Multivariate Trend Filtering for Lattice Data

Veeranjaneyulu Sadhanala[a] Yu-Xiang Wang[b] Addison J. Hu[c] Ryan J. Tibshirani[c]

[a]Google [b]University of California Santa Barbara [c]Carnegie Mellon University

Abstract

We study a multivariate version of trend filtering, called Kronecker trend filtering or KTF, for the case in which the design points form a lattice in d dimensions. KTF is a natural extension of univariate trend filtering (Steidl et al., 2006; Kim et al., 2009; Tibshirani, 2014), and is defined by minimizing a penalized least squares problem whose penalty term sums the absolute (higher-order) differences of the parameter to be estimated along each of the coordinate directions. The corresponding penalty operator can be written in terms of Kronecker products of univariate trend filtering penalty operators, hence the name Kronecker trend filtering. Equivalently, one can view KTF in terms of an ℓ_1-penalized basis regression problem where the basis functions are tensor products of falling factorial functions, a piecewise polynomial (discrete spline) basis that underlies univariate trend filtering.

This paper is a unification and extension of the results in Sadhanala et al. (2016, 2017). We develop a complete set of theoretical results that describe the behavior of k^{th} order Kronecker trend filtering in d dimensions, for every $k \geq 0$ and $d \geq 1$. This reveals a number of interesting phenomena, including the dominance of KTF over linear smoothers in estimating heterogeneously smooth functions, and a phase transition at $d = 2(k+1)$, a boundary past which (on the high dimension-to-smoothness side) linear smoothers fail to be consistent entirely. We also leverage recent results on discrete splines from Tibshirani (2020), in particular, discrete spline interpolation results that enable us to extend the KTF estimate to any off-lattice location in constant-time (independent of the size of the lattice n).

1 Introduction

We consider a standard nonparametric regression model, relating real-valued responses $y_i \in \mathbb{R}$, $i = 1, \dots, n$ to design points $x_i \in \mathcal{X} \subseteq \mathbb{R}^d$, $i = 1, \dots, n$,

$$y_i = f_0(x_i) + \epsilon_i, \quad i = 1, \dots, n, \tag{1}$$

where $f_0 : \mathcal{X} \to \mathbb{R}$ is the (unknown) regression function to be estimated, and ϵ_i, $i = 1, \dots, n$ are mean zero stochastic errors. In this paper, we will focus on functions f_0 that display heterogeneous smoothness across the domain $\mathcal{X}$, in a sense we will make precise later. We will also focus on the case in which the design points form a d-dimensional lattice: that is, we assume $n = N^d$, and

$$\{x_1, \dots, x_n\} = \{1/N, 2/N, \dots, 1\}^d := Z_{n,d}. \tag{2}$$

The lattice structure is important, but the assumption about uniform spacing over the unit cube $[0,1]^d$ is used only for simplicity. Essentially all of our results (both methods and theory) translate over to the case of a more general lattice structure, a Cartesian product $\{z_{i1}\}_{i=1}^{N_1} \times \{z_{i2}\}_{i=1}^{N_2} \times \cdots \times \{z_{id}\}_{i=1}^{N_d}$, where $n = \prod_{j=1}^d N_j$, and the sets in this product are otherwise arbitrary. We return to this point in Section 9.1.

This paper is a unification and extension of Sadhanala et al. (2016, 2017) (more will be said about the relationship to these papers in Section 1.3). The models of smoothness for f_0 that we will study are based on *total variation* (TV). For a univariate function $g : [a,b] \to \mathbb{R}$, recall that its total variation is defined as

$$\mathrm{TV}(g; [a,b]) = \sup_{a < z_1 < \cdots < z_{m+1} < b} \sum_{i=1}^m |g(z_i) - g(z_{i+1})|.$$

For a multivariate function $f : [0,1]^d \to \mathbb{R}$, we will consider notions of smoothness that revolve around the following *discrete* version of multivariate total variation:

$$\mathrm{TV}(f; Z_{n,d}) = \sum_{j=1}^d \sum_{\substack{x, z \in Z_{n,d} \\ z = x + e_j/N}} |f(x) - f(z)|.$$

Here we use e_j to denote the j^{th} standard basis vector in $\mathbb{R}^d$, and hence the inner sum is taken over all pairs of lattice points $x, z \in Z_{n,d}$ that differ in the j^{th} coordinate by $1/N$ (and match in all other coordinates). We will also consider higher-order versions of discrete multivariate TV, which are based on higher-order differences in the summands in the above display. We will connect our discrete notions of TV smoothness to standard continuum notions of total variation in Section 3.

Broadly speaking, there are many multivariate nonparametric regression methods available. Many of the methods in common use are *linear smoothers*: estimators of the form $\hat{f}(x) = w(x)^\mathsf{T} y$, for a suitable weight function $w : \mathcal{X} \to \mathbb{R}^n$ (which can depend on the design $x_1, \dots, x_n$), where we use $y = (y_1, \dots, y_n) \in \mathbb{R}^n$ for the response vector. Examples include kernel smoothing, thin-plate splines, and reproducing kernel Hilbert space estimators. A critical shortcoming of linear smoothers is that they cannot be *locally adaptive*—they cannot adapt to different local levels of smoothness exhibited by f_0 over $\mathcal{X}$. This is a phenomenon that has been well-documented in various settings; see Section 1.3.

The limitations of linear smoothers—and the need for nonlinear adaptive methods—is a major theme in this paper. The central method that drives this story is a multivariate extension of trend filtering, which is indeed nonlinear and locally adaptive, a claim that will be supported by experiments and theory in the coming sections. We must note at the outset that all of the developments in this paper hinge on the assumption of lattice data (meaning, a lattice structure for the design points). A multivariate extension of trend filtering for scattered data would require a completely different approach (unlike, say, kernel smoothing or reproducing kernel Hilbert space methods, which apply regardless of the structure of the design points). The intersection of multivariate nonparametric regression methods and locally adaptive methods is actually quite small, especially when we further intersect this with the set of simple methods that are easy to use in practice, are well-understood theoretically. For this reason, we see the contributions of the current paper, though limited to lattice data, as being worthwhile. The development of new multivariate trend filtering methods for scattered data is important, and an extension is discussed in Section 9.5, but a comprehensive study is left to future work.

1.1 Review: trend filtering

Before describing the main proposal, we review *trend filtering*, a relatively recent method for univariate nonparametric regression, independently proposed by Steidl et al. (2006); Kim et al. (2009). For a univariate design, equally-spaced (say) on the unit interval, $x_i = i/n$, $i = 1, \dots, n$, and an integer $k \geq 0$, the k^{th} order trend filtering estimate is defined by the solution of the optimization problem:

$$\underset{\theta \in \mathbb{R}^n}{\text{minimize}} \ \frac{1}{2}\|y - \theta\|_2^2 + \lambda \|D_n^{(k+1)} \theta\|_1. \tag{3}$$

Here $\lambda \geq 0$ denotes a tuning parameter, $y = (y_1, \dots, y_n) \in \mathbb{R}^n$ is the response vector, and $D_n^{(k+1)} \in \mathbb{R}^{(n-k-1) \times n}$ is the difference operator of order $k+1$, which we will also loosely call discrete derivative operator of order $k+1$. This can be defined recursively in the following manner:

$$\begin{aligned} D_n^{(1)} &= \begin{bmatrix} -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & \dots & 0 & 0 \\ \vdots & & & & & \\ 0 & 0 & 0 & \dots & -1 & 1 \end{bmatrix}, \\ D_n^{(k+1)} &= D_{n-k}^{(1)} D_n^{(k)}, \quad k = 1, 2, 3, \dots. \end{aligned} \tag{4}$$

The intuition behind problem (3) is as follows. The penalty term, which penalizes the discrete derivatives of θ of order $k+1$, can be equivalently seen as penalizing the *differences* in k^{th} discrete derivatives of θ at adjacent design points (due to (4)). By the sparsity-inducing property of the ℓ_1 norm, the k^{th} discrete derivatives of the trend filtering solution $\hat{\theta}$ will be exactly equal at a subset of adjacent design points, and $\hat{\theta}$ will therefore exhibit the structure of a k^{th} degree piecewise polynomial, with adaptively-chosen knots.

Here is a summary of the properties of trend filtering, as a nonparametric regression tool.[1]

- The discrete trend filtering estimate, which is defined over the design points, can be "naturally" extended to a k^{th} degree piecewise polynomial function, in fact, a k^{th} degree discrete spline, on $[0, 1]$.

[1] We have defined it in this subsection for an evenly-spaced design, for simplicity, but trend filtering can still be defined for an arbitrary design, and all of the following properties still hold; see Section 9.1.

- Trend filtering is computationally efficient (several fast algorithms exist for the structured, convex problem (3)), and is not much slower to compute than (say) the smoothing spline.
- Trend filtering is more locally adaptive than the smoothing spline (or any linear smoother). This not only carries theoretical backing (next point), but is clearly noticeable in practice as well.
- Trend filtering attains the minimax rate (in squared empirical norm) of $n^{-(2k+2)/(2k+3)}$ for estimating a function f_0 whose k^{th} weak derivative has bounded total variation. The minimax linear rate (the best worst-case risk that can be attained by a linear smoother) over this class is $n^{-(2k+1)/(2k+2)}$.

Support for the above facts can be found in Tibshirani (2014), and the discrete spline (numerical analytic) perspective behind trend filtering is further developed in in Tibshirani (2020). More will be said about all of these properties in the coming sections, as analogous properties will be developed for a multivariate extension of trend filtering.

To prepare for this multivariate extension, it helps to recast the discrete problem (3) in just a slightly different form. First, some notation: for a vector $\theta \in \mathbb{R}^n$, we will (when convenient) index it by the underlying design points, and write its components as $\theta(x_i)$, $i = 1, \dots, n$ in place of θ_i, $i = 1, \dots, n$. Next, we define a difference operator, which we will again loosely refer to as a discrete derivative operator, by

$$(\Delta\theta)(x_i) = \begin{cases} \theta(x_{i+1}) - \theta(x_i) & \text{if } i \leq n-1, \\ 0 & \text{else.} \end{cases}$$

Naturally, we can view $\Delta\theta$ as a vector in $\mathbb{R}^n$ with components $(\Delta\theta)(x_i)$, $i = 1, \dots, n$. Higher-order discrete derivatives are obtained by repeated application of the same formula; we abbreviate $(\Delta^2\theta)(x_i) = (\Delta(\Delta\theta))(x_i)$, and so on. In this new notation, we can now rewrite problem (3) as

$$\underset{\theta \in \mathbb{R}^n}{\text{minimize}} \ \frac{1}{2} \sum_{i=1}^n \big(y_i - \theta(x_i)\big)^2 + \lambda \sum_{i=1}^n |(\Delta^{k+1}\theta)(x_i)|. \tag{5}$$

1.2 Kronecker trend filtering

Let us return to the setting of multivariate design points on a lattice $Z_{n,d}$ as in (2) (where recall we denote $N = n^{1/d}$, assumed to be integral). The multivariate setting brings a number challenges (methodological, theoretical, computational, and so on). To begin, at each multivariate $x \in Z_{n,d}$, there are many (partial) derivatives one could consider regularizing, and one must decide which ones to penalize and how to compute them discretely. Not all constructions will be equally tractable, and it is unclear which of these—if any—will enable us to carry forward the key characteristics of univariate trend filtering described above, such as the local piecewise polynomial structure (discrete spline with adaptively chosen knots), local adaptivity, computational efficiency, and minimax rate optimality over TV classes. Our approach will be to penalize all axis-aligned derivatives, possible because of the lattice structure of the design points. As we will see, this leads to suitable generalizations of many of the properties in the univariate case.

Building from the univariate notation and definitions at the end of the last subsection, for a vector $\theta \in \mathbb{R}^n$, we will (when convenient) index its components by their lattice positions, denoted $\theta(x)$, $x \in Z_{n,d}$. For each $j = 1, \dots, d$, we define the discrete derivative of θ in the j^{th} coordinate direction at a location x by

$$(\Delta_{x_j}\theta)(x) = \begin{cases} \theta(x + e_j/N) - \theta(x) & \text{if } x, x + e_j/N \in Z_{n,d}, \\ 0 & \text{else.} \end{cases}$$

We write $\Delta_{x_j}\theta \in \mathbb{R}^n$ for the vector with components $(\Delta_{x_j}\theta)(x)$, $x \in Z_{n,d}$. As before, higher-order discrete derivatives are simply defined by repeated application of the above definition; we use abbreviations $(\Delta_{x_j^2}\theta)(x) = (\Delta_{x_j}(\Delta_{x_j}\theta))(x)$, $(\Delta_{x_j,x_\ell}\theta)(x) = (\Delta_{x_j}(\Delta_{x_\ell}\theta))(x)$, and so on.

With this notation in place, we define a multivariate version of trend filtering, that we call *Kronecker trend filtering* (KTF). Given an integer $k \geq 0$, the k^{th} order KTF estimate is defined by the solution of the optimization problem:

$$\underset{\theta \in \mathbb{R}^n}{\text{minimize}} \ \frac{1}{2} \sum_{i=1}^n \big(y_i - \theta(x_i)\big)^2 + \lambda \sum_{j=1}^d \sum_{x \in Z_{n,d}} |(\Delta_{x_j^{k+1}}\theta)(x)|. \tag{6}$$

Note the close analogy between (5) and (6): the latter extends the former by adding up absolute discrete derivatives of θ of order $k+1$ along each one of the d coordinate directions. A similar intuition carries over from the univariate case, regarding the role of the penalty in (6), and the structure of the solution. As we can see, the KTF problem penalizes the differences in k^{th} discrete derivatives of θ at lattice positions x and z, for all x and z that are adjacent along any one of the d coordinate directions. By the sparsifying nature of the ℓ_1 norm, the KTF solution $\hat{\theta}$ will have equal k^{th} discrete derivatives between neighboring points on the lattice (and more so for larger λ, generally speaking). Hence, along any line segment parallel to one of the coordinate axes, the KTF solution $\hat{\theta}$ will have the structure of a k^{th} degree piecewise polynomial, with adaptively-chosen knots. This intuition will be made rigorous in Section 2.3.

We can also rewrite the KTF problem (6) in a more compact form, so that it resembles (3):

$$\underset{\theta \in \mathbb{R}^n}{\text{minimize}} \ \frac{1}{2}\|y-\theta\|_2^2 + \lambda \|D_{n,d}^{(k+1)}\theta\|_1, \tag{7}$$

where we define

$$D_{n,d}^{(k+1)} = \begin{bmatrix} D_N^{(k+1)} \otimes I_N \otimes \cdots \otimes I_N \\ I_N \otimes D_N^{(k+1)} \otimes \cdots \otimes I_N \\ \vdots \\ I_N \otimes I_N \otimes \cdots \otimes D_N^{(k+1)} \end{bmatrix}. \tag{8}$$

Here $D_N^{(k+1)} \in \mathbb{R}^{(N-k-1)\times N}$ is the discrete derivative matrix from (4) (as would be used in k^{th} order univariate trend filtering on N points); $I_N \in \mathbb{R}^{N\times N}$ denotes the identity matrix; and $A \otimes B$ denotes the Kronecker product of matrices A, B. Each block of rows in (8) is made up of a total of $d-1$ Kronecker products (a total of d matrices). The Kronecker product structure behind the penalty matrix in (8) is what inspires the name Kronecker trend filtering. In a similar vein, we will also refer to $\|D_{n,d}^{(k+1)}\theta\|_1$ as the k^{th} order *Kronecker total variation* (KTV) of θ. Note that when $k=0$, the KTF problem (7) reduces to anisotropic TV denoising on the d-dimensional lattice, and when $d=1$, it reduces to k^{th} order univariate trend filtering (3).

Before we delve any deeper into its properties, which will start in Section 2, we give a simple example of KTF in Figure 1. The example portrays an underlying function f_0 in $d=2$ dimensions that has two peaks in opposite corners of the unit square $[0,1]^2$, one larger and one smaller, and is otherwise very smooth. Based on noisy observations of f_0 over a 2d lattice, the estimates from KTF of orders $k=0,1,2$ are able to capture this behavior. Meanwhile, estimates from kernel smoothing are not, and they either oversmooth the larger peak or undersmooth the valleys, depending on the choice of the bandwidth. Advocates of kernel smoothing might say that this problem can be solved by giving the kernel a locally varying bandwidth (that is, modeling the bandwidth itself as a function of $x \in [0,1]^d$). While this can work in principle, for example, using Lepski's method (see Section 1.3), it can often be hard to implement in practice. More to the point: the comparison in Figure 1 is not meant to portray the kernel smoothing framework as being wholly untenable; rather, it is just meant to portray the differences in adaptivity of KTF versus kernel smoothing when each method is allowed only a single tuning parameter.

1.3 Related work

Our paper builds from a line of work on trend filtering, which, recall, enforces smoothness by penalizing the ℓ_1 norm of discrete derivatives of a given order (Steidl et al., 2006; Kim et al., 2009; Tibshirani, 2014; Wang et al., 2014; Ramdas and Tibshirani, 2016). This can be seen as a discrete analog of the *locally adaptive regression spline* estimator, which penalizes the TV of the k^{th} derivative of a function. This estimator was formally studied (and named) by Mammen and van de Geer (1997), though the same core idea—regularization via the TV of derivatives—can also be found in Koenker et al. (1994), and even earlier in Nemirovski et al. (1984, 1985). Notably, the latter paper derives rates of convergence and proves that linear smoothers cannot be minimax rate optimal over TV classes, a result that was later generalized to Besov spaces by Donoho and Johnstone (1998).

For an overview of trend filtering, its connection to classical theory on splines and divided differences, and related results that bridge discrete and continuum representations (such as that connecting trend filtering and locally adaptive regression splines), we refer to Tibshirani (2020). Trend filtering was extended to general graphs by Wang et al. (2016) (more on this below), and was analyzed in an additive model setting by Sadhanala and Tibshirani (2019). The Kronecker trend filtering estimator was proposed by Sadhanala et al. (2017). Rates of convergence for KTF have been developed in different special cases, distributed among several papers. Minimax results for TV denoising on lattices can be found in Hutter and Rigollet (2016) (upper bounds) and Sadhanala et al. (2016) (lower bounds); with respect to KTF and KTV

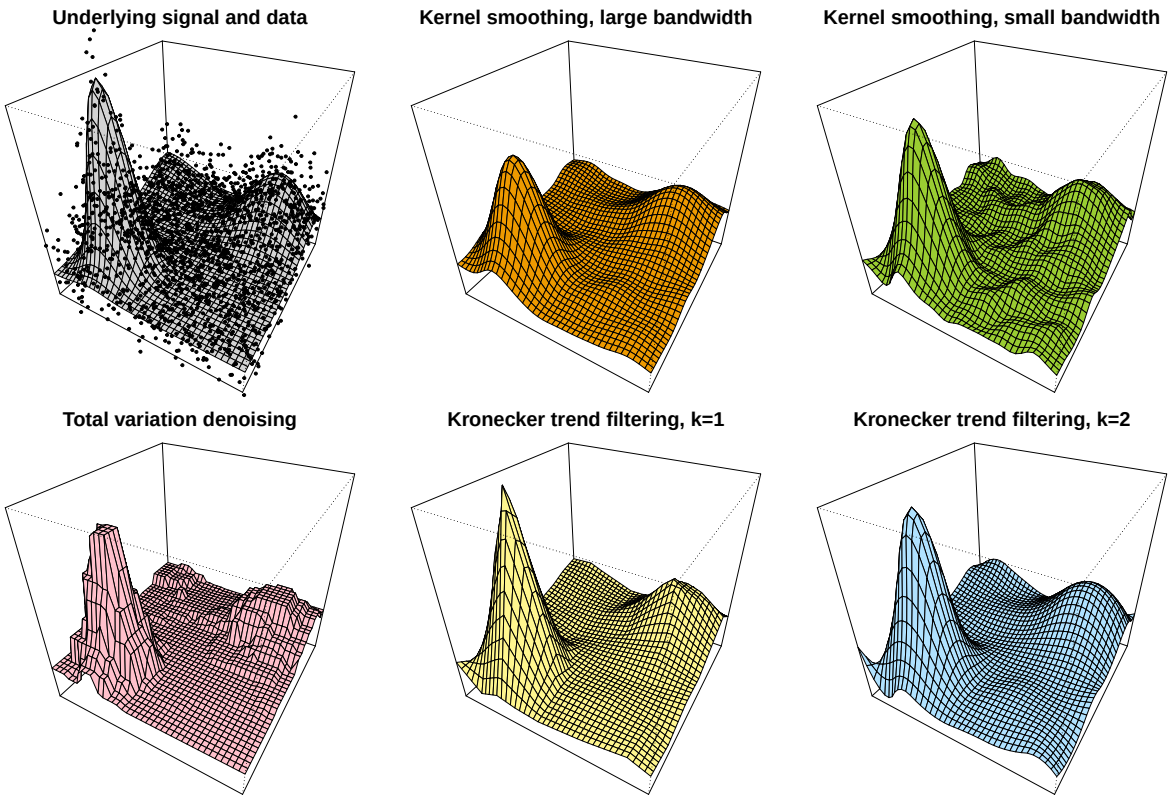

Figure 1: *Top left: underlying regression function f_0 evaluated over a square lattice with $n = 40^2 = 1600$ points, and associated responses (formed by adding noise) shown as black points. Top middle and top right: kernel smoothing (with a spherical Gaussian kernel) fit to this data using large and small bandwidth values, respectively. Bottom left, middle, and right: Kronecker trend filtering estimates of orders $k = 0, 1, 2$, respectively (recall, KTF with $k = 0$ reduces to anisotropic total variation denoising). We see that, in order to capture the larger of the two peaks in f_0, kernel smoothing must significantly undersmooth the other peak (and surrounding areas); instead, with more regularization, it undersmooths throughout. The KTF estimates are able to adapt to heterogeneity in the smoothness of f_0. Also, each exhibits a distinct structure based on the polynomial order k.*

classes, this covers the case of $k = 0$ and all dimensions d. Meanwhile, Sadhanala et al. (2017) gave minimax results (matching upper and lower bounds) for all k and $d = 2$. The current paper completes the landscape, deriving minimax theory for all smoothness orders k and all dimensions d. The majority of this paper (including the minimax theory) was completed in 2018 and can be found in the Ph.D. thesis of the first author (Sadhanala, 2019).

Graph-based TV methods. Our paper is complementary to the line of research on locally adaptive nonparametric estimation over graphs, such as Gavish et al. (2010); Sharpnack et al. (2013); Wang et al. (2016); Göbel et al. (2018); Padilla et al. (2018, 2020); Ye and Padilla (2021). The lattice structure that we consider in this paper can be cast as a particular d-dimensional grid graph with n nodes (that is, with all side lengths equal to $N = n^{1/d}$). However, many of the methods proposed in the aforementioned references seek to be far more general, and operate over arbitrary graph structures. This generality comes with several challenges, from conceptual to theoretical.

For example, Wang et al. (2016) developed *graph trend filtering* (GTF), which estimates a signal that takes values over the nodes of a general graph by penalizing the ℓ_1 norm of graph derivatives, defined by iterating the graph Laplacian. Translated to our setting and notation, the penalty term used by k^{th} order GTF for a signal θ at a point $x \in Z_{n,d}$ is:

$$\begin{cases} \displaystyle\sum_{j_1=1}^{d} \Big| \sum_{j_2,\ldots,j_q=1}^{d} \Big(\Delta_{x_{j_1},x_{j_2}^2,\ldots,x_{j_q}^2}\,\theta\Big)(x) \Big| & \text{for } k \text{ even, where } q = k/2, \\ \displaystyle\Big| \sum_{j_1,\ldots,j_q=1}^{d} \Big(\Delta_{x_{j_1}^2,x_{j_2}^2,\ldots,x_{j_q}^2}\,\theta\Big)(x) \Big| & \text{for } k \text{ odd, where } q = (k+1)/2. \end{cases}$$

Compared to the analogous penalty term used in KTF (6), which is just $\sum_{j=1}^d |(\Delta_{x_j^{k+1}}\theta)(x)|$, we see that the above is much is harder to interpret. GTF clearly considers some form of mixed derivatives (whereas KTF does not and is anisotropic), but it is generally unclear what kind of smoothness GTF is promoting. Moreover, Sadhanala et al. (2017) argue that using the GTF penalty operator in order to define a smoothness class for the analysis of multivariate signals is problematic, in the following sense: for any $k \geq 1$ and any dimension d, there are k^{th} order Holder smooth functions whose discretization to the lattice is arbitrarily nonsmooth as measured by the k^{th} order GTF penalty operator. This is due to issues in the way the GTF penalty operator measures smoothness on the boundaries of the lattice.

Continuous-time multivariate TV methods. There is a rich body of work in applied mathematics on the denoising of signals or images by promoting total variation smoothness, beginning with the seminal paper by Rudin et al. (1992), which gave rise to the so-called Rudin-Osher-Fatemi (ROF) functional. This was then further developed and extended by Rudin and Osher (1994); Vogel and Oman (1996); Chambolle and Lions (1997); Chan et al. (2000); Candès and Guo (2002); Chan and Esedoglu (2005); Dong et al. (2011). Papers in this line of work tend to be cast in continuous-time,[2] which means that the estimand, estimator, and typically even the data itself are each functions of one or more variables on a continuous domain (such as $[0,1]^d$). A related line of work, inspired by ROF, considers discretization as a step in numerical optimization, see, for example, Chambolle (2004, 2005); Almansa et al. (2008).

More recently, del Álamo et al. (2021) studied estimation of a multivariate function of bounded variation under the white noise model, in arbitrary dimension d. This may be interpreted as a continuum analog of our setting, albeit for $k = 0$ only. They derive minimax rates for L^p estimation of TV bounded functions. When $p = 2$ (matching our analysis), their estimator obtains (up to log factors) the minimax rate of $n^{-1/d}$ on the squared L^2 error scale, for any $d \geq 2$, which agrees with the minimax rate in our discrete setting. We note that our work, while motivated from discrete principles, bears rigorous connections to continuous-time formulations of multivariate TV; see Sections 2.3 and 3.

Alternative models for multivariate TV smoothness. Recently, there has been a stream of work studying different generalizations of TV and trend filtering penalties to multiple dimensions, including Bibaut and van der Laan (2019); Fang et al. (2021); Ortelli and van de Geer (2021b); Ki et al. (2021). These papers are based on notions of smoothness related to the Hardy-Krause variation of a multivariate function. For a function to be smooth in the Hardy-Krause sense, it must exihibit an order of smoothness that scales with the dimension d,[3] which leads to error rates that are (nearly) dimension-free. However, practically speaking, assuming that the inherent smoothness of the regression function is on par with the ambient dimension may not be reasonable in some applications. A distinct feature of our work is that the smoothness order k (which translates into the max degree of the local polynomial in the fitted model) is a user-defined parameter, and is not tied in any way to the dimension d.

One may adopt a different perspective and model the regression function as being multivariate piecewise constant, or more generally multivariate piecewise polynomial, which can be broadly interpreted as a "strong sparsity" analog to the "weak sparsity" assumption that underlies TV smoothness. In the univariate case, trend filtering (or TV denoising) has been analyzed under such "strong sparsity" assumptions by Dalalyan et al. (2017); Lin et al. (2017); Guntuboyina et al. (2020); Ortelli and van de Geer (2021a), and others, yielding faster rates of convergence under a certain min-length condition on the polynomial pieces in the true signal. The multivariate case is much more subtle; for example, for a 2d piecewise constant signal model, Chatterjee and Goswami (2021b) prove that the error rate of TV denoising depends on the orientation of the boundary of the true pieces (whether they are axis-aligned or not). Currently, it appears the most comprehensive theory for the multivariate piecewise polynomial model is given by Chatterjee and Goswami (2021a), who propose and analyze CART-style estimators, inspired by earlier work of Donoho (1997). While we believe that the KTF estimator will exhibit some degree of adaptivity to multivariate piecewise polynomials, we also believe it will have some nontrivial failure cases (suboptimality), given what the existing 1d and 2d analyses have shown.

Locally adaptive kernel smoothing. Lepski's method, which originated in the seminal paper by Lepskii (1991), is a procedure for selecting a local bandwidth in kernel smoothing; roughly speaking, at each point x in the domain, it chooses the largest bandwidth (from a discrete set of possible values) such that the kernel estimate at x is within a carefully-defined error tolerance to estimates at smaller bandwidths. Since its introduction, many papers have studied and generalized Lepski's method, see, for example, Lepskii (1992, 1993); Lepski et al. (1997); Lepski and Spokoiny (1997); Kerkyacharian et al. (2001, 2008); Goldenshluger and Lepski (2008, 2009, 2011, 2013); Lepski (2015). Most

[2] For $d \geq 2$, it may be more appropriate to call this "continuous-space", but we stick with the term continuous-time for simplicity.

[3] For example, for a smooth function f, its Hardy-Krause variation can be expressed in terms of the L^1 norm of $\partial^d f / \prod_{j=1}^d \partial x_j$.

related to the current paper is Kerkyacharian et al. (2001, 2008), who consider estimation in anisotropic Besov classes using Lepski's method, under the white noise model. The "dense" and "sparse" zones in their work roughly correspond to the cases $s > 1/2$ and $s \leq 1/2$ in our theory, respectively (see Figure 2). This will be revisited in Remark 12.

The work referenced above—and the broader literature on locally adaptive kernel methods—is focused on establishing sharp theoretical guarantees. Practical and computational considerations are not a focus. These are some nontrivial barriers to practical implementation, even as basic as the fact that there are effectively many tuning parameters (leading to somewhat arbitrary practical design choices that could be made) in the proposed methods.

Tensor product and hyperbolic wavelets. Wavelets have a rich history in signal processing, approximation theory, and other disciplines; classic references include Daubechies (1992); Chui et al. (1992a); Meyer and Roques (1993); Mallat (2009). Seminal work by Donoho and Johnstone (1998), on minimax estimation over univariate Besov spaces using wavelet-based estimators, contributed greatly to the popularity of wavelets in statistics. It appears that the earliest work on multivariate wavelet approximation is Meyer (1987, 1990) and Mallat (1989b,a), which focuses on separable wavelet bases, formed from tensor products of univariate wavelet basis functions within each resolution level. A second approach, which is well-studied in approximation theory but less so in statistics, instead constructs "truly multivariate" wavelet bases by generating multiresolution subspaces in the ambient domain, which gives rise to nonseparable bases; see Meyer (1990); Riemenschneider and Shen (1992); Chui et al. (1992b); Lorentz and Madych (1992); DeVore and Lucier (1992). In both cases described above, the support of the wavelet function has the same scale in each coordinate direction. The desire to effectively represent functions with different degrees of smoothness in different directions thus led to the development of hyperbolic wavelets, whose basis functions are formed by taking tensor products of univariate wavelet basis functions *across* resolution levels (Neumann and von Sachs, 1997; DeVore et al., 1998; Neumann, 2000). Of particular relevance to our work is the latter paper, and connections will be drawn in Remark 12.

Practically speaking, we generally find multivariate wavelet denoising to be more sensitive to the level of noise than an estimator like KTF, and for wavelet denoising to suffer worse performance when the signal-to-noise ratio is at low or moderate levels. Evidence for this is given later in Figure 7 in Section 8.1. This is also consistent with what is observed in the univariate setting in Tibshirani (2014).

1.4 Summary and outline

A summary of results in this paper and outline for this paper is given below.

- In Section 2, we derive some basic properties of KTF, including an equivalent continuous-time formulation for (7), which provides insights into the local structure of KTF estimates.
- In Section 3, we derive an expression for higher-order multivariate TV (in the standard measure-theoretic sense) in terms of an integrating univariate TV on line segments running parallel to the coordinate axes. We use this to motivate the definition of KTV smoothness, the central notion of smoothness used in this paper
- In Section 4, we introduce the smoothness classes of interest for our study of minimax theory, and examine the relationships between them.
- In Section 5, we derive a complete set of results on the minimax estimation risk, as measured in the squared ℓ_2 norm, over the set $\mathcal{T}^k_{n,d}(C_n)$ of vectors θ with k^{th} order KTV smoothness satisfying $\|D^{(k+1)}_{n,d}\theta\|_1 \leq C_n$, for a given sequence $C_n > 0$. We prove that KTF is minimax rate optimal (up to log factors) for any k, d, and derive lower bounds on the minimax linear risk (that is, the best worst-case risk over all linear smoothers) which show that linear estimators are suboptimal for any k, d. Interestingly, the minimax rates reveal a phase transition at $2(k+1) = d$, and in the low smoothness-to-dimension regime, linear smoothers fail to be consistent altogether. See Figure 2 for a more detailed summary.
- In Section 6, we study specialized convex optimization algorithms for solving the KTF problem (7).
- In Section 7, we present an extremely efficient and simple algorithm for interpolating the discrete KTF estimate $\hat{\theta}$ (the solution in (7)), defined over the lattice, into a function $\hat{f}$ (the solution in (12)), defined over the underlying continuum domain $[0,1]^d$. Remarkably, this interpolation method runs in *constant-time* (independent of n).
- In Section 8, we carry out empirical experiments that compare KTF and various other nonparametric regression estimators, and examine whether the empirical error rates match the minimax theory derived in Section 5.

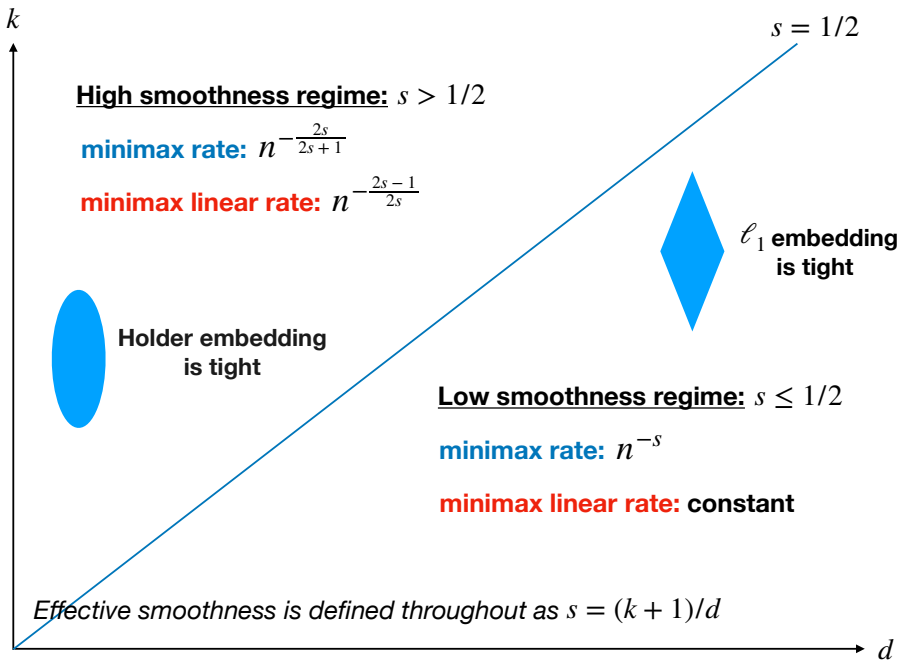

Figure 2: *Summary of the minimax results developed in this paper. The central object of our study is the set $\mathcal{T}^k_{n,d}(C^*_n)$ of vectors θ defined over the d-dimensional lattice $Z_{n,d}$, with k^{th} order KTV smoothness satisfying $\|D^{(k+1)}_{n,d}\theta\|_1 \leq C^*_n$, for a sequence $C^*_n > 0$ obeying what we call the canonical scaling, to be made precise later. The following two statements hold, generally (regardless of k, d):*

1. *KTF achieves the minimax rate (up to log factors) over $\mathcal{T}^k_{n,d}(C^*_n)$; and*
2. *no linear smoother is able to achieve the minimax rate over this class.*

However, the story is more interesting, due to a phase transition occurring at $2(k+1) = d$. Defining a notion of effective smoothness *by $s = (k+1)/d$, this can be explained as follows. When $s > 1/2$, the minimax rate has the more classical form $n^{-2s/(2s+1)}$, matching the minimax rate for a k^{th} Holder class in dimension d (or an s^{th} order Holder class in the univariate case). Indeed, the lower bound on the minimax rate that we derive is given by embedding a Holder class into $\mathcal{T}^k_{n,d}(C^*_n)$. Meanwhile, the minimax linear risk (the best worst-case risk among linear smoothers) scales as $n^{-(2s-1)/(2s)}$, which can be interpreted as the rate of KTF (or any other minimax optimal method) for a problem with a half less degree of effective smoothness. When $s \leq 1/2$, the minimax rate takes on the less classical form n^{-s}, and the lower bound is obtained by embedding a suitable ℓ_1 ball into $\mathcal{T}^k_{n,d}(C^*_n)$. Further, the gap between the minimax linear and nonlinear rates is even more dramatic: the minimax linear rate is constant, which means no linear smoother is even consistent over $\mathcal{T}^k_{n,d}(C^*_n)$ (in the sense of worst-case risk). Finally, though not reflected in the figure, we note that when $s < 1/2$ the KTV class and its embedded Holder class exhibit different minimax rates, n^{-s} versus $n^{-2s/(2s+1)}$, respectively. Whether KTF can adapt to the latter (faster) Holder rate in the low smoothness-to-dimension regime, $s < 1/2$, is an open question.*

- In Section 9, we conclude with a discussion, and cover some extensions and directions for future work.

2 Basic properties

In this section, we cover a number of basis properties that reflect the structure and complexity of KTF estimates.

2.1 Unpenalized component

We start by examining the null space of the KTF penalty matrix in (8). A word on notation here, and in general: when convenient, we will use x_j to denote the j^{th} component of a vector x, which should not be confused with our use of x_i to denote the i^{th} design point (itself a d-dimensional vector). While this is an unfortunate clash of notation, the meaning should always be clear form the context (and further, we will keep the use of indices i, j for the two cases consistent

throughout to aid the interpretation—hence x_j will always be univariate, the j^{th} component of a vector x, and x_i will always be d-dimensional, the i^{th} design point).

Proposition 1. *The null space of the KTF penalty matrix in (8) has dimension $(k+1)^d$. Furthermore, it is spanned by a polynomial basis made up of elements*

$$p(x) = x_1^{a_1} x_2^{a_2} \cdots x_d^{a_d}, \quad x \in Z_{n,d},$$

for all $a_1, \ldots, a_d \in \{0, \ldots, k\}$.

The proof is elementary and is deferred to Appendix A (all proofs in this paper are deferred to the appendix). This proposition reveals that the KTF penalty matrix has quite a rich null space, thus KTF lets a significant component of the response vector y "pass through" unpenalized. In contrast to univariate trend filtering, which preserves univariate polynomials of degree k (precisely, it preserves the projection of y onto this subspace), KTF preserves "much more" than multivariate polynomials of degree k: it preserves multivariate polynomials of *max degree* k.[4] To see an example, when $k = 1$ and $d = 2$, the KTF estimator—which we might be tempted to call "linear-order" KTF (to use an analogous term as we do in univariate trend filtering)—preserves any polynomial of the form $p(x) = ax_1 + bx_2 + cx_1x_2$. This is of course *not* a linear function, but a bilinear one (due to the cross-product term x_1x_2).

2.2 Review: trend filtering in continuous-time

For univariate trend filtering (3), an equivalent continuous-time formulation was derived in Tibshirani (2014):

$$\underset{f \in \mathcal{H}_n^k}{\text{minimize}} \; \frac{1}{2} \sum_{i=1}^n \big(y_i - f(x_i)\big)^2 + \lambda \, \mathrm{TV}(f^{(k)}). \tag{9}$$

Problems (3), (9) are equivalent in the sense that their solutions $\hat{\theta}$, $\hat{f}$, respectively, satisfy $\hat{\theta}_i = \hat{f}(x_i)$, $i = 1, \ldots, n$. In (9), we use $f^{(k)}$ to denote the k^{th} weak derivative of f, and $\mathrm{TV}(\cdot) = \mathrm{TV}(\cdot; [a, b])$ to denote the total variation operator defined with respect to any interval $[a, b]$ containing the design points (henceforth, when convenient, we will drop the underlying domain from our notation for univariate TV). The optimization in (9) is performed over all functions f that lie in a space $\mathcal{H}_n^k$, which is the span of $h_{n,j}^k$, $j = 1, \ldots, n$, the following k^{th} degree piecewise polynomials:

$$\begin{aligned} h_{n,i}^k(x) &= \frac{1}{(i-1)!} \prod_{j=1}^{i-1} (x - j/n), \quad i = 1, \ldots, k+1, \\ h_{n,i}^k(x) &= \frac{1}{k!} \prod_{j=i-k}^{i-1} (x - j/n) \cdot 1\{x > (i-1)/n\}, \quad i = k+2, \ldots, n. \end{aligned} \tag{10}$$

(Here, for convenience, we interpret the empty product to be equal to 1.) The functions in (10) are called the k^{th} degree *falling factorial basis*. Observe that they depend on the n underlying design points $1/n, 2/n, \ldots, 1$, which we express through the first subscript n in $h_{n,i}^k$. Note also the similarity between the above basis and the standard truncated power basis for splines; in fact, when $k = 0$ or $k = 1$, the two bases are equal and $\mathcal{H}_n^k$ is just a space of splines with knots at the design points. However, when $k \geq 2$, this is no longer true—the falling factorial functions are k^{th} degree piecewise polynomials with (mildly) discontinuous derivatives of all orders $1, \ldots, k-1$, and therefore they span a different space than that of k^{th} degree splines—they space the space of k^{th} degree *discrete splines*, which are piecewise polynomials that have continuous discrete derivatives (rather than derivatives) at their knot points. See Tibshirani (2020).

To be clear, the original formulation (3) is more computationally convenient (it is a structured convex problem for which several fast algorithms exist, discussed in Section 1.3). But the variational formulation (9) is important because it provides rigorous backing to the intuition that a k^{th} order trend filtering estimate exhibits the structure of a k^{th} degree piecewise polynomial, with adaptively-chosen knots (a feature of the ℓ_1 penalty in (3) or TV penalty in (9)); moreover, it shows how to extend the trend filtering estimate from a discrete sequence, defined over the design points, to a function on the continuum interval $[0, 1]$.

[4] By a multivariate polynomial of degree k, we mean (adhering to the standard classification) a sum of terms of the form $b \prod_{j=1}^d x_j^{a_j}$, where the sum of degrees satisfies $\sum_{j=1}^d a_j \leq k$. By a multivariate polynomial of max degree k, we mean the same, but where the degrees satisfy $a_j \leq k$, $j = 1, \ldots, d$.

2.3 Continuous-time

We now develop a similar continuous-time representation for KTF.

Proposition 2. *Let $h^k_{N,i} : [0,1] \to \mathbb{R}$, $i = 1, \dots, N$ denote the k^{th} degree falling factorial functions (10) with respect to design points $1/N, 2/N, \dots, 1$, and $\mathcal{H}^k_{n,d}$ denote the space spanned by all d-wise tensor products of these functions. That is, abbreviating $h_i = h^k_{N,i}$, $i = 1, \dots, N$, this is the space of all functions $f : [0,1]^d \to \mathbb{R}$ of the form*

$$f(x) = \sum_{i_1,\dots,i_d=1}^{N} \alpha_{i_1,\dots,i_d} h_{i_1}(x_1) h_{i_2}(x_2) \cdots h_{i_d}(x_d), \quad x \in [0,1]^d, \tag{11}$$

for coefficients $\alpha \in \mathbb{R}^n$ (whose components we denote as $\alpha_{i_1,\dots,i_d}$, $i_1, \dots, i_d \in \{1, \dots, N\}$). Then the KTF estimator defined in (7) is equivalent to the optimization problem:

$$\underset{f \in \mathcal{H}^k_{n,d}}{\text{minimize}} \ \frac{1}{2} \sum_{i=1}^{n} \big(y_i - f(x_i)\big)^2 + \lambda \sum_{j=1}^{d} \sum_{x_{-j}} \mathrm{TV}\bigg(\frac{\partial^k f(\cdot, x_{-j})}{\partial x_j^k}\bigg), \tag{12}$$

where $f(\cdot, x_{-j})$ denotes f as function of the j^{th} coordinate with all other dimensions fixed at x_{-j}, $(\partial^k / \partial x_j^k)(\cdot)$ denotes the k^{th} partial weak derivative operator with respect to x_j, and inner sum in the second term in (12) is interpreted as a sum over $Z_{m,d-1}$, where $m = N^{d-1}$ (that is, a sum over the $(d-1)$-dimensional uniformly-spaced lattice with N^{d-1} total points). The discrete (7) and continuous-time (12) problems are equivalent in the sense that at their solutions $\hat{\theta}$, $\hat{f}$, respectively, we have: $\hat{\theta}_i = \hat{f}(x_i)$, $i = 1, \dots, n$.

Remark 1. Similar to the discrete-time terminology, we will refer to $\sum_{j=1}^{d} \sum_{x_{-j}} \mathrm{TV}(\partial^k f(\cdot, x_{-j}) / \partial x_j^k)$, the penalty functional in the continuous-time representation (12) of the KTF problem, as the k^{th} order *Kronecker total variation* (KTV) of f. A key implication of Proposition 2 is that the solution $\hat{f}$ in (12) not only interpolates the solution $\hat{\theta}$ in (7), but it is exactly as smooth in continuous-time as $\hat{\theta}$ is in discrete-time, as measured by KTV:

$$\|D^{(k+1)}_{n,d} \hat{\theta}\|_1 = \sum_{j=1}^{d} \sum_{x_{-j}} \mathrm{TV}\bigg(\frac{\partial^k \hat{f}(\cdot, x_{-j})}{\partial x_j^k}\bigg).$$

How to form the interpolant $\hat{f}$ is discussed briefly in the remark after the next, and covered in detail in Section 7.

Remark 2. From (11), the basis underlying the continuous-time representation (12) of the KTF optimization problem, we can see that a k^{th} order KTF estimate exhibits the structure of a tensor product of k^{th} degree discrete splines, with adaptively knots, chosen to promote higher-order TV smoothness along the coordinate axes. In other words, locally, it exhibits the structure of a multivariate polynomial of max degree k. When $k = 1$ and $d = 2$, for example, the structure is locally of the form $\hat{f}(x) = ax_1 + bx_2 + cx_1x_2$, which is a bilinear function and has local curvature. Such curvature is somewhat visible in Figure 1 (bottom row, middle panel), and it will be even more apparent later when we discuss interpolation, see Figure 6.

Remark 3. The proof of Proposition 2 reveals that (12) has an equivalent form, transcribed here for convenience:

$$\underset{\alpha \in \mathbb{R}^n}{\text{minimize}} \ \frac{1}{2} \Big\| y - \Big(H^{(k+1)}_N \otimes \cdots \otimes H^{(k+1)}_N\Big)\alpha \Big\|_2^2 + \lambda k! \left\| \begin{bmatrix} I^0_N \otimes H^{(k+1)}_N \otimes \cdots \otimes H^{(k+1)}_N \\ H^{(k+1)}_N \otimes I^0_N \otimes \cdots \otimes H^{(k+1)}_N \\ \vdots \\ H^{(k+1)}_N \otimes H^{(k+1)}_N \otimes \cdots \otimes I^0_N \end{bmatrix} \alpha \right\|_1, \tag{13}$$

where $H^{(k+1)}_N \in \mathbb{R}^{N \times N}$ is the falling factorial basis matrix (with columns given by evaluations of the falling factorial functions at the design points) and $I^0_N \in \mathbb{R}^{(N-k-1) \times N}$ denotes the last $N-k-1$ rows of the identity I_N. Interestingly, the penalty in (13) is not a pure ℓ_1 penalty on the coefficients α (as it would be in basis form in the univariate case) but an ℓ_1 penalty on aggregated (positive linear combinations of) coefficients.

The basis formulation in (13) gives us a natural recipe for how to extend a KTF estimate from a discrete sequence, defined over the lattice points, to a function on the hypercube $[0,1]^d$. This is simply:

$$\hat{f}(x) = \sum_{i_1,\dots,i_d=1}^{N} \hat{\alpha}_{i_1,\dots,i_d} h_{i_1}(x_1) h_{i_2}(x_2) \cdots h_{i_d}(x_d), \quad x \in [0,1]^d, \tag{14}$$

where $\hat{\alpha}$ is the solution in (13). Though it is not obvious from the basis expansion in (14), it turns out that at any point $x \in [0,1]^d$, we can form the prediction $\hat{f}(x)$ in constant-time, starting from the fitted values $\hat{\theta}_i = \hat{f}(x_i)$, $i = 1, \dots, n$. This leverages recent advances in discrete spline interpolation from Tibshirani (2020), and is covered in Section 7.

2.4 Degrees of freedom

Given data from a model (1), where the errors ϵ_i, $i = 1, \dots, n$ are i.i.d. with mean zero and variance σ^2, recall that the *degrees of freedom* of an estimator $\hat{\theta}_i = \hat{f}(x_i)$, $i = 1, \dots, n$ of the means $\theta_{0i} = f_0(x_i)$, $i = 1, \dots, n$ is a quantitative reflection of its complexity, defined as (Efron, 1986; Hastie and Tibshirani, 1990):

$$\mathrm{df}(\hat{\theta}) = \frac{1}{\sigma^2} \sum_{i=1}^n \mathrm{Cov}(y_i, \hat{\theta}_i).$$

When the errors are Gaussian, Tibshirani and Taylor (2011, 2012) derived an expression for the degrees of freedom of any generalized lasso estimator, based on Stein's formula (Stein, 1981). This covers KTF in (7) as a special case, and thus translates into the following result for our setting: if $\epsilon_i \sim N(0, \sigma^2)$, $i = 1, \dots, n$ in (1), and $\hat{\theta}$ denotes the solution in (7) with active set

$$A = \mathrm{supp}\big(D^{(k+1)}_{n,d} \hat{\theta}\big) = \big\{ i : \big[D^{(k+1)}_{n,d} \theta\big]_i \neq 0 \big\},$$

then

$$\mathrm{df}(\hat{\theta}) = \mathbb{E}\Big[\mathrm{nullity}\Big(\big[D^{(k+1)}_{n,d}\big]_{-A} \Big) \Big], \tag{15}$$

where $\mathrm{nullity}(M)$ denotes the nullity (dimension of the null space) of a matrix M, and M_{-S} denotes the submatrix of M given by removing all rows indexed by a set S. From the above, we of course have the natural estimator of degrees of freedom

$$\widehat{\mathrm{df}}(\hat{\theta}) = \mathrm{nullity}\Big(\big[D^{(k+1)}_{n,d}\big]_{-A} \Big), \tag{16}$$

which is unbiased for (15).

The expression in (16) is easy to interpret when $k = 0$: in this case, it reduces to the number of connected constant pieces in the KTF solution $\hat{\theta}$, where connectivity is interpreted with respect to the underlying d-dimensional grid graph. This follows from the fact that the penalty matrix $D^{(1)}_{n,d}$ in this case is the edge incidence operator on the grid graph, and any submatrix of this penalty matrix (defined by removing a subset of rows) is itself the edge incidence operator with respect to a subgraph of the original grid (induced by removing a subset of the edges). This result and its interpretation was already given in Tibshirani and Taylor (2011, 2012) in the context of TV denoising on a graph.

When $k \geq 1$, the unbiased estimator of degrees of freedom in (16) is not as easy to interpret. This is because, at a high level, there is no longer a clear link between the local structure exhibited by $\hat{\theta}$ and whether or not a particular entry of $D^{(k+1)}_{n,d} \hat{\theta}$ is nonzero. However, we show in Appendix D that it is possible to compute the right-hand side in (16) with a simple, direct algorithm that runs in linear time (more precisely, the algorithm requires $O(ndk)$ operations).

3 Interlude: total variation on lines

In this section, we take a continuum perspective, looking at total variation defined over functions in $\mathbb{R}^d$, and connect it to the discrete notion of total variation used in the previous sections used to define the KTF estimator.

3.1 Measure-theoretic total variation

Let U be an open, bounded subset of $\mathbb{R}^d$ and $L^p(U)$ denote the space of real-valued functions on U with finite L^p norm, $\int_U |f(x)|^p \, dx < \infty$. A function $f \in L^1(U)$ is said to be of *bounded variation* (BV) provided $\mathrm{TV}(f; U) < \infty$, where

$$\mathrm{TV}(f; U) = \sup \bigg\{ \int_U f(x) \, \mathrm{div} \phi(x) \, dx : \phi \in C^\infty_c(U; \mathbb{R}^d), \ \|\phi(x)\|_\infty \leq 1 \text{ for all } x \in U \bigg\}. \tag{17}$$

Above, $C^\infty_c(U; \mathbb{R}^d)$ denotes the space of infinitely continuously differentiable functions from U to $\mathbb{R}^d$ with compact support, and $\mathrm{div}(\cdot)$ denotes the divergence operator, $\mathrm{div} \phi = \sum_{j=1}^d \partial \phi_j / \partial x_j$. We call $\mathrm{TV}(f; U)$ the *total variation* of f; this is the standard measure-theoretic definition used in modern analysis; see, for example, Chapter 5 of Evans and

Gariepy (2015). To be clear, it would be more precise to call our definition in (17) the *aniostropic* total variation of f (due to the use of the ℓ_∞ norm in the constraint in (17) on the test function ϕ), but we often drop the reference to the anisotropic qualifier for simplicity.

To build intuition, we note that if $f \in W^{1,1}(U)$, that is, f is in $L^1(U)$ and it is weakly differentiable and its weak derivative ∇f is also in $L^1(U)$, then

$$\mathrm{TV}(f; U) = \int_U \|\nabla f(x)\|_1 \, dx. \tag{18}$$

Writing $\mathrm{BV}(U)$ for the space of bounded variation functions, the above shows that $W^{1,1}(U) \subseteq \mathrm{BV}(U)$. Importantly, this is a strict inclusion, because, for example, the indicator function of a set that has smooth boundary is of bounded variation, but it is not in $W^{1,1}(U)$ (it is not weakly differentiable).

3.2 Univariate total variation revisited

In order to connect the discrete notions of TV that we use in this paper to the standard measure-theoretic definition of TV defined in (17), we must first refine our definition of univariate TV. For a function $g : [a, b] \to \mathbb{R}$, we define its total variation as:

$$\mathrm{TV}(g; [a, b]) = \sup_{\substack{a < z_1 < \cdots < z_{m+1} < b \\ z_1, \ldots, z_{m+1} \in \mathrm{AC}(g)}} \sum_{i=1}^{m} |g(z_i) - g(z_{i+1})|, \tag{19}$$

where the supremum is only taken over the set $\mathrm{AC}(g)$ of points of approximate continuity of g. Approximate continuity is a weak notion of continuity that excludes, for example, point discontinuities; see Section 1.7.2 of Evans and Gariepy (2015). Observe that the definition in (19) differs from that given in the introduction in that the latter does not require that the supremum be taken over points of approximate continuity. Some authors, including Evans and Gariepy (2015), differentiate these definitions by calling the latter the *variation* of g and (19) the *essential variation* of g. An intuitive way of interpreting their connection is as follows: the essential variation of g is the infimum of the variation achievable by any function $\tilde{g}$ that agrees with g Lebesgue almost everywhere.

The reason the refinement in (19) is important, when using the measure-theoretic definition in (17) as a basis for defining the BV space, is that BV functions (as with L^p functions and Sobolev functions) are only well-defined up to a set of Lebesgue measure zero. That is, if f and $\tilde{f}$ agree Lebesgue almost everywhere, then their TV as defined in (17) (as with L^p norms or Sobolev norms) must also agree. Therefore, it should be clear that (19) is the proper univariate notion here, as otherwise redefining g at a point would change its univariate TV (without restricting the supremum to points of approximate continuity). Lastly, and reassuringly, the multivariate measure-theoretic definition in (17) reduces to the univariate definition in (19) once we take $U = (a, b)$ (see Theorem 5.21 in Evans and Gariepy (2015)).

3.3 Total variation on lines

We now proceed in the opposite direction to the end of the last subsection: instead of reducing the multivariate definition to the univariate case, we will use the univariate definition of TV to approach the multivariate one. Interestingly, as we will see next, it turns out that (19) can be used as a building block for (17) for an open, bounded set $U \subseteq \mathbb{R}^d$. In words, the next result says that the multivariate notion of TV on U is given by aggregating the univariate notion on all line segments parallel to the coordinate axes, anchored at boundary points of U.

Theorem 1. *Let $U \subseteq \mathbb{R}^d$ be an open, bounded, convex set. Then for any $f \in \mathrm{BV}(U)$,*

$$\mathrm{TV}(f; U) = \sum_{j=1}^{d} \int_{U_{-j}} \mathrm{TV}\big(f(\cdot, x_{-j}); I_{x_{-j}}\big) \, dx_{-j}, \tag{20}$$

where for each $j = 1, \ldots, d$, we define $U_{-j} = \{x_{-j} : (x_j, x_{-j}) \in U \text{ for some } x_j\}$, and $I_{x_{-j}} = [a_{x_{-j}}, b_{x_{-j}}]$, with

$$a_{x_{-j}} = \inf\{x_j : (x_j, x_{-j}) \in U\},$$
$$b_{x_{-j}} = \sup\{x_j : (x_j, x_{-j}) \in U\}.$$

Recall $f(\cdot, x_{-j})$ denotes f as function of the j^{th} coordinate with all other dimensions fixed at x_{-j}. Lastly, the univariate TV operator in the integrand in (20) is to be interpreted in the essential variation sense, as in (19).

Remark 4. The assumption of convexity of U in Theorem 1 is used for simplicity, to ensure that each coordinatewise slice of U—intersecting it with a line segment parallel to the coordinate axis—is an interval. The proof trivially extends to the case in which each slice is a finite union of intervals; more complex structures could likely be handled via more complex arguments.

Remark 5. The above result is inspired by Theorem 5.22 of Evans and Gariepy (2015). In the proof, we mollify f in order to invoke the representation in (18) for the TV of a smooth function, and then we leverage the separability of the ℓ_1 norm (that is, we leverage the fact that the integrand in (18) decomposes into a sum of absolute partial derivatives) in order to derive (20). Curiously, an analogous result does not seem straightforward to derive for the case of isotropic TV; this being defined by using an ℓ_2 norm constraint on the test function ϕ in (17) (and for functions in $W^{1,2}(U)$, it would reduce to the integral of ℓ_2 norm of the weak derivative, instead of the ℓ_1 norm as in (18)).

3.4 Connection to KTV smoothness

We connect the representation in (20) to the KTF penalty functional, which we call KTV smoothness. First note that we can rewrite the definition of the anisotropic TV of a function f in (17) as

$$\mathrm{TV}(f;U) = \sum_{j=1}^{d} \underbrace{\sup\left\{ \int_U f(x)\, \frac{\partial \phi(x)}{\partial x_j}\, dx : \phi \in C_c^\infty(U),\ |\phi(x)| \le 1 \text{ for all } x \in U \right\}}_{V_j(f;U)},$$

where $C_c^\infty(U)$ is the space of infinitely continuously differentiable real-valued functions on U. Note that $V_j(f;U)$, as defined above, measures the variation of f along the j^{th} coordinate direction. Now consider the following definition of k^{th} order multivariate TV, for an integer $k \ge 0$:

$$\mathrm{TV}^k(f;U) = \sum_{j=1}^{d} V_j\left(\frac{\partial^k f}{\partial x_j^k}; U\right), \tag{21}$$

where $\partial^k f / \partial x_j^k$ denotes the k^{th} partial weak derivative of f with respect to x_j. The formulation in (21) is, in a sense, among the many possible options for higher-order TV in the multivariate setting, the "most" anisotropic. It only looks at the variation in the partial derivatives along the coordinate directions with respect to which they are defined (that is, the variation in the j^{th} partial derivative along the j^{th} coordinate axis).

Thanks to the representation in Theorem 1, we can rewrite the definition of k^{th} order TV in (21) as:

$$\mathrm{TV}^k(f;U) = \sum_{j=1}^{d} \int_{U_{-j}} \mathrm{TV}\left(\frac{\partial^k f(\cdot, x_{-j})}{\partial x_j^k}\right) dx_{-j}, \tag{22}$$

where we have made the dependence on the domain $I_{x_{-j}}$ in the TV operator in the integrand implicit. Finally, we are ready to draw the connection to KTF. Observe that the penalty functional underlying KTF, the second term in the criterion of its continuous-time formulation (12), is given by taking the notion of k^{th} order TV in (22) and approximating the integral via discretization; that is, the integral over U_{-j} is simply replaced by a sum over an embedded lattice.

4 Smoothness classes

We present various discrete smoothness classes, then connect them to each other and to traditional Holder smoothness classes defined in continuous-time, to derive what we refer to as *canonical scalings* for the radii of the discrete classes. This paves the way for the minimax analysis in the next section.

4.1 Discrete TV and Sobolev classes

First we define the k^{th} order *Kronecker total variation* (KTV) class, for a radius $\rho > 0$, by

$$\mathcal{T}_{n,d}^k(\rho) = \big\{\theta \in \mathbb{R}^n : \|D_{n,d}^{(k+1)}\theta\|_1 \le \rho\big\}. \tag{23}$$

It is a priori unclear what scaling for the radius ρ in (23) makes for an "interesting" smoothness class for theoretical analysis. This is discussed at length in Sadhanala et al. (2016), and was one of the original motivations for that paper. There, it is shown that taking ρ to be a constant (as $n \to \infty$) leads to seemingly very fast minimax rates, however, in this regime trivial estimators turn out to be rate optimal (such as the sample mean estimator $\hat{\theta}_i = \bar{y}$, $i = 1, \dots, n$).

In order to begin reasoning about scalings for ρ in (23), it helps to define the order $k+1$ discrete ℓ_2-Sobolev class:

$$\mathcal{W}_{n,d}^{k+1}(\rho) = \big\{\theta \in \mathbb{R}^n : \|D_{n,d}^{(k+1)}\theta\|_2 \leq \rho\big\}. \tag{24}$$

Observe that $\mathcal{W}_{n,d}^{k+1}(\rho)$ only considers partial derivatives of order $k+1$ aligned with one of the coordinate axes, rather than considering all mixed derivatives of total order $k+1$, as we would in a traditional Sobolev class. For simplicity, we drop reference to the ℓ_2 prefix when referring to (24) henceforth.

By the inequality $\|v\|_2 \leq \sqrt{p}\|\theta\|_1$ for vectors $v \in \mathbb{R}^p$, and the fact that the number of rows of $D_{n,d}^{(k+1)}$ can be upper bounded by dn, we have the following embedding:

$$\mathcal{W}_{n,d}^{k+1}(\rho) \subseteq \mathcal{T}_{n,d}^{k}\big(\sqrt{dn}\rho\big), \quad \text{for any } \rho > 0.$$

This shows that any reasonable regime for analysis must have ρ varying with n in (23), or in (24) (or both), because a constant radius in one class would translate into a growing or diminishing radius in the other, by the above display. However, it still leaves unspecified what precise scalings for the radii in (23) and (24), would correspond to "interesting" classes, comparable in some sense to choices of radii in analogous continuous-time TV or Sobolev smoothness classes. We answer this question in the next subsection, by introducing discrete and continuum Holder classes, and pursuing further embeddings.

4.2 Discrete and continuum Holder classes

Now we recall the traditional definition for the k^{th} order Holder class of functions from $[0,1]^d$ to $\mathbb{R}$, of radius $L > 0$:

$$C^k(L;[0,1]^d) = \Bigg\{ f : [0,1]^d \to \mathbb{R} : f \text{ is } k \text{ times differentiable and for all integers } \alpha_1, \dots, \alpha_d \geq 0,$$
$$\text{with } \alpha_1 + \cdots + \alpha_d = k, \ \left| \frac{\partial^k f(x)}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}} - \frac{\partial^k f(z)}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}} \right| \leq L\|x - z\|_2, \ \text{ for all } x, z \in [0,1]^d \Bigg\}.$$

We define a discretized version of this class by simply evaluating the functions in $C^k(L;[0,1]^d)$ on the lattice $Z_{n,d}$:

$$\mathcal{C}_{n,d}^{k}(L) = \big\{\theta \in \mathbb{R}^n : \text{there exists some } f \in C^k(L;[0,1]^d) \text{ such that } \theta(x) = f(x), x \in Z_{n,d}\big\}. \tag{25}$$

The next proposition derives embeddings for the discrete Holder class (25) into the discrete Sobolev (24) and KTV (23) classes. It is a direct consequence of Lemma A.6 in Sadhanala et al. (2017) (which is a classical result of sorts that quantifies the error of the forward difference approximation of the derivative of a Holder function).

Proposition 3 (Sadhanala et al. 2017)**.** *The discrete classes in* (23)–(25) *satisfy, for any* $L > 0$*,*

$$\mathcal{C}_{n,d}^{k}(L) \subseteq \mathcal{W}_{n,d}^{k+1}\big(c_1 L n^{\frac{1}{2} - \frac{k+1}{d}}\big) \subseteq \mathcal{T}_{n,d}^{k}\big(c_2 L n^{1 - \frac{k+1}{d}}\big), \tag{26}$$

where $c_1, c_2 > 0$ *are constants depending only on* k, d*.*

Motivated by the last result, as in Sadhanala et al. (2017), we define the *canonical scalings* for the discrete Sobolev and KTV classes as

$$B_n^* = n^{\frac{1}{2} - \frac{k+1}{d}}, \tag{27}$$
$$C_n^* = n^{1 - \frac{k+1}{d}}, \tag{28}$$

so that $\mathcal{C}_{n,d}^{k}(1) \subseteq \mathcal{W}_{n,d}^{k+1}(c_1 B_n^*) \subseteq \mathcal{T}_{n,d}^{k}(c_2 C_n^*)$, for constants $c_1, c_2 > 0$ that depend only on k, d. Thus, by analogy to classical results on nonparametric estimation over Holder spaces, we should expect the minimax rate over $\mathcal{W}_{n,d}^{k+1}(B_n^*)$ (in the squared ℓ_2 norm) to be $n^{-2(k+1)/(2(k+1)+d)}$. This is indeed the case, as we will show at the end of Section 5. The minimax rate over $\mathcal{T}_{n,d}^{k}(C_n^*)$, on the other hand, will turn out to be more exotic, and is the focus of the majority of the next section.

5 Estimation theory

We derive a number of results on estimation theory over KTV classes. We begin by deriving upper bounds on the error of the KTF estimator, and then study lower bounds. Throughout, we assume the data model in (1) with $\theta_{0i} = f_0(x_i)$, $i = 1, \dots, n$ and i.i.d. normal errors, to be precise:

$$y_i \sim N(\theta_{0,i}, \sigma^2), \quad \text{independently, for } i = 1, \dots, n. \tag{29}$$

To set some basic notation, based on estimators $\hat{\theta}$ of the mean θ_0 in (29), we define for a subset $\mathcal{K} \subseteq \mathbb{R}^n$,

$$R(\mathcal{K}) = \inf_{\hat{\theta}} \sup_{\theta_0 \in \mathcal{K}} \frac{1}{n} \mathbb{E} \|\hat{\theta} - \theta_0\|_2^2,$$

which is called the *minimax risk* over $\mathcal{K}$. Also of interest will be

$$R_L(\mathcal{K}) = \inf_{\hat{\theta} \text{ linear}} \sup_{\theta_0 \in \mathcal{K}} \frac{1}{n} \mathbb{E} \|\hat{\theta} - \theta_0\|_2^2,$$

called the *minimax linear risk* over $\mathcal{K}$, the infimum being restricted to linear estimators $\hat{\theta}$ (that is, of the form $\hat{\theta} = Sy$ for a matrix $S \in \mathbb{R}^{n \times n}$). To finish our discussion of notation, for deterministic sequences a_n, b_n we write $a_n = O(b_n)$ when a_n/b_n is upper bounded for large enough n, we write $a_n = \Omega(b_n)$ when $a_n^{-1} = O(b_n^{-1})$, and $a_n \asymp b_n$ when both $a_n = O(b_n)$ and $a_n = \Omega(b_n)$. For random sequences A_n, B_n, we write $A_n = O_{\mathbb{P}}(B_n)$ when A_n/B_n is bounded in probability. In the theory that follows, all asymptotics are for $n \to \infty$ with k, d fixed.

As a side remark, although not the focus of the current paper, analogous theory can be established for graph trend filtering on grids (see Section 1.3 for a discussion of its relation to KTF), which we defer to Appendix B.

5.1 Upper bounds on estimation risk

To derive upper bounds on the risk of KTF, we leverage the following simple generalization of a key result from Wang et al. (2016). Here and henceforth, for an integer $a \geq 1$, we abbreviate $[a] = \{1, \dots, a\}$.

Theorem 2 (Wang et al. 2016)**.** *Consider the generalized lasso estimator $\hat{\theta}$ with penalty matrix $D \in \mathbb{R}^{r \times n}$, defined by the solution of*

$$\underset{\theta \in \mathbb{R}^n}{\text{minimize}} \ \frac{1}{2} \|y - \theta\|_2^2 + \lambda \|D\theta\|_1 \tag{30}$$

Suppose that D has rank q, and denote by $\xi_1 \leq \dots \leq \xi_q$ its nonzero singular values. Also let $u_1, \dots, u_q \in \mathbb{R}^r$ be the corresponding left singular vectors. Assume that these vectors, except possibly for those in a set $I \subseteq [q]$, are incoherent*, meaning that for a constant $\mu \geq 1$,*

$$\|u_i\|_\infty \leq \mu/\sqrt{n}, \quad i \in [q] \setminus I.$$

Then under the data model (29), choosing

$$\lambda \asymp \mu \sqrt{\frac{\log r}{n} \sum_{i \in [q] \setminus I} \frac{1}{\xi_i^2}},$$

the generalized lasso estimator satisfies

$$\frac{1}{n} \|\hat{\theta} - \theta_0\|_2^2 = O_{\mathbb{P}} \left(\frac{\text{nullity}(D)}{n} + \frac{|I|}{n} + \frac{\mu}{n} \sqrt{\frac{\log r}{n} \sum_{i \in [q] \setminus I} \frac{1}{\xi_i^2}} \cdot \|D\theta_0\|_1 \right). \tag{31}$$

We will now apply this result to KTF in (7), and choose the set I in order to balance the second and third terms on the right-hand side in (31). Throughout, we will make reference to the *effective degree of smoothness* (or effective smoothness for short), defined by

$$s = \frac{k+1}{d}.$$

Theorem 3. *Let $\hat{\theta}$ denote the KTF estimator in (7). Under the data model (29), denote $C_n = \|D_{n,d}^{(k+1)}\theta_0\|_1$, and assume $C_n > 0$. Choosing*

$$\lambda \asymp \begin{cases} \sqrt{\log n} & \text{if } s < 1/2, \\ \log n & \text{if } s = 1/2, \\ (\log n)^{\frac{1}{2s+1}} (n/C_n)^{\frac{2s-1}{2s+1}} & \text{if } s > 1/2, \end{cases}$$

the KTF estimator satisfies

$$\frac{1}{n}\|\hat{\theta} - \theta_0\|_2^2 = O_{\mathbb{P}}\left(\frac{1}{n} + \frac{\lambda}{n} C_n\right).$$

Remark 6. The result in the above theorem for the case of $k = 0$ (TV denoising) and any d was already established in Hutter and Rigollet (2016). The result for $d = 2$ and any k was given in Sadhanala et al. (2017). For $d = 1$ and any k, the result was established in Mammen and van de Geer (1997); Tibshirani (2014) (though the results in the latter papers are sharper by log factors).

Theorem 3 covers all k and d. Its proof, an application of Theorem 2 to KTF, requires checking the incoherence of the penalty matrix $D = D_{n,d}^{(k+1)}$, and bounding the partial sum of squared reciprocal singular values $\sum_{i \in [q] \setminus I} \xi_i^{-2}$, for an appropriate set I that corresponds to small eigenvalues. The former—checking the incoherence of the left singular vectors of the KTF penalty matrix—turns out to be the harder calculation. However, the hardest part of this calculation was already done in Sadhanala et al. (2017), who established the incoherence of $D_n^{(k+1)}$ (the trend filtering penalty matrix, or equivalently, the KTF penalty matrix when $d = 1$), using complex approximation results for the eigenvectors of Toeplitz matrices. To handle KTF in arbitrary dimension d, in the current paper, we use careful arguments that relate singular vectors of Kronecker products to singular vectors of their constituent matrices.

Remark 7. The choice of tuning parameter λ in Theorem 3 generally depends on the smoothness level C_n, as well as the order k of the underlying KTV smoothness class (which appears through $s = (k+1)/d$). In this sense, our KTF bounds are weaker than some of the sharpest results in the literature on nonparametric estimation, which are adaptive to underlying smoothness parameters (k and C_n in our setting). However, it is interesting to note that the case $s \leq 1/2$, that is, $2k + 2 \leq d$, appears to be special: KTF ends up being adaptive to C_n. In other words, the prescribed choice of tuning parameter is $\lambda \asymp \sqrt{\log n}$ when $s < 1/2$, and $\lambda \asymp \log n$ when $s = 1/2$, neither of which depend on C_n.

Under the canonical scaling (28), the error bound in Theorem 3 reduces to the following.

Corollary 1. *Assume the conditions of Theorem 3. When $C_n \asymp L_n C_n^* = L_n n^{1-s}$, where C_n^* is the canonical scaling in (28) for the KTV smoothness level, the KTF estimator satisfies*

$$\frac{1}{n}\|\hat{\theta} - \theta_0\|_2^2 = \begin{cases} O_{\mathbb{P}}(L_n n^{-s} \sqrt{\log n}\,) & \text{if } s < 1/2, \\ O_{\mathbb{P}}(L_n n^{-s} \log n) & \text{if } s = 1/2, \\ O_{\mathbb{P}}\big(L_n^{\frac{2}{2s+1}} n^{-\frac{2s}{2s+1}} (\log n)^{\frac{1}{2s+1}}\big) & \text{if } s > 1/2. \end{cases} \tag{32}$$

Remark 8. Recall the continuous-time analog of the KTF penalty in (12). It turns out, as we show in Appendix A.6, that the assumption that the mean vector θ_0 is KTV smooth in Theorem 3 and Corollary 1 can be broadened into an assumption that the KTV of the mean *function* f_0 is KTV smooth, in the sense of the penalty functional in (12).

5.2 Lower bounds on estimation risk

Next we give lower bounds on the minimax risk over KTV classes.

Theorem 4. *The minimax risk for KTV class defined in (23) satisfies, for any sequence $C_n \leq n$,*

$$R\big(\mathcal{T}_{n,d}^k(C_n)\big) = \Omega\left(\frac{1}{n} + \frac{C_n}{n} + \left(\frac{C_n}{n}\right)^{\frac{2}{2s+1}}\right). \tag{33}$$

Remark 9. The result in Theorem 4 for $s \geq 1/2$ (that is, $2k + 2 \geq d$) was already derived in Sadhanala et al. (2017). More precisely, these authors established the third term on the right-hand side in (33), by using the Holder embedding

in (26) of Proposition 3, and suitably adapting classical results on minimax bounds for Holder spaces (Korostelev and Tsybakov, 2003; Tsybakov, 2009), which ends up being tight in the $s > 1/2$ regime. Moreover, the lower bound in the middle term in (33) was derived by Sadhanala et al. (2016), for $k = 0$ and all d, which was obtained by embedding an appropriate ℓ_1 ball into $\mathcal{T}^k_{n,d}(C_n)$, and appealing to minimax theory over ℓ_1 balls from Birge and Massart (2001). This ends up being tight for $k = 0$ and all d.

In the current work, we establish tight lower bounds over all k, d, by essentially combining these two strategies. As we will see comparing to upper bounds in the next remark, the Holder embedding ends up being tight for $s > 1/2$, and the ℓ_1 embedding for $s \leq 1/2$.

Remark 10. Plugging in $C_n \asymp L_n C_n^* = L_n n^{1-s}$, where C_n^* is the canonical scaling in (28), and simplifying to keep only the dominant terms, we see that (33) becomes

$$R\big(\mathcal{T}^k_{n,d}(C_n^*)\big) = \begin{cases} \Omega(L_n n^{-s}) & \text{if } s \leq 1/2, \\ \Omega\big(L_n^{\frac{2}{2s+1}} n^{-\frac{2s}{2s+1}}\big) & \text{if } s > 1/2. \end{cases}$$

By comparing this to (32), we can see that KTF is minimax rate optimal for estimation over KTV classes, under the canonical scaling, up to log factors.

Remark 11. From the upper bound in (32) and the Holder embedding in (26), we see that for $s \geq 1/2$, KTF achieves the rate of $n^{-2s/(2s+1)}$ (up to log factors) over $\mathcal{H}^k_d(1)$. This matches the optimal rate for estimation over Holder classes (see Sadhanala et al. (2017) for a formal statement and proof for the discretized class $\mathcal{H}^k_d(1)$), which means that KTF automatically adapts Holder smooth signals.

However, when $s < 1/2$, the same results show that KTF achieves a rate of n^{-s} (ignoring log factors) over $\mathcal{H}^k_d(1)$, which is slower than the optimal rate of $n^{-2s/(2s+1)}$. Whether this upper bound is pessimistic and KTF can actually adapt to $\mathcal{H}^k_d(1)$, or whether this upper bound is tight and KTF fails to adapt to Holder smooth signals, remains to be formally resolved. Later, we investigate this empirically in Section 8.3.

Remark 12. It is interesting to compare minimax results for anisotropic Besov spaces, under the white noise model. Past work on this topic includes Neumann (2000) with a focus on hyperbolic wavelets, as well as Kerkyacharian et al. (2001, 2008) with a focus on Lepski's method applied to kernel smoothing. A nice summary of past work, and what appears to be the most comprehensive results, can be found in Lepski (2015). It should be noted that this line of work considers a more general setup than ours (albeit in the white noise model), in a few ways: anisotropic Besov classes with an arbitrary smoothness index in each coordinate direction; error measured in L^p norm, for arbitrary $p \geq 1$; and so on. That said, translating their results to match our setting as best as we can (recalling general embeddings of BV spaces into Besov spaces; for example, DeVore and Lorentz (1993)), we find that the minimax rate under the squared L^2 loss, for the anisotropic Besov class with integrability index 1, smoothness index $k + 1$ in each coordinate direction, and any third index $q \geq 1$, is indeed $n^{-2s/(2s+1)}$ for $s > 1/2$ (or $2k + 2 > d$). This regime is what Lepski and others refer to as the "dense zone".

In what they refer to as the "sparse zone", $s \leq 1/2$, the minimax risk under the L^2 loss for the same Besov class is a constant. This is due to the fact that this Besov space fails to embed compactly into L^2 (see Section 5.5 of Johnstone (2015) for a general discussion of the implications of this phenomenon). If the underlying regression function is itself additionally assumed to be bounded in L^∞ norm, then we believe the minimax rate in their white noise setting will be n^{-s}, matching that in our discrete setting. Evidence of this claim includes the result in del Álamo et al. (2021) for the BV space (under the white noise model), who derive a rate of $n^{-1/d}$, assuming such L^∞ boundedness; as well as the Besov results in Goldenshluger and Lepski (2014), who also assume L^∞ boundedness, but study density estimation.

5.3 Minimax rates for linear smoothers

We now study whether linear smoothers can achieve the minimax rate over the KTV class in (23). Before stating our result, we define a truncated eigenmaps estimator based on $D^{(k+1)}_{n,d}$ as follows. Denote by $\xi_i \geq 0$, $i \in [N]^d$ its singular values (noting that $(k+1)^d$ of these are zero), where along each dimension in the multi-index, the singular values are sorted in increasing order. Denote also by $v_i \in \mathbb{R}^n$, $i \in [N]^d$ its corresponding right singular vectors. Then, for a subset $Q \subseteq [N]^d$, we denote by $V_Q \in \mathbb{R}^{n \times |Q|}$ the matrix with columns given by v_i, $i \in Q$, and define the projection estimator

$$\hat{\theta} = V_Q V_Q^\mathsf{T} y. \tag{34}$$

Note that this reduces to the Laplacian eigenmaps estimator (here the Laplacian is that of the grid graph) when $k = 0$.

Theorem 5. *The minimax linear risk over the KTV class in (23) satisfies, for any sequence $C_n \leq \sqrt{n}$,*

$$R_L\big(\mathcal{T}^k_{n,d}(C_n)\big) = \begin{cases} \Omega(1/n + C_n^2/n) & \text{if } s < 1/2, \\ \Omega(1/n + C_n^2/n \log(1 + n/C_n^2)) & \text{if } s = 1/2, \\ \Omega\big(1/n + (C_n^2/n)^{\frac{1}{2s}}\big) & \text{if } s > 1/2. \end{cases} \tag{35}$$

This is achieved in rate by the projection estimator in (34), where we set $Q = [\tau]^d$ for $\tau^d \asymp (C_n n^{s-1/2})^{1/s}$, in the case $s > 1/2$. When $s < 1/2$, the simple polynomial projection estimator, which projects onto all multivariate polynomials of max degree k (equivalently, the estimator in (34) with $Q = [k+1]^d$), achieves the rate in (35). When $s = 1/2$, either estimator achieves the rate in (35) up to a log factor. Lastly, if $C_n^2 = O(n^\alpha)$ for $\alpha < 1$, and still $s = 1/2$, then either estimator achieves the rate in (35) without the additional log factor.

Remark 13. Plugging in $C_n \asymp L_n C_n^* = L_n n^{1-s}$, where C_n^* is the canonical scaling in (28), and simplifying to keep only the dominant terms (when $L_n \geq 1$), we see that (35) becomes

$$R_L\big(\mathcal{T}^k_{n,d}(C_n^*)\big) = \begin{cases} \Omega(1) & \text{if } s \leq 1/2, \\ \Omega\big(L_n^{\frac{1}{s}} n^{-\frac{2s-1}{2s}}\big) & \text{if } s > 1/2. \end{cases}$$

These minimax linear rates display a stark difference to the minimax rates for the KTV class in Remark 10 (achieved by KTF up to log factors, in (32)). When $s > 1/2$, the minimax linear rate of $L_n^{\frac{1}{s}} n^{-(2s-1)/(2s)}$ can be interpreted as the rate of the optimal nonlinear estimator for a problem whose effective smoothness level has been decremented by a half ($s - 1/2$ in place of s). More dramatically, when $s \leq 1/2$, we see that *no linear smoother is consistent* over the KTV class, in the sense of worst-case risk. This contributes an interesting addition to the line of work on the suboptimality of linear smoothers for nonparametric regression over heterogeneous smoothness classes, dating back to Nemirovski et al. (1985); Donoho and Johnstone (1998).

5.4 Summary of rates

Table 1 presents a summary of the minimax rates derived in the previous three subsections. (It offers a more detailed summary than Figure 2.) The upper bound on minimax risk is from Corollary 1, the lower bound on the minimax risk from Theorem 4 and Remark 10, and the minimax linear risk is from Theorem 5 and Remark 13.

Regime	R, upper bound	R, lower bound	R_L, linear risk
$s < 1/2$	$n^{-s}\sqrt{\log n}$	n^{-s}	1
$s = 1/2$	$n^{-\frac{1}{2}} \log n$	$n^{-\frac{1}{2}}$	1
$s > 1/2$	$n^{-\frac{2s}{2s+1}} (\log n)^{\frac{1}{2s+1}}$	$n^{-\frac{2s}{2s+1}}$	$n^{-\frac{2s-1}{2s}}$

Table 1: *Minimax rates over the KTV class $\mathcal{T}^k_{n,d}(C_n^*)$, where the canonical scaling is $C_n^* = n^{1-s}$, and recall $s = (k+1)/d$. We use the abbreviations $R = R(\mathcal{T}^k_{n,d}(C_n^*))$ and $R_L = R_L(\mathcal{T}^k_{n,d}(C_n^*))$.*

5.5 Minimax rates over Sobolev classes

For completeness, we establish the minimax rate for the (discrete) Sobolev class defined in (24). The lower bounds are simply those from the Holder class (due to the embedding in Proposition 3), and as we show next, this is achieved in rate by the eigenmaps estimator in (34).

Theorem 6. *The minimax risk over the (discrete) Sobolev class in (24) satisfies, for any sequence $B_n \leq \sqrt{n}$,*

$$R\big(\mathcal{W}^{k+1}_{n,d}(B_n)\big) \asymp \frac{1}{n} + \left(\frac{B_n^2}{n}\right)^{\frac{1}{2s+1}}.$$

The lower bound is due to the Holder embedding in (26) (and the lower bound on the discretized Holder class derived in Sadhanala et al. (2017)), and the upper bound is from the estimator in (34), with $Q = [\tau]^d$ for $\tau^d \asymp (B_n^2 n^{2s})^{1/(2s+1)}$. Finally, when $B_n \asymp L_n B_n^*$ where B_n^* is the canonical scaling in (27), the minimax rate is $L_n^{2/(2s+1)} n^{-2s/(2s+1)}$.

6 Optimization algorithms

In this section, we describe a number of numerical algorithms for solving the convex KTF problem (7), analyze their asymptotic time complexity, and benchmark their performance. In particular, we will describe a family of specialized ADMM algorithms that adapt to the structure of the KTF problem, and as we will show, can find moderately accurate solutions much faster than a general purpose "off-the-shelf" solver (this could be applied to the dual of (7), which is a simple box-constrained quadratic program, and is amenable to standard interior point methods).

6.1 Specialized ADMM algorithms

Motivated by the popularity of operating splitting methods in machine learning over the last decade, and specifically by their success in application to trend filtering and TV denoising problems (for example, see Ramdas and Tibshirani (2016); Wang et al. (2016); Barbero and Sra (2018)), we consider the application of similar methods to Kronecker trend filtering. We consider a proximal Dykstra algorithm, Douglas-Rachford splitting, and a family of specialized ADMM algorithms. For brevity, the details on the former two are deferred to Appendix E.

Our specialized ADMM approach is inspired by that of Ramdas and Tibshirani (2016), for univariate trend filtering. To reformulate (7) into "ADMM form", where the criterion decomposes as a sum of functions of separate optimization variables, we must introduce auxiliary variables. To do so, we rely on the following observation that decomposes the KTF penalty operator into the product of a block-diagonal matrix and a lower-order KTF penalty operator.

Proposition 4. *For each $j = 1, 2, \ldots, k+1$, the KTF penalty operator in (8) obeys (where $D_N^{(0)} = I_N$ for convenience):*

$$D_{n,d}^{(k+1)} = \underbrace{\begin{bmatrix} D_N^{(k+1-j)} \otimes I_N \otimes \cdots \otimes I_N & & & \\ & I_N \otimes D_N^{(k+1-j)} \otimes \cdots \otimes I_N & & \\ & & \ddots & \\ & & & I_N \otimes I_N \otimes \cdots \otimes D_N^{(k+1-j)} \end{bmatrix}}_{M_{n,d}^{(k+1-j)}} D_{n,d}^{(j)}.$$

This follows directly from the the univariate recursion in (4), and the Kronecker structure in (8) (particularly, the mixed-product property of Kronecker products), and therefore we omit its proof. Observe that each diagonal block of the block-diagonal matrix $M_{n,d}^{(k+1-j)}$ is itself—possibly after appropriate permutation of the row and column order—a block-diagonal matrix with all diagonal blocks equal to $D_N^{(k+1-j)}$. This is a key fact that we will leverage shortly.

Now fix any $j \in \{1, \ldots, k+1\}$. We can reformulate (7) as:

$$\begin{aligned} \underset{\theta, z}{\text{minimize}} \quad & \frac{1}{2}\|y - \theta\|_2^2 + \lambda \|M_{n,d}^{(k+1-j)} z\|_1 \\ \text{subject to} \quad & z = D_{n,d}^{(j)} \theta. \end{aligned} \tag{36}$$

The augmented Lagrangian associated with (36), for an augmented Lagrangian parameter $\rho \geq 0$, is:

$$L_\rho(\theta, z, u) = \frac{1}{2}\|y - \theta\|_2^2 + \lambda \|M_{n,d}^{(k+1-j)} z\|_1 + \frac{\rho}{2}\|z - D_{n,d}^{(j)}\theta + u\|_2^2 - \frac{\rho}{2}\|u\|_2^2.$$

ADMM iteratively performs a separate minimization over the primal variables θ, z, and then updates the dual variable u via gradient ascent. Namely, given some initialization $\theta^{(0)}, z^{(0)}, u^{(0)}$, it repeats the following, for $t = 1, 2, 3, \ldots$:

$$\theta^{(t)} = \Big(I_n + \rho [D_{n,d}^{(j)}]^\mathsf{T} D_{n,d}^{(j)}\Big)^{-1} \Big(y + \rho [D_{n,d}^{(j)}]^\mathsf{T} (z^{(t-1)} + u^{(t-1)})\Big), \tag{37}$$

$$z^{(t)} = \mathsf{prox}_{\frac{\lambda}{\rho}\|M_{n,d}^{(k+1-j)}(\cdot)\|_1}\Big(D_{n,d}^{(j)}\theta^{(t)} - u^{(t-1)}\Big), \tag{38}$$

$$u^{(t)} = u^{(t-1)} + z^{(t)} - D_{n,d}^{(j)}\theta^{(t)}. \tag{39}$$

Here we use the notation $\mathsf{prox}_h(\cdot)$ for the proximal operator associated with a function h. Below we make a few remarks about the computational costs associated with the updates (37)–(39).

- When $j = 1$, the matrix $[D^{(j)}_{n,d}]^\mathsf{T} D^{(j)}_{n,d}$ in (37) is the graph Laplacian of the d-dimensional grid, which decomposes into the Kronecker sum of Laplacians of (univariate) chain graphs. This can be diagonalized by a d-dimensional discrete cosine transform (DCT) (see, for example, the proof of Corollary 8 in Wang et al. (2016)). Computationally, this is essentially sequentially applying univariate DCTs to every dimension. This implies that the θ-update in (37) can be done in $O(n \log n)$ time. Further improvements (to linear-time) should be possible with multi-grid methods.

- By the key fact mentioned after the proposition, the z-update in (38) can be decomposed into dN^{d-1} univariate trend filtering problems, each of order $k + 1 - j$ and each with sample size N. These can be solved in parallel, in a total time that is nearly-linear in n, using either the univariate ADMM approach of Ramdas and Tibshirani (2016) or the primal-dual interior point method (PDIP) of Kim et al. (2009). PDIP is likely the best option for reasonably small N, and using it, the update (38) can be done in $O(dN^{d-1}N^{1.5}) = O(n^{1+1/(2d)})$ time.

- When $j = k$ or $k + 1$, the z-update (38) can be performed even more efficiently, in $O(n)$ time. This is because it reduces to soft-thresholding for $j = k + 1$, and reduces to separate univariate TV denoising problems for $j = k$. In the former case, the linear time complexity is obvious; in the latter, it is due to the dynamic programming (DP) method of Johnson (2013).

- The case $k = 0$ is quite favorable, as we can use DCT in (37) and soft-thresholding in (38), by choosing $j = 1$, or simple coordinatewise shrinkage in (37) and DP in (38), by choosing $j = 0$. Both are efficient, but the latter ends up being generally the better approach, and can be seen as the ADMM-analog of Barbero and Sra (2018).

- Among the higher-order cases $k \geq 1$, the case $k = 1$ ends up being quite special, because $j = 1$ simultaneously supports the DCT solver in (37) and DP in (38). When $k \geq 2$, we essentially need to decide in between these highly efficient subroutines (choosing either $j = 1$ or $j = k$).

In summary, each iteration (cycle of updates over θ, z, u) of the proposed ADMM algorithm is $O(n)$ for $k = 0, 1$. For $k \geq 2$, the time complexity is $O(n^{1+1/(2d)})$ when we choose $j = 1$, which we refer to as ADMM *Type I*. When we choose $j = k$, which we refer to as ADMM *Type II*, the time complexity is dominated by the sparse linear system solve in (37). This linear system should be well-conditioned for reasonable ranges of ρ, thus the standard conjugate gradient method will be able to solve it in approximately linear-time (proportional to the number of nonzero elements).

6.2 Empirical comparisons

We now compare the ADMM algorithms developed in the last subsection to Douglas-Rachford and proximal Dykstra algorithms applied to (7), as well as the Gurobi general purpose solver (free for academic use) applied to the dual of (7).[5] From the family of specialized ADMM algorithms, we pay particular attention to ADMM Types I and II, which correspond to $j = 1$ and $j = k$, respectively. We also consider $j = 0$, which we ADMM *Type 0*, mainly because it is closely related and should perform similarly to the Douglas-Rachford and proximal Dykstra methods. In all ADMM algorithms, we adopt an adaptive choice of ρ that balances the primal and dual suboptimality (Boyd et al., 2011).

To ensure a fair comparison between ADMM Types I and II, which we will see are generally the best performing methods (and thus the comparison between them is of particular interest), we use optimized C++ implementations for each of their prox subroutines; for Type I, this is the DP algorithm for univariate TV denoising, and for Type II, this is the PDIP algorithm for univariate trend filtering; and in both cases, we use C++ implementations from Ramdas and Tibshirani (2016). Aside from specialized subroutines, the implementation of all iterative algorithms is in MATLAB.

The results are presented in Figures 3 and 4. Figure 3 compares the operator splitting algorithms for denoising the standard "Lena" method at a resolution of 256×256. The KTF orders are taken to be $k = 1, 2, 3$, corresponding to the columns in the figure. For each k, the solution returned by Gurobi is used to define the optimal criterion value, which is is then used to measure the suboptimality gap of solutions returned the iterative methods. ADMM Type I is generally the winner in all cases, whether measured by iteration or (especially) by wall-clock time.

Figure 4 compares ADMM Type I to Gurobi for varying k, and also for varying resolutions of the underlying Lena image. In all cases, ADMM Type I obtains a moderate-quality solution in less one second—which is sometimes two orders of magnitude faster than the off-the-shelf solver provided by Gurobi. It should be further noted that Gurobi is

[5] We add tiny amount of regularization to the dual problem to avoid numerical issues that cause Gurobi to fail. The solution of this regularized problem is first used to reconstruct the primal solution, but then evaluated on the objective function of the original problem, when computing the suboptimality gaps.

highly-optimized, whereas our ADMM Type I implementation is not—recall, only the prox subroutine is optimized, and the outer looping is performed in MATLAB. Transporting the entire algorithm to C++ would clearly yield further improvements in efficiency. Of course, if a truly high-accuracy solution is required, then Gurobi may be the best option. However, its strong performance in this subsection suggests that ADMM Type I is an efficient and useful approach for many applications in statistics and machine learning, where moderate-accuracy solutions suffice.

7 Interpolation algorithm

In this section, we derive an algorithm to extend the KTF solution in (7), defined only at points in the lattice $Z_{n,d}$, to a function defined on all of $[0,1]^d$. As discussed in Remark 3, this is made possible by the continuous-time formulation for KTF in (12) of Proposition 2, which, recall, relates to the original discrete-time problem (7) in that at their solutions $\hat{f}, \hat{\theta}$, respectively, we have $\hat{f}(x_i) = \theta_i$, for $i = 1, \dots, n$. Given the coefficients that define the function $\hat{f} \in \mathcal{H}^k_{n,d}$ in its expansion the tensor product basis of univariate falling factorial functions, that is, given the solution $\hat{\alpha}$ in (13), we can form the interpolated prediction $\hat{f}(x)$ at an arbitrary point $x \in [0,1]^d$ by simply evaluating this basis expansion at x, as shown in (14).

While this is conceptually the easiest way to interpolate the fitted values $\hat{f}(x_i)$, $i = 1, \dots, n$ to form the prediction at an arbitrary $x \in [0,1]^d$, it is not the most efficient. The expression in (14) takes $O(\|\hat{\alpha}\|_0 (k+1)^d)$ operations, where $\|\hat{\alpha}\|_0$ denotes the number of nonzero elements in $\hat{\alpha}$ (the number of active basis functions), because evaluating each univariate basis function $h_{i_j}(x_j)$ takes $O(k+1)$ operations (it being a product of $k+1$ terms, recall (10)). In what follows, we present an interpolation algorithm that takes only $O((k+1)^{d+1})$ operations, a big savings over (14) when $\|\hat{\alpha}\|_0$ is large. Moreover, our algorithm acts directly on the solution $\hat{\theta}$ in (7), meaning that we never have to solve (13) in the first place.

7.1 Review: univariate interpolation

Our interpolation algorithm for KTF in the multivariate case builds from the univariate discrete spline interpolation algorithm derived in Corollary 2 of Tibshirani (2020). For completeness, we transcribe this in Algorithm 1. Here and henceforth, we use the abbreviation $x_{a:b} = (x_a, \dots, x_b)$ for integers $a \leq b$. Also, we use $f[z_1, \dots, z_r]$ for the divided difference of a function f at distinct points $z_1, \dots, z_r$. Recall, this is defined for $r = 2$ by

$$f[z_1, z_2] = \frac{f(z_2) - f(z_1)}{z_2 - z_1},$$

and for any $r \geq 3$ by the recursion

$$f[z_1, \dots, z_r] = \frac{f[z_2, \dots, z_r] - f[z_1, \dots, z_{r-1}]}{z_r - z_1}.$$

Note that for evenly-spaced points, this simply coincides with a scaled forward difference; in particular, recalling the notation introduced in Section 1.1, we have

$$f[z, \dots, z + (r-1)/n] = \frac{(r-1)!}{n^r} (\Delta^r f)(z).$$

For more background on divided differences, discrete splines, their connection to the falling factorial basis and to trend filtering, we refer to Tibshirani (2020).

Algorithm 1 takes $O((k+1)^2)$ operations, as (40), (41) are each linear systems in just one unknown, and forming the coefficients in either linear system can be done in $O((k+1)^2)$ operations (this uses a representation of a divided difference as an explicit linear combination of the underlying function evaluations; refer to, for example, Section 2.1 of Tibshirani (2020)). Note that this assumes $x_{1:n}$ are evenly-spaced design points, because in this case identifying the smallest index i such that $x_i > x$ can be done with integer division. For general design points, the identification step takes $O(\log n)$ operations via binary search, so the total cost would be $O(\log n + (k+1)^2)$.

Moreover, Corollary 2 in Tibshirani (2020) establishes that the value $f(x)$ returned by Algorithm 1 is equal to that produced by the falling factorial basis representation in (11) (for $d = 1$), where α is the unique coefficient vector such that $f(x_i) = \theta_i$, $i = 1, \dots, n$.

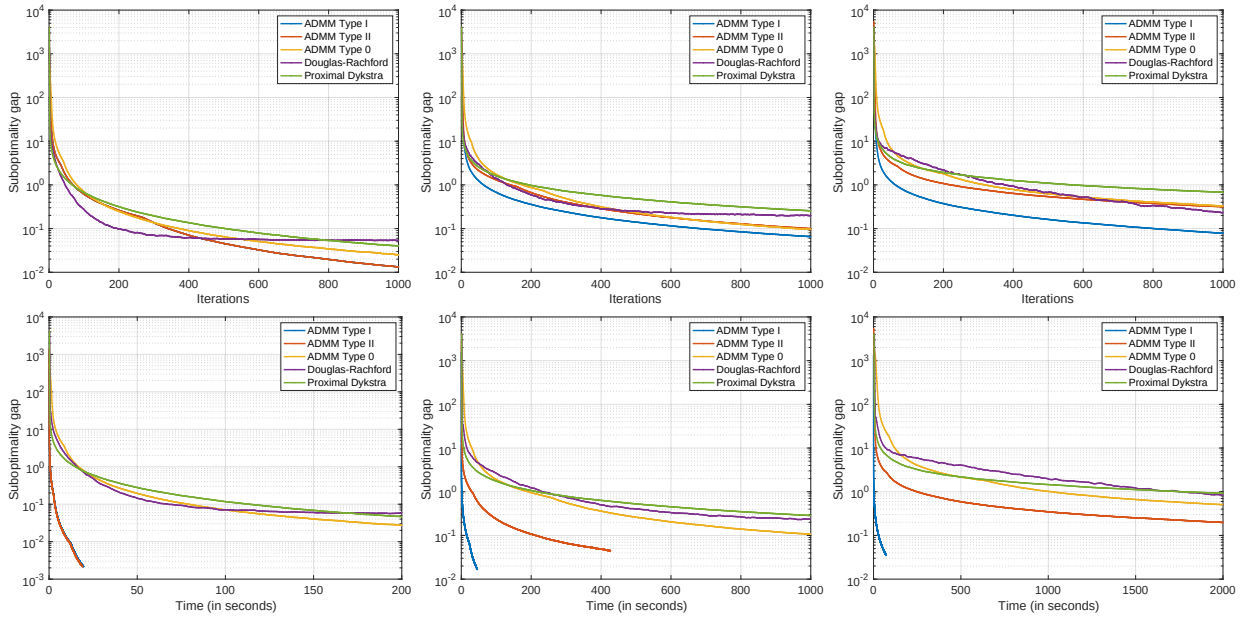

Figure 3: *Comparison of iterative algorithms for KTF on the standard Lena image of resolution* 256×256 *(that is,* $n = 65536$*), when* $k = 1, 2, 3$*, corresponding to the three columns, from left to right. The top row compares the convergence of the suboptimality gap as a function of the number of iterations. The bottom row shows the same but parametrized by wall-clock time in seconds. While these methods have similar sublinear convergence rates (top row), ADMM Types I and II are clearly the fastest (bottom row) to reach a small suboptimality gap, due to their low per-iteration cost. Type I is the overall winner. (Recall that when* $k = 1$*, Types I and II coincide, so the blue curve is hidden behind the red curve).*

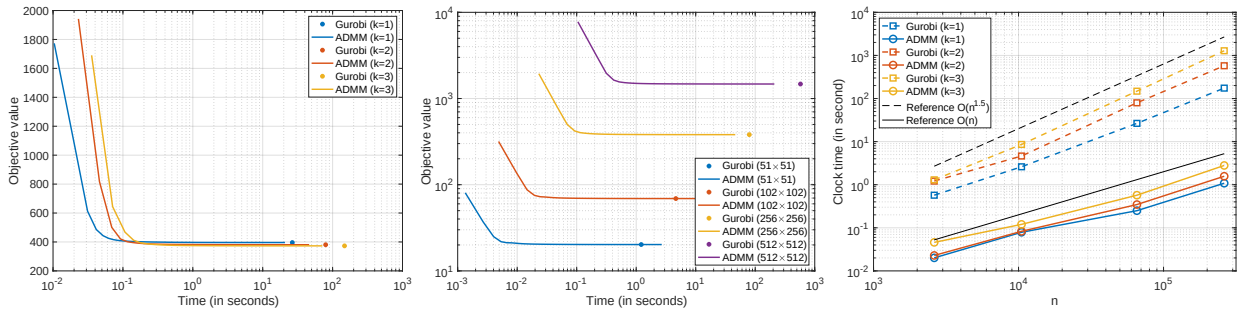

Figure 4: *Comparison of ADMM Type I to Gurobi for the same Lena problem. The left panel compares the two methods for varying* k *at a fixed resolution of* 256×256*. The middle panel fixes* $k = 2$ *and compares them across varying resolutions. The right panel plots the time needed to achieve a certain relative error, defined by the ratio of the suboptimality gap to the objective value being less than* 10^{-2}*. Altogether, we see that ADMM Type I achieves a moderate-quality solution several orders of magnitude faster than Gurobi. Furthermore, the right panel shows that Gurobi appears to scale as* $O(n^{1.5})$ *(to be expected, if it is based on interior point methods internally) whereas ADMM Type I appears to scale closer to* $O(n)$*.*

Algorithm 1 INTERPOLATE-1D$(x_{1:n}, \theta_{1:n}, x, k)$

Input: design points $x_{1:n}$ with entries in increasing order; values $\theta_{1:n}$ to interpolate; query point x; integer $k \geq 0$.
Output: interpolated value $f(x)$, where f is the unique k^{th} order discrete spline with knots in $x_{(k+1):(n-1)}$, such that $f(x_i) = \theta_i$, $i = 1, \dots, n$.

1. If $x = x_i$ for some $i = 1, \dots, n$, then return θ_i.
2. Else, if $x > x_{k+1}$ and i is the smallest index such that $x_i > x$ (with $i = n$ when $x > x_n$), then return $f(x)$ as the unique solution of the linear system:
$$f[x_{i-k}, \dots, x_i, x] = 0. \tag{40}$$
3. Else, if $x < x_{k+1}$, then return $f(x)$ as the unique solution of the linear system:
$$f[x_1, \dots, x_{k+1}, x] = 0. \tag{41}$$

(Note that both (40), (41) are linear systems in just one unknown, $f(x)$, since we interpret $f(x_i) = \theta_i$, $i = 1, \dots, n$.)

Algorithm 2 INTERPOLATE$\big(\{z_{i1}\}_{i=1}^{N_1} \times \cdots \times \{z_{id}\}_{i=1}^{N_d},\ \{\theta_i\}_{i \in [N_1] \times \cdots \times [N_d]},\ x,\ k\big)$

Input: lattice $\{z_{i1}\}_{i=1}^{N_1} \times \cdots \times \{z_{id}\}_{i=1}^{N_d}$ where each set $\{z_{ij}\}_{i=1}^{N_j}$ in the Cartesian product is sorted in increasing order; values $\{\theta_i\}_{i \in [N_1] \times \cdots \times [N_d]}$ over the lattice to interpolate; query point x; integer $k \geq 0$.
Output: interpolated value $f(x)$, for the unique function f in the tensor product space of k^{th} degree discrete splines with knots in $\{z_{i1}\}_{i=k+1}^{N_1-1} \times \cdots \times \{z_{id}\}_{i=k+1}^{N_d-1}$, such that $f(z_{i_1,1}, \dots, z_{i_d,d}) = \theta_{i_1,\dots,i_d}$, $(i_1, \dots, i_d) \in [N_1] \times \cdots \times [N_d]$.

1. If $d = 1$, then return INTERPOLATE-1D$(z_{1:N_1}, \theta_{1:N_1}, x, k)$.
2. Else, let i_1 denote the smallest index such that $x_{i_1,1} \geq z_{i_1,1}$.
3. Let $\ell_1 = \min\{\max\{i_1 - k, 1\}, N_1 - k\}$.
4. Let $\vartheta_p = \text{INTERPOLATE}\big(\{z_{i2}\}_{i=1}^{N_2} \times \cdots \times \{z_{id}\}_{i=1}^{N_d},\ \{\theta_i\}_{i \in \{i_1+p-1\} \times [N_2] \times \cdots \times [N_d]},\ x_{2:d},\ k\big)$, for $p \in [k+1]$.
5. Return INTERPOLATE-1D$(z_{\ell_1:(\ell_1+k),1}, \vartheta_{1:(k+1)}, x_1, k)$.

7.2 Multivariate interpolation

In the multivariate case, it turns out that we can interpolate within the space of tensor products of k^{th} degree discrete splines in $O((k+1)^{d+1})$ time, assuming a uniformly-spaced lattice. The idea is to reduce the d-dimensional problem calculation to $k+1$ interpolation problems, each one in dimension $d-1$. Figure 5 gives the intuition for $d = 2$. The algorithm in described in Algorithm 2, where we recall the notation from Section 5, abbreviating $[a] = \{1, \dots, a\}$ for an integer $a \geq 1$.

Algorithm 2 assumes each side length of the lattice is at least $k+1$. Its proof of correctness, as well as the running time of $O((k+1)^{d+1})$ for a uniformly-spaced lattice, follows from a straightforward inductive argument over d, whose proof we omit. We note that for an arbitrary lattice, the running time is $O(\sum_{j=1}^{d} \log N_j + (k+1)^{d+1})$. Figures 5 and 6 give an illustration and examples when $d = 2$.

8 Experiments

We explore the empirical properties of KTF through a series of experiments that compare its performance against other nonparametric methods.

8.1 Comparison of methods

We study the performance of KTF against other nonparametric estimators by analyzing their empirical mean squared error (MSE) in estimating the function f_0 defined in Figure 1, over a square lattice with $n = 256^2 = 65536$ points. In the experiments that follow, we generate data by adding Gaussian noise to evaluations of f_0, of the same magnitude illustrated in Figure 1, yielding a signal-to-noise ratio (SNR) of 0.5. Aside from KTF and graph trend filtering (GTF) of orders $k = 0, 1, 2$ (for $k = 0$, they both reduce to TV denoising), we also consider second-order Laplacian smoothing (using the squared Laplacian L^2, where L is the Laplacian matrix of the 2d grid graph), the eigenmaps estimator in (34)

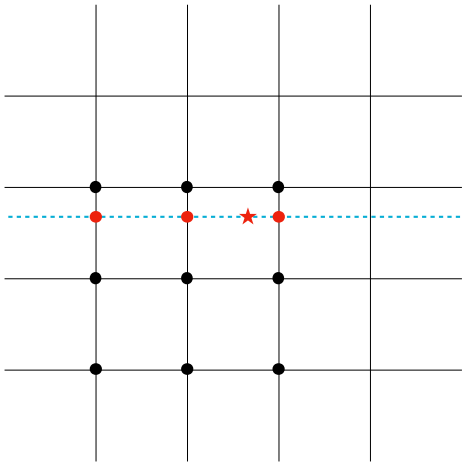

Figure 5: *Illustration of multivariate interpolation from Algorithm 2, when $d = 2$ and $k = 2$. The value to be interpolated is marked by a red star. The algorithm first interpolates the $k + 1 = 3$ values indicated by red dots, each time using univariate interpolation along the y-axis, from Algorithm 1. As the final step, these red dots are used to interpolate the value at the red star, using univariate interpolation along the x-axis, again from Algorithm 1.*

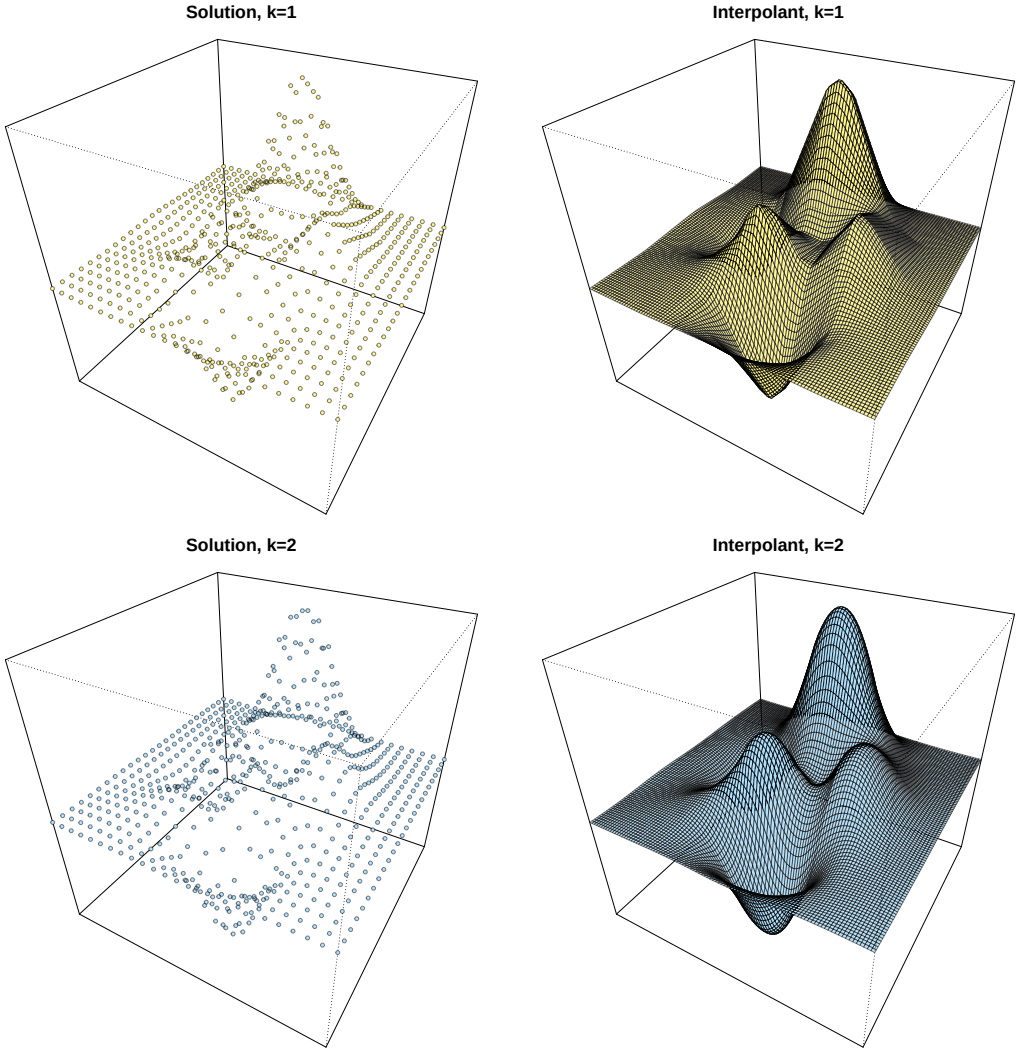

Figure 6: *Top left: KTF solution, with $k = 1$, for a problem over a square lattice with side length $N = 25$ (that is, $n = 25^2 = 625$ points). Top right: the interpolated surface from Algorithm 2 (itself evaluated over a grid with 4x the resolution along each dimension; that is, $N = 100$). Bottom row: analogous but for $k = 2$. In either case the interpolated function is in the tensor space of k^{th} order discrete splines.*

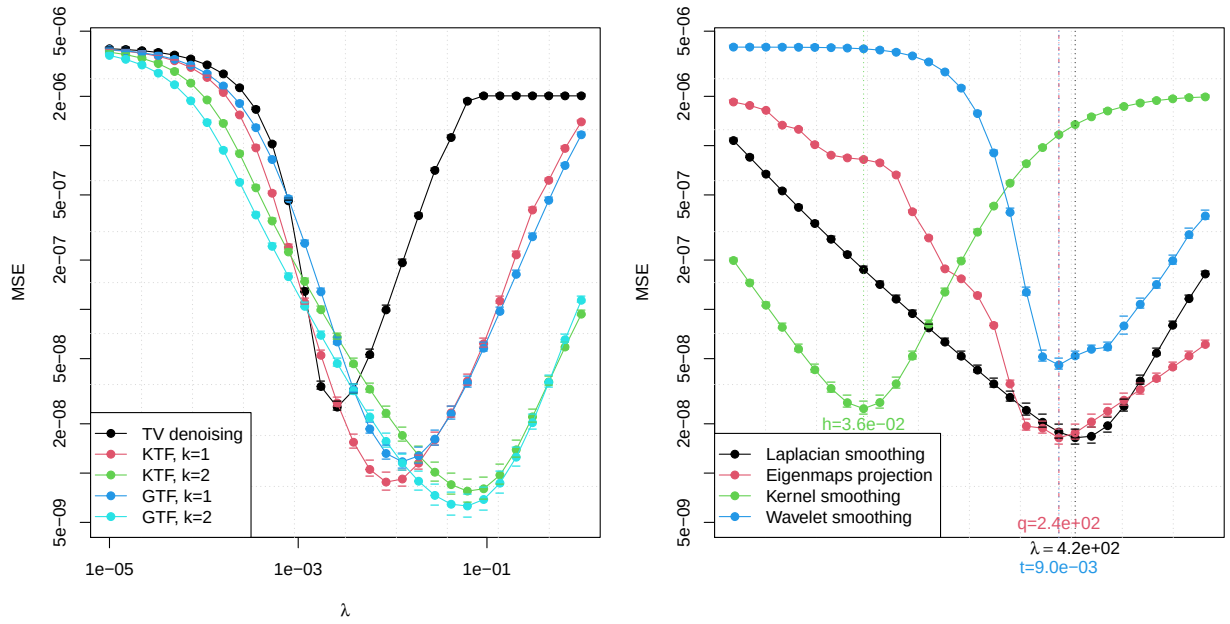

Figure 7: *MSE of various methods for estimating the signal function f_0 in Figure 1, over a square lattice with $n = 256^2 = 65536$ points, and subject to Gaussian noise that yields an SNR of 0.5. Left panel: the average performance (over 20 repetitions) of trend filtering estimators over a common range of tuning parameters λ. Right panel: the average performance of other nonparametric methods. As their tuning parameters lie on different scales, their values are scaled to fit onto a single x-axis. The takeaway is that KTF and GTF, in particular for $k = 2$, achieve clearly the best performance.*

with $k = 1$, kernel smoothing (using a Gaussian kernel), and wavelet smoothing (using Daubechies' least asymmetric wavelets, ten vanishing moments, and hard thresholding). For the latter two estimators, we use the implementations from the R packages `np` and `wavethresh`, respectively.

Each estimator is fit over a range of tuning parameters, and its MSE as a function of the tuning parameter is shown in Figure 7. The MSE curves here are averaged over 20 repetitions (each repetition forms a data set by adding noise to f_0). Error bars are shown as well, denoting the standard deviations of the MSE over these 20 repetitions. We can see that the trend filtering estimators (with the exception of TV denoising, which returns a piecewise constant fit that is not well-suited to an underlying signal with this level of smoothness) perform quite a bit better than all competitors, and factoring in the variability over repetitions, KTF and GTF achieve essentially equivalent performance (for common values of k). The superiority of KTF over the linear estimators (Laplacian smoothing, eigenmaps projection, and kernel smoothing), should not come as a surprise—our theory prescribes that KTF should perform better in a minimax sense than any linear smoother when the underlying signal displays heterogeneous smoothness. These results thus serve as a quantitative complement to the qualitative findings in Figure 1, and to the theoretical findings in Section 5.

8.2 Rates for heterogeneous signals

We now examine the empirical error rates of KTF and a linear smoother in estimating signals belonging to KTV classes, for $k = 0, 1$. In both cases, we choose k, d so that the effective degree of smoothness remains $s = 1/2$, and we use the canonical scaling in (28), so that the two classes under consideration are:

$$\mathcal{T}^0_{n,2}(\sqrt{n}) \quad \text{and} \quad \mathcal{T}^1_{n,4}(\sqrt{n}). \tag{42}$$

As representatives for "hard" signals within these two classes, we take the true mean θ_0 to be a "one-hot" signal when $k = 0$, and a linear "spike" signal when $k = 1$ (each scaled to the appropriate magnitude). For each sample size n, we compute the MSE of KTF, and the eigenmaps projection estimator in (34) with $k = 0, 1$, averaged over 20 repetitions, where each method is tuned to have the optimal average MSE over a range of tuning parameter values.

Figure 8 shows these average MSE curves as functions of n, where the error bars again denote standard deviations. We can see that in both cases, the KTF error decays faster than the minimax rate (suggesting that the particular signals

under consideration do not achieve the worst-case rate for KTF), while the linear method exhibits a perfectly flat error curve, suggesting that it fails to be consistent entirely.

8.3 Rates for homogeneous signals

Lastly, we examine the empirical error rates of KTF and the eigenmaps projector for estimating homogeneously smooth signals. Recall we established that the latter is minimax rate optimal over Sobolev (and Holder) classes in Theorem 6, whereas for $s < 1/2$, it is not clear (from its minimax rate over the inscribing KTV class) that KTF achieves the faster rate for Sobolev (or Holder) signals. We fix $k = 0$ and consider two values for the dimension, $d = 2$ and $d = 3$, which correspond to $s = 1/2$ and $s < 1/2$, respectively. The true mean θ_0 was taken to be the evaluations of a linear function over the lattice, scaled appropriately (according to the canonical scaling). Figure 9 reports the MSE curves from each method, under the same general setup as in the last subsection (averaged over 20 repetitions, optimal tuning per n). Moving from $s = 1/2$ to $s < 1/2$, we do not find that the empirical performance of TV denoising becomes markedly worse than its linear competitor, and the adaptivity of KTF to the smooth signals remains an open question.

9 Discussion

We proposed and studied a method, Kronecker trend filtering, that extends univariate trend filtering to the multivariate setting, where the design points lie on a uniformly-spaced lattice. We derived a continuous-time representation for the optimization problem defining the KTF estimator, which led to a method for rapid interpolation to off-lattice locations. We also established a comprehensive set of minimax results which show that KTF is rate optimal over heterogeneous classes of signals defined in terms of higher-order TV regularity, whereas linear estimators fail to be optimal (and even consistent, in some cases) over such classes. Lastly, we presented fast and specialized ADMM methods for computing KTF solutions, which were shown to perform favorably against numerous alternatives.

There are numerous possible directions for future work. We finish by describing five such directions.

9.1 General lattice structures

All along we have assumed a uniformly-spaced lattice with equal side lengths, but much of this paper carries over to a more general lattice structure, namely a Cartesian product $\{z_{i1}\}_{i=1}^{N_1} \times \{z_{i2}\}_{i=1}^{N_2} \times \cdots \times \{z_{id}\}_{i=1}^{N_d}$, with $n = \prod_{j=1}^{d} N_j$. The KTF penalty operator over this general lattice becomes:

$$D^{(k+1)}_{z_1,\ldots,z_d} = \begin{bmatrix} D^{(k+1)}_{z_1} \otimes I_{N_2} \otimes \cdots \otimes I_{N_d} \\ I_{N_1} \otimes D^{(k+1)}_{z_2} \otimes \cdots \otimes I_{N_d} \\ \vdots \\ I_{N_1} \otimes I_{N_2} \otimes \cdots \otimes D^{(k+1)}_{z_d} \end{bmatrix}, \tag{43}$$

where for each $j = 1, \ldots, d$, we abbreviate $z_j = \{z_{ij}\}_{i=1}^{N_j}$, and denote by $D^{(k+1)}_{z_j}$ the k^{th} order univariate trend filtering penalty matrix defined with respect to z_j, which (assuming the points in z_j are sorted in increasing order) is given for $k = 1$ by $D^{(1)}_{z_j} = D^{(1)}_{N_j}$, and for all other k by:

$$D^{(k+1)}_{z_j} = D^{(1)}_{N_j-k} \cdot \operatorname{diag}\left(\frac{k}{z_{k+1,j} - z_{1j}}, \frac{k}{z_{k+2,j} - z_{2j}}, \ldots, \frac{k}{z_{N_j,j} - z_{N_j-k,j}} \right) \cdot D^{(k)}_{z_j}, \quad k = 1, 2, 3, \ldots.$$

Many of the properties and results derived in this paper carry over more or less immediately to the KTF estimator on a general lattice defined using the penalty matrix (43). All properties in Section 2 carry over with suitable adjustments, as do the specialized ADMM algorithms in Section 6, and the fast interpolation routine in Section 7 (Algorithm 2 is in fact already written assuming a general lattice structure). Extending the theory in Section 5 is certainly less trivial, though we expect that this should be possible under mild assumptions on the spacings between points in z_j, $j = 1, \ldots, d$.

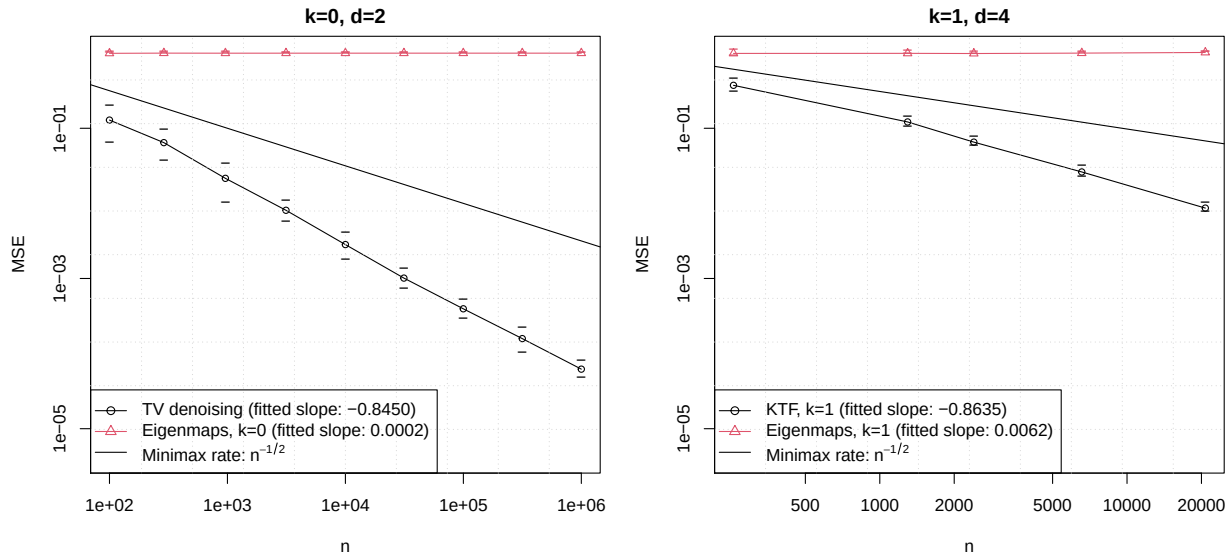

Figure 8: *Empirical error rates of KTF versus the eigenmaps projection estimator, for two cases:* $k = 0$, $d = 2$, *and* $k = 1$, $d = 4$. *The true mean in each case was chosen to be representative of a "hard" signal in the appropriate KTV class in* (42). *KTF converges faster than the minimax rate, whereas the eigenmaps estimator fails to be consistent entirely.*

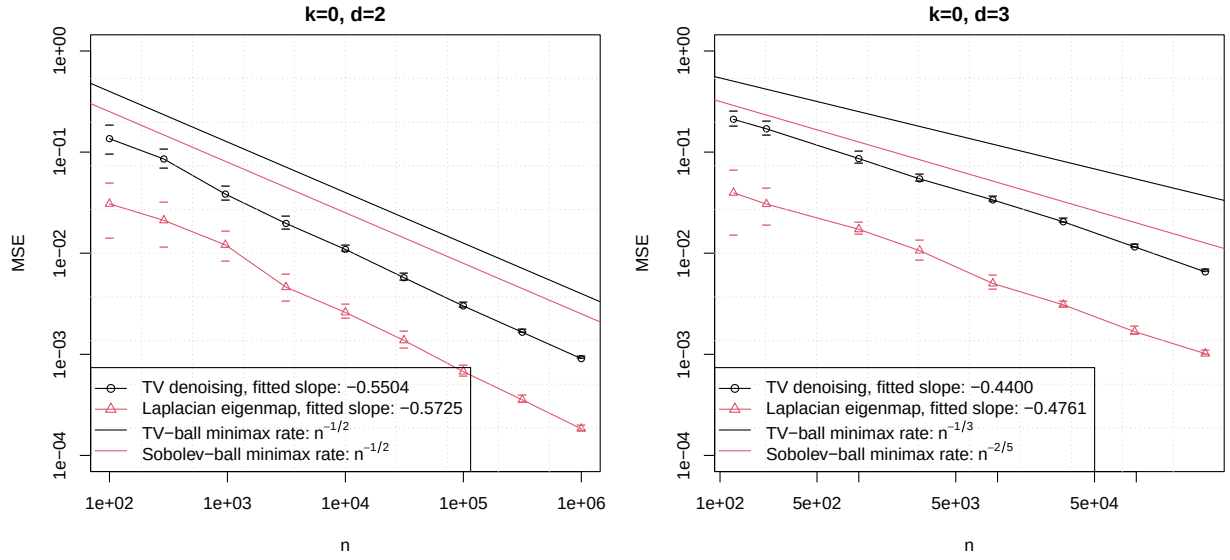

Figure 9: *Empirical error rates of KTF versus the eigenmaps projection estimator, for two cases:* $k = 0$, $d = 2$, *and* $k = 0$, $d = 3$. *The true mean was chosen to be a linear function in either case. KTF does not appear to be outperformed (in rate) by the eigenmaps estimator, leaving its adaptivity to smooth signals when* $s < 1/2$ *an open question.*

9.2 Generalized linear models

The KTF problem may be extended by replacing the squared loss in (7) with a generalized linear model (GLM) loss:

$$\underset{\theta \in \mathbb{R}^n}{\text{minimize}} \; -\sum_{i=1}^n T(y_i)\theta_i + \psi(\theta) + \lambda \| D_{n,d}^{(k+1)} \theta \|_1, \tag{44}$$

where $\psi : \mathbb{R}^n \to \mathbb{R}$ is convex and twice differentiable. The loss in (44) corresponds to the negative log-likelihood of an exponential family model for y_i, $i = 1, \dots, n$, with log-partition function ψ, natural sufficient statistic T, and natural parameters θ_i, $i = 1, \dots, n$. The squared loss in (7) corresponds to the working model where each y_i is Gaussian with mean θ_i and a common variance. Under a working model where each y_i is Poisson with mean θ_i, the GLM problem (44) becomes:

$$\underset{\theta \in \mathbb{R}^n}{\text{minimize}} \; \sum_{i=1}^n \big(-y_i\theta_i + \exp(\theta_i) \big) + \lambda \| D_{n,d}^{(k+1)} \theta \|_1. \tag{45}$$

Not far afield from this is an approach for density estimation, in which we take data supported on a bounded domain in $\mathbb{R}^d$, discretize the domain using a lattice of our choosing, and then bin the data to form counts—so that now each y_i represents the count in a Poisson model for bin i (lattice point x_i). This is closely related to Padilla and Scott (2016); Bassett and Sharpnack (2019) who study similar ideas in different contexts.

Generally, we note that the properties in Section 2, as well as the interpolation algorithm in Section 7, all carry over directly to (44), since they are driven entirely by the structure of the KTF penalty (which remains unchanged in (44)). The optimization algorithms from Section 6 could be applied to (44), as the inner loop of a proximal Newton method (where the outler loop makes a weighted quadratic approximation to the loss in (44)). Meanwhile, estimation theory will be more difficult to extend, though for upper bounds, simple truncation arguments (as in, for example, Lin et al. (2017)) may be a viable approach.

9.3 Mixed discrete derivatives

The KTF penalty operator computes discrete derivatives of order $k+1$ along each coordinate direction. This can be extended by considering *mixed* discrete derivatives of total order $k+1$; in the notation of Section 1.2, this is

$$\sum_{|\alpha|=k+1} \sum_{x \in Z_{n,d}} \big| \big(\Delta_{x_1^{\alpha_1}, \dots, x_d^{\alpha_d}} \theta\big)(x) \big|,$$

where the outer sum is over all multi-indices $\alpha \in \mathbb{R}_+^d$ of size $|\alpha| = \alpha_1 + \cdots + \alpha_d = k+1$. The analog of the penalty operator in (8) is now:

$$\tilde{D}_{n,d}^{(k+1)} = \begin{bmatrix} D_N^{(\alpha_1^1)} \otimes D_N^{(\alpha_2^1)} \otimes \cdots D_N^{(\alpha_d^1)} \\ \vdots \\ D_N^{(\alpha_1^P)} \otimes D_N^{(\alpha_2^P)} \otimes \cdots D_N^{(\alpha_d^P)} \end{bmatrix}, \tag{46}$$

where $P = \binom{k+d}{d-1}$, and $\alpha^p \in \{0, \dots, k+1\}^d$, $p = 1, \dots, P$ enumerate all multi-indices of size $k+1$. (Recall that we use $D_N^{(0)} = I_N$ by convention.) Compared to (8), the mixed penalty operator in (46) has many more rows, which leads to computational difficulty even for moderate k, d. However, we have found KTF with mixed derivatives to work well in certain problems, likely explained by the fact that it is is less anisotropic (bringing it closer, in a sense to the GTF penalty operator), and thus can work well when some degree of isotropy is warranted.

We note that the null space of (46) has dimension $\binom{k+d}{d}$ and consists of evaluations of all polynomials of degree k, which provides the analog of Proposition 1. However, a continuous-time representation as in Proposition 2 is likely not possible, and extensions of results in the rest of the paper would be highly nontrivial, and a topic for future work.

9.4 Coordinate-specific smoothness

Proceeding in some sense in an opposite direction to the extension in the last section, we can also generalize KTF to make it *more* anisotropic, by allowing the degree of modeled smoothness to differ for each coordinate. In the notation

of Section 1.2, the extended penalty would be

$$\sum_{j=1}^{d} \sum_{x \in Z_{n,d}} |(\Delta_{x_j^{k_j+1}} \theta)(x)|.$$

where $k_j \geq 0$ are (possibly) distinct integers, for $j = 1, \ldots, d$, and the analogous penalty operator would be

$$\tilde{D}_{n,d}^{(k_1,\ldots,k_d)} = \begin{bmatrix} D_N^{(k_1)} \otimes I_N \otimes \cdots \otimes I_N \\ I_N \otimes D_N^{(k_2)} \otimes \cdots \otimes I_N \\ \vdots \\ I_N \otimes I_N \otimes \cdots \otimes D_N^{(k_d)} \end{bmatrix}. \tag{47}$$

The properties and results derived Sections 2, 6, and 7 translate with straightforward modification (in many instances, merely notational) to the generalized KTD estimator with the penalty operator as in (47). With some work, we expect much of the estimation theory in Section 5 to carry over as well, where we conjecture that the appropriate notation of effective smoothness will be $s = (\sum_{j=1}^{d} 1/(k_j+1))^{-1}$, by analogy to minimax theory on anisotropic Besov spaces.

9.5 Scattered data

Saving the most ambitious extension for last, the continuous-time representation of KTF from Section 2.3, specifically the basis form in (13), suggests an approach for extending KTF to scattered data. For arbitrary design points $x_i \in \mathbb{R}^d$, $i = 1, \ldots, n$, we form any lattice of our choosing $z_1 \times \cdots \times z_d$, where $z_j = \{z_{ij}\}_{i=1}^{N_j}$, $j = 1, \ldots, d$ and $m = \prod_{j=1}^{d} N_j$. We then define for each $i = 1, \ldots, n$ and $j = 1, \ldots, d$:

$$h_{x_i,z_j} = \big(h^k_{z_j,1}(x_{ij}), h^k_{z_j,2}(x_{ij}), \ldots, h^k_{z_j,N_j}(x_{ij})\big) \in \mathbb{R}^{N_j},$$

where $h^k_{z_j,\ell}$, $\ell = 1, \ldots, N_j$ are the falling factorial basis functions defined over the knot set z_j (see Tibshirani (2020) for the general definition). The extended basis form of KTF for scattered data is now:

$$\underset{\alpha \in \mathbb{R}^m}{\text{minimize}} \; \frac{1}{2} \left\| y - \begin{bmatrix} h_{x_1,z_1} \otimes h_{x_1,z_2} \otimes \cdots \otimes h_{x_1,z_d} \\ h_{x_2,z_1} \otimes h_{x_2,z_2} \otimes \cdots \otimes h_{x_2,z_d} \\ \vdots \\ h_{x_n,z_1} \otimes h_{x_3,z_2} \otimes \cdots \otimes h_{x_n,z_d} \end{bmatrix} \alpha \right\|_2^2 + \lambda \left\| D^{(k+1)}_{z_1,\ldots,z_d} \left[H^{(k)}_{z_1} \otimes H^{(k)}_{z_2} \otimes \cdots H^{(k)}_{z_d} \right] \alpha \right\|_1. \tag{48}$$

Here $D^{(k+1)}_{z_1,\ldots,z_d}$ denotes the KTF penalty operator over the lattice $z_1 \times \cdots \times z_d$, as defined in (43), and each $H^{(k)}_{z_j}$ is the k^{th} order univariate falling factorial basis matrix defined with respect to z_j, for $j = 1, \ldots, d$. A solution $\hat{\alpha}$ in (48) for the coefficients in the basis expansion can be used to form fitted values as well as predictions at arbitrary $x \in \mathbb{R}^d$.

Acknowledgements

The authors thank James Sharpnack and Alden Green for numerous insightful discussions. This material is based upon work supported by the National Science Foundation under grant DMS-1554123 and Graduate Research Fellowship Program award DGE1745016.

References

Andrés Almansa, Coloma Ballester, Vicent Caselles, and Gloria Haro. A TV based restoration model with local constraints. *Journal of Scientific Computing*, 34(3):209–236, 2008.

Alvaro Barbero and Suvrit Sra. Modular proximal optimization for multidimensional total-variation regularization. *Journal of Machine Learning Research*, 19(56):1–82, 2018.

Robert Bassett and James Sharpnack. Fused density estimation: Theory and methods. *Journal of Royal Statistical Society: Series B*, 81(5):839–860, 2019.

Aurélien F. Bibaut and Mark J. van der Laan. Fast rates for empirical risk minimization over càdlàg functions with bounded sectional variation norm. arXiv: 1907.09244, 2019.

Lucien Birge and Pascal Massart. Gaussian model selection. *Journal of the European Mathematical Society*, 3(3): 203–268, 2001.

Steve Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternative direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1): 1–122, 2011.

Emmanuel J. Candès and Franck Guo. New multiscale transforms, minimum total variation synthesis: Applications to edge-preserving image reconstruction. *Signal Processing*, 82(11):1519–1543, 2002.

Antonin Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1):89–97, 2004.

Antonin Chambolle. Total variation minimization and a class of binary MRF models. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 136–152. Springer, 2005.

Antonin Chambolle and Pierre-Louis Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188, 1997.

Tony Chan, Antonio Marquina, and Pep Mulet. High-order total variation-based image restoration. *SIAM Journal on Scientific Computing*, 22(2):503–516, 2000.

Tony F. Chan and Selim Esedoglu. Aspects of total variation regularized L^1 function approximation. *SIAM Journal on Applied Mathematics*, 65(5):1817–1837, 2005.

Sabyasachi Chatterjee and Subhajit Goswami. Adaptive estimation of multivariate piecewise polynomials and bounded variation functions by optimal decision trees. *Annals of Statistics*, 49(5):2531–2551, 2021a.

Sabyasachi Chatterjee and Subhajit Goswami. New risk bounds for 2d total variation denoising. *IEEE Transactions on Information Theory*, 67(6):4060–4091, 2021b.

Charles K. Chui, Jeffrey M. Lemm, and Sahra Sedigh. *An Introduction to Wavelets*. Academic Press, 1992a.

Charles K. Chui, Joachim Stöckler, and Joseph D. Ward. Compactly supported box-spline wavelets. *Approximation Theory and its Applications*, 8(3):77–100, 1992b.

Arnak S. Dalalyan, Mohamed Hebiri, and Johannes Lederer. On the prediction performance of the Lasso. *Bernoulli*, 23(1):552–581, 2017.

Ingrid Daubechies. *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, 1992.

Miguel del Álamo, Housen Li, and Axel Munk. Frame-constrained total variation regularization for white noise regression. *Annals of Statistics*, 49(3), 2021.

Ronald DeVore and George Lorentz. *Constructive Approximation*. Springer, 1993.

Ronald A. DeVore and Bradley J. Lucier. Wavelets. *Acta Numerica*, 1:1–56, 1992.

Ronald A. DeVore, Sergei V. Konyagin, and Vladimir N. Temlyakov. Hyperbolic wavelet approximation. *Constructive Approximation*, 14(1):1–26, 1998.

Yiqiu Dong, Michael Hintermüller, and M. Monserrat Rincon-Camacho. Automated regularization parameter selection in multi-scale total variation models for image restoration. *Journal of Mathematical Imaging and Vision*, 40(1): 82–104, 2011.

David L. Donoho. CART and best-ortho-basis: a connection. *Annals of Statistics*, 25(5):1870–1911, 1997.

David L. Donoho and Iain M. Johnstone. Minimax estimation via wavelet shrinkage. *Annals of Statistics*, 26(8): 879–921, 1998.

Bradley Efron. How biased is the apparent error rate of a prediction rule? *Journal of the American Statistical Association*, 81(394):461–470, 1986.

Lawrence C. Evans and Ronald F. Gariepy. *Measure Theory and Fine Properties of Functions*. CRC Press, 2015. Revised edition.

Billy Fang, Adityanand Guntuboyina, and Bodhisattva Sen. Multivariate extensions of isotonic regression and total variation denoising via entire monotonicity and Hardy-Krause variation. *Annals of Statistics*, 49(2):769–792, 2021.

Matan Gavish, Boaz Nadler, and Ronald Coifman. Multiscale wavelets on trees, graphs and high dimensional data: theory and applications to semi supervised learning. In *Proceedings of the Annual Conference on Learning Theory*, 2010.

Franziska Göbel, Gilles Blanchard, and Ulrike von Luxburg. Construction of tight frames on graphs and application to denoising. In *Handbook of Big Data Analytics*, pages 503–522. Springer, 2018.

Alexander Goldenshluger and Oleg Lepski. Universal pointwise selection rule in multivariate function estimation. *Bernoulli*, 14(4):1150–1190, 2008.

Alexander Goldenshluger and Oleg Lepski. Structural adaptation via L_p-norm oracle inequalities. *Probability Theory and Related Fields*, 143(1–2):41–71, 2009.

Alexander Goldenshluger and Oleg Lepski. Bandwidth selection in kernel density estimation: Oracle inequalities and adaptive minimax optimality. *Annals of Statistics*, 39(3):1608–1632, 2011.

Alexander Goldenshluger and Oleg Lepski. General selection rule from a family of linear estimators. *Theory of Probability & Its Applications*, 57(2):209–226, 2013.

Alexander Goldenshluger and Oleg Lepski. On adaptive minimax density estimation on R^d. *Probability Theory and Related Fields*, 159(3):479–543, 2014.

Adityanand Guntuboyina, Donovan Lieu, Sabyasachi Chatterjee, and Bodhisattva Sen. Adaptive risk bounds in univariate total variation denoising and trend filtering. *Annals of Statistics*, 48(1):205–209, 2020.

Trevor Hastie and Robert Tibshirani. *Generalized Additive Models*. Chapman & Hall, 1990.

Jan-Christian Hutter and Philippe Rigollet. Optimal rates for total variation denoising. In *Proceedings of the Annual Conference on Learning Theory*, 2016.

Nicholas Johnson. A dynamic programming algorithm for the fused lasso and l_0-segmentation. *Journal of Computational and Graphical Statistics*, 22(2):246–260, 2013.

Iain M. Johnstone. *Gaussian Estimation: Sequence and Wavelet Models*. Cambridge University Press, 2015. Draft version.

Gérard Kerkyacharian, Oleg V. Lepski, and Dominique Picard. Nonlinear estimation in anisotropic multi-index denoising. *Probability Theory and Related Fields*, 121(2):137–170, 2001.

Gérard Kerkyacharian, Oleg V. Lepski, and Dominique Picard. Nonlinear estimation in anisotropic multi-index denoising. Sparse case. *Theory of Probability & Its Applications*, 52(1):58–77, 2008.

Dohyeong Ki, Billy Fang, and Adityanand Guntuboyina. MARS via LASSO. arXiv: 2111.11694, 2021.

Seung-Jean Kim, Kwangmoo Koh, Stephen Boyd, and Dimitry Gorinevsky. ℓ_1 trend filtering. *SIAM Review*, 51(2): 339–360, 2009.

Roger Koenker, Pin Ng, and Stephen Portnoy. Quantile smoothing splines. *Biometrika*, 81(4):673–680, 1994.

Aleksandr P. Korostelev and Alexandre B. Tsybakov. *Minimax Theory of Image Reconstructions*. Springer, 2003.

Oleg V. Lepski. Adaptive estimation over anisotropic functional classes via oracle approach. *Annals of Statistics*, 43(3): 1178–1242, 2015.

Oleg V. Lepski and Vladimir G. Spokoiny. Optimal pointwise adaptive methods in nonparametric estimation. *Annals of Statistics*, 25(6):2512–2546, 1997.

Oleg V. Lepski, Enno Mammen, and Vladimir G. Spokoiny. Optimal spatial adaptation to inhomogeneous smoothness: An approach based on kernel estimates with variable bandwidth selectors. *Annals of Statistics*, 25(3):929–947, 1997.

Oleg V. Lepskii. On a problem of adaptive estimation in Gaussian white noise. *Theory of Probability & Its Applications*, 35(3):454–466, 1991.

Oleg V. Lepskii. Asymptotically minimax adaptive estimation. I: Upper bounds. Optimally adaptive estimates. *Theory of Probability & Its Applications*, 36(4):682–697, 1992.

Oleg V. Lepskii. Asymptotically minimax adaptive estimation. II. Schemes without optimal adaptation: Adaptive estimators. *Theory of Probability & Its Applications*, 37(3):433–448, 1993.

Kevin Lin, James Sharpnack, Alessandro Rinaldo, and Ryan J. Tibshirani. A sharp error analysis for the fused lasso, with application to approximate changepoint screening. In *Advances in Neural Information Processing Systems*, 2017.

Rudolph A. H. Lorentz and Wolodymyr R. Madych. Wavelets and generalized box splines. *Applicable Analysis*, 44 (1–2):51–76, 1992.

Stephane Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 2009. Third edition.

Stephane G. Mallat. Multiresolution approximations and wavelet orthonormal bases of $L^2(R)$. *Transactions of the American Mathematical Society*, 315(1):69–87, 1989a.

Stephane G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989b.

Enno Mammen and Sara van de Geer. Locally apadtive regression splines. *Annals of Statistics*, 25(1):387–413, 1997.

Yves Meyer. Principe d'incertitude, bases Hilbertiennes et algebres d'operateurs. *Séminaire Bourbaki*, 145–146(662): 209–223, 1987.

Yves Meyer. *Ondelettes et Opérateurs*. Hermann, 1990.

Yves Meyer and Sylvie Roques. *Progress in Wavelet Analysis and Applications*. Atlantica Séguier Frontières, 1993.

Arkadi S. Nemirovski, Boris T. Polyak, and Alexandre B. Tsybakov. Signal processing by the nonparametric maximum likelihood. *Problems on Information Transmission*, 20(3):29–46, 1984.

Arkadi S. Nemirovski, Boris T. Polyak, and Alexandre B. Tsybakov. Rate of convergence of nonparametric estimates of maximum-likelihood type. *Problems on Information Transmission*, 21(4):17–33, 1985.

Michael H. Neumann. Multivariate wavelet thresholding in anisotropic function spaces. *Statistica Sinica*, 10(2): 399–431, 2000.

Michael H. Neumann and Rainer von Sachs. Wavelet thresholding in anisotropic function classes and application to adaptive estimation of evolutionary spectra. *Annals of Statistics*, 25(1):38–76, 1997.

Francesco Ortelli and Sara van de Geer. Prediction bounds for higher order total variation regularized least squares. *Annals of Statistics*, 49(5), 2021a.

Francesco Ortelli and Sara van de Geer. Tensor denoising with trend filtering. arXiv: 2101.10692, 2021b.

Oscar Hernan Madrid Padilla and James G. Scott. Nonparametric density estimation by histogram trend filtering. arXiv: 1509.04348, 2016.

Oscar Hernan Madrid Padilla, James Sharpnack, James G. Scott, and Ryan J. Tibshirani. The DFS fused lasso: Linear-time denoising over general graphs. *Journal of Machine Learning Research*, 18:176–1, 2018.

Oscar Hernan Madrid Padilla, James Sharpnack, Yanzhen Chen, and Daniela Witten. Adaptive non-parametric regression with the k-nn fused lasso. *Biometrika*, 107(2):293–310, 2020.

Aaditya Ramdas and Ryan J. Tibshirani. Fast and flexible ADMM algorithms for trend filtering. *Journal of Computational and Graphical Statistics*, 25(3):839–858, 2016.

Sherman D. Riemenschneider and Zuowei Shen. Wavelets and pre-wavelets in low dimensions. *Journal of Approximation Theory*, 71(1):18–38, 1992.

Leonid I. Rudin and Stanley Osher. Total variation based image restoration with free local constraints. In *Proceedings of the International Conference on Image Processing*, pages 31–35, 1994.

Leonid I. Rudin, Stanley Osher, and Emad Faterni. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

Veeranjaneyulu Sadhanala. *Nonparametric Methods with Total Variation Type Regularization*. PhD thesis, Machine Learning Department, Carnegie Mellon University, 2019.

Veeranjaneyulu Sadhanala and Ryan J. Tibshirani. Additive models via trend filtering. *Annals of Statistics*, 47(6): 3032–3068, 2019.

Veeranjaneyulu Sadhanala, Yu-Xiang Wang, and Ryan J. Tibshirani. Total variation classes beyond 1d: Minimax rates, and the limitations of linear smoothers. In *Advances in Neural Information Processing Systems*, 2016.

Veeranjaneyulu Sadhanala, Yu-Xiang Wang, James Sharpnack, and Ryan J. Tibshirani. Higher-total variation classes on grids: Minimax theory and trend filtering methods. In *Advances in Neural Information Processing Systems*, 2017.

James Sharpnack, Aarti Singh, and Akshay Krishnamurthy. Detecting activations over graphs using spanning tree wavelet bases. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2013.

Gabriel Steidl, Stephan Didas, and Julia Neumann. Splines in higher order TV regularization. *International Journal of Computer Vision*, 70(3):214–255, 2006.

Charles Stein. Estimation of the mean of a multivariate normal distribution. *Annals of Statistics*, 9(6):1135–1151, 1981.

Ryan J. Tibshirani. Adaptive piecewise polynomial estimation via trend filtering. *Annals of Statistics*, 42(1):285–323, 2014.

Ryan J. Tibshirani. Divided differences, falling factorials, and discrete splines: Another look at trend filtering and related problems. arXiv: 2003.03886, 2020.

Ryan J. Tibshirani and Jonathan Taylor. The solution path of the generalized lasso. *Annals of Statistics*, 39(3): 1335–1371, 2011.

Ryan J. Tibshirani and Jonathan Taylor. Degrees of freedom in lasso problems. *Annals of Statistics*, 40(2):1198–1232, 2012.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer, 2009.

Curtis R. Vogel and M. E. Oman. Iterative methods for total variation denoising. *SIAM Journal on Scientific Computing*, 17(1):227–238, 1996.

Yu-Xiang Wang, Alexander Smola, and Ryan J. Tibshirani. The falling factorial basis and its statistical applications. In *Proceedings of the International Conference on Machine Learning*, 2014.

Yu-Xiang Wang, James Sharpnack, Alex Smola, and Ryan J. Tibshirani. Trend filtering on graphs. *Journal of Machine Learning Research*, 17(105):1–41, 2016.

Steven Siwei Ye and Oscar Hernan Madrid Padilla. Non-parametric quantile regression via the k-nn fused lasso. *Journal of Machine Learning Research*, 22(111):1–38, 2021.