \theta_{n,k} * x_k = \phi_k w_k \\ y_k = x_k + v_k \end{cases}. \quad (26)$$

The optimal mean-square error estimator that estimates x_k from y_k is the optimal Wiener smoothing filter, whose transfer function is given by

$$G_{opt}(z) = \frac{S_{xx}(z)}{S_{xx}(z) + S_{vv}(z)}, \quad (27)$$

where $S_{xx}(z)$ and $S_{vv}(z)$ denote the power spectral densities

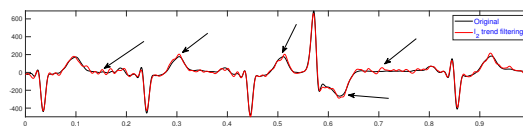


Figure 5: The Estimated ECG using ℓ_2 trend filtering with $\lambda = 20$ (NSR = 0.13).

(PSDs) of the signal and observation noise process, respectively. According to the process model, the PSD of x is equal to

$$S_{xx}(z) = \frac{\Phi(z) * \Phi(z) * S_{ww}(z)}{\Theta_n(z)\Theta_n(\frac{1}{z})}. \quad (28)$$

Substituting (28) in (27) and after some simplifications, we find

$$G_{opt}(z) = \frac{\Phi(z) * \Phi(z) * S_{ww}(z)}{\Phi(z) * \Phi(z) * S_{ww}(z) + S_{vv}(z)\Theta_n(z)\Theta_n(\frac{1}{z})}. \quad (29)$$

Let us consider a specific case where the observation and process noises are Gaussian white noises with variances σ_v^2 and σ_w^2 , respectively. The output of the Wiener smoothing filter in frequency domain is

$$\sigma_w^2 \Phi(z) * \Phi(z) * Y(z) = \sigma_w^2 \Phi(z) * \Phi(z) * X(z) + \sigma_v^2 \Theta_n(z)\Theta_n(\frac{1}{z})X(z). \quad (30)$$

Taking the inverse transform of (30) and after some manipulation, we find

$$\phi_k^2 y_k = \phi_k^2 x_k + \frac{\sigma_v^2}{\sigma_w^2} \theta_{n,k} * \theta_{n,-k} * x_k. \quad (31)$$

Comparing (31) with (25), we conclude that the ℓ_1 trend filtering is a special case of Wiener smoothing filter when the regularization parameter is chosen as

$$\lambda = \frac{\sigma_v^2}{\sigma_w^2}. \quad (32)$$

References

- [1] Collin A Politsch, Jessi Cisewski-Kehe, Rupert A C Croft, and Larry Wasserman. Trend filtering I. A modern statistical tool for time-domain astronomy and astronomical spectroscopy. *Monthly Notices of the Royal Astronomical Society*, 492(3):4005–4018, 2020.
- [2] Seung-Jean Kim, Kwangmoo Koh, Stephen Boyd, and Dmitry Gorinevsky. ℓ_1 trend filtering. *SIAM Review*, 51(2):339–360, 2009.
- [3] Arman Kheirati Roonizi and Christian Jutten. Improved smoothness priors using bilinear transform. *Signal Processing*, 169:107381, 2020.
- [4] Arman Kheirati Roonizi and Christian Jutten. Forward-backward filtering and penalized least-squares optimization: A unified framework. *Signal Processing*, 178:107796, 2021.
- [5] Arman Kheirati Roonizi and Christian Jutten. Band-stop smoothing filter design. *IEEE Transactions on Signal Processing*, 69:1797–1810, 2021.

In a particular case, when $\phi_k = 1$, (31) becomes the optimal solution of (2a):

$$y_k = x_k + \frac{\sigma_v^2}{\sigma_w^2} \theta_{n,k} * \theta_{n,-k} * x_k. \quad (33)$$

It confirms that the ℓ_2 trend filtering is a special case of Wiener smoothing filter [4, section 4].

Acknowledgment

I would like to sincerely thank Professor Christian Jutten, Emeritus Professor of Université Grenoble Alpes, Grenoble, France, for his insightful and motivating comments throughout this work.

Author

Arman Kheirati Roonizi (ebad.kheirati.roonizi@gmail.com) received his Ph.D. degree in computer science from the University of Milan, Italy, in 2017. During 2017-2018, he held a Postdoctoral Research position with GIPSA-lab, Grenoble, France. He is currently an assistant Professor in the Department of Computer Science, Fasa University, Iran. His research interests include signal modeling, processing, filtering and separation, multimodal signal processing, and functional data analysis.

- [6] Max A. Little and Nick S. Jones. Sparse bayesian step-filtering for high-throughput analysis of molecular machine dynamics. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4162–4165, 2010.
- [7] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60:259 – 268, 1992.
- [8] Arman Kheirati Roonizi. A New Approach to ARMAX Signals Smoothing: Application to Variable-Q ARMA Filter Design. *IEEE Transactions on Signal Processing*, 67(17):4535–4544, 2019.
- [9] Y. Sun, P. Babu, and D. P. Palomar. Majorization-minimization algorithms in signal processing, communications, and machine learning. *IEEE Transactions on Signal Processing*, 65(3):794–816, 2017.
- [10] M. A. T. Figueiredo, J. B. Dias, J. P. Oliveira, and R. D. Nowak. On total variation denoising: A new majorization-minimization algorithm and an experimental comparison with wavelet denoising. In *2006 International Conference on Image Processing*, pages 2633–2636, 2006.
- [11] D. Simon. *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. Wiley, 2006.
- [12] Ary L. Goldberger, Luis A. N. Amaral, Leon Glass, Jeffrey M. Hausdorff, Plamen Ch. Ivanov, Roger G. Mark, Joseph E. Mietus, George B. Moody, Chung-Kang Peng, and H. Eugene Stanley. Physiobank, Physiokit, and Physionet: Components of a new research resource for complex physiologic signals. *Circulation*, 101:e215–e220, 2000.