Total Variation Denoising (An MM Algorithm)*

Ivan Selesnick

Abstract

Total variation denoising (TVD) is an approach for noise reduction developed so as to preserve sharp edges in the underlying signal. Unlike a conventional low-pass filter, TV denoising is defined in terms of an optimization problem. This module describes an algorithm for TV denoising derived using the majorization-minimization (MM) approach, developed by Figueiredo et al. [ICIP 2006]. To keep it simple, this module addresses TV denoising of 1-D signals only. For computational efficiency, the algorithm may use a solver for sparse banded systems.

1 Introduction

Total variation denoising (TVD) is an approach for noise reduction developed so as to preserve sharp edges in the underlying signal [13]. Unlike a conventional low-pass filter, TV denoising is defined in terms of an optimization problem. The output of the TV denoising 'filter' is obtained by minimizing a particular cost function. Any algorithm that solves the optimization problem can be used to implement TV denoising. However, it is not trivial because the TVD cost function is non-differentiable. Numerous algorithms have been developed to solve the TVD problem, e.g. [5], [6], [4], [15], [16].

Total variation is used not just for denoising, but for more general signal restoration problems, including deconvolution, interpolation, in-painting, compressed sensing, etc. [2]. In addition, the concept of total variation has been generalized and extended in various ways [12], [10], [3].

These notes describe an algorithm for TV denoising derived using the majorization-minimization (MM) approach, developed by Figueiredo et al. [9]. To keep it simple, these notes address TV denoising of 1-D signals only (ref. [9] considers 2D TV denoising for images). Interestingly, it is possible to obtain the exact solution to the TV denoising problem (for the 1-D case) without optimization, but instead using a direct algorithm based on a characterization of the solution. Recently, a fast algorithm has been developed and C code made available [7].

Total variation denoising assumes that the noisy data $y(n)$ is of the form

$$y(n) = x(n) + w(n), \qquad n = 0, \cdots, N-1 \tag{1}$$

*Version 1.4: Dec 3, 2012 11:08 am -0600

†http://creativecommons.org/licenses/by/3.0/

where $x(n)$ is a (approximately) piecewise constant signal and $w(n)$ is white Gaussian noise. TV denoising estimates the signal $x(n)$ by solving the optimization problem:

$$\arg\min_{x} \sum_{n=0}^{N-1} |y(n) - x(n)|^2 + \lambda \sum_{n=1}^{N-1} |x(n) - x(n-1)|. \quad (2)$$

The regularization parameter $\lambda > 0$ controls the degree of smoothing. Increasing λ gives more weight to the second term which measures the fluctuation of the signal $x(n)$.

MATLAB software is available online at:

`http://eeweb.poly.edu/iselesni/lecture_notes/TVDmm/index.html`

1.1 Notation

The N-point signal x is represented by the vector

$$\mathbf{x} = [x(0), \cdots, x(N-1)]^t. \quad (3)$$

The ℓ_1 norm of a vector $\mathbf{v}$ is defined as

$$\| \mathbf{v} \|_1 = \sum_n |v(n)|. \quad (4)$$

The ℓ_2 norm of a vector $\mathbf{v}$ is defined as

$$\| \mathbf{v} \|_2 = \left[\sum_n |v(n)|^2 \right]^{\frac{1}{2}}. \quad (5)$$

The matrix $\mathbf{D}$ is defined as

$$\mathbf{D} = \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & & \ddots & \\ & & & -1 & 1 \end{bmatrix}. \quad (6)$$

The first-order difference of an N-point signal $\mathbf{x}$ is given by $\mathbf{Dx}$ where $\mathbf{D}$ is of size $(N-1) \times N$.

Note, for later, that $\mathbf{DD}^t$ is a tridiagonal matrix of the form:

$$\mathbf{DD}^t = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & & \ddots & & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}. \quad (7)$$

The total variation of the N-point signal $x(n)$ is given by

$$\mathrm{TV}(\mathbf{x}) := \| \mathbf{Dx} \|_1 = \sum_{n=1}^{N-1} |x(n) - x(n-1)|. \quad (8)$$

With this notation, the TV denoising problem (2) can be written compactly as

$$\arg \underset{\mathbf{x}\in\mathbb{R}^N}{min} \ \parallel \mathbf{y} - \mathbf{x} \parallel_2^2 + \lambda \parallel \mathbf{D}\mathbf{x} \parallel_1 . \tag{9}$$

2 Majorization-Minimization

Majorization-minimization (MM) is an approach to solve optimization problems that are too difficult to solve directly. Instead of minimizing the cost function $F(\mathbf{x})$ directly, the MM approach solves a sequence of optimization problems, $G_k(\mathbf{x})$, $k = 0, 1, 2, \cdots$. The idea is that each $G_k(\mathbf{x})$ is easier to solve that $F(\mathbf{x})$. The MM approach produces a sequence $\mathbf{x}_k$, each being obtained by minimizing $G_{k-1}(\mathbf{x})$. To use MM, one must specify the functions $G_k(\mathbf{x})$. The trick is to choose the $G_k(\mathbf{x})$ so that they are easy to solve, but they should also each approximate $F(\mathbf{x})$.

The MM approach requires that each function $G_k(\mathbf{x})$ is a majorizor of $F(\mathbf{x})$, i.e.,

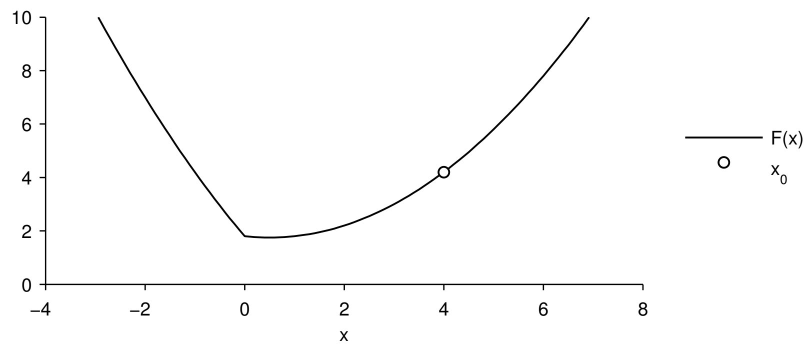
$$G_k(\mathbf{x}) \geq F(\mathbf{x}), \qquad \forall \mathbf{x} \tag{10}$$

and that it agrees with $F(\mathbf{x})$ at $\mathbf{x}_k$,

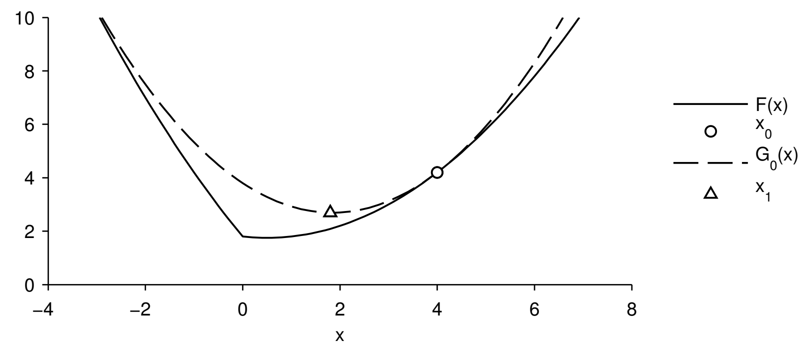
$$G_k(\mathbf{x}_k) = F(\mathbf{x}_k). \tag{11}$$

In addition, $G_k(\mathbf{x})$ should be convex functions. The MM approach then obtains $\mathbf{x}_{k+1}$ by minimizing $G_k(\mathbf{x})$.

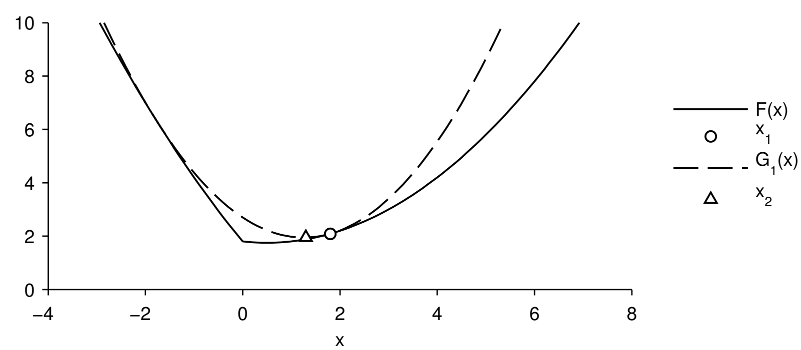
Figure 11 illustrates the MM procedure with a simple example. For clarity, the figure illustrates the minimization of a univariate function. However, the MM procedure works in the same way for the minimization of multivariate functions, and it is in the multivariate case where the MM procedure is especially useful.



(a)



(b)



(c)

Figure 11: Illustration of majorization-minimization (MM) procedure. (a) Cost function $F(x)$ to be minimized; and initialization, x_0. (b) Iteration 1. Majorizor $G_0(x)$ coincides with $F(x)$ at x_0. Minimize $G_0(x)$ to get x_1. (c) Iteration 2. Majorizor $G_1(x)$ coincides with $F(x)$ at x_1. Minimize $G_1(x)$ to get x_2. The x_k converge to the minimizer of $F(x)$.

The majorization-minimization approach to minimize the function $F(\mathbf{x})$ can be summarized as:

1. Set $k = 0$. Initialize $\mathbf{x}_0$.
2. Choose $G_k(\mathbf{x})$ such that
   a. $G_k(\mathbf{x}) \geq F(\mathbf{x})$ for all $\mathbf{x}$
   b. $G_k(\mathbf{x}_k) = F(\mathbf{x}_k)$
3. Set $\mathbf{x}_{k+1}$ as the minimizer of $G_k(\mathbf{x})$.

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}} G_k(\mathbf{x}) \tag{12}$$

4. Set $k = k + 1$ and go to step (2.)

When $F(\mathbf{x})$ is convex, then under mild conditions, the sequence $\mathbf{x}_k$ produced by MM converges to the minimizer of $F(\mathbf{x})$. More details about the majorization-minimization procedure can be found in [8] and references therein.

Example majorizor. An upper bound (majorizor) of $f(t) = |t|$ that agrees with $f(t)$ at $t = t_k$ is

$$g(t) = \frac{1}{2|t_k|} t^2 + \frac{1}{2}|t_k| \tag{13}$$

as illustrated in Figure 15. The figure makes clear that

$$g(t) \geq f(t), \quad \forall t \tag{14}$$

$$g(t_k) = f(t_k) \tag{15}$$

The derivation of the majorizor in (13) is left as exercise list, p. 12.

It is convenient to use second-order polynomials as majorizors because they are easy to minimize. Setting the derivatives to zero gives linear equations. A higher order polynomial could be used to give a closer fit to the function $f(t)$ to be minimized, however, then the minimization will be more difficult (involving polynomial root finding, etc.)

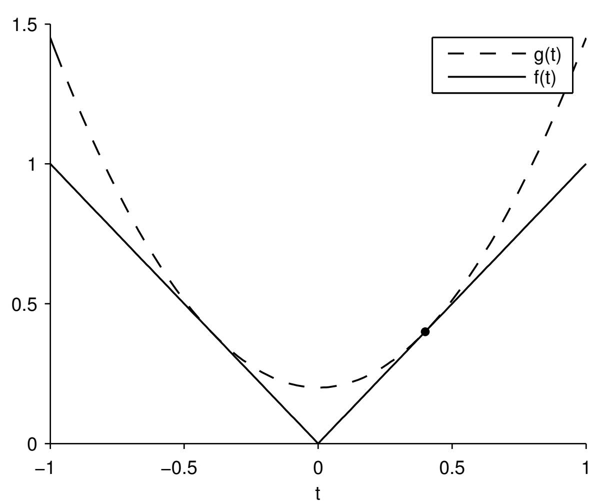


Figure 15: Majorization of $f(t) = |t|$ by $g(t) = at^2 + b$.

3 TV Denoising Algorithm

One way to apply MM to TV denoising is to majorize $\mathrm{TV}(\mathbf{x})$ by a quadratic function of $\mathbf{x}$, as described in ref. [9]. Then the TVD cost function $F(x)$ can be majorized by a quadratic function, which can in turn be minimized by solving a system of linear equations.

To that end, using (13), we can write

$$\frac{1}{2|t_k|}t^2 + \frac{1}{2}|t_k| \geq |t| \qquad \forall\, t \in \mathbb{R} \tag{16}$$

Using $v(n)$ for t and summing over n gives

$$\sum_n \left[\frac{1}{2|v_k(n)|}v^2(n) + \frac{1}{2}|v_k(n)|\right] \geq \sum_n |v(n)| \tag{17}$$

which can be written compactly as

$$\boxed{\tfrac{1}{2}\mathbf{v}^t\Lambda_k^{-1}\mathbf{v} + \tfrac{1}{2}\|\mathbf{v}_k\|_1 \geq \|\mathbf{v}\|_1} \tag{18}$$

where Λ_k is the diagonal matrix

$$\Lambda_k := \begin{bmatrix} |v_k(1)| & & & \\ & |v_k(2)| & & \\ & & \ddots & \\ & & & |v_k(N)| \end{bmatrix} = \mathrm{diag}(|\mathbf{v}_k|). \tag{19}$$

In the notation, $\mathrm{diag}(|\mathbf{v}|)$, the absolute value is applied element-wise to the vector $\mathbf{v}$.

Using $\mathbf{Dx}$ for $\mathbf{v}$, we can write

$$\frac{1}{2}\mathbf{x}^t\mathbf{D}^t\Lambda_k^{-1}\mathbf{Dx} + \frac{1}{2}\|\mathbf{Dx}_k\|_1 \geq \|\mathbf{Dx}\|_1 \tag{20}$$

where

$$\Lambda_k := \mathrm{diag}(|\mathbf{Dx}_k|). \tag{21}$$

Note in (20) that the majorizor of $\|\mathbf{Dx}\|_1$ is a quadratic function of $\mathbf{x}$. Also note that the term $\|\mathbf{Dx}_k\|_1$ in (20) should be considered a constant — it is fixed as $\mathbf{x}_k$ is the value of $\mathbf{x}$ at the previous iteration. Similarly, Λ_k in (20) is also not a function of $\mathbf{x}$.

A majorizor of the TV cost function (9) can be obtained from (20) by adding $\|\mathbf{y}-\mathbf{x}\|_2^2$ to both sides,

$$\|\mathbf{y}-\mathbf{x}\|_2^2 + \lambda\frac{1}{2}\mathbf{x}^t\mathbf{D}^t\Lambda_k^{-1}\mathbf{Dx} + \lambda\frac{1}{2}\|\mathbf{Dx}_k\|_1 \geq \|\mathbf{y}-\mathbf{x}\|_2^2 + \lambda\|\mathbf{Dx}\|_1. \tag{22}$$

Therefore a majorizor $G_k(\mathbf{x})$ for the TV cost function is given by

$$G_k(\mathbf{x}) = \|\mathbf{y}-\mathbf{x}\|_2^2 + \lambda\frac{1}{2}\mathbf{x}^t\mathbf{D}^t\Lambda_k^{-1}\mathbf{Dx} + \lambda\frac{1}{2}\|\mathbf{Dx}_k\|_1, \qquad \Lambda_k = \mathrm{diag}(|\mathbf{Dx}_k|) \tag{23}$$

which satisfies $G_k(\mathbf{x}_k) = F(\mathbf{x}_k)$ by design. Using MM, we obtain $\mathbf{x}_k$ by minimizing $G_k(\mathbf{x})$,

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}}\ \|\mathbf{y}-\mathbf{x}\|_2^2 + \lambda\frac{1}{2}\mathbf{x}^t\mathbf{D}^t\Lambda_k^{-1}\mathbf{Dx} + \lambda\frac{1}{2}\|\mathbf{Dx}_k\|_1. \tag{24}$$

An explicit solution to (24) is given by

$$\mathbf{x}_{k+1} = \left(\mathbf{I} + \frac{\lambda}{2}\mathbf{D}^t\Lambda_k^{-1}\mathbf{D}\right)^{-1}\mathbf{y}. \tag{25}$$

A problem with update (25) is that as the iterations progress, some values of $\mathbf{D}\mathbf{x}_k$ will generally go to zero, and therefore some entries of Λ_k^{-1} in (25) will go to infinity. This issue is addressed in Ref. [9] by rewriting the equation using the matrix inverse lemma. By the matrix inverse lemma,

$$\left(\mathbf{I} + \frac{\lambda}{2}\mathbf{D}^t \Lambda_k^{-1}\,\mathbf{D}\right)^{-1} = \mathbf{I} - \mathbf{D}^t \left(\frac{2}{\lambda}\Lambda_k + \mathbf{D}\mathbf{D}^t\right)^{-1} \mathbf{D} \tag{26}$$

where

$$\Lambda_k = \operatorname{diag}\left(|\mathbf{D}\mathbf{x}_k|\right). \tag{27}$$

Now the update equation (25) becomes

$$\boxed{\mathbf{x}_{k+1} = \mathbf{y} - \mathbf{D}^t \left(\tfrac{2}{\lambda}\operatorname{diag}\left(|\mathbf{D}\mathbf{x}_k|\right) + \mathbf{D}\mathbf{D}^t\right)^{-1} \mathbf{D}\mathbf{y}.} \tag{28}$$

Observe that even if some elements of $\mathbf{D}\mathbf{x}_k$ are zero, no division by zero arises in (28).

The update (28) calls for the solution to a linear system of equations. In general, it is desirable to avoid such a computation in an iterative filtering algorithm due to the high computational cost of solving linear systems (especially when the signal $\mathbf{y}$ is very long and the system is very large). However, the matrix $\left[\frac{2}{\lambda}\operatorname{diag}\left(|\mathbf{D}\mathbf{x}_k|\right) + \mathbf{D}\mathbf{D}^t\right]$ in (28) is a sparse banded matrix; it consists of only three diagonals — the main diagonal, one upper diagonal, and one lower diagonal. This is because $\mathbf{D}\mathbf{D}^t$ is tridiagonal as shown in (7). Therefore, the linear system in (28) can be solved very efficiently [11]. Further, the whole matrix need not be stored in memory, only the three diagonals.

The MATLAB function `TVD_mm` implements TV denoising based on the update (28). The function uses the sparse matrix structure in MATLAB so as to avoid high memory requirements and so as to invoke sparse banded system solvers. MATLAB uses LAPACK [1] to solve the sparse banded system in the program `TVD_mm`. The algorithm used by MATLAB to solve a sparse linear system can be monitored using the command `spparms('spumoni',3)`.

```
function [x, cost] = TVD_mm(y, lam, Nit)
% [x, cost] = TVD_mm(y, lam, Nit)
% Total variation denoising using majorization-minimization
% and a banded linear systems.
%
% INPUT
%   y - noisy signal
%   lam - regularization parameter
%   Nit - number of iterations
%
% OUTPUT
%   x - denoised signal
%   cost - cost function history
%
% Reference
% 'On total-variation denoising: A new majorization-minimization
% algorithm and an experimental comparison with wavalet denoising.'
% M. Figueiredo, J. Bioucas-Dias, J. P. Oliveira, and R. D. Nowak.
% Proc. IEEE Int. Conf. Image Processing, 2006.

% Ivan Selesnick, selesi@poly.edu, 2011

y = y(:);                                                 % Ensure column vector
cost = zeros(1, Nit);                                     % Cost function history
N = length(y);

e = ones(N-1, 1);
DDT = spdiags([-e 2*e -e], [-1 0 1], N-1, N-1);          % D*D' (sparse matrix)
D = @(x) diff(x);                                         % D (operator)
DT = @(x) [-x(1); -diff(x); x(end)];                      % D'

x = y;                                                    % Initialization
Dx = D(x);

for k = 1:Nit
    F = 2/lam * spdiags(abs(Dx), 0, N-1, N-1) + DDT;      % F : Sparse matrix structure
    % F = 2/lam * diag(abs(D(x))) + DDT;                  % Not stored as sparse matrix

    x = y - DT(F\D(y));                                   % Solve sparse linear system
    Dx = D(x);

    cost(k) = sum(abs(x-y).^2) + lam * sum(abs(Dx));      % Save cost function history
end
```

Listing 1: MATLAB program for TV denoising using majorization-minimization. The program is based on update (28).

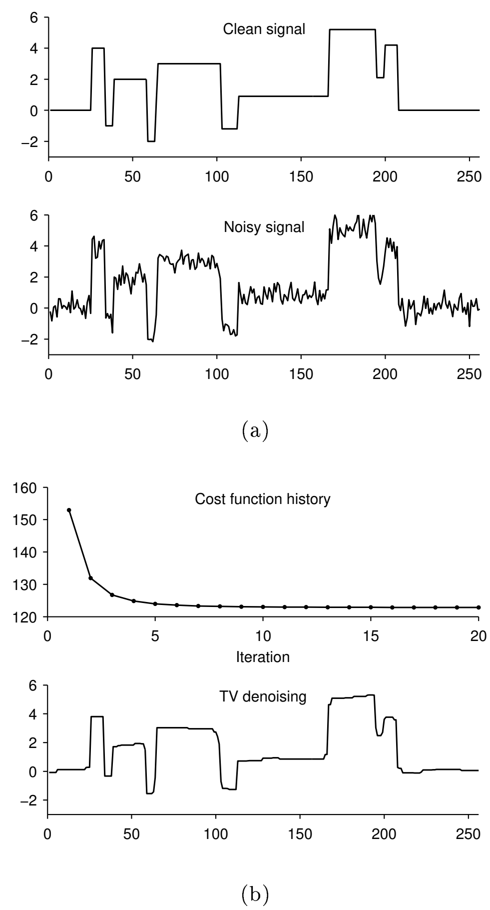


Figure 28: TV denoising example.

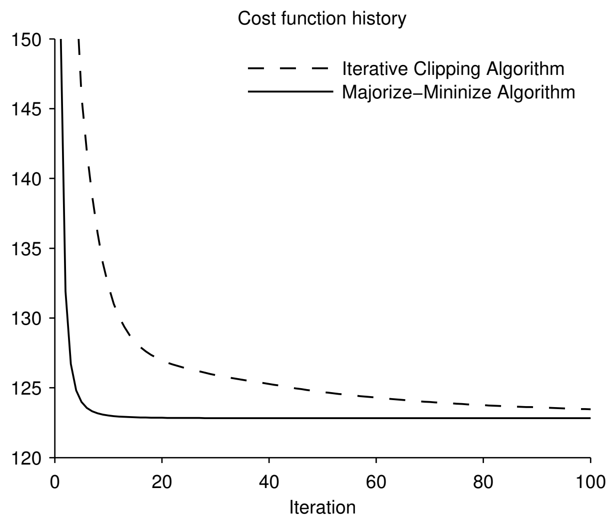


Figure 28: Comparison of convergence behavior of two TV denoising algorithms.

An example of TV denoising is shown in Figure 28. The history of the cost function through the progression of the algorithm is shown in the figure. It can be seen that after 20 iterations the cost function has leveled out, suggesting that the algorithm has practically converged.

Another algorithm for 1-D TV denoising is Chambolle's algorithm [5], a variant of which is the 'iterative clipping' algorithm [14]. This algorithm is computationally simpler than the MM algorithm because it does not call for the solution to a linear system at each iteration. However, it may converge slowly. For the denoising problem illustrated in Figure 28, the convergence of both the iterative clipping and MM algorithms are shown in Figure 28. It can be seen that the MM algorithm converges in fewer iterations.

4 Optimality Condition

It turns out that the solution to the TV denoising problem can be concisely characterized [7]. Suppose the noisy data is $\mathbf{y}$ and the regularization parameter is λ. If $\mathbf{x}$ is the solution to the TV denoising problem, then it must satisfy

$$|s(n)| \leq \frac{\lambda}{2}, \quad n = 0, \cdots, N-1 \tag{29}$$

where $s(n)$ is the 'cumulative sum' of the residual, i.e.

$$s(n) := \sum_{k=0}^{n} \left(y(k) - x(k) \right). \tag{30}$$

The condition (29) is illustrated in Figure 32a for the TV denoising example of Figure 28.

The condition (29) by itself is not sufficient for $x(n)$ to be the solution to the TV denoising problem. It is further necessary that $x(n)$ satisfy

$$\begin{array}{ll} d(n) > 0, & s(n) = \lambda/2 \\ d(n) < 0, & s(n) = -\lambda/2 \\ d(n) = 0, & |s(n)| < \lambda/2 \end{array} \qquad (31)$$

where $d(n)$ is the first-order difference function of $x(n)$, i.e.

$$d(n) = x(n+1) - x(n). \qquad (32)$$

The condition (31) is illustrated in Figure 32b. The figure shows $(d(n), s(n))$ as a scatter plot. It can be seen that this condition requires the points to lie on a curve consisting of three line segments (a 'double-L' shape). Notice in the figure that $d(n)$ is mostly zero, reflecting the sparsity of the derivative of $x(n)$.

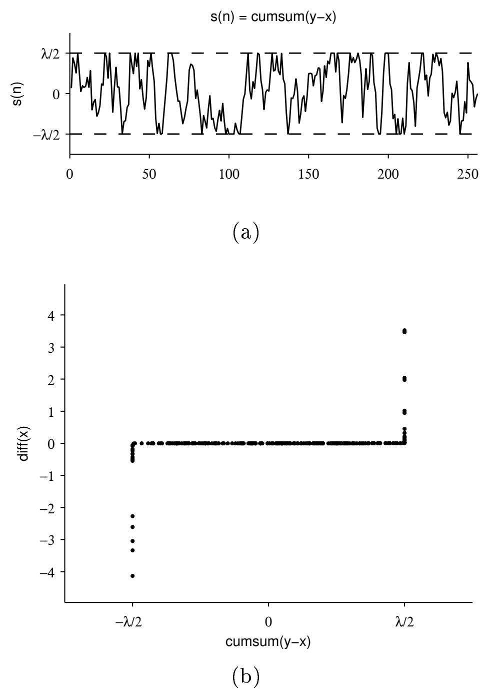


Figure 32: Optimality condition for TV denoising. (a) The cumulative sum $s(n)$ is bounded by $\lambda/2$. (b) Scatter plot of $d(n)$ versus $s(n)$. The points lie on a 'double-L' curve.

5 Conclusion

Total variation (TV) denoising is a method to smooth signals based on a sparse-deriviative signal model. TV denoising is formulated as the minimization of a non-differentiable cost function. Unlike a conventional low-pass filter, the output of the TV denoising 'filter' can only be obtained through a numerical algorithm. Total variation denoising is most appropriate for piecewise constant signals, however, it has been modified and extended so as to be effective for more general signals.

6 Exercises

1. Reproduce figures like those of the example (using a 'blocky' signal). Try different values of λ. How does the solution change as λ is increased or decreased?
2. Compare TV denoising with low-pass filtering (e.g. a Butterworth or FIR filter, etc). Apply each method to the same signal. Plot the denoised/filtered signals using each method and discuss the differences you observe.
3. Perform TV denoising on a signal that is not 'blocky' (which has slopes or oscillatory behavior). You should see 'stair-case' artifacts in the denoised signal. Show these artifacts in a figure and explain why they arise.
4. Is TV denoising linear? (Conventional low-pass filters are linear, e.g. Butterworth filter.) Illustrate that TV denoising satisfies (or does not) the superposition property by performing TV denoising on each of two signals and their sum.
5. Find a majorizor of the function $f(t) = |t|$ of the form

$$g(t) = a\,t^2 + b\,t + c, \tag{33}$$

that coincides with $f(t)$ at $t = t_k$. As illustrated in Figure 15, the function $g(t)$ should satisfy

$$g(t) \geq f(t) \quad \forall\, t \in \mathbb{R}, \tag{34}$$

$$g(t_k) = f(t_k)\,. \tag{35}$$

6. Show the solution to problem (24) is given by (25).
7. For denoising a noisy signal using TV denoising, devise a method or formula to set the regularization parameter λ. You can assume that the variance σ^2 of the noise is known. Show examples of your method.
8. Explain why $\mathbf{D}^t$ can be implemented in MATLAB by the command

```
DT~=~@(x)~[-x(1);~-diff(x);~x(end)];
```

9. Modify the TV denoising MATLAB program so that the matrix $\mathbf{F}$ is not sparse. Measure the run-times of the original and modified programs. Is the sparse version faster? Use a long signal and many iterations to see the difference more clearly.
10. Second-order TV denoising is based the second-order difference instead of the first-order difference. Modify the algorithm and MATLAB program so that it performs second-order TV denoising. Compare first and second order TV denoising using 'blocky' and non-'blocky' signals, and comment on your observations.

References

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen. *LAPACK's user's guide*. SIAM, 1992. http://www.netlib.org/lapack.

[2] J. Bect, L. Blanc-Feaud, G. Aubert, and A. Chambolle. A l1-unified variational framework for image restoration. In *European Conference on Computer Vision, Lecture Notes in Computer Sciences*, volume 3024, pages 1–13, 2004.

[3] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM J. Imag. Sci.*, 3(3):492–526, 2010.

[4] A. Chambolle. An algorithm for total variation minimization and applications. *J. of Math. Imaging and Vision*, 20:89–97, 2004.

[5] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76:1678211;188, 1997.

[6] T. F. Chan, S. Osher, and J. Shen. The digital tv filter and nonlinear denoising. *IEEE Trans. Image Process.*, 10(2):231–241, February 2001.

[7] L. Condat. A direct algorithm for 1d total variation denoising. Technical report, Hal-00675043, 2012. http://hal.archives-ouvertes.fr/.

[8] M. Figueiredo, J. Bioucas-Dias, and R. Nowak. Majorization-minimization algorithms for wavelet-based image restoration. *IEEE Trans. Image Process.*, 16(12):2980–2991, December 2007.

[9] M. Figueiredo, J. Bioucas-Dias, J. P. Oliveira, and R. D. Nowak. On total-variation denoising: A new majorization-minimization algorithm and an experimental comparison with wavalet denoising. In *Proc.IEEE Int.Conf.Image Processing*, 2006.

[10] Y. Hu and M. Jacob. Higher degree total variation (hdtv) regularization for image recovery. *IEEE Trans. Image Process.*, 21(5):2559–2571, May 2012.

[11] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C: the art of scientific computing (2nd ed.).* Cambridge University Press, 1992.

[12] P. Rodriguez and B. Wohlberg. Efficient minimization method for a generalized total variation functional. *IEEE Trans. Image Process.*, 18(2):322–332, February 2009.

[13] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[14] I. Selesnick and I. Bayram. Total variation filtering. *Connexions Web site*, 2009. http://cnx.org/content/m31292/1.1/.

[15] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM J. on Imaging Sciences*, 1(3):248–272, 2008.

[16] M. Zhu, S. J. Wright, and T. F. Chan. Duality-based algorithms for total-variation-regularized image restoration. *J. Comp. Optimization and Applications*, 2008.