Properties of Laplacian Pyramids for Extension and Denoising

William Leeb*

Abstract

We analyze the Laplacian pyramids algorithm of Rabin and Coifman for extending and denoising a function sampled on a discrete set of points. We provide mild conditions under which the algorithm converges, and prove stability bounds on the extended function. We also consider the iterative application of truncated Laplacian pyramids kernels for denoising signals by non-local means.

1 Introduction

This paper analyzes the Laplacian pyramids (LP) algorithm for extending a function sampled on a discrete set of points to outside values. This method was introduced in the context of machine learning by Rabin and Coifman in [20], and is modeled after the classical Laplacian pyramids algorithm of Burt and Adelson [10], which is a standard technique in image processing. The LP extension algorithm has been considered in a variety of applications [14, 11, 24, 1, 18, 12, 2], and several variants have been proposed [15, 21, 22].

The LP algorithm constructs a multiscale decomposition of the estimated function, consisting of averaged differences at successive levels. At each level, the residuals from the previous approximation are averaged and extended to the entire domain. The level 0 approximation is just a weighted average of the observed values. At each sampled point, the residual is then computed, and the average residual is added to form the level 1 approximation. The residuals are computed again, and the average residuals are again added back. This process is repeated, constructing a sequence of approximations at each successive level.

A key observation driving the extension method is that to compute the average residuals, we may use a kernel that is defined on points outside the samples. That is, while the residuals necessarily make use of the observed values, the *averaged* residuals are well-defined everywhere, because the averaging kernel may be computed out-of-sample. Furthermore, a different averaging kernel may be used at each level. The sequence of bandwidths defining the extent of each kernel is typically chosen to be decreasing, with a large initial bandwidth to permit wide extrapolation.

In this paper, we prove certain properties about the LP extension method. First, we show that the scheme does in fact interpolate the observed values (to arbitrarily high precision), and show how the rate of convergence, i.e. the number of levels used in the extension scheme, is controlled by the choice of bandwidths. In particular, we show that the scheme may converge even when the kernel bandwidths do not shrink to 0, or equivalently, when the averaging kernels do not converge to the identity matrix on the sampled points. This permits avoiding the use of small-bandwidth kernels which can introduce spurious artifacts into the extension.

Second, we show that for certain sequences of bandwidths the algorithm is stable, in the sense that the output function is bounded in terms of the maximum value of the input data. The stability bounds we derive are analogous to the stability bound from [13] for classical kernel interpolation methods that involve a single kernel at one scale. Our bound increases with the ratio of the maximum bandwidth to the minimum distance between the sample points, raised to a power that scales inversely to the rate of bandwidth decay.

Third, we consider the use of iterated truncated LP kernels for signal denoising by non-local means (NL means). The two-level version of a truncated LP kernel was employed in this fashion in [23], and was shown to have advantages over a traditional NL means kernel. We consider the advantages of iterating higher-step kernels as well.

Our results are mainly derived from a simple formula for the residual terms of LP at each iteration. This formula expresses the residual operator at each level as a product of differencing operators from the previous scales. Similar decompositions have previously been observed for certain examples of boosting [7, 17, 13, 8, 5, 9, 6, 16], though the applicability to the LP extension algorithm appears to be new.

The rest of the paper is organized as follows. In the remainder of Section 1, we review the LP extension algorithm, and compare it to other kernel-based methods for function extension. In Section 2, we state and prove the main analytical results, namely the factorization of the residual operators, convergence, and stability. In Section 3, we illustrate the use of truncated LP kernels for denoising by non-local means. In Section 4 we provide a brief conclusion.

1.1 The Laplacian pyramids algorithm

In this section, we first review the method for Laplacian pyramids extension, as described in [20]. We are given n samples $\mathcal{X} = \{x_1, \ldots, x_n\}$ from $\mathbb{R}^p$. For any point $x \in \mathbb{R}^p$, we are given a family of kernels $P_0(x, x_j), P_1(x, x_j), P_2(x, x_j) \ldots$, defined

*School of Mathematics, University of Minnesota, Twin Cities. Minneapolis, MN.

on $\mathbb{R}^p \times \mathcal{X}$, which capture the affinity between points $x \in \mathbb{R}^p$ to the sampled points $x_j \in \mathcal{X}$. In this paper we will define the affinities by a radial kernel $\Phi(r)$; that is, we first define:

$$G_\ell(w) = \Phi(\|w\|/\sigma_\ell), \tag{1}$$

for some bandwidth $\sigma_\ell > 0$, and then define the kernel P_ℓ by

$$P_\ell(x, x_j) = \frac{G_\ell(x - x_j)}{\sum_{j'=1}^n G_\ell(x - x_{j'})}. \tag{2}$$

For instance, the function $\Phi(r)$ may be taken to be a Gaussian, $\Phi(r) = e^{-r^2}$ (as suggested in [20], and frequently used in applications). In [20] and most applications we have seen, the sequence of bandwidths σ_ℓ are taken to be geometrically decreasing; that is,

$$\sigma_\ell = \sigma_0/\mu^\ell, \quad \ell \geq 0, \tag{3}$$

for some value $\mu > 1$; $\mu = 2$ is a typical choice.

We are given the values $y_j = f(x_j)$ of a function f at the points x_j. Given a new point $x \in \mathbb{R}^p$, the LP scheme extends f to x by defining a sequence of approximations as follows. The first approximation to $f(x)$ is defined as

$$s_0(x) = \sum_{j=1}^n P_0(x, x_j) f(x_j). \tag{4}$$

If P_0 is row-stochastic over the x_j's then $s_0(x)$ is a weighted average of the observed values $f(x_j)$. We will also denote this by $f_0(x) = s_0(x)$.

At the sample points x_j, $f_0(x_j)$ is an average over all the points $x_1, \ldots, x_n$, and so generally will not be equal to $f(x_j)$. At each sample point x_j, we compute the residual term defined by

$$d_1(x_j) = f(x_j) - f_0(x_j). \tag{5}$$

By definition, $f(x_j) = s_0(x_j) + d_1(x_j)$; so our next task is to extend d_1 to the out-of-sample point x. To extend d_1, we use the next kernel P_1, defining

$$s_1(x) = \sum_{j=1}^n P_1(x, x_j) d_1(x_j). \tag{6}$$

We now can define the level 1 approximation to $f(x)$ as the sum of $s_0(x)$ and $s_1(x)$, namely

$$f_1(x) = s_0(x) + s_1(x). \tag{7}$$

The entire procedure may now be repeated again, at every level. We construct a sequence of estimators $f_\ell(x) = s_0(x) + \cdots + s_\ell(x)$, where

$$s_\ell(x) = \sum_{j=1}^n P_\ell(x, x_j) d_\ell(x_j), \tag{8}$$

and

$$d_\ell(x_j) = f(x_j) - f_{\ell-1}(x_j) = f(x_j) - (s_0(x_j) + \ldots s_{\ell-1}(x_j)). \tag{9}$$

In other words, starting with the level $\ell - 1$ approximation, $f_{\ell-1}(x)$, we find its residuals $d_\ell(x_j)$ at the known points, and define s_ℓ by approximately extrapolating these residuals everywhere using kernel P_ℓ, and then form our refined estimate f_ℓ by adding the estimated residual s_ℓ to $f_{\ell-1}$.

1.2 Other kernel-based methods

The LP algorithm is similar to other kernel-based methods for extending functions sampled on discrete points. We mention two approaches in particular. Kernel interpolation takes a fixed radial function $G(w)$, and seeks to approximate f by writing

$$f(x) = \sum_{i=1}^n \alpha_i G(x - x_i). \tag{10}$$

Because this expression is linear in the coeffcients α_i, they may be fit by least-squares, to ensure that $f(x_i) = y_i$ on the sampled points x_i.

One drawback of this class of methods is that they may suffer from numerical instabilities due to the fitting procedure. This is especially true if the kernel G is chosen to have a large bandwidth, since in this case the functions $G(x - x_i)$ may be nearly linearly dependent if the x_i are too close, and the resulting linear system for the α_i is ill-conditioned.

An alternative approach that is used primarily in the statistics community is known as the Nadaraya-Watson (NW) estimator [26, 19]. This takes a kernel G, and writes the estimated function f as the weighted average of observed values:

$$f(x) = \frac{\sum_{i=1}^{n} G(x - x_i) y_i}{\sum_{i=1}^{n} G(x - x_i)}. \tag{11}$$

A modification of NW is proposed in [16] using the method of L_2 boosting [7]. The residuals at each level are fit using the same kernel G, and the process is then iterated several times. This method can be seen as a special case of LP, where the same bandwidth σ_ℓ is used at every scale, although there is no extra work in introducing variable bandwidths. In this sense, LP and NW with L_2 boosting are essentially identical methods.

1.3 Notation

We will denote by $\overline{P}_k$ the n-by-n matrix with $(i,j)^{th}$ entry $P_k(x_i, x_j)$. The matrix $\overline{P}_k$ is the discretization of the kernel P_k on the n sampled points $x_1, \ldots, x_n$.

Similarly, for any function g defined on all of $\mathbb{R}^p$, we will denote by $\overline{g}$ the vector of samples:

$$\overline{g} = (g(x_1), \ldots, g(x_n))^T. \tag{12}$$

We will also define the following matrices. Let A_ℓ be the ℓ^{th} level LP operator, mapping the vector $\overline{f}$ of observed values to the ℓ^{th} level approximation f_ℓ; that is, $f_\ell = A_\ell \overline{f}$. Following our previous notation, denote by $\overline{A}_\ell$ the n-by-n matrix whose rows are restricted to $x_1, \ldots, x_n$; in this notation, $\overline{f}_\ell = \overline{A}_\ell \overline{f}$.

Define S_ℓ to be the operator mapping $\overline{f}$ to s_ℓ, defined by (8); that is, $s_\ell = S_\ell \overline{f}$. Again, we will let $\overline{S}_\ell$ be the n-by-n matrix whose rows are restricted to $x_1, \ldots, x_n$.

Finally, we let D_ℓ denote the the differencing operator $I - \overline{A}_{\ell-1}$, so that $d_\ell = \overline{f} - \overline{f}_{\ell-1} = (I - \overline{A}_{\ell-1})\overline{f} = D_\ell \overline{f}$. Note that the differencing operators are only defined on the in-sample points x_j, which is why we do not use extra notation in this case.

2 Analysis of LP: convergence and stability

In this section we will address several basic questions about the LP extension algorithm. First, it is not obvious under what conditions the scheme will converge to the observed values y_j on the in-sample points x_j. At level ℓ the residual vectors d_ℓ are averaged using the kernel P_ℓ, and these averaged residuals are added to the approximation. To guarantee convergence of $f_\ell(x_j)$ to y_j, one might suppose that at high levels the residuals d_ℓ must be approximated arbitrarily well – that is, that the matrices $\overline{P}_\ell$ should approach the identity matrix, or equivalently that the bandwidths σ_ℓ approach 0.

As we will show, it turns out that this is not necessary. The LP scheme will interpolate the given points so long as the $\overline{P}_\ell$ are sufficiently close to the identity; however, they do not need to approach the identity. In particular, the sequence of bandwidths may plateau at a sufficiently small value instead of approaching 0 and the scheme will still converge. (The convergence rate, however, will depend on the decay of the bandwidths.) We will also demonstrate on a numerical example that there can be advantages to not using arbitrarily small bandwidths, as small-bandwidth kernels may introduce high-frequency artifacts into the extension.

We will also show that under the same conditions on the bandwidths, the LP algorithm is stable. More precisely, the infinity norm of the exended function cannot exceed a constant times the infinity norm of the input values. Phrased differently, treating LP as an operator that maps the input vector $y = (y_1, \ldots, y_n)^T$ to the extended function f_K, we show that LP is a bounded operator from ℓ_∞ to L^∞. The bound on the operator norm we derive exhibits a similar scaling as bounds for classical kernel interpolation methods shown in [13].

2.1 Factorization of the residual operators D_k

This section derives a factorization of the residual operators D_k, which will be used repeatedly throughout the rest of paper. A similar formula has been shown for certain boosting methods in statistics; see [7, 17, 13, 8]. For completeness we provide a self-contained statement and derivation here.

Proposition 2.1. *The operators D_ℓ may be factored as follows:*

$$D_\ell = (I - \overline{P}_{\ell-1}) \cdots (I - \overline{P}_0), \tag{13}$$

for each $\ell \geq 1$.

Proof. By definition, $A_0 = P_0$, and so $D_1 = I - \overline{A}_0 = I - \overline{P}_0$, proving the claim when $\ell = 1$. We now proceed by induction. Suppose we have shown that $D_\ell = (I - \overline{P}_{\ell-1}) \cdots (I - \overline{P}_0)$ for some $\ell \geq 1$. Because $\overline{A}_{\ell-1} = I - D_\ell$ and $S_\ell = P_\ell D_\ell$, we have:

$$\overline{A}_\ell = \overline{A}_{\ell-1} + \overline{S}_\ell = I - D_\ell + \overline{P}_\ell D_\ell = I - (I - \overline{P}_\ell) D_\ell = I - (I - \overline{P}_\ell)(I - \overline{P}_{\ell-1}) \cdots (I - \overline{P}_0) \tag{14}$$

and consequently

$$D_{\ell+1} = I - \overline{A}_\ell = (I - \overline{P}_\ell)(I - \overline{P}_{\ell-1}) \cdots (I - \overline{P}_0), \tag{15}$$

proving the factorization formula for all ℓ. □

2.2 Convergence of LP

The factorization (13) of D_ℓ from Proposition 2.1 has a trivial corollary, which implies convergence of the LP scheme (and bounds on its error) for a broad range of operators P_ℓ.

Proposition 2.2. *The relative error of the ℓ^{th} level LP approximation on the x_j's is bounded by:*

$$\frac{\|\overline{f}_\ell - \overline{f}\|}{\|\overline{f}\|} \leq \prod_{k=0}^{\ell-1} \|I - \overline{P}_k\|. \tag{16}$$

Here, $\|\cdot\|$ denotes any norm on $\mathbb{R}^p$ when applied to a vector, and the corresponding induced matrix norm when applied to a matrix.

Corollary 2.3. *If for some $0 < \epsilon < 1$ and $L \geq 1$ we have $\|I - \overline{P}_\ell\| \leq \epsilon$ for $\ell > L$, then $\overline{f}_\ell \to \overline{f}$ as $\ell \to \infty$. In fact,*

$$\|\overline{f}_{\ell+1} - \overline{f}\| \leq \epsilon \|\overline{f}_\ell - \overline{f}\|, \tag{17}$$

for all $\ell > L$.

In particular, Corollary 2.3 shows that the $\overline{P}_\ell$ do not need to converge to the identity in order for LP to extend $\overline{f}$. It is enough that $\overline{P}_\ell$ be sufficiently close to I in some norm.

We next show that when the bandwidth σ_ℓ is sufficiently small, the infinity norm of $I - \overline{P}_\ell$ is indeed less than 1, allowing us to invoke Corollary 2.3 to show convergence. We define $\delta = \delta(\mathcal{X})$ to be the minimum Euclidean distance separating any two distinct points in $\mathcal{X}$:

$$\delta = \min_{1 \leq i \neq j \leq n} \|x_i - x_j\|_2. \tag{18}$$

We will assume that the radial kernel $\Phi(r)$ is decreasing as a function of $r \geq 0$, and satisfies the following decay condition:

$$\Phi(r) \leq C r^{-q}, \quad r > 0, \tag{19}$$

for some parameter $q > p$ and constant $C > 0$. This family includes the Gaussian kernels (for any value of q).

We then have the following result:

Proposition 2.4. *Assume $\Phi(0) = 1$, $\Phi(r)$ decreases as a function of $r \geq 0$, and Φ satisfies condition (19). Then for $0 < \epsilon < 1$ there is a constant $c = c(p, \epsilon)$ such that*

$$\|I - \overline{P}_\ell\|_\infty < \epsilon \tag{20}$$

if the bandwidth of P_ℓ satisfies $\sigma_\ell < c\delta$. In particular, if $\sigma_\ell < c\delta$ for all $\ell > L$, then $\overline{f}_\ell$ will converge to $\overline{f}$ as $\ell \to \infty$; in fact $\|\overline{f}_{\ell+1} - \overline{f}\| \leq \epsilon \|\overline{f}_\ell - \overline{f}\|$ for $\ell > L$.

Proof. The infinity norm of a matrix is the largest ℓ_1 norm of its rows. Since $\overline{P}_\ell$ is row-stochastic, this implies

$$\|I - \overline{P}_\ell\|_\infty = 2 \max_{1 \leq i \leq n} (1 - \overline{P}_\ell(x_i, x_i)). \tag{21}$$

This is less than ϵ precisely when

$$\sum_{j \neq i} G_\ell(x_i - x_j) \leq \frac{\epsilon}{2 - \epsilon} \equiv \eta \tag{22}$$

for all $i = 1, \ldots, n$.

Fix a value i. Because the x_j's are all at least δ from each other, the number of points N_r contained in any ball $B(x_i, r)$ cannot exceed $(2r/\delta + 1)^p$. Indeed, since $|B(x_i, r)| = C_p r^p$ and the balls $B(x_j, \delta/2)$ are disjoint, we have

$$N_r C_p (\delta/2)^p \leq C_p (r + \delta/2)^p. \tag{23}$$

Consequently, setting $R_k = B(x_i, 2^{k+1}\delta) \setminus B(x_i, 2^k \delta)$, we have the bound:

$$\begin{aligned} \sum_{j \neq i} G_\ell(x_i - x_j) &= \sum_{k=0}^{\infty} \sum_{x_j \in R_k} G_\ell(x_i - x_j) \leq C \sum_{k=0}^{\infty} (2^{k+2} + 1)^p \Phi(2^k \delta / \sigma_\ell) \\ &\leq C_p \left(\frac{\sigma_\ell}{\delta}\right)^q \sum_{k=0}^{\infty} 2^{kp} 2^{-kq} = C_p \left(\frac{\sigma_\ell}{\delta}\right)^q, \end{aligned} \tag{24}$$

where C_p denotes a constant depending on the dimension p and the kernel Φ. The expression on the right of (24) will be less than η whenever

$$\sigma_\ell \leq (\eta / C_p)^{1/q} \delta, \tag{25}$$

which is the desired result. □

2.3 Stability of LP

In this section, we will show that the LP scheme is stable, in the sense that the extended function f_K can be bounded by the size of the input vectors $\overline{f}$. We will consider the same class of radial kernel $G_\ell(x-y) = \Phi(\|x-y\|/\sigma_\ell)$ considered in Section 2.2, where $\Phi(r)$ is decreasing and satisfies the decay condition (19).

Stability estimates like the ones we will prove have been shown previously for interpolating methods of the form

$$f(x) = \sum_{j=1}^{n} \alpha_j \Phi(\|x - x_j\|), \tag{26}$$

where Φ satisfies a specified decay condition, and the coefficients α_j are found by least squares; see, for example, [13]. We note, however, that the condition imposed in [13] does not apply to as broad a family of kernels as we assume here, specifically Gaussian kernels.

We define δ to be the minimum distance between distinct points in $\mathcal{X}$, as in (18). We first prove a general estimate.

Proposition 2.5. *Suppose LP is performed with the sequence of bandwidths $\sigma_0, \sigma_1, \ldots$. Take $\sigma^* < c\delta$, where $c = c(p, 1/2)$ is the constant from Proposition 2.4, and suppose for some m,*

$$\sigma_j \le \sigma^*, \quad j \ge m. \tag{27}$$

Then for all $\ell \ge 0$ we have the bound

$$\|\overline{f}_\ell\|_\infty \le C 2^m \|\overline{f}\|_\infty \tag{28}$$

where C is a universal constant.

Taking a geometrically-decaying sequence of bandwidths, we immediately obtain the following corollary:

Corollary 2.6. *Suppose $\sigma_\ell = \sigma_0/\mu^\ell$, where $\mu > 1$. Then for all $\ell \ge 0$ we have the bound*

$$\|\overline{f}_\ell\|_\infty \le C_p \left(\frac{\sigma_0}{\delta}\right)^t \|\overline{f}\|_\infty \tag{29}$$

where C_p is a constant depending on the dimension p and the kernel Φ, and where $t = \log_\mu(2)$. The same estimate also holds if $\sigma_\ell = \max\{\sigma_0/\mu^\ell, \sigma^*\}$, where $\sigma^* < c\delta$.

Proof of Proposition 2.5. We write the expansion $f_\ell = \sum_{k=0}^{\ell} P_k D_k \overline{f}$. Since $\sum_{j=1}^{n} P_k(x, x_j) = 1$, it follows that

$$\|f_\ell\|_\infty \le \sum_{k=0}^{\ell} \|D_k \overline{f}\|_\infty = \sum_{k=0}^{m} \|D_k \overline{f}\|_\infty + \sum_{k=m+1}^{\ell} \|D_k \overline{f}\|_\infty. \tag{30}$$

We bound the first term:

$$\sum_{k=0}^{m} \|D_k \overline{f}\|_\infty \le 2^{m+1} \|\overline{f}\|_\infty. \tag{31}$$

Indeed, for any vector $v \in \mathbb{R}^p$, $\|(I - \overline{P}_\ell)v\|_\infty \le 2\|v\|_\infty$. Consequently, $\|D_k \overline{f}\|_\infty \le 2^k \|\overline{f}\|_\infty$, and summing a geometric series we obtain (31).

Next, for any $k \ge 0$, the choice of σ^* and Proposition 2.4 tells us that $\|I - \overline{P}_{m+k}\|_\infty \le 1/2$. Consequently, for any $j > 0$ we have:

$$\|D_{m+j} \overline{f}\|_\infty \le \|D_m \overline{f}\|_\infty \prod_{k=0}^{j-1} \|I - \overline{P}_{m+k}\|_\infty \le \|D_m \overline{f}\|_\infty 2^{-j}. \tag{32}$$

From summing a geometric series we then obtain the bound

$$\sum_{k=m+1}^{\ell} \|D_k \overline{f}\|_\infty \le \|D_m \overline{f}\|_\infty \le 2^m \|\overline{f}\|_\infty. \tag{33}$$

Combining (30), (31) and (33) yields the result. □

2.4 Example: interpolation on the circle

In this section we demonstrate on a numerical example how LP can result in qualitatively different extensions depending on the choice of bandwidth sequence. In particular, kernels with small bandwidths are close to the identity on the sampled points x_j, and so can introduce high-frequency components into the extension not present in the original data, even when they perfectly interpolate the observed values.

To illustrate this phenomenon, we sample $n = 16$ equispaced points $x_k = k/n$ from the circle $S^1 \subset \mathbb{R}^2$ of circumference 1, and evaluate the function $f(x) = \cos(10\pi x)$. In this case, there is enough information from the samples to perfectly interpolate f on the entire circle. We plot the function f in the left panel of Figure 1, along with the sampled values.

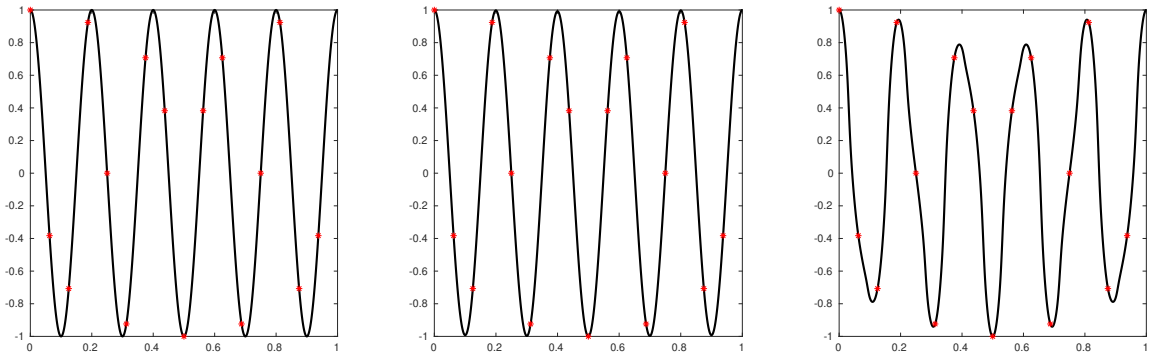

Figure 1: Left: The function $f(x) = \cos(10\pi x)$. Middle: The LP reconstruction with bandwidths $\sigma_\ell = \max\{2^{-\ell+1}, 1/2\}$. Right: The LP extension with bandwidths $\sigma_\ell = 2^{-\ell+1}$. On all figures the sampled points are highlighted in red.

In the right panel of Figure 1, we plot the LP extension of f using the geometrically-decreasing sequence of bandwidths $\sigma_\ell = 2^{-\ell+1}$, $\ell \geq 0$. In the middle panel of Figure 1, we plot the LP extension of f using the sequence of bandwidths $\sigma_\ell = \max\{2^{-\ell+1}, 1/2\}$, $\ell \geq 0$; in other words, the bandwidths plateau after the third scale. The relative errors are, respectively, 2.14×10^{-1} and 6.14×10^{-3}. The geometrically-decreasing sequence requires only 6 levels until convergence to machine precision (approximately 10^{-14} in this case) on the sampled values, whereas the plateaued sequence requires 136 levels until convergence.

The reason for the higher error in the first scheme is that kernels with smaller bandwidth put more weight on the higher frequencies. In other words, these kernels introduce greater aliasing into the reconstruction. By choosing the plateaued sequence of bandwidths, we are able to mitigate the aliasing, at the expense of introducing more levels into the reconstruction.

2.5 Example: extrapolation from an interval

In this example, we illustrate the stability estimate from Proposition 2.5 on an example. We take $n = 16$ equispaced points on the interval $[0, 1]$, and assign them alternating values ± 1, so that $y_0 = 1$, $y_1 = -1$, and so forth. We apply the LP extension procedure for geometrically decreasing bandwidths, $\sigma_\ell = \sigma_0/\mu^\ell$. We plot an example of the extrapolated function, for $\mu = 2$ and $\sigma_0 = 1$, in Figure 2.

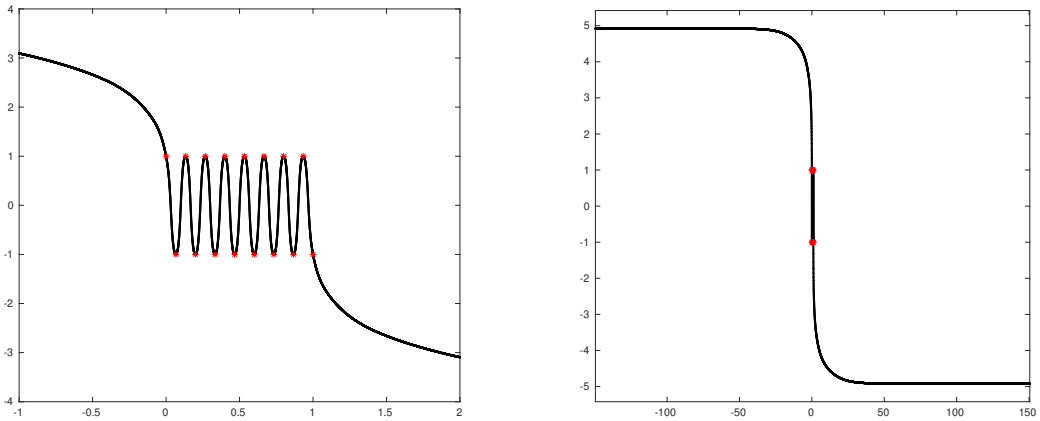

Figure 2: The extrapolated function, displayed at different scales. The observed values are highlighted in red.

We are interested in exploring the size of the extrapolation as functions of the parameters σ_0 and μ. Proposition 2.5 predicts that larger values of σ_0 and smaller values of μ will result in larger extrapolated values. In the left panel of Figure 3, we plot the maximum value (to within precision 10^{-7}) of the extrapolated function as a function of σ_0. Indeed, we see that as σ_0 grows, the infinity norm of the extrapolation increases.

Similarly, in the right panel of Figure 3 we plot the maximum value (to within precision 10^{-7}) of the extrapolated function as a function of the decay rate μ. The infinity norm of the extrapolation increases with decreasing μ. Again, this is the qualitative behavior expected from Proposition 2.5.

3 Laplacian pyramids and denoising

In this section, we consider the problem of denoising the in-sample observations, rather than extending the function to new values. In [15], it is proposed that when the observed data is noisy, the LP algorithm should be truncated before

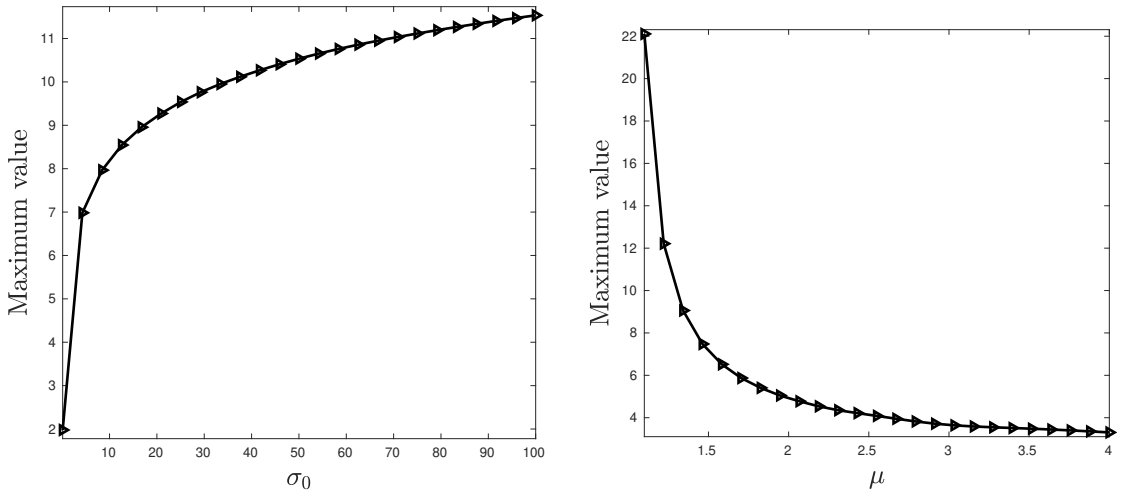

Figure 3: Left: The maximum value of the extrapolation as a function of σ_0. Right: The maximum value of the extrapolation as a function of μ.

convergence to avoid overfitting; a method that approximates cross-validation is used to determine the stopping level. If K levels are used, then from Proposition 2.1, the denoised vector is $\overline{A}_K y$, where y is the observed vector and

$$\overline{A}_K = I - (I - \overline{P}_K) \cdots (I - \overline{P}_0). \tag{34}$$

If each $\overline{P}_\ell$ is row-stochastic, then so too is the denoising kernel $\overline{A}_K$.

In this special case where $\overline{P}_K = \cdots = \overline{P}_0 = Q$, the denoising kernel $\overline{A}_K$ takes on a particularly simple form. Changing notation to $Q_K \equiv \overline{A}_K$, we have

$$Q_K = I - (I - Q)^K. \tag{35}$$

(Note that $Q = Q_1$.) As we noted in Section 1.2, Q_K is the same kernel used when applying L_2 boosting to kernel regression, as described in [16]. In this section, we will consider using the kernels Q_K in the context of non-local means denoising [4, 3]. We will first review the basic non-local means algorithm, and then compare the use of the iterated kernels Q_K within the non-local means framework.

3.1 Non-local means

Given a signal $s \in \mathbb{R}^M$, we suppose that we observe s in the presence of noise:

$$y = s + \varepsilon \tag{36}$$

where the entries of ε are noise, e.g. shot noise or Gaussian. NL means (in its simplest incarnation) performs the following procedure to remove the noise ε. First, patches of adjacent samples (or pixels, in the case of an image) are extracted from the long signal; we call these vectors $x_1, \ldots, x_n$. We will suppose each $x_i \in \mathbb{R}^m$, where $m \ll M$.

Second, an affinity between the patches x_i is defined. For concreteness, we will use the common choice of a Gaussian kernel to specify the affinity, writing

$$G(x_i, x_j) = \exp\{-\|x_i - x_j\|^2/\sigma^2\} \tag{37}$$

where $\sigma > 0$ is a specified parameter.

Third, the affinities $G(x_i, x_j)$ are normalized to form the row-stochastic matrix Q:

$$Q(x_i, x_j) = \frac{G(x_i, x_j)}{\sum_{j'} G(x_i, x_{j'})}. \tag{38}$$

With this Markov kernel now defined, one iteration of NL means is performed by taking

$$s_{NL}^{(1)} = Qy. \tag{39}$$

In words, each entry of y is replaced by a weighted average of the other entries, where the weights are determined by local patches.

Of course, this process can be iterated multiple times by repeated application of Q. In this way, we obtain a sequence of denoised images:

$$s_{NL}^{(\ell)} = Q^\ell y. \tag{40}$$

A physical interpretation of this algorithm is provided in [23] . $s_{NL}^{(\ell)}[i]$ is equal to the expected value of a random process that takes ℓ steps along the patches x_j starting at patch x_i, with transition probabilities specified by Q, where the value of the process at patch x_j is y_j.

3.2 The choice of kernel

The transition probabilities along the patches x_j can be specified by any Markov matrix, not just the local diffusion matrix Q. In particular, [23] proposes the alternative matrix

$$Q_2 = 2Q - Q^2 = I - (I - Q)^2. \tag{41}$$

As we have seen, the kernel Q_2 is equal to a two-step truncated LP kernel, or equivalently a two-step L_2 boosting kernel [7, 16]. As has been observed previously in [7, 17, 16], Q_2 is also equal to the "twicing" kernel introduced by Tukey [25]. It is illustrated in [23] on several examples that iteratively applying Q_2 may achieve better denoising than iteratively applying the original kernel $Q_1 \equiv Q$.

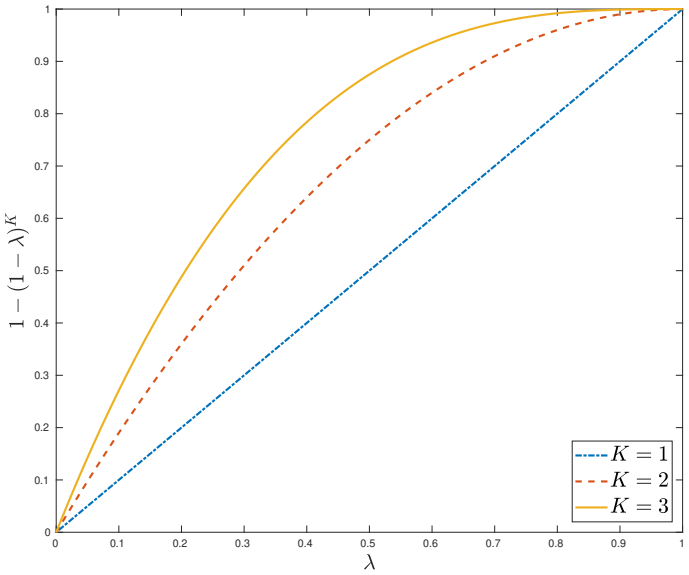

Figure 4: The eigenvalues of Q_K as functions of the eigenvalues of Q.

Of course, one may also consider running NL means by iteratively applying the higher-step LP kernels $Q_K = I-(I-Q)^K$ as well. Because Q is diagonalizable with eigenvalues contained between 0 and 1, the truncated LP kernels Q_K are also row stochastic, with eigenvalues

$$1 - (1 - \lambda)^K, \quad \lambda \in \mathrm{spec}(Q). \tag{42}$$

In Figure 4, we plot the functions $1-(1-\lambda)^K$ for several values of K. Larger values of K result in kernels Q_K closer to the identity I. Consequently, the iterations of NL means will converge more slowly to 0, allowing a more refined denoising procedure.

3.3 Example: step function

We illustrate the behavior of NL means with different kernels for denoising a 1D signal. The signal s, which was considered in [23], has length $M = 100$, which assumes two values, -1 and $+1$, and is observed with additive Gaussian noise of standard deviation 0.5. The signal is plotted in the left side of Figure 5, and the signal with noise is plotted in the right side.

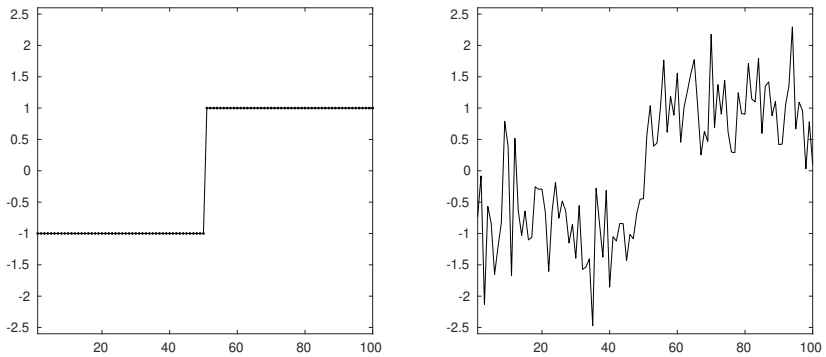

Figure 5: Left: The clean step function. Right: The step function with Gaussian noise with standard deviation 0.5.

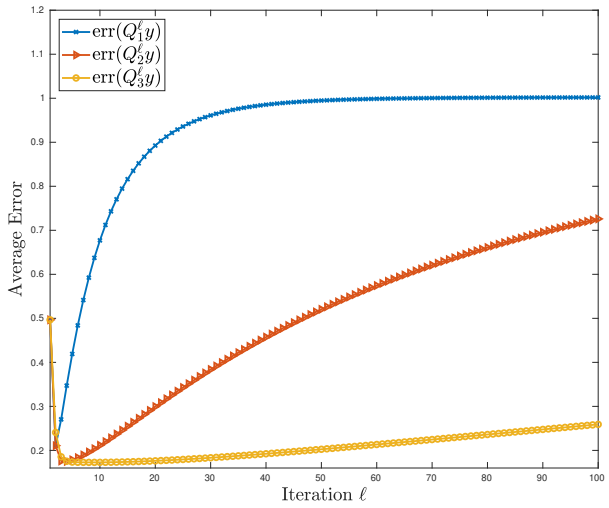

Figure 6: The average errors when denoising by kernels Q_1^ℓ, Q_2^ℓ, and Q_3^ℓ, as a function of the iteration ℓ. Errors are averaged over 500 runs.

We build the NL means kernel using subintervals of size $m = 3$. The Gaussian kernel matrix for the step function is built using the median squared distance between all pairs of points divided by 3. This scaling value is somewhat arbitrary, and was manually chosen to ensure that the graph defined by Q is not too connected.

In Figure 6, we plot the errors of NL means as a function of the number of iterations, for kernels Q_K with parameters $K = 1, 2, 3$. That is, we plot:

$$\text{err}(Q_K^\ell y) = \frac{\|Q_K^\ell y - s\|_2}{\|s\|_2}, \tag{43}$$

as a function of the iteration number ℓ, where s is the step function. For comparison, we also plot the average errors $\text{err}(Q_\ell y) = \|Q_\ell y - s\|_2/\|s\|_2$ of applying the non-iterated LP kernels Q_ℓ, as a function of the level ℓ; these are the iterates we obtain by boosting. We emphasize that the scheme we propose uses a fixed value of K, and iteratively applies Q_K; the resulting denoising kernel is then Q_K^ℓ, where ℓ is the number of iterates. The curves displayed are averaged over 500 runs of the experiment, where each experiment is run with a different realization of the noise. The minimal errors of Q_2^ℓ and Q_3^ℓ (over ℓ) are both smaller than the minimal error for Q_1^ℓ. The average minimal error for Q_1^ℓ is 0.210, while they are 0.171 and 0.168 for Q_2^ℓ and Q_3^ℓ, respectively.

For any choice of Markov kernel, the iterations of NL means will both average out the noise and the signal. While the effect of the noise will be reduced, it will also result in smoothing of the signal by shrinking all the values towards the mean. In other words, increasing the iterations will increase the bias and decrease the variance. In general, given only the noisy signal y, it may be difficult to estimate the optimal number of iterations that minimizes the overall error.

In light of these considerations, while the minimal errors achieved by Q_2^ℓ and Q_3^ℓ are nearly identical, more interesting is that, because it takes longer for the spectrum of Q_3^ℓ to decay, there is a much larger range of iterations for which it does not yet oversmooth the signal, and hence where the error is smaller than the error for Q_2^ℓ. In this sense, the sequence Q_3^ℓ is less sensitive to the number of iterations ℓ chosen by the user, and hence more rubust to the specification of this parameter.

4 Conclusion

We have proven several properties of the Laplacian pyramids extension algorithm. Based on the factorization formula from Proposition 2.1, we showed that the method always converges to an interpolator of the observed data if the kernel bandwidths drop below a certain threshold. We also proved a stability estimate for the extension, which exhibits similar qualitative behavior as prior estimates from [13] for classical kernel interpolation methods.

We also considered iterating the truncated LP kernels to denoise signals by non-local means. A scheme of this kind for a two-step kernel was proposed in [23]. Here, we have shown on numerical examples that using higher-step kernels may be advantageous, as they are less sensitive to the number of iterations chosen by the user, and may also achieve lower error overall with an optimal number of iterations. In future work, we plan to further explore the properties and behavior of these denoising kernels.

Acknowledgements

I acknowledge support from the NSF BIGDATA program, IIS 1837992.

References

[1] Yariv Aizenbud, Amit Bermanis, and Amir Averbuch. PCA-based out-of-sample extension for dimensionality reduction. *arXiv preprint arXiv:1511.00831*, 2015.

[2] Romeo Alexander, Zhizhen Zhao, Eniko Székely, and Dimitrios Giannakis. Kernel analog forecasting of tropical intraseasonal oscillations. *Journal of the Atmospheric Sciences*, 74(4):1321–1342, 2017.

[3] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A non-local algorithm for image denoising. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 60–65. IEEE, 2005.

[4] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005.

[5] Peter Bühlmann. Boosting for high-dimensional linear models. *The Annals of Statistics*, 34(2):559–583, 2006.

[6] Peter Bühlmann and Torsten Hothorn. Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22(4):477–505, 2007.

[7] Peter Bühlmann and Bin Yu. Boosting With the L_2 Loss: Regression and Classification. *Journal of the American Statistical Association*, 98(462):324–339, 2003.

[8] Peter Bühlmann and Bin Yu. Sparse boosting. *Journal of Machine Learning Research*, 7:1001–1024, 2006.

[9] Peter Bühlmann and Bin Yu. Boosting. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2:69–74, 2010.

[10] Peter Burt and Edward Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31(4):532–540, 1983.

[11] Eliodoro Chiavazzo, Charles Gear, Carmeline Dsilva, Neta Rabin, and Ioannis Kevrekidis. Reduced models in chemical kinetics via nonlinear data-mining. *Processes*, 2(1):112–140, 2014.

[12] Darin Comeau, Dimitrios Giannakis, Zhizhen Zhao, and Andrew J. Majda. Predicting regional and pan-arctic sea ice anomalies with kernel analog forecasting. *Climate Dynamics*, 52(9-10):5507–5525, 2019.

[13] Stefano De Marchi and Robert Schaback. Stability of kernel-based interpolation. *Advances in Computational Mathematics*, 32(2):155–161, 2010.

[14] Carmeline J. Dsilva, Ronen Talmon, Neta Rabin, Ronald R. Coifman, and Ioannis G. Kevrekidis. Nonlinear intrinsic variables and state reconstruction in multiscale simulations. *The Journal of Chemical Physics*, 139(18):184109–1–13, 2013.

[15] Angela Fernández, Neta Rabin, Dalia Fishelov, and José R. Dorronsoro. Auto-adaptative Laplacian pyramids for high-dimensional data analysis. *arXiv preprint arXiv:1311.6594*, 2013.

[16] Marco Di Marzio and Charles C. Taylor. On boosting kernel regression. *Journal of Statistical Planning and Inference*, 138(8):2483–2498, 2008.

[17] Peyman Milanfar. A tour of modern image filtering. *IEEE Signal Processing Magazine*, January 2011.

[18] Gal Mishne and Israel Cohen. Multiscale anomaly detection using diffusion maps and saliency score. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2823–2827. IEEE, 2014.

[19] E. A. Nadaraya. On estimating regression. *Theory of Probability & Its Applications*, 9(1):141–142, 1963.

[20] Neta Rabin and Ronald R. Coifman. Heterogeneous datasets representation and learning using diffusion maps and Laplacian pyramids. In *Proceedings of the 2012 SIAM International Conference on Data Mining*, pages 189–199. Society for Industrial and Applied Mathematics, 2012.

[21] Neta Rabin and Dalia Fishelov. Multi-scale kernels for Nyström based extension schemes. *Applied Mathematics and Computation*, 319:165–177, 2018.

[22] Neta Rabin and Dalia Fishelov. Two directional laplacian pyramids with application to data imputation. *Advances in Computational Mathematics*, pages 1–24, 2019.

[23] Amit Singer, Yoel Shkolnisky, and Boaz Nadler. Diffusion interpretation of nonlocal neighborhood filters for signal denoising. *SIAM Journal on Imaging Sciences*, 2(1):118–139, 2009.

[24] Nurit Spingarn, Saman Mousazadeh, and Israel Cohen. Voice activity detection in transient noise environment using Laplacian pyramid algorithm. In *14th International Workshop on Acoustic Signal Enhancement*, pages 238–242. IEEE, 2014.

[25] John W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, 1977.

[26] Geoffrey S. Watson. Smooth regression analysis. *Sankhya: The Indian Journal of Statistics, Series A*, 26(4):359–372, 1964.