Adaptive Piecewise Polynomial Estimation via Trend Filtering

Ryan J. Tibshirani

Abstract

We study trend filtering, a recently proposed tool of Kim et al. (2009) for nonparametric regression. The trend filtering estimate is defined as the minimizer of a penalized least squares criterion, in which the penalty term sums the absolute kth order discrete derivatives over the input points. Perhaps not surprisingly, trend filtering estimates appear to have the structure of kth degree spline functions, with adaptively chosen knot points (we say "appear" here as trend filtering estimates are not really functions over continuous domains, and are only defined over the discrete set of inputs). This brings to mind comparisons to other nonparametric regression tools that also produce adaptive splines; in particular, we compare trend filtering to smoothing splines, which penalize the sum of squared derivatives across input points, and to locally adaptive regression splines (Mammen & van de Geer 1997), which penalize the total variation of the kth derivative. Empirically, we discover that trend filtering estimates adapt to the local level of smoothness much better than smoothing splines, and further, they exhibit a remarkable similarity to locally adaptive regression splines. We also provide theoretical support for these empirical findings; most notably, we prove that (with the right choice of tuning parameter) the trend filtering estimate converges to the true underlying function at the minimax rate for functions whose kth derivative is of bounded variation. This is done via an asymptotic pairing of trend filtering and locally adaptive regression splines, which have already been shown to converge at the minimax rate (Mammen & van de Geer 1997). At the core of this argument is a new result tying together the fitted values of two lasso problems that share the same outcome vector, but have different predictor matrices.

Keywords: *trend filtering, nonparametric regression, smoothing splines, locally adaptive regression splines, minimax convergence rate, lasso stability*

1 Introduction

Per the usual setup in nonparametric regression, we assume that we have observations $y_1, \ldots y_n \in \mathbb{R}$ from the model

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

where $x_1, \ldots x_n \in \mathbb{R}$ are input points, f_0 is the underlying function to be estimated, and $\epsilon_1, \ldots \epsilon_n$ are independent errors. For most of this work, we will further assume that the inputs are evenly spaced over the interval $[0, 1]$, i.e., $x_i = i/n$ for $i = 1 \ldots n$. (In Section 6, we relax this assumption.) The literature on nonparametric regression is rich and diverse, and there are many methods for estimating f_0 given observations from the model (1); some well-known examples include methods based on local polynomials, kernels, splines, sieves, and wavelets.

This paper focuses on a relative newcomer in nonparametric regression: trend filtering, proposed by Kim et al. (2009). For a given integer $k \geq 0$, the kth order trend filtering estimate $\hat{\beta} = (\hat{\beta}_1, \ldots \hat{\beta}_n)$ of $(f_0(x_1), \ldots f_0(x_n))$ is defined by a penalized least squares optimization problem,

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^n}{\operatorname{argmin}} \ \frac{1}{2}\|y - \beta\|_2^2 + \frac{n^k}{k!}\lambda \|D^{(k+1)}\beta\|_1, \tag{2}$$

where $\lambda \geq 0$ is a tuning parameter, and $D^{(k+1)} \in \mathbb{R}^{(n-k-1)\times n}$ is the discrete difference operator of order $k+1$. (The constant factor $n^k/k!$ multiplying λ is unimportant, and can be absorbed into the tuning parameter λ, but it will facilitate comparisons in future sections.) When $k = 0$,

$$D^{(1)} = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & \dots & 0 & 0 \\ \vdots & & & & & \\ 0 & 0 & 0 & \dots & -1 & 1 \end{bmatrix} \in \mathbb{R}^{(n-1)\times n}, \tag{3}$$

and so $\|D^{(1)}\beta\|_1 = \sum_{i=1}^{n-1} = |\beta_i - \beta_{i+1}|$. Hence the 0th order trend filtering problem, which we will also call constant trend filtering, is the same as 1-dimensional total variation denoising (Rudin et al. 1992), or the 1-dimensional fused lasso (Tibshirani et al. 2005) (with pure fusion penalty, i.e., without an additional ℓ_1 penalty on the coefficients themselves). In this case, $k = 0$, the components of the trend filtering estimate form a piecewise constant structure, with break points corresponding to the nonzero entries of $D^{(1)}\hat{\beta} = (\hat{\beta}_2 - \hat{\beta}_1, \dots \hat{\beta}_n - \hat{\beta}_{n-1})$. See Figure 1 for an example.

For $k \geq 1$, the operator $D^{(k+1)} \in \mathbb{R}^{(n-k-1)\times n}$ is most easily defined recursively, as in

$$D^{(k+1)} = D^{(1)} \cdot D^{(k)}. \tag{4}$$

[Above, $D^{(1)}$ is the $(n-k-1) \times (n-k)$ version of the 1st order discrete difference operator (3)]. In words, the definition (4) says that the $(k+1)$st order difference operator is built up by evaluating differences of differences, a total of $k+1$ times. Therefore, the matrix $D^{(k+1)}$ can be thought of as the discrete analogy to the $(k+1)$st order derivative operator, and the penalty term in (2) penalizes the discrete $(k+1)$st derivative of the vector $\beta \in \mathbb{R}^n$, i.e., the changes in the discrete kth derivative of β. Accordingly, one might expect the components of the kth order trend filtering estimate to exhibit the structure of a piecewise polynomial of order k—e.g., for 1st order trend filtering, the estimate would be piecewise linear, for 2nd order, it would be piecewise quadratic, etc. Figure 1 gives empirical evidence towards this claim. Later, in Section 4, we provide a more definitive confirmation of this piecewise polynomial structure when we examine a continuous-time representation for trend filtering.

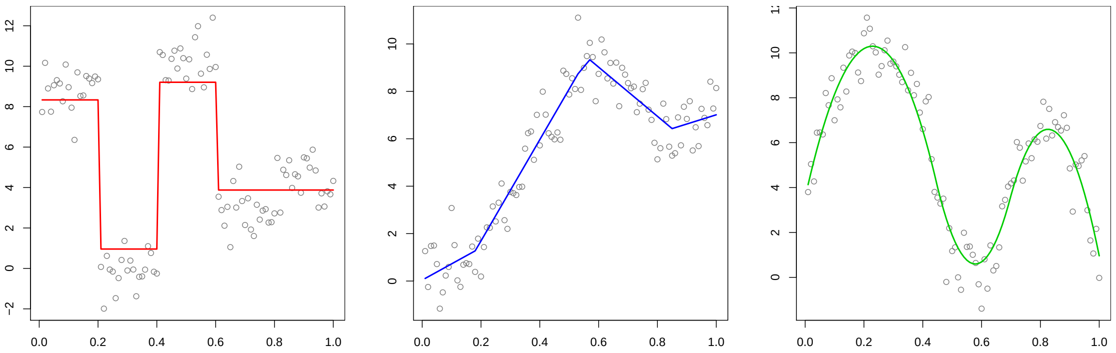

Figure 1: *Simple examples of trend filtering for constant, linear, and quadratic orders ($k = 0, 1, 2$, respectively), shown from left to right. Although the trend filtering estimates are only defined at the discrete inputs $x_i = i/n$, $i = 1, \dots n$, we use linear interpolation to extend the estimates over* $[0, 1]$ *for visualization purposes (this is the default for all figures in this paper).*

It is straightforward to check that

$$D^{(2)} = \begin{bmatrix} 1 & -2 & 1 & 0 & \dots & 0 \\ 0 & 1 & -2 & 1 & \dots & 0 \\ 0 & 0 & 1 & -2 & \dots & 0 \\ \vdots & & & & & \end{bmatrix}, \quad D^{(3)} = \begin{bmatrix} -1 & 3 & -3 & 1 & \dots & 0 \\ 0 & -1 & 3 & -3 & \dots & 0 \\ 0 & 0 & -1 & 3 & \dots & 0 \\ \vdots & & & & & \end{bmatrix},$$

and in general, the nonzero elements in each row of $D^{(k)}$ are given by the $(k+1)$st row of Pascal's triangle, but with alternating signs. A more explicit (but also more complicated-looking) expression for the kth order trend filtering problem is therefore

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^n}{\operatorname{argmin}} \; \frac{1}{2} \sum_{i=1}^{n} (y_i - \beta_i)^2 + \frac{n^k}{k!} \lambda \sum_{i=1}^{n-k-1} \left| \sum_{j=i}^{i+k+1} (-1)^{j-i} \binom{k+1}{j-i} \beta_j \right| .$$

The penalty term above sums over successive linear combinations of $k+2$ adjacent coefficients, i.e., the discrete difference operator $D^{(k+1)}$ is a banded matrix with bandwidth $k+2$.

1.1 The generalized lasso and related properties

For any order $k \geq 0$, the trend filtering estimate $\hat{\beta}$ is uniquely defined, because the criterion in (2) is strictly convex. Furthermore, the trend filtering criterion is of generalized lasso form with an identity predictor matrix $X = I$ (this is called the signal approximator case) and a specific choice of penalty matrix $D = D^{(k+1)}$. Some properties of the trend filtering estimate therefore follow from known results on the generalized lasso (Tibshirani & Taylor 2011, 2012), e.g., an exact representation of the trend filtering estimate in terms of its active set and signs, and also, a formula for its degrees of freedom:

$$\mathrm{df}(\hat{\beta}) = \mathbb{E}[\text{number of knots in } \hat{\beta}] + k + 1, \tag{5}$$

where the number of knots in $\hat{\beta}$ is interpreted to mean the number of nonzero entries in $D^{(k+1)}\hat{\beta}$ (the basis and continuous-time representations of trend filtering, in Sections 3.3 and 4, provide a justification for this interpretation). To repeat some of the discussion in Tibshirani & Taylor (2011), Tibshirani & Taylor (2012), the result in (5) may seem somewhat remarkable, as a fixed-knot kth degree regression spline with d knots also has $d+k+1$ degrees of freedom—and trend filtering does not employ fixed knots, but rather, chooses them adaptively. So why does trend filtering not have a larger degrees of freedom? At a high level, the answer lies in the shrinkage due to the ℓ_1 penalty in (2): the nonzero entries of $D^{(k+1)}\hat{\beta}$ are shrunken toward zero, compared to the same quantity for the corresponding equality-constrained least squares estimate. In other words, within each interval defined by the (adaptively chosen) knots, trend filtering fits a kth degree polynomial whose kth derivative is shrunken toward its kth derivatives in neighboring intervals, when compared to a kth degree regression spline with the same knots. Figure 2 gives a demonstration of this phenomenon for $k = 1$ and $k = 3$.

In terms of algorithms, the fact that the discrete difference operator $D^{(k+1)}$ is banded is of great advantage for solving the generalized lasso problem in (2). Kim et al. (2009) describe a primal-dual interior point method for solving (2) at a fixed value of λ, wherein each iteration essentially reduces to solving a linear system in $D^{(k+1)}(D^{(k+1)})^T$, costing $O(n)$ operations. In the worst case, this algorithm requires $O(n^{1/2})$ iterations, so its complexity is $O(n^{3/2})$.[1] [Kim et al. (2009) focus mainly on linear trend filtering, the case $k = 1$, but their arguments carry over to the general case as well.] On the other hand, instead of solving (2) at a fixed λ, Tibshirani & Taylor (2011) describe a path algorithm to solve (2) over all values of $\lambda \in [0, \infty)$, i.e., to compute the entire solution path $\hat{\beta} = \hat{\beta}(\lambda)$ over λ. This path is piecewise linear as a function of λ [not to be confused with the estimate itself at any fixed λ, which has a piecewise polynomial structure over the input points $x_1, \ldots x_n$]. Again, the bandedness of $D^{(k+1)}$ is key here for efficient computations, and Tibshirani & Arnold (2013) describe

[1] It should be noted that hidden in the $O(\cdot)$ notation here is a factor depending on the prespecified error tolerance ϵ, namely, a term of the form $\log(1/\epsilon)$. We emphasize here that the primal-dual interior point method is a different type of algorithm than the path algorithm, in the sense that the latter returns an exact solution up to computer precision, whereas the former returns an ϵ-suboptimal solution, as measured by the difference in its achieved criterion value and the optimal criterion value. Essentially all general purpose convex optimization techniques (that are applicable to the trend filtering problem) fall into the same class as the primal-dual interior point method, i.e., they return ϵ-suboptimal solutions; only specialized techniques like the path algorithm can deliver exact solutions.

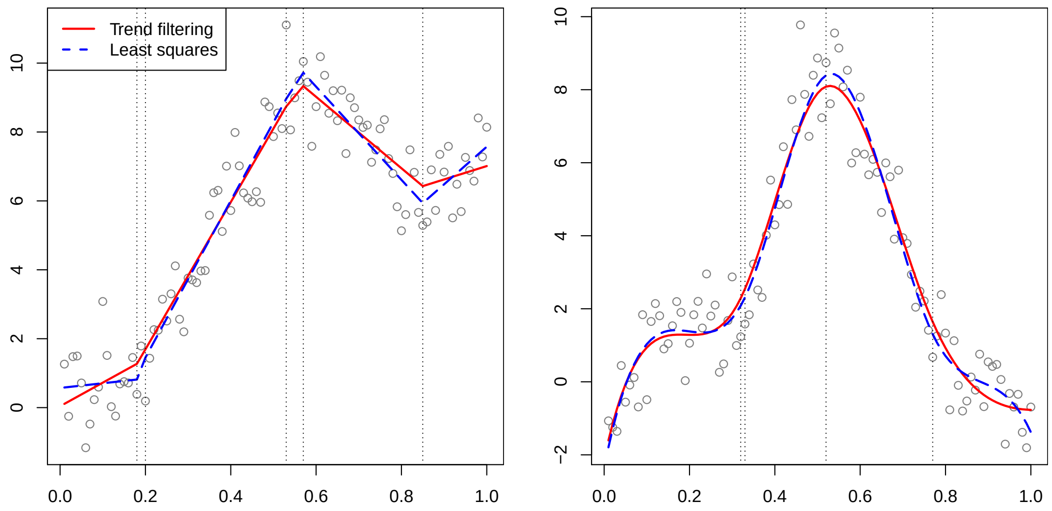

Figure 2: *Examples of the shrinkage effect for linear trend filtering ($k = 1$, left panel) and for cubic trend filtering ($k = 3$, right panel). In each panel, the solid red line is the trend filtering estimate (at a particular value of λ), and the dashed blue line is the regression spline estimate of the same order and with the same knots, with the vertical lines marking the locations of the knots. The trend filtering estimates on the left and right have shrunken 1st and 3rd derivatives, respectively, compared to their regression spline counterparts.*

an implementation of the path algorithm in which computing the estimate at each successive critical point in the path requires $O(n)$ operations.

Software for both of these algorithms is freely available online. For the primal-dual interior point method, see `http://stanford.edu/~boyd/l1_tf`, which provides Matlab and C implementations (these only cover the linear trend filtering case, but can be extended to the general polynomial case); for the path algorithm, see the function `trendfilter` in the R package `genlasso`, available on the CRAN repository.

1.2 Summary of our results

Little is known about trend filtering—mainly, the results due to its generalized lasso form, e.g., the degrees of freedom result (5) discussed in the previous section—and much is unknown. Examining the trend filtering fits in Figures 1 and 2, it appears that the estimates not only have the structure of piecewise polynomials, they furthermore have the structure of splines: these are piecewise polynomial functions that have continuous derivatives of all orders lower than the leading one [i.e., a kth degree spline is a kth degree piecewise polynomial with continuous 0th through $(k-1)$st derivatives at its knots]. Figure 3 plots an example cubic trend filtering estimate, along with its discrete 1st, 2nd, and 3rd derivatives (given by multiplication by $D^{(1)}$, $D^{(2)}$, and $D^{(3)}$, respectively). Sure enough, the lower order discrete derivatives appear "continuous" across the knots, but what does this really mean for such discrete sequences? Does trend filtering have an analogous continuous-time representation, and if so, are the estimated functions really splines?

Besides these questions, one may also wonder about the performance of trend filtering estimates compared to other methods. Empirical examples (like those in Section 2) show that trend filtering estimates achieve a significantly higher degree of local adaptivity than smoothing splines, which are arguably the standard tool for adaptive spline estimation. Other examples (like those in Section 3)

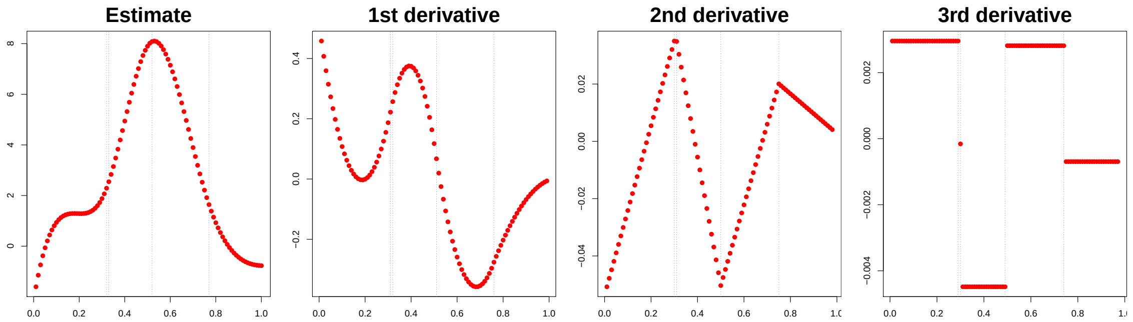

Figure 3: *The leftmost panel shows the same cubic trend filtering estimate as in Figure 2 (but here we do not use linear interpolation to emphasize the discrete nature of the estimate). The components of this estimate appear to be the evaluations of a continuous piecewise polynomial function. Moreover, its discrete 1st and 2nd derivatives (given by multiplying by $D^{(1)}$ and $D^{(2)}$, respectively) also appear to be continuous, and its discrete third derivative (from multiplication by $D^{(3)}$) is piecewise constant within each interval. Hence we might believe that such a trend filtering estimate actually represents the evaluations of a 3rd degree spline over the inputs $x_i = i/n$, $i = 1, \ldots n$. We address this idea in Section 4.*

show that trend filtering estimates display a comparable level of adaptivity to locally adaptive regression splines, another well-known technique for adaptive spline estimation, proposed by Mammen & van de Geer (1997) on the basis of being more locally adaptive (as their name would suggest). Examples are certainly encouraging, but a solely empirical conclusion here would be unsatisfactory—fixing as a metric the squared error loss in estimating the true function f_0, averaged over the input points, can we say more definitively that trend filtering estimates actually outperform smoothing splines, and perform as well as locally adaptive regression splines?

We investigate the questions discussed above in this paper. To summarize our results, we find that:

- for $k = 0, 1$ (constant or linear orders), the continuous-time analogues of trend filtering estimates are indeed kth degree splines; moreover, they are exactly the same as kth order locally adaptive regression splines;
- for $k \geq 2$ (quadratic or higher orders), the continuous-time versions of trend filtering estimates are not quite splines, but piecewise polynomial functions that are "close to" splines (with small discontinuities in lower order derivatives at the knots); hence they are not the same as kth order locally adaptive regression splines;
- for any k, if the kth derivative of true function f_0 is of bounded variation, then the kth order trend filtering estimate converges to f_0 (in terms of squared error loss) at the minimax rate; this rate is achieved by locally adaptive regression splines (Mammen & van de Geer 1997), but not by smoothing splines nor any other estimate linear in y (Donoho & Johnstone 1998).

We note that, although trend filtering and locally adaptive regression splines are formally different estimators for $k \geq 2$, they are practically indistinguishable by eye in most examples. Such a degree of similarity, in finite samples, goes beyond what we are able to show theoretically. However, we do prove that trend filtering estimates and locally adaptive regression spline estimates converge to each other asymptotically (Theorem 1). The argument here boils down to a bound on the difference in the fitted values of two lasso problems that have the same outcome vector, but different predictor matrices (because both trend filtering and locally adaptive regression splines can be represented as lasso problems, see Section 3). To the best of our knowledge, this general bound is a new result (Appendix B). Further, we use this asymptotic pairing between trend filtering and locally adaptive regression splines to prove the minimax convergence rate for trend filtering (Corollary 1). The idea is simple: trend filtering and locally adaptive regression splines converge to each other at the minimax rate, locally adaptive regression splines converge to the true function at the minimax rate (Mammen & van de Geer 1997), and hence so does trend filtering.

1.3 Why trend filtering?

Trend filtering estimates, we argue, enjoy the favorable theoretical performance of locally adaptive regression splines; but now it is fair to ask: why would we ever use trend filtering, over, say, the latter estimator? The main reason is that trend filtering estimates are much easier to compute, due to the bandedness of the discrete derivative operators, as explained previously. The computations for locally adaptive regression splines, meanwhile, cannot exploit such sparsity or structure, and are considerably slower. To be more concrete, the primal-dual interior point method described in Section 1.1 above can handle problems of size on the order of $n = 1,000,000$ points (and the path algorithm, on the order of $n = 100,000$ points), but even for $n = 10,000$ points, the computations for locally adaptive regression splines are prohibitively slow. We discuss this in Section 3.

Of course, the nonparametric regression toolbox is highly-developed and already offers plenty of good methods. We do not presume that trend filtering should be regarded as the preferred method in every nonparametric regression problem, but simply that it represents a useful contribution to the toolbox, being both fast and locally adaptive, i.e., balancing the strengths of smoothing splines and locally adaptive regression splines. This manuscript mainly focuses on the comparison to the aforementioned estimators because they too, like trend filtering, fit piecewise polynomials functions, and they are widely used. Though we do not compare wavelets or smoothing splines with a spatially variable tuning parameter in as much detail, we consider them in Section 7 in an analysis of astrophysics data. It should be mentioned that for trend filtering to become a truly all-purpose nonparametric regression tool, it must be able to handle arbitrary input points $x_1, \dots x_n$ (not just evenly spaced inputs). We give an extension to this case in Section 6. Our analysis of trend filtering with arbitrary inputs shows promising computational and theoretical properties, but still, a few questions remain unanswered. This will be the topic of future work.

As a separate point, another distinguishing feature of trend filtering is that it falls into what is called the *analysis* framework with respect to its problem formulation, whereas locally adaptive regression splines, smoothing splines, and most others fall into the *synthesis* framework. Synthesis and analysis are two terms used in signal processing that describe different approaches for defining an estimator with certain desired characteristics. In the synthesis approach, one builds up the estimate constructively from a set of characteristic elements or atoms; in the analysis approach, the strategy is instead to define the estimate deconstructively, via an operator that penalizes undesirable or uncharacterisic behavior. Depending on the situation, it can be more natural to implement the former rather than the latter, or vice versa, and hence both are important. We discuss the importance of the analysis framework in the context of nonparametric regression estimators in Section 8.2, where we define extensions of trend filtering that would be difficult to construct from the synthesis perspective, e.g., a sparse variant of trend filtering.

Here is an outline for the rest of this article (though we have discussed its contents throughout the introduction, we list them here in proper order). In Sections 2 and 3, we compare trend filtering to smoothing splines and locally adaptive regression splines, respectively. We give data examples that show trend filtering estimates are more locally adaptive than smoothing splines, and that trend filtering and locally adaptive regression splines are remarkably similar, at any common value of their tuning parameters. We also discuss the differing computational requirements for these methods. In Section 3, we show that both locally adaptive regression splines and trend filtering can be posed as lasso problems, with identical predictor matrices when $k = 0$ or 1, and with similar but slightly different predictor matrices when $k \geq 2$. This allows us to conclude that trend filtering and locally adaptive regression splines are exactly the same for constant or linear orders, but not for quadratic or higher orders. Section 4 develops a continuous-time representation for the trend filtering problem, which reveals that (continuous-time) trend filtering estimates are always kth order piecewise polynomials, but for $k \geq 2$, are not kth order splines. In Section 5, we derive the minimax convergence rate of trend filtering estimates, under the assumption that the kth derivative of the true function has bounded total variation. We do this by bounding the difference between trend filtering estimates and locally adaptive regression splines, and invoking the fact that the latter are already known to converge at the minimax rate (Mammen & van de Geer 1997). We also study convergence rates for a true function with growing total variation. Section 6 discusses the extension to arbitrary input points. In Section 7, we consider an astrophysics data example, and compare the performance of several commonly used nonparametric regression tools. Section 8 presents ideas for future work: multivariate trend filtering, sparse trend filtering, and the synthesis versus analysis perspectives. Most proofs are deferred until Appendices A, B, C.

2 Comparison to smoothing splines

Smoothing splines are a popular tool in nonparametric regression, and have been extensively studied in terms of both computations and theory [some well-known references are de Boor (1978), Wahba (1990), Green & Silverman (1994)]. Given input points $x_1, \ldots x_n \in [0,1]$, which we assume are unique, and observations $y_1, \ldots y_n \in \mathbb{R}$, the kth order smoothing spline estimate is defined as

$$\hat{f} = \underset{f \in \mathcal{W}_{(k+1)/2}}{\operatorname{argmin}} \sum_{i=1}^{n} \big(y_i - f(x_i)\big)^2 + \lambda \int_0^1 \big(f^{(\frac{k+1}{2})}(t)\big)^2 \, dt, \tag{6}$$

where $f^{(\frac{k+1}{2})}(t)$ is the derivative of f of order $(k+1)/2$, $\lambda \geq 0$ is a tuning parameter, and the domain of minimization here is the Sobolev space

$$\mathcal{W}_{(k+1)/2} = \left\{ f : [0,1] \to \mathbb{R} \; : \; f \text{ is } (k+1)/2 \text{ times differentiable, and } \int_0^1 \big(f^{(\frac{k+1}{2})}(t)\big)^2 \, dt < \infty \right\}.$$

Unlike trend filtering, smoothing splines are only defined for an odd polynomial order k. In practice, it seems that the case $k = 3$ (i.e., cubic smoothing splines) is by far the most common case considered. In the next section, we draw a high-level comparison between smoothing splines and trend filtering, by writing the smoothing spline minimization problem (6) in finite-dimensional form. Following this, we make empirical comparisons, and then discuss computational efficiency.

2.1 Generalized ridge regression and Reinsch form

Remarkably, it can be shown that the infinite-dimensional problem in (6) is has a unique minimizer, which is a kth degree natural spline with knots at the input points $x_1, \ldots x_n$ [see, e.g., Wahba (1990), Green & Silverman (1994), Hastie et al. (2008)]. Recall that a kth degree natural spline is a simply a kth degree spline that reduces to a polynomial of degree $(k-1)/2$ before the first knot and after the last knot; it is easy to check the set of natural splines of degree k, with knots at $x_1, \ldots x_n$, is spanned by precisely n basis functions. Hence to solve (6), we can solve for the coefficients $\theta \in \mathbb{R}^n$ in this basis expansion:

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^n}{\operatorname{argmin}} \; \|y - N\theta\|_2^2 + \lambda \theta^T \Omega \theta, \tag{7}$$

where $N \in \mathbb{R}^{n \times n}$ contains the evaluations of kth degree natural spline basis functions over the knots $x_1, \ldots x_n$, and $\Omega \in \mathbb{R}^{n \times n}$ contains the integrated products of their $((k+1)/2)$nd derivatives; i.e., if

$\eta_1, \ldots \eta_n$ denotes a collection of basis functions for the set of kth degrees natural splines with knots at $x_1, \ldots x_n$, then

$$N_{ij} = \eta_j(x_i) \quad \text{and} \quad \Omega_{ij} = \int_0^1 \eta_i^{(\frac{k+1}{2})}(t) \cdot \eta_j^{(\frac{k+1}{2})}(t)\, dt \quad \text{for all } i, j = 1, \ldots n. \tag{8}$$

The problem in (7) is a generalized ridge regression, and from its solution $\hat{\theta}$, the function $\hat{f}$ in (6) is simply given at the input points $x_1, \ldots x_n$ by

$$\big(\hat{f}(x_1), \ldots \hat{f}(x_n)\big) = N\hat{\theta}.$$

More generally, the smoothing spline estimate $\hat{f}$ at an arbitrary input $x \in [0,1]$ is given by

$$\hat{f}(x) = \sum_{j=1}^n \hat{\theta}_j \eta_j(x).$$

To compare the smoothing spline problem, as expressed in (7), with trend filtering, it helps to rewrite the smoothing spline fitted values as follows:

$$\begin{aligned} N\hat{\theta} &= N(N^T N + \lambda\Omega)^{-1} N^T y \\ &= N\Big(N^T (I + \lambda N^{-T}\Omega N^{-1})N\Big)^{-1} N^T y \\ &= (I + \lambda K)^{-1} y, \end{aligned} \tag{9}$$

where $K = N^{-T}\Omega N^{-1}$. The expression in (9) is called the *Reinsch* form for the fitted values. From this expression, we can view $\hat{u} = N\hat{\theta}$ as the solution of the minimization problem

$$\hat{u} = \underset{u \in \mathbb{R}^n}{\operatorname{argmin}} \; \|y - u\|_2^2 + \lambda u^T K u, \tag{10}$$

which is of similar form to the trend filtering problem in (2), but here the ℓ_1 penalty $\|D^{(k+1)}\beta\|_1$ is replaced by the quadratic penalty $u^T K u = \|K^{1/2}u\|_2^2$. How do these two penalties compare? First, the penalty matrix $K^{1/2}$ used by smoothing splines is similar in nature to the discrete derivative operators [we know from its continuous-time analog in (6) that the term $\|K^{1/2}u\|_2^2$ penalizes wiggliness in something like the $((k+1)/2)$nd derivative of u] but is still strictly different. For example, for $k = 3$ (cubic smoothing splines) and input points $x_i = i/n$, $i = 1, \ldots n$, it can be shown (Green & Yandell 1985) that the smoothing spline penalty is $\|K^{1/2}u\|_2^2 = \|C^{-1/2}D^{(2)}u\|_2^2/n^3$ where $D^{(2)}$ is the second order discrete derivative operator, and $C \in \mathbb{R}^{n \times n}$ is a tridiagonal matrix, with diagonal elements equal to $2/3$ and off-diagonal elements equal to $1/6$.

A second and more important difference is that smoothing splines utilize a (squared) ℓ_2 penalty, while trend filtering uses an ℓ_1 penalty. Analogous to the usual comparisons between ridge regression and the lasso, the former penalty shrinks the components of $K^{1/2}\hat{u}$, but does not set any of the components to zero unless $\lambda = \infty$ (in which case all components are zero), whereas the latter penalty shrinks and also adaptively sets components of $D^{(k+1)}\hat{\beta}$ to zero. One might imagine, recalling that $K^{1/2}$ and $D^{(k+1)}$ both act in a sense as derivative operators, that trend filtering estimates therefore exhibit a finer degree of local adaptivity than do smoothing splines. This idea is supported by the examples in the next section, which show that trend filtering estimates outperform smoothing splines (when both are optimally tuned) in estimating functions with spatially inhomogeneous smoothness. The idea is also supported by our theory in Section 5, where we prove that trend filtering estimates have a better rate of convergence than smoothing splines (in fact, they achieve the optimal rate) over a broad class of underlying functions.

2.2 Empirical comparisons

We compare trend filtering and smoothing spline estimates on simulated data. We fix $k = 3$ (i.e., we compare cubic trend filtering versus cubic smoothness splines), because the `smooth.spline` function in the R programming language provides a fast implementation for smoothing splines in this case. Generally speaking, smoothing splines and trend filtering provide similar estimates when the underlying function f_0 has spatially homogeneous smoothness, or to put it simply, is either entirely smooth or entirely wiggly throughout its domain. Hence, to illustrate the difference between the two estimators, we consider two examples of functions that display varying levels of smoothness at different spatial locations.

Our first example, which we call the "hills" example, considers a piecewise cubic function f_0 over $[0, 1]$, whose knots are spaced farther apart on the left side of the domain, but bunched closer together on the right side. As a result, $f_0(x)$ is smooth for x between 0 and about 0.8, but then abruptly becomes more wiggly—see the top left panel of Figure 4. We drew $n = 128$ noisy observations from f_0 over the evenly spaced inputs $x_i = i/n$, $i = 1, \ldots n$ (with independent, normal noise), and fit a trend filtering estimate, tuned to have 19 degrees of freedom, as shown in the top right panel.[2] We can see here that the estimate adapts to the appropriate levels of smoothness at both the left and right sides of the domain. But this is not true of the smoothing spline estimate with 19 degrees of freedom, displayed in the bottom left panel: the estimate is considerably oversmoothed on the right side. As we increase the allowed flexibility, the smoothing spline estimate is able to fit the small hills on the right, with a total of 30 degrees of freedom; however, this causes undersmoothing on the left side, as shown in the bottom right panel.

For our second example, we take f_0 to be the "Doppler" function [as considered in, e.g., Donoho & Johnstone (1995), Mammen & van de Geer (1997)]. Figure 5, clockwise from the top left, displays the Doppler function and corresponding $n = 1000$ noisy observations, the trend filtering estimate with 50 degrees of freedom, the smoothing spline estimate with 50 degrees of freedom, and the smoothing spline estimate with 90 degrees of freedom. The same story, as in the hills example, holds here: trend filtering adapts to the local level of smoothness better than smoothing splines, which have trouble with the rapidly increasing frequency of the Doppler function (as x decreases).

In Figure 6, we display the average squared error losses[3]

$$\frac{1}{n}\sum_{i=1}^{n}\big(\hat{\beta}_i - f_0(x_i)\big)^2 \quad \text{and} \quad \frac{1}{n}\sum_{i=1}^{n}\big(\hat{f}(x_i) - f_0(x_i)\big)^2,$$

for the trend filtering and smoothing spline estimates $\hat{\beta}$ and $\hat{f}$, respectively, on the hills and Doppler examples. We considered a wide range of model complexities indexed by degrees of freedom, and averaged the results over 50 simulated data sets for each setup (the dotted lines show plus or minus one standard deviations). Aside from the visual evidence given in Figures 4 and 5, Figure 6 shows that from the perspective of squared error loss, trend filtering outperforms smoothing splines in estimating underlying functions with variable spatial smoothness. As mentioned previously, we will prove in Section 5 that for a large class of underlying functions f_0, trend filtering estimates have a sharper convergence rate than smoothing splines.

2.3 Computational considerations

Recall that the smoothing spline fitted values are given by

$$N\hat{\theta} = N(N^T N + \lambda\Omega)^{-1}N^T y, \tag{11}$$

[2] To be precise, this is an unbiased estimate of its degrees of freedom; see (5) in Section 1.1.

[3] For the Doppler data example, we actually average the squared error loss only over inputs $x_i \geq 0.175$, because for $x_i < 0.175$, the true Doppler function f_0 is of such high frequency that neither trend filtering nor smoothing splines are able to do a decent job of fitting it.

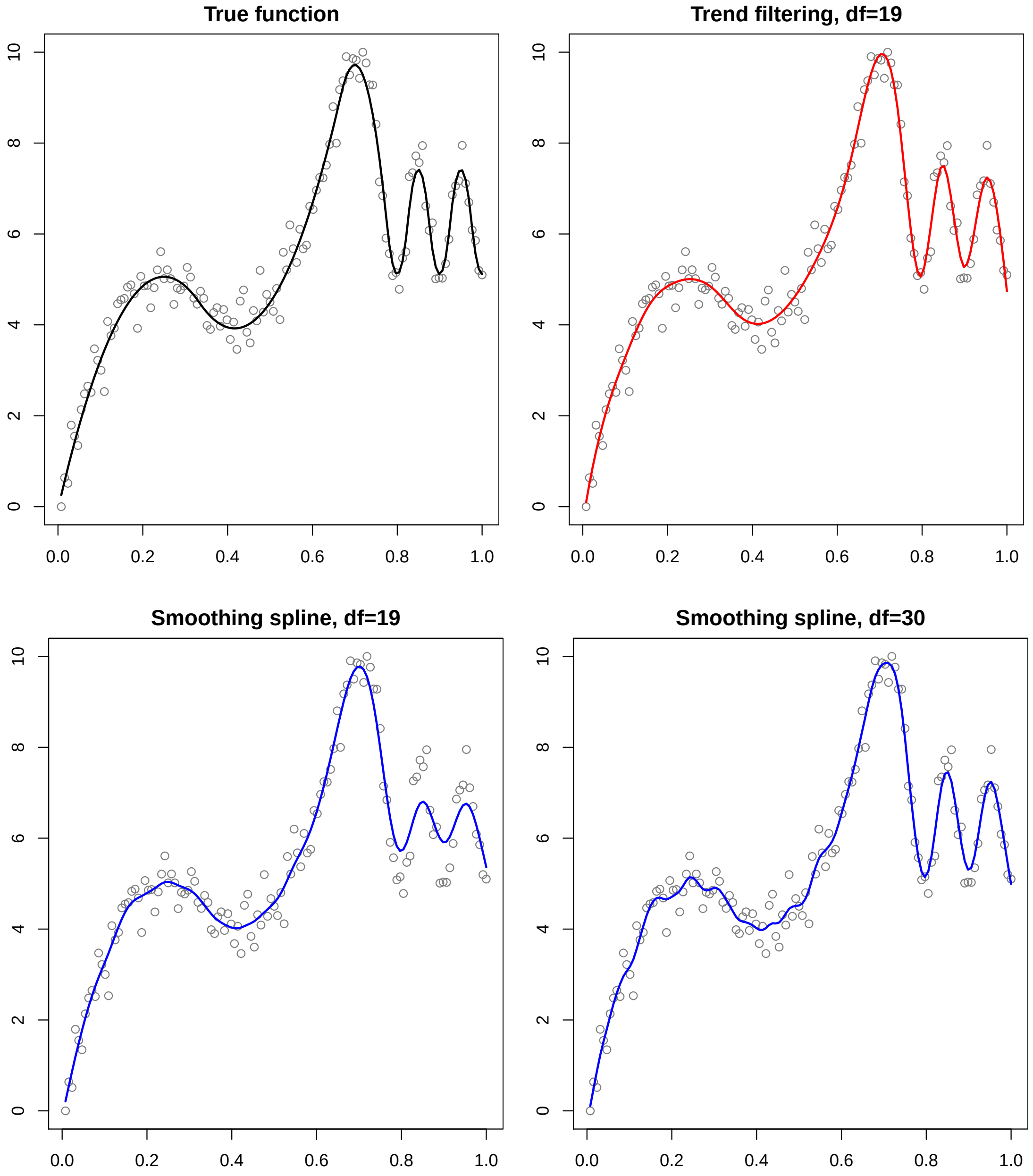

Figure 4: *An example with $n = 128$ observations drawn from a model where the underlying function has variable spatial smoothness, as shown in the top left panel. The cubic trend filtering estimate with 19 degrees of freedom, shown in the top right panel, picks up the appropriate level of smoothness at different spatial locations: smooth at the left side of the domain, and wiggly at the right side. When also allowed 19 degrees of freedom, the cubic smoothing spline estimate in the bottom left panel grossly underestimates the signal on the right side of the domain. The bottom right panel shows the smooth spline estimate with 30 degrees of freedom, tuned so that it displays the appropriate level of adaptivity on the right side; but now, it is overly adaptive on the left side.*

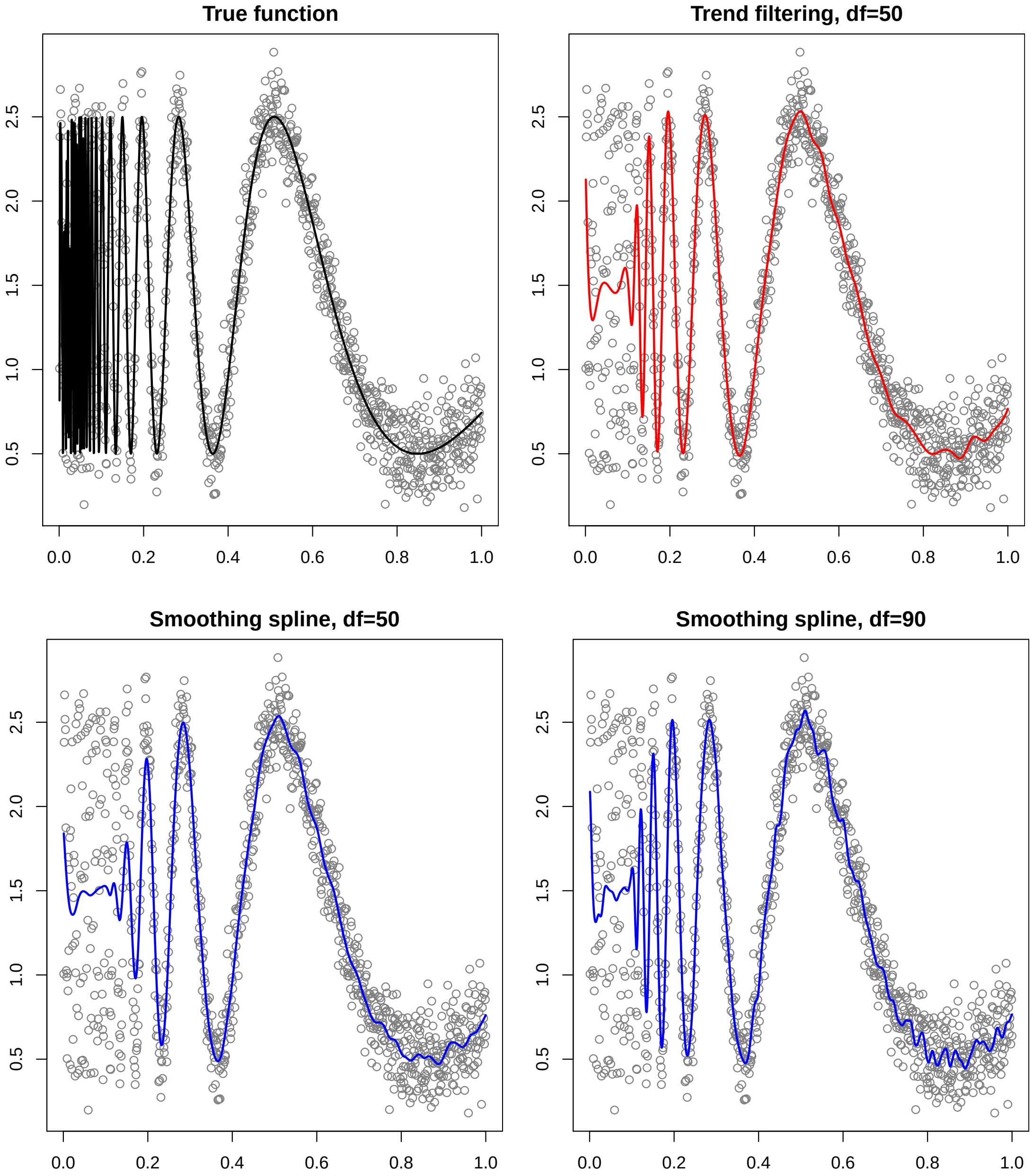

Figure 5: *An example with* $n = 1000$ *noisy observations of the Doppler function,* $f(x) = \sin(4/x) + 1.5$*, drawn in the top left panel. The top right and bottom left panels show the cubic trend filtering and smoothing spline estimates, each with 50 degrees of freedom; the former captures approximately 4 cycles of the Doppler function, and the latter only 3. If we nearly double the model complexity, namely, we use 90 degrees of freedom, then the smoothing spline estimate is finally able to capture 4 cycles, but the estimate now becomes very jagged on the right side of the plot.*

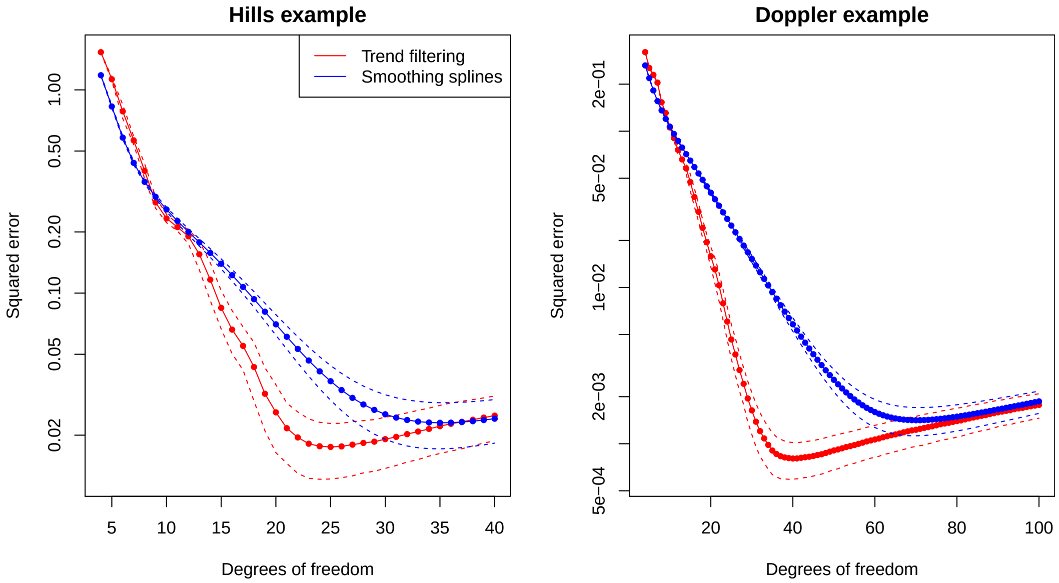

Figure 6: *Shown is the squared error loss in predicting the true function* f_0*, averaged over the input points, for the hills data example on the left, and the Doppler example on the right. In each setup, trend filtering and smoothing spline estimators were fit over a range of degrees of freedom values; the red curves display the loss for trend filtering, and the blue curves for smoothing splines. The results were averaged over 50 simulated data sets, and the standard deviations are denoted by dotted lines. In these examples, trend filtering has a generally better predictive accuracy than smoothing splines, especially for models of low to intermediate complexity (degrees of freedom).*

where $N \in \mathbb{R}^{n\times n}$ contains the evaluations of basis functions $\eta_1, \ldots \eta_n$ for the subspace of kth degree natural splines with knots at the inputs, and $\Omega \in \mathbb{R}^{n\times n}$ contains their integrated products of their $((k+1)/2)$nd order derivatives, as in (8). Depending on exactly which basis we choose, computation of (11) can be fast or slow; by choosing the B-spline basis functions, which have local support, the matrix $N^TN + \lambda\Omega$ is banded, and so the smoothing spline fitted values can be computed in $O(n)$ operations [e.g, see de Boor (1978)]. In practice, these computations are extremely fast.

By comparison, Kim et al. (2009) suggest a primal-dual interior point method, as mentioned in Section 1.1, that computes the trend filtering estimate (at any fixed value of the tuning parameter λ) by iteratively solving a sequence of banded linear systems, rather than just a single one. Theoretically, the worst-case number of iterations scales as $O(n^{1/2})$, but the authors report that in practice the number of iterations needed is only a few tens, almost independent of the problem size n. Hence trend filtering computations with the primal-dual path interior point method are slower than those for smoothing splines, but not by a huge margin.

To compute the trend filtering estimates for the examples in the previous section, we actually used the dual path algorithm of Tibshirani & Taylor (2011), which was also discussed in Section 1.1. Instead of solving the trend filtering problem at a fixed value of λ, this algorithm constructs the solution path as λ varies from ∞ to 0. Essentially, it does so by stepping through a sequence of estimates, where each step either adds one knot to or deletes one knot from the fitted piecewise polynomial structure. The computations at each step amount to solving two banded linear systems, and hence require $O(n)$ operations; the overall efficiency depends on how many steps along the path are needed before the estimates of interest have been reached (at which point the path algorithm can be terminated early). But because knots can be both added and deleted to the fitted piecewise polynomial structure at each step, the algorithm can take much more than k steps to reach an estimate with k knots. Consider the Doppler data example, in the last section, with $n = 1000$ points: the path algorithm used nearly 4000 steps to compute the trend filtering estimate with 46 knots (50 degrees of freedom) shown in the upper right panel of Figure 5. This took approximately 28 seconds on a standard desktop computer, compared to the smoothing spline estimates shown in the bottom left and right panels of Figure 5, which took about 0.005 seconds each. We reiterate that in this period of time, the path algorithm for trend filtering computed a total of 4000 estimates, versus a single estimate computed by the smoothing spline solver. (A quick calculation, $28/4000 = 0.007$, shows that the time per estimate here is comparable.) For the hills data set in the last section, where $n = 128$, the dual path algorithm constructed the entire path of trend filtering estimates (consisting of 548 steps) in less than 3 seconds; both smoothing spline estimates took under 0.005 seconds each.

3 Comparison to locally adaptive regression splines

Locally adaptive regression splines are an alternative to smoothing splines, proposed by Mammen & van de Geer (1997). They are more computationally intensive than smoothing splines but have better adaptivity properties (as their name would suggest). Let $x_1, \ldots x_n \in [0, 1]$ denote the inputs, assumed unique and ordered as in $x_1 < x_2 < \ldots < x_n$, and $y_1, \ldots y_n \in \mathbb{R}$ denote the observations. For the kth order locally adaptive regression spline estimate, where $k \geq 0$ is a given arbitrary integer (not necessarily odd, as is required for smoothing splines), we start by defining the knot superset

$$T = \begin{cases} \{x_{k/2+2}, \ldots x_{n-k/2}\} & \text{if } k \text{ is even,} \\ \{x_{(k+1)/2+1}, \ldots x_{n-(k+1)/2}\} & \text{if } k \text{ is odd.} \end{cases} \tag{12}$$

This is essentially just the set of inputs $\{x_1, \ldots x_n\}$, but with points near the left and right boundaries removed. We then define the kth order locally adaptive regression spline estimate as

$$\hat{f} = \underset{f \in \mathcal{G}_k}{\operatorname{argmin}} \; \frac{1}{2} \sum_{i=1}^{n} \big(y_i - f(x_i)\big)^2 + \frac{\lambda}{k!} \cdot \mathrm{TV}(f^{(k)}), \tag{13}$$

where $f^{(k)}$ is now the kth weak derivative of f, $\lambda \geq 0$ is a tuning parameter, $\mathrm{TV}(\cdot)$ denotes the total variation operator, and $\mathcal{G}_k$ is the set

$$\mathcal{G}_k = \big\{ f : [0, 1] \to \mathbb{R} \;:\; f \text{ is } k\text{th degree spline with knots contained in } T \big\}. \tag{14}$$

Recall that for a function $f : [0, 1] \to \mathbb{R}$, its total variation is defined as

$$\mathrm{TV}(f) = \sup \left\{ \sum_{i=1}^{p} |f(z_{i+1}) - f(z_i)| \;:\; z_1 < \ldots < z_p \text{ is a partition of } [0, 1] \right\},$$

and this reduces to $\mathrm{TV}(f) = \int_0^1 |f'(t)|\, dt$ if f is (strongly) differentiable.

Next, we briefly address the difference between our definition of locally adaptive regression splines in (13) and the original definition found in Mammen & van de Geer (1997); this discussion can be skipped without interrupting the flow of ideas. After this, we rewrite problem (13) in terms of the coefficients of f with respect to a basis for the finite-dimensional set $\mathcal{G}_k$. For an arbitrary choice of basis, this new problem is of generalized lasso form, and in particular, if we choose the truncated power series as our basis for $\mathcal{G}_k$, it simply becomes a lasso problem. We will see that trend filtering, too, can be represented as a lasso problem, which allows for a more direct comparison between the two estimators.

3.1 Unrestricted locally adaptive regression splines

For readers familiar with the work of Mammen & van de Geer (1997), it may be helpful to explain the difference between our definition of locally adaptive regression splines and theirs: these authors define the locally adaptive regression spline estimate as

$$\hat{f} \in \operatorname*{argmin}_{f \in \mathcal{F}_k} \frac{1}{2} \sum_{i=1}^n \big(y_i - f(x_i)\big)^2 + \frac{\lambda}{k!} \cdot \mathrm{TV}(f^{(k)}), \tag{15}$$

where $\mathcal{F}_k$ is the set

$$\mathcal{F}_k = \Big\{ f : [0,1] \to \mathbb{R} \ : \ f \text{ is } k \text{ times weakly differentiable and } \mathrm{TV}(f^{(k)}) < \infty \Big\}.$$

[The element notation in (15) emphasizes the fact that the minimizer is not generally unique.] We call (15) the *unrestricted* locally adaptive regression spline problem, in reference to its minimization domain compared to that of (13). Mammen & van de Geer (1997) prove that the minimum in this unrestricted problem is always achieved by a kth degree spline, and that this spline has knots contained in T if $k = 0$ or 1, but could have knots outside of T (and in fact, outside of the input set $\{x_1, \ldots x_n\}$) if $k \geq 2$. In other words, the solution in (13) is always a solution in (15) when $k = 0$ or 1, but this need not be true when $k \geq 2$; in the latter case, even though there exists a kth degree spline that minimizes (15), its knots could occur at non-input points.

The unrestricted locally adaptive regression estimate (15) is the main object of theoretical study in Mammen & van de Geer (1997), but practically speaking, this estimate is difficult to compute when $k \geq 2$, because the possible knot locations are generally not easy to determine [see also Rosset & Zhu (2007)]. On the other hand, the restricted estimate as defined in (13) is more computationally tractable. Fortunately, Mammen & van de Geer (1997) show that essentially all of their theoretical results for the unrestricted estimate also apply to the restricted estimate, as long as the input points $x_1, \ldots x_n$ are not spaced too far apart. In particular, for evenly spaced inputs, $x_i = i/n$, $i = 1, \ldots n$, the convergence rates of the unrestricted and restricted estimates are the same. We therefore focus on the restricted problem (13) in the current paper, and mention the unrestricted problem (15) purely out of interest. For example, to anticipate results to come, we will show in Lemma 3 of Section 3.3 that the trend filtering estimate (2) for $k = 0$ or 1 is equal to the locally adaptive regression spline estimate (13) (i.e., they match at the input points $x_1, \ldots x_n$); hence, from what we discussed above, it also solves the unrestricted problem in (15).

3.2 Generalized lasso and lasso form

We note that the knot set T in (12) has $n - k - 1$ elements, so $\mathcal{G}_k$ is spanned by n basis functions, call them $g_1, \ldots g_n$. Since each g_j, $j = 1, \ldots n$ is a kth degree spline with knots in T, we know that its kth weak derivative is piecewise constant and (say) right-continuous, with jump points contained in T; therefore, writing $t_0 = 0$ and $T = \{t_1, \ldots t_{n-k-1}\}$, we have

$$\mathrm{TV}(g_j^{(k)}) = \sum_{i=1}^{n-k-1} \big| g_j^{(k)}(t_i) - g_j^{(k)}(t_{i-1}) \big|.$$

Similarly, any linear combination of $g_1, \ldots g_n$ satisfies

$$\mathrm{TV}\bigg(\Big(\sum_{j=1}^n \theta_j g_j \Big)^{(k)} \bigg) = \sum_{i=1}^{n-k-1} \bigg| \sum_{j=1}^n \Big(g_j^{(k)}(t_i) - g_j^{(k)}(t_{i-1}) \Big) \cdot \theta_j \bigg|.$$

Hence we can reexpress (13) in terms of the coefficients $\theta \in \mathbb{R}^n$ in its basis expansion with respect to $g_1, \ldots g_n$,

$$\hat{\theta} = \operatorname*{argmin}_{\theta \in \mathbb{R}^n} \frac{1}{2} \|y - G\theta\|_2^2 + \frac{\lambda}{k!} \|C\theta\|_1, \tag{16}$$

where $G \in \mathbb{R}^{n \times n}$ contains the evaluations of $g_1, \dots g_n$ over the inputs $x_1, \dots x_n$, and $C \in \mathbb{R}^{(n-k-1) \times n}$ contains the differences in their kth derivatives across the knots, i.e.,

$$G_{ij} = g_j(x_i) \qquad \text{for } \ i, j = 1, \dots n, \tag{17}$$

$$C_{ij} = g_j^{(k)}(t_i) - g_j^{(k)}(t_{i-1}) \qquad \text{for } \ i = 1, \dots n-k-1, \ \ j = 1, \dots n. \tag{18}$$

Given the solution $\hat{\theta}$ in (16), we can recover the locally adaptive regression spline estimate $\hat{f}$ in (13) over the input points by

$$\big(\hat{f}(x_1), \dots \hat{f}(x_n)\big) = G\hat{\theta},$$

or, at an arbitrary point $x \in [0,1]$ by

$$\hat{f}(x) = \sum_{j=1}^{n} \hat{\theta}_j g_j(x).$$

The problem (16) is a generalized lasso problem, with predictor matrix G and penalty matrix C; by taking $g_1, \dots g_n$ to be the truncated power basis, we can turn (a block of) C into the identity, and hence (16) into a lasso problem.

Lemma 1. *Let $T = \{t_1, \dots t_{n-k-1}\}$ denote the set defined in* (12)*, and let $g_1, \dots g_n$ denote the kth order truncated power basis with knots in T,*

$$\begin{aligned} g_1(x) = 1, \ g_2(x) = x, \ \dots \ g_{k+1}(x) = x^k, \\ g_{k+1+j}(x) = (x - t_j)^k \cdot 1\{x \geq t_j\}, \quad j = 1, \dots n-k-1. \end{aligned} \tag{19}$$

(For the case $k = 0$, we interpret $0^0 = 1$.) Then the locally adaptive regression spline problem (13) *is equivalent to the following lasso problem:*

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^n}{\mathrm{argmin}} \ \frac{1}{2}\|y - G\theta\|_2^2 + \lambda \sum_{j=k+2}^{n} |\theta_j|, \tag{20}$$

in that $\hat{f}(x) = \sum_{j=1}^{n} \hat{\theta}_j g_j(x)$ for $x \in [0,1]$. Here $G \in \mathbb{R}^{n \times n}$ is the basis matrix as in (17).

Proof. This follows from the fact that for the truncated power basis, the penalty matrix C in (18) satisfies $C_{i,i+k+1} = k!$ for $i = 1, \dots n-k-1$, and $C_{ij} = 0$ otherwise. □

It is worth noting that Osborne et al. (1998) investigate a lasso problem similar to (20) for the purposes of knot selection in regression splines. Note that (20) is of somewhat nonstandard form for a lasso problem, because the ℓ_1 penalty is only taken over the last $n-k-1$ components of θ. We will see next that the trend filtering problem in (2) can also be written in lasso form (again with the ℓ_1 penalty summing over the last $n-k-1$ coefficients), and we will compare these two formulations. First, however, it is helpful to express the knot superset T in (12) and the basis matrix G in (17) in a more explicit form, for evenly spaced input points $x_i = i/n$, $i = 1, \dots n$ (this being the underlying assumption for trend filtering). These become:

$$T = \begin{cases} ((k+2)/2 + i)/n & \text{for } \ i = 1, \dots n-k-1, \ \text{ if } k \text{ is even}, \\ ((k+1)/2 + i)/n & \text{for } \ i = 1, \dots n-k-1, \ \text{ if } k \text{ is odd}, \end{cases} \tag{21}$$

and

$$G_{ij} = \begin{cases} \begin{cases} 0 & \text{for } \ i < j, \\ 1 & \text{for } \ i \geq j, \end{cases} & \text{if } k = 0, \\[1em] \begin{cases} i^{j-1}/n^{j-1} & \text{for } \ i = 1, \ldots n, \ \ j = 1, \ldots k+1, \\ 0 & \text{for } \ i \leq j - k/2, \ \ j \geq k+2, \\ (i - j + k/2)^k/n^k & \text{for } i > j - k/2, \ \ j \geq k+2, \end{cases} & \text{if } k > 0 \text{ is even}, \\[1em] \begin{cases} i^{j-1}/n^{j-1} & \text{for } \ i = 1, \ldots n, \ \ j = 1, \ldots k+1, \\ 0 & \text{for } \ i \leq j - (k+1)/2, \ \ j \geq k+2, \\ (i - j + (k+1)/2)^k/n^k & \text{for } i > j - (k+1)/2, \ \ j \geq k+2. \end{cases} & \text{if } k > 0 \text{ is odd}. \end{cases} \tag{22}$$

(It is not really important to separate the definition of G for $k = 0$ from that for $k > 0$, k even; this is only done to make transparent the structure of G.)

3.3 Trend filtering in lasso form

We can transform the trend filtering problem in (2) into lasso form, just like the representation for locally adaptive regression splines in (20).

Lemma 2. *The trend filtering problem in* (2) *is equivalent to the lasso problem*

$$\hat{\alpha} = \operatorname*{argmin}_{\alpha \in \mathbb{R}^n} \frac{1}{2}\|y - H\alpha\|_2^2 + \lambda \sum_{j=k+2}^{n} |\alpha_j|, \tag{23}$$

in that the solutions satisfy $\hat{\beta} = H\hat{\alpha}$. Here, the predictor matrix $H \in \mathbb{R}^{n \times n}$ is given by

$$H_{ij} = \begin{cases} i^{j-1}/n^{j-1} & \textit{for } \ i = 1, \ldots n, \ \ j = 1, \ldots k+1, \\ 0 & \textit{for } \ i \leq j-1, \ \ j \geq k+2, \\ \sigma_{i-j+1}^{(k)} \cdot k!/n^k & \textit{for } \ i > j-1, \ \ j \geq k+2, \end{cases} \tag{24}$$

where we define $\sigma_i^{(0)} = 1$ for all i and

$$\sigma_i^{(k)} = \sum_{j=1}^{i} \sigma_j^{(k-1)} \quad \textit{for } k = 1, 2, 3, \ldots,$$

i.e., $\sigma_i^{(k)}$ is the kth order cumulative sum of $(1, 1, \ldots 1) \in \mathbb{R}^i$.

The proof of this lemma basically inverts the $(k+1)$st order discrete difference operator $D^{(k+1)}$; it is not difficult, but requires some calculation, so we defer it until Appendix A.1. We remark that this result, in the special case of $k = 1$, can be found in Kim et al. (2009).

It is not hard to check that in the case $k = 0$ or 1, the definitions of G in (22) and H in (24) coincide, which means that the locally adaptive regression spline and trend filtering problems (20) and (23) are the same. But when $k \geq 2$, we have $G \neq H$, and hence the problems are different.

Lemma 3. *Consider evenly spaced inputs $x_i = i/n$, $i = 1, \ldots n$, and the basis matrices G, H defined in* (22), (24)*. If $k = 0$ or 1, then $G = H$, so the lasso representations for locally adaptive regression splines and trend filtering,* (20) *and* (23)*, are the same. Therefore their solutions are the same, or in other words,*

$$\hat{\beta}_i = \hat{f}(x_i) \quad \textit{for } i = 1, \ldots n,$$

where $\hat{\beta}$ and $\hat{f}$ are the solutions of the original trend filtering and locally adaptive regression spline problems, (2) *and* (13)*, at any fixed common value of the tuning parameter λ.*

If $k \geq 2$, however, then $G \neq H$, so the problems (20) *and* (23) *are different, and hence the trend filtering and locally adpative regression spline estimators are generically different.*

Proof. By inspection of (22) and (24), we have

$$G = H = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 1 & \dots & 0 \\ \vdots & & & \\ 1 & 1 & \dots & 1 \end{bmatrix} \text{ if } k = 0, \quad G = H = \frac{1}{n} \cdot \begin{bmatrix} n & 1 & 0 & 0 & \dots & 0 \\ n & 2 & 0 & 0 & \dots & 0 \\ n & 3 & 1 & 0 & \dots & 0 \\ n & 4 & 2 & 1 & \dots & 0 \\ \vdots & & & & & \\ n & n & n-2 & n-3 & \dots & 1 \end{bmatrix} \text{ if } k = 1,$$

and $G \neq H$ if $k \geq 2$ [in this case G and H differ by more than a scalar factor, so problems (20) and (23) cannot be equated by simply modifying the tuning parameters]. □

Though the trend filtering and locally adaptive regression spline estimates are formally different for polynomial orders $k \geq 2$, they are practically very similar (at all common values of λ). We give examples of this next, and then compare the computational requirements for the two methods.

3.4 Empirical comparisons

We revisit the hills and Doppler examples of Section 2.2. Figure 7 displays, for $k = 3$ (cubic order), the trend filtering and locally adaptive regression spline estimates at matching values of the tuning parameter λ. The estimates are visually identical in both examples (but not numerically identical—the average squared difference between the estimates across the input points is around 10^{-5} for the hills example, and 10^{-7} for the Doppler example). This remains true for a wide range of common tuning parameter values, and only for very small values of λ do slight differences between the two estimators become noticeable.

Nothing is special about the choice $k = 3$ here or about these data sets in particular: as far as we can tell, the same phenomenon occurs for any polynomial order k, and any set of observations. This extreme similarity between the two estimators, holding in finite sample and across essentially all common tuning parameter values, is beyond what we show theoretically in Section 5. In this section, we prove that for tuning parameters of a certain order, the two estimators converge asymptotically at a fast rate. Sharper statements are a topic for future work.

3.5 Computational considerations

Both the locally adaptive regression spline and trend filtering problems can be represented as lasso problems with dense, square predictor matrices, as in (20) and (23). For trend filtering, we do this purely for analytical reasons, and computationally it is much more efficient to work from its original representation in (2), where the penalty operator $D^{(k+1)}$ is sparse and banded. As discussed in Sections 1.1 and 2.3, two efficient algorithms for trend filtering are the primal-dual interior point method of Kim et al. (2009) and the dual path algorithm of Tibshirani & Taylor (2011); the former computes the trend filtering estimate at a fixed value of λ, in $O(n^{3/2})$ worst-case complexity [the authors claim that the practical complexity is closer to $O(n)$]; the latter computes the entire solution path over λ, with each critical point in this piecewise linear path requiring $O(n)$ operations.

For locally adaptive regression splines, on the other hand, there is not a better computational alternative than solving the lasso problem in (20). One can check that the inverse of the truncated power basis matrix G is dense, so if we converted (20) to generalized lasso form [to match the form of trend filtering in (2)], then it would have a dense penalty matrix. And if we were to choose, e.g.,

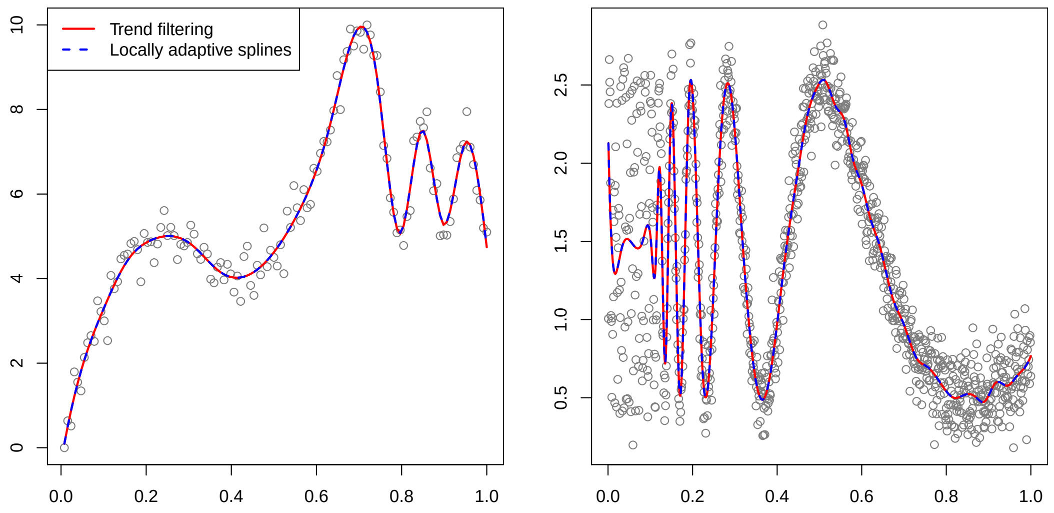

Figure 7: *Trend filtering and locally adaptive regression spline estimates, using the same values of the tuning parameter λ, for the hills and Doppler data examples considered in Section 2.2. The trend filtering estimates are drawn as solid red lines, and locally adaptive regression splines as dotted blue lines; in both examples, the two estimates are basically indistinguishable by eye.*

the B-spline basis over the truncated power basis to parameterize $\mathcal{G}_k(T)$ in (13), then although the basis matrix G would be sparse and banded, the resulting penalty matrix C in (16) would be dense. In other words, to compute the locally adaptive regression spline estimate, we are more or less stuck with solving the lasso problem in (20), where G is the dense predictor matrix in (22). This task is manageable for small or moderately sized problems, but for large problems, dealing with the $n \times n$ dense matrix G, and even holding it in memory, becomes burdensome.

To compute the locally adaptive regression spline estimates for the examples in the last section, we solved the lasso problem in (20) using the LARS algorithm for the lasso path (Efron et al. 2004), as implemented by the `lars` R package.[4] This particular algorithm was chosen for the sake of a fair comparison to the dual path algorithm used for trend filtering. For the Doppler data example with $n = 1000$ points, the LARS algorithm computed the locally adaptive regression spline estimate (shown in the right panel of Figure 7) in a comparable amount of time to that taken by the dual path algorithm for trend filtering—in fact, it was faster, at approximately 16 versus 28 seconds on a standard desktop computer. The real issue, however, is scalability. For $n = 1000$ points, each of these algorithms required about 4000 steps to compute their respective estimates; for $n = 10,000$ noisy observations from the Doppler curve, the dual path algorithm completed 4000 steps in a little under 2.5 minutes, whereas the LARS algorithm completed 4000 steps in 1 hour. Furthermore, for problem sizes n somewhat larger than $n = 10,000$, just fitting the $n \times n$ basis matrix G used by the LARS algorithm into memory becomes an issue.

4 Continuous-time representation

Section 3.3 showed that the trend filtering minimization problem (2) can be expressed in lasso form (23), with a predictor matrix H as in (24). The question we consider is now: is there a set of basis

[4] To fit the problem in (20) into standard lasso form, i.e., a form in which the ℓ_1 penalty is taken over the entire coefficient vector, we can solve directly for the first $k+1$ coefficients (in terms of the last $n-k-1$ coefficients), simply by linear regression.

functions whose evaluations over the inputs $x_1, \ldots x_n$ give this matrix H? Our next lemma answers this question affirmatively.

Lemma 4. *Given inputs $x_1 < \ldots < x_n$, consider the functions $h_1, \ldots h_n$ defined as*

$$\begin{gathered} h_1(x) = 1,\ h_2(x) = x,\ \ldots\ h_{k+1}(x) = x^k, \\ h_{k+1+j}(x) = \prod_{\ell=1}^{k} (x - x_{j+\ell}) \cdot 1\{x \geq x_{j+k}\}, \quad j = 1, \ldots n-k-1. \end{gathered} \tag{25}$$

If the input points are evenly spaced over $[0,1]$, $x_i = i/n$ for $i = 1, \ldots n$, then the trend filtering basis matrix H in (24) is generated by evaluating the functions $h_1, \ldots h_n$ over $x_1, \ldots x_n$, i.e.,

$$H_{ij} = h_j(x_i), \quad i, j = 1, \ldots n. \tag{26}$$

The proof is given in Appendix A.2. As a result of the lemma, we can alternatively express the trend filtering basis matrix H in (24) as

$$H_{ij} = \begin{cases} i^{j-1}/n^{j-1} & \text{for } i = 1, \ldots n,\ j = 1, \ldots k+1, \\ 0 & \text{for } i \leq j-1,\ j \geq k+2, \\ \prod_{\ell=1}^{k} (i - (j-k-1+\ell))/n^k & \text{for } i > j-1,\ j \geq k+2. \end{cases} \tag{27}$$

This is a helpful form for bounding the difference between the entries of G and H, which is needed for our convergence analysis in the next section. Moreover, the functions defined in (25) give rise to a natural continuous-time parameterization for trend filtering.

Lemma 5. *For inputs $x_1 < \ldots < x_n$, and the functions $h_1, \ldots h_n$ in (25), define the linear subspace of functions*

$$\mathcal{H}_k = \mathrm{span}\{h_1, \ldots h_n\} = \bigg\{ \sum_{j=1}^{n} \alpha_j h_j : \alpha_1, \ldots \alpha_n \in \mathbb{R} \bigg\}. \tag{28}$$

If the inputs are evenly spaced, $x_i = i/n$, $i = 1, \ldots n$, then the continuous-time minimization problem

$$\hat{f} = \underset{f \in \mathcal{H}_k}{\mathrm{argmin}}\ \frac{1}{2} \sum_{i=1}^{n} \big(y_i - f(x_i)\big)^2 + \frac{\lambda}{k!} \cdot \mathrm{TV}(f^{(k)}) \tag{29}$$

(where as before, $f^{(k)}$ is understood to mean the kth weak derivative of f) is equivalent to the trend filtering problem in (2), i.e., their solutions match at the input points,

$$\hat{\beta}_i = \hat{f}(x_i) \quad \textit{for } i = 1, \ldots n.$$

Proof. Expressing f in terms of the finite-dimensional parameterization $f = \sum_{j=1}^{n} \alpha_j h_j$ transforms (29) into the lasso problem (23), with H the basis matrix as in (26); this is then equivalent to the trend filtering problem in (2) by Lemmas 4 and 2. □

Lemma 5 says that the components of trend filtering estimate, $\hat{\beta}_1, \ldots \hat{\beta}_n$, can be seen as the evaluations of a function $\hat{f} \in \mathcal{H}_k$ over the input points, where $\hat{f}$ solves the continuous-time problem (29). The function $\hat{f}$ is a piecewise polynomial of degree k, with knots contained in $\{x_{k+1}, \ldots x_{n-1}\}$, and for $k \geq 1$, it is continuous since each of the basis functions $h_1, \ldots h_n$ are continuous. Hence for $k = 0$ or 1, the continuous-time trend filtering estimate $\hat{f}$ is a spline (and further, it is equal to the locally adaptive regression spline estimate by Lemma 3). But $\hat{f}$ is not necessarily a spline when $k \geq 2$, because in this case it can have discontinuities in its lower order derivatives (of orders $1, \ldots k-1$) at the input points. This is because each basis function h_j, $j = k+2, \ldots n$, though infinitely (strongly)

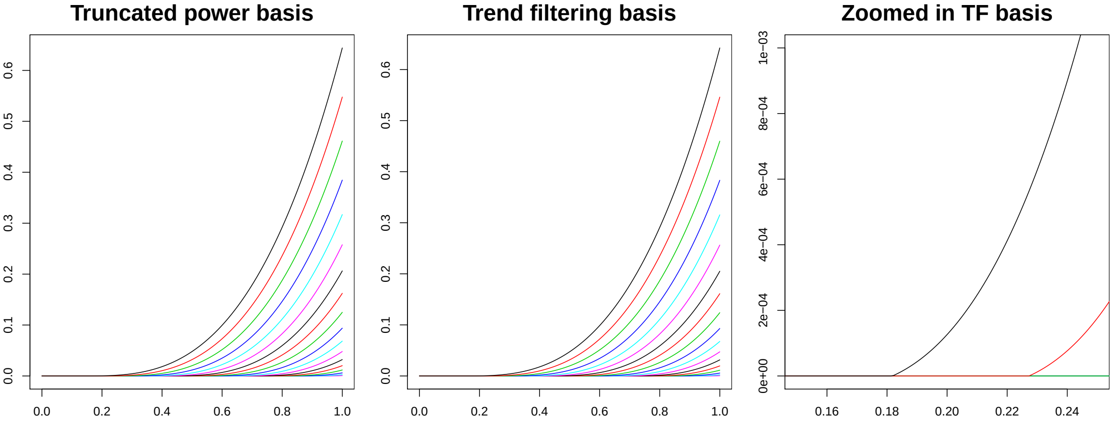

Figure 8: *For $n = 22$ inputs (evenly spaced over $[0, 1]$) and $k = 3$, the left panel shows the truncated power basis functions in* (19) *and the center panel shows the basis functions in* (25) *utilized by trend filtering. The two sets of basis functions appear very similar. The right plot is a zoomed in version of the center plot, and shows the nonsmooth nature of the trend filtering basis functions—here (for $k = 3$) they have discontinuous first and second derivatives.*

differentiable in between the inputs, has discontinuous derivatives of all lower orders $1, \ldots k-1$ at the input point x_{j-1}. These discontinuities are visually quite small in magnitude, and the basis functions $h_1, \ldots h_n$ look extremely similar to the truncated power basis functions $g_1, \ldots g_n$; see Figure 8 for an example.

Loosely speaking, the basis functions $h_1, \ldots h_n$ in (25) can be thought of as the falling factorial analogues of the truncated power basis $g_1, \ldots g_n$ in (19). One might expect then, that the subspaces of kth degree piecewise polynomial functions $\mathcal{H}_k$ and $\mathcal{G}_k$ are fairly close, and that the (continuous-time) trend filtering and locally adaptive regression spline problems (29) and (13) admit similar solutions. In the next section, we prove that (asymptotically) this is indeed the case, though we do so by instead studying the discrete-time parameterizations of these problems. We show that over a broad class of true functions f_0, trend filtering estimates inherit the minimax convergence rate of locally adaptive regression splines, because the two estimators converge to each other at this same rate.

5 Rates of convergence

In this section, we assume that the observations are drawn from the model

$$y_i = f_0(x_i) + \epsilon_i, \quad i = 1, \ldots n, \tag{30}$$

where $x_i = i/n$, $i = 1, \ldots n$ are evenly spaced input points, $f_0 : [0, 1] \to \mathbb{R}$ is an unknown regression function to be estimated, and ϵ_i, $i = 1, \ldots n$ are i.i.d. sub-Gaussian errors with zero mean, i.e.,

$$\mathbb{E}[\epsilon_i] = 0, \;\; \mathbb{P}(|\epsilon_i| > t) \le M \exp\left(-t^2/(2\sigma^2)\right) \;\text{ for all } t > 0, \quad i = 1, \ldots n, \tag{31}$$

for some constants $M, \sigma > 0$. [We will write $A_n = O_{\mathbb{P}}(B_n)$ to denote that A_n/B_n is bounded in probability, for random sequences A_n, B_n. We will also write $a_n = \Omega(b_n)$ to denote $1/a_n = O(1/b_n)$ for constant sequences a_n, b_n, and finally $a_n = \Theta(b_n)$ to denote $a_n = O(b_n)$ and $a_n = \Omega(b_n)$.]

In Mammen & van de Geer (1997), the authors consider the same setup, and study the performance of the locally adaptive regression spline estimate (13) when the true function f_0 belongs to the set

$$\mathcal{F}_k(C) = \Big\{ f : [0,1] \to \mathbb{R} \; : \; f \text{ is } k \text{ times weakly differentiable and } \mathrm{TV}(f^{(k)}) \leq C \Big\},$$

for some order $k \geq 0$ and constant $C > 0$. Theorem 10 of Mammen & van de Geer (1997) shows that the kth order locally adaptive regression spline estimate $\hat{f}$ in (13), with $\lambda = \Theta(n^{1/(2k+3)})$, satisfies

$$\frac{1}{n} \sum_{i=1}^n \big(\hat{f}(x_i) - f_0(x_i)\big)^2 = O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)}), \tag{32}$$

and also that $\mathrm{TV}(\hat{f}) = O_{\mathbb{P}}(1)$.

5.1 Minimax convergence rate

We note that the rate $n^{-(2k+2)/(2k+3)}$ in (32) is the minimax rate for estimation over the function class $\mathcal{F}_k(C)$, provided that $C > 1$. To see this, define the Sobolev smoothness class

$$\mathcal{W}_k(C) = \Big\{ f : [0,1] \to \mathbb{R} \; : \; f \text{ is } k \text{ times differentiable and } \int_0^1 \big(f^{(k)}(t)\big)^2 \, dt \leq C \Big\},$$

Minimax rates over the Sobolev classes are well-studied, and it is known [e.g., see Nussbaum (1985)] that

$$\min_{\hat{f}} \max_{f_0 \in \mathcal{W}_k(C)} \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^n \big(\hat{f}(x_i) - f_0(x_i)\big)^2 \right] = \Omega(n^{-2k/(2k+1)}).$$

Recalling that $\mathrm{TV}(f) = \int_0^1 |f'(t)| \, dt$ for differentiable f, it follows that $\mathcal{F}_k(C) \supseteq \mathcal{W}_{k+1}(C-1)$, and

$$\min_{\hat{f}} \max_{f_0 \in \mathcal{F}_k(C)} \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^n \big(\hat{f}(x_i) - f_0(x_i)\big)^2 \right] = \Omega(n^{-(2k+2)/(2k+3)}).$$

In words, one cannot hope to do better than $n^{-(2k+2)/(2k+3)}$ for function estimation over $\mathcal{F}_k(C)$.

On the other hand, the work of Donoho & Johnstone (1998) provides a lower bound on the rate of convergence over $\mathcal{F}_k(C)$ for any estimate linear in y—by this, we mean that the vector of its fitted values over the inputs is a linear function of y. This covers smoothing splines [recall the expression (11) for the smoothing splines fitted values] and also, e.g., kernel regression estimators. Letting $B^\alpha_{p,q}$ denote the three parameter Besov space as in Donoho & Johnstone (1998), and $\|\cdot\|_{B^\alpha_{p,q}}$ denote the correspsonding norm, we have

$$\begin{aligned} \mathcal{F}_k(C) &\supseteq \Big\{ f : [0,1] \to \mathbb{R} \; : \; \|f^{(k)}\|_\infty + \mathrm{TV}(f^{(k)}) \leq C \Big\} \\ &\supseteq \Big\{ f : [0,1] \to \mathbb{R} \; : \; \|f^{(k)}\|_{B^1_{1,1}} \leq C' \Big\} \\ &\supseteq \Big\{ f : [0,1] \to \mathbb{R} \; : \; \|f\|_{B^{k+1}_{1,1}} \leq C'' \Big\}, \end{aligned} \tag{33}$$

where we write $\|f\|_\infty = \max_{t\in[0,1]} |f(t)|$ for the L_∞ function norm, and C', C'' are constants. The second containment above follows from a well-known embedding of function spaces [e.g., see Mallat (2008), Johnstone (2011)]. The third containment is given by applying the Johnen-Scherer bound on the modulus of continuity [e.g., Theorem 3.1 of DeVore & Lorentz (1993)] when working with the usual definition of the Besov norms.[5] Since the minimax linear risk over the Besov ball in (33) is of order $n^{-(2k+1)/(2k+2)}$ (Donoho & Johnstone 1998), we have[6]

$$\min_{\hat{f} \text{ linear}} \; \max_{f_0 \in \mathcal{F}_k(C)} \; \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n \big(\hat{f}(x_i) - f_0(x_i)\big)^2\right] = \Omega(n^{-(2k+1)/(2k+2)}).$$

Hence, in terms of their convergence rate over $\mathcal{F}_k(C)$, smoothing splines are suboptimal.

5.2 Trend filtering convergence rate

Here we show that trend filtering also achieves the minimax convergence rate over $\mathcal{F}_k(C)$. The arguments used by Mammen & van de Geer (1997) for locally adaptive regression splines cannot be directly applied here, as they are based some well-known interpolating properties of splines that do not easily extend to the trend filtering setting. Our strategy is hence to show that, as $n \to \infty$, trend filtering estimates lie close enough to locally adaptive regression spline estimates to share their favorable asymptotic properties. (Note that for $k = 0$ or $k = 1$, the trend filtering and locally adaptive regression spline estimates are exactly the same for any given problem instance, as shown in Lemma 3 in Section 3; therefore, the arguments here are really directed toward establishing a convergence rate for trend filtering in the case $k \geq 2$.) Using the triangle inequality (actually, using $\|a+b\|_2^2 = \|a\|_2^2 + \|b\|_2^2 + 2a^Tb \leq 2\|a\|_2^2 + 2\|b\|_2^2$), we have

$$\frac{1}{n}\sum_{i=1}^n \big(\hat{\beta}_i - f_0(x_i)\big)^2 \leq \frac{2}{n}\sum_{i=1}^n \big(\hat{\beta}_i - \hat{f}(x_i)\big)^2 + \frac{2}{n}\sum_{i=1}^n \big(\hat{f}(x_i) - f_0(x_i)\big)^2, \tag{34}$$

where $\hat{\beta}, \hat{f}$ are the trend filtering and locally adaptive regression spline estimates in (2), (13), respectively. The second term above is $O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)})$ by (32); if we could show that the first term above is also $O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)})$, then it would follow that trend filtering converges (in probability) to f_0 at the minimax rate.

Recall from Section 3 that both the trend filtering and locally adaptive regression spline estimates can be expressed in terms of the fitted values of lasso problems,

$$\hat{\beta} = H\hat{\alpha}, \quad \big(\hat{f}(x_1), \ldots \hat{f}(x_n)\big) = G\hat{\theta},$$

where $G, H \in \mathbb{R}^{n \times n}$ are the basis matrices in (22), (24), and $\hat{\alpha}, \hat{\theta}$ are the solutions in lasso problems (20), (23). Hence we seek a bound for $\sum_{i=1}^n (\hat{\beta}_i - \hat{f}(x_i))^2 = \|H\hat{\alpha} - G\hat{\theta}\|_2^2$, the (squared norm) difference in fitted values between two lasso problems with the same outcome y, but different predictor matrices G, H. Intuitively, a tight bound is plausible here because G and H have such similar entries (again, for $k = 0$ or $k = 1$, we know that indeed $G = H$).

While there are existing results on the stability of the lasso fit as a function of the outcome vector y [e.g., Tibshirani & Taylor (2012) show that for any fixed predictor matrix and tuning parameter value, the lasso fit is nonexpansive as a function of y], as far as we can tell, general stability results do not exist for the lasso fit as a function of its predictor matrix. To this end, in Appendix B, we develop bounds for the difference in fitted values of two lasso problems that have different predictor matrices, but the same outcome. The bounds are asymptotic in nature, and are driven primarily by the maximum elementwise difference between the predictor matrices. We can apply this work in the current setting to show that the trend filtering and locally adaptive regression spline estimates converge (to each other) at the desired rate, $n^{-(2k+2)/(2k+3)}$; essentially, this amounts to showing that the elements of $G - H$ converge to zero quickly enough.

[5] Thanks to Iain Johnstone for pointing this out.

[6] These authors actually study minimax rates under the L_2 function norm, instead of the discrete (input-averaged) norm that we consider here. However, these two norms are close enough over the Besov spaces that the rates do not change; see Section 15.5 of Johnstone (2011).

Theorem 1. *Assume that $y \in \mathbb{R}^n$ is drawn from the model* (30)*, with evenly spaced inputs $x_i = i/n$, $i = 1, \ldots n$ and i.i.d. sub-Gaussian errors* (31)*. Assume also that $f_0 \in \mathcal{F}_k(C)$, i.e., for a fixed integer $k \geq 0$ and constant $C > 0$, the true function f_0 is k times weakly differentiable and $\mathrm{TV}(f_0^{(k)}) \leq C$. Let $\hat{f}$ denote the kth order locally adaptive regression spline estimate in* (13) *with tuning parameter $\lambda = \Theta(n^{1/(2k+3)})$, and let $\hat{\beta}$ denote the kth order trend filtering estimate in* (2) *with tuning parameter $(1+\delta)\lambda$, for any fixed $\delta > 0$. Then*

$$\frac{1}{n}\sum_{i=1}^{n}\big(\hat{\beta}_i - \hat{f}(x_i)\big)^2 = O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)}).$$

Proof. We use Corollary 4 in Appendix B, with $X = G$ and $Z = H$. First note that our sub-Gaussian assumption in (31) implies that $\mathbb{E}[\epsilon_i^4] < \infty$ (indeed, it implies finite moments of all orders), and with $\mu = (f_0(x_1), \ldots f_0(x_n))$, we know from the result of Mammen & van de Geer (1997), paraphrased in (32), that

$$\|\mu - G\hat{\theta}\|_2 = O_{\mathbb{P}}(n^{-(k+1)/(2k+3)+1/2}) = O_{\mathbb{P}}(\sqrt{n}).$$

Furthermore, the locally adaptive regression spline estimate $\hat{f}$ has total variation

$$\mathrm{TV}(\hat{f}) = \|\hat{\theta}_2\|_1 = O_{\mathbb{P}}(1),$$

where $\hat{\theta}_2$ denotes the last $p_2 = n-k-1$ components of $\hat{\theta}$. Therefore, recalling that $\lambda = \Theta(n^{1/(2k+3)})$, the remaining conditions (57) needed for Corollary 4 reduce to

$$n^{(2k+2)/(2k+3)}\|G_2 - H_2\|_\infty \to 0 \quad \text{as } n \to \infty,$$

where G_2 and H_2 denote the last $n-k-1$ columns of G and H, respectively, and $\|A\|_\infty$ denotes the maximum absolute element of a matrix A. The above limit can be established by using Stirling's formula (and controlling the approximation errors) to bound the elementwise differences in G_2 and H_2; see Lemma 10 in Appendix C. Therefore we apply Corollary 4 to conclude that

$$\|G\hat{\theta} - H\hat{\alpha}\|_2 = O_{\mathbb{P}}(\sqrt{n^{1/(2k+3)}}).$$

Squaring both sides and dividing by n gives the result. □

Remark. It may seem a little strange to choose different tuning parameter values for the locally adaptive regression spline and trend filtering problems [Theorem 1 chooses a tuning parameter λ for the locally adaptive regression spline problem, and a tuning parameter $(1+\delta)\lambda$ for trend filtering], given that we want to equate the fitted values of these two problems. However, it turns out that this "extra" amount of regularization for trend filtering is needed in order to remove the dependence of the final bound on the total variation of the trend filtering estimate ($\|\hat{\alpha}_2\|_1$, in the notation of the proof). See the remark following Lemma 7 in Appendix B.

Now, using the triangle inequality (34) [and recalling the convergence rate of the locally adaptive regression spline estimate (32)], we arrive at the following result.

Corollary 1. *Under the assumptions of Theorem 1, for a tuning parameter value $\lambda = \Theta(n^{1/(2k+3)})$, the kth order trend filtering estimate $\hat{\beta}$ in* (2) *satisfies*

$$\frac{1}{n}\sum_{i=1}^{n}\big(\hat{\beta}_i - f_0(x_i)\big)^2 = O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)}).$$

Hence the trend filtering estimate converges in probability to f_0 at the minimax rate.

Remark. Mammen & van de Geer (1997) prove the analogous convergence result (32) for locally adaptive regression splines using an elegant argument involving metric entropy and the interpolating properties of splines. In particular, a key step in their proof uses the fact that for every $k \geq 0$, and every function $f : [0,1] \to \mathbb{R}$ that has k weak derivatives, there exists a spline $g \in \mathcal{G}_k$ [i.e., g is a spline of degree k with knots in the set T, as defined in (14)] such that

$$\max_{x \in [x_1, x_n]} |f(x) - g(x)| \leq d_k \mathrm{TV}(f^{(k)}) n^{-k} \quad \text{and} \quad \mathrm{TV}(g^{(k)}) \leq d_k \mathrm{TV}(f^{(k)}), \tag{35}$$

where d_k is a constant depending only on k (not on the function f). Following this line of argument for trend filtering would require us to establish the same interpolating properties (35) with $h \in \mathcal{H}_k$ in place of g, where $\mathcal{H}_k$, as defined in (25), (28), is the domain of the continuous-time trend filtering minimization problem in (29). This gets very complicated, as $\mathcal{H}_k$ does not contain spline functions, but instead functions that can have discontinuous lower order derivatives at the input points $x_1, \ldots x_n$. We circumvented such a complication by proving that trend filtering estimates converge to locally adaptive regression spline estimates at a rate equal to the minimax convergence rate (Theorem 1), therefore “piggybacking” on the locally adaptive regression splines rate due to Mammen & van de Geer (1997).

5.3 Functions with growing total variation

We consider an extension to estimation over the function class $\mathcal{F}_k(C_n)$, where now $C_n > 0$ is not necessarily a constant and can grow with n. As in the last section, we rely on a result of Mammen & van de Geer (1997) for locally adaptive regression splines in the same situation, and prove that trend filtering estimates and locally adaptive regression spline estimates are asymptotically very close.

Theorem 2. *Assume that $y \in \mathbb{R}^n$ is drawn from the model* (30)*, with inputs $x_i = i/n$, $i = 1, \ldots n$ and i.i.d. sub-Gaussian errors* (31)*. Assume also that $f_0 \in \mathcal{F}_k(C_n)$, i.e., for a fixed integer $k \geq 0$ and $C_n > 0$ (depending on n), the true function f_0 is k times weakly differentiable and* $\mathrm{TV}(f_0^{(k)}) \leq C_n$*. Let $\hat{f}$ denote the kth order locally adaptive regression spline estimate in* (13) *with tuning parameter $\lambda = \Theta(n^{1/(2k+3)} C_n^{-(2k+1)/(2k+3)})$, and let $\hat{\beta}$ denote the kth order trend filtering estimate in* (2) *with tuning parameter $(1+\delta)\lambda$, for any fixed $\delta > 0$. If C_n does not grow too quickly,*

$$C_n = O(n^{(k+2)/(2k+2)}), \tag{36}$$

then

$$\frac{1}{n} \sum_{i=1}^{n} \big(\hat{\beta}_i - \hat{f}(x_i)\big)^2 = O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)} C_n^{2/(2k+3)}).$$

Proof. The arguments here are similar to the proof of Theorem 1. We invoke Theorem 10 of Mammen & van de Geer (1997), for the present case of growing total variation C_n: this says that

$$\frac{1}{n} \sum_{i=1}^{n} \big(\hat{f}(x_i) - f_0(x_i)\big)^2 = O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)} C_n^{2/(2k+3)}), \tag{37}$$

and also $\mathrm{TV}(\hat{f}) = \|\hat{\theta}_2\|_1 = O_{\mathbb{P}}(C_n)$. Now the conditions for Corollary 4 in Appendix B reduce to

$$n^{(2k+2)/(4k+6)} C_n^{(2k+2)/(2k+3)} \|G_2 - H_2\|_\infty = O(1), \tag{38}$$

$$n^{(2k+2)/(2k+3)} \|G_2 - H_2\|_\infty \to 0 \quad \text{as } n \to \infty. \tag{39}$$

Applying the assumption (36) on C_n, it is seen that both (38), (39) are implied by the condition $n\|G_2 - H_2\|_\infty = O(1)$, which is shown in Lemma 10 in Appendix C. Therefore, we conclude using

Corollary 4 that

$$\sqrt{\sum_{i=1}^{n}\big(\hat{\beta}_i - \hat{f}(x_i)\big)^2} = \|H\hat{\alpha} - G\hat{\theta}\|_2 = O_{\mathbb{P}}\Big(\sqrt{n^{1/(2k+3)}C_n^{2/(2k+3)}}\Big),$$

which gives the rate in the theorem after squaring both sides and dividing by n. □

Finally, we employ the same triangle inequality (34) [and the locally adaptive regression splines result (37) of Mammen & van de Geer (1997)] to derive a rate for trend filtering.

Corollary 2. *Under the assumptions of Theorem 2, for $C_n = O(n^{(k+2)/(2k+2)})$ and a tuning parameter value $\lambda = \Theta(n^{1/(2k+3)}C_n^{-(2k+1)/(2k+3)})$, the kth order trend filtering estimate $\hat{\beta}$ in* (2) *satisfies*

$$\frac{1}{n}\sum_{i=1}^{n}\big(\hat{\beta}_i - f_0(x_i)\big)^2 = O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)}C_n^{2/(2k+3)}).$$

Remark. Although we manage to show that trend filtering achieves the same convergence rate as locally adaptive regression splines in the case of underlying functions with growing total variation, we require the assumption that C_n grows no faster than $O(n^{(k+2)/(2k+2)})$, which is not required for the locally adaptive regression spline result proved in Mammen & van de Geer (1997). But it is worth pointing out that for $k = 0$ or $k = 1$, the restriction $C_n = O(n^{(k+2)/(2k+2)})$ for the trend filtering convergence result is not needed, because in these cases trend filtering and locally adaptive regression splines are exactly the same by Lemma 3 in Section 3.

6 Unevenly spaced inputs

So far, our implicit assumption with trend filtering has been that the inputs $x_1, \ldots x_n$ are evenly spaced. [We have been writing this assumption as $x_i = i/n$ for $i = 1, \ldots n$, but really, it is only the spacings that matter; for a common spacing of $d > 0$ between inputs, if we wanted to compare the trend filtering problem in (2) with, say, the locally adaptive regression spline problem in (13) across λ values, then we would simply replace the factor of n^k in (2) by $1/d^k$.] How could we extend the trend filtering criterion in (2) to account for arbitrarily spaced inputs? One nice feature of the continuous-time representation of trend filtering in (29) is that it provides a natural answer to this question.

For arbitrary input points $x_1 < x_2 < \ldots < x_n$, consider defining the basis matrix H as in (25), (26), and defining the trend filtering estimate by the fitted values $H\hat{\alpha}$ of the problem in (23). Aside from its connection to the continuous-time representation, this definition is supported by the fact that the trend filtering estimates continue to match those from locally adaptive regression splines for polynomial orders $k = 0$ or 1, as they did in the evenly spaced input case. [This follows from the fact that for $k = 0$ or 1, the basis functions $h_1, \ldots h_n$ defined in (25) match the truncated power basis functions $g_1, \ldots g_n$ in (19), with knots as in (12).] Let us write the trend filtering basis matrix as $H^{(x)}$ to emphasize its dependence on the inputs $x_1, \ldots x_n$. To express the fitted values $H^{(x)}\hat{\alpha}$ in a more familiar form, we seek a matrix $D^{(x,k+1)} \in \mathbb{R}^{(n-k-1)\times n}$ so that the estimate

$$\hat{\beta} = \underset{\beta\in\mathbb{R}^n}{\mathrm{argmin}}\ \frac{1}{2}\|y - \beta\|_2^2 + \frac{\lambda}{k!}\|D^{(x,k+1)}\beta\|_1 \tag{40}$$

satisfies $\hat{\beta} = H^{(x)}\hat{\alpha}$. Note that $D^{(x,k+1)}$ is given precisely by the last $n-k-1$ rows of $(H^{(x)})^{-1}/k!$. For $k = 0$, it is easy to see that $D^{(x,1)} = D^{(1)}$, the first difference matrix (for evenly spaced inputs) given in (3). For $k \geq 1$, an inductive calculation (similar to the proofs of Lemmas 2, 4) shows that

$$D^{(x,k+1)} = D^{(1)} \cdot \mathrm{diag}\left(\frac{k}{x_{k+1} - x_1}, \frac{k}{x_{k+2} - x_2}, \ldots \frac{k}{x_n - x_{n-k}}\right) \cdot D^{(x,k)}. \tag{41}$$

[The leading $D^{(1)}$ above is the $(n-k-1) \times (n-k)$ version of the first difference matrix in (3).] This parallels the definition of the $(k+1)$st difference matrix in (4), and hence (as our notation would suggest), $D^{(x,k+1)}$ can still be thought of as a difference operator of order $k+1$, but adjusted to account for the unevenly spaced inputs $x_1, \ldots x_n$.

In (41), multiplication by the diagonal matrix of spacing weights does not cause any change in structure, so $D^{(x,k+1)}$ is still banded with bandwidth $k+2$. Therefore, in principle, the trend filtering problem in (40) is not any harder computationally than the original problem in (2) for evenly spaced inputs, because algorithms like the primal-dual interior point method of Kim et al. (2009) and the dual path algorithm of Tibshirani & Taylor (2011) only rely on such bandedness for efficient calculations. In practice, too, these algorithms are able to efficiently handle the extension to unevenly spaced input points, in the majority of cases. However, when dealing with inputs that have highly irregular spacings, numerical accuracy can become an issue.[7] Robustifying trend filtering algorithms to handle these difficult cases is a direction for future work.

On the theoretical side, the same strategy used to derive the convergence rates in Section 5 for evenly spaced input points can be applied to the unevenly spaced case, as well. Recall that the basic idea was to tie trend filtering estimates together asymptotically with locally adaptive regression splines, at a tight enough rate that trend filtering estimates inherit the (known) convergence rates of the latter estimators. Appendix B provides the technical framework for tying together these two estimators, which can be seen as the fitted values of two lasso problems. The asymptotic bounds between the two estimators, in the current setting, depend primarily on the maximum elementwise difference between $G^{(x)}$, the truncated power basis matrix in (19), (17) evaluated at the inputs $x_1, \ldots x_n$, and $H^{(x)}$, the trend filtering basis matrix in (25), (26) evaluated at the inputs $x_1 \ldots x_n$. We state the following convergence result without proof, since it follows from similar arguments to those in Section 5.

Theorem 3. *Assume that $y \in \mathbb{R}^n$ is drawn from the model* (30), *with inputs $x_1 < \ldots < x_n \in [0,1]$ and i.i.d. sub-Gaussian errors* (31). *Assume also that $f_0 \in \mathcal{F}_k(C_n)$, i.e., for a fixed integer $k \geq 0$ and $C_n > 0$ (depending on n), the true function f_0 is k times weakly differentiable and* $\mathrm{TV}(f_0^{(k)}) \leq C_n$. *Let $\hat{f}$ denote the kth order locally adaptive regression spline estimate in* (13) *with tuning parameter $\lambda = \Theta(n^{1/(2k+3)} C_n^{-(2k+1)/(2k+3)})$, and let $\hat{\beta}$ denote the kth order trend filtering estimate in* (40) *with tuning parameter $(1+\delta)\lambda$, for any fixed $\delta > 0$. Finally, if $k \geq 2$, then we must assume that the following conditions are met: C_n does not grow too quickly,*

$$C_n = O(n^{(k+2)/(2k+2)}),$$

and the input points $x_1, \ldots x_n$ satisfy

$$\max_{i=1,\ldots n-1} (x_{i+1} - x_i) = O(n^{-(k+1)/(k(2k+3))} C_n^{-(2k+2)/(k(2k+3))}), \tag{42}$$

$$\max_{i=1,\ldots n-k-1} \left| \prod_{\ell=1}^{k} (x_n - x_{i+\ell}) - (x_n - x_{i+\lfloor (k+2)/2 \rfloor})^k \right| = O(1/n). \tag{43}$$

(Above, we use $\lfloor \cdot \rfloor$ to denote the floor function.) Then

$$\frac{1}{n} \sum_{i=1}^{n} \big(\hat{\beta}_i - \hat{f}(x_i)\big)^2 = O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)} C_n^{2/(2k+3)}),$$

[7] Recall that these algorithms iteratively solve linear systems in $D^{(x,k+1)}(D^{(x,k+1)})^T$; each system requires $O(n)$ operations, regardless of the inputs $x_1, \ldots x_n$. However, if $x_1, \ldots x_n$ have highly irregular spacings, with some points being very close together and some quite far apart, then $D^{(x,k+1)}$ can contain both very large and very small elements, which can cause numerical inaccuracies when solving the linear systems.

and furthermore

$$\frac{1}{n}\sum_{i=1}^{n}\big(\hat{\beta}_i - f_0(x_i)\big)^2 = O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)}C_n^{2/(2k+3)}).$$

Hence, if the kth derivative of f_0 has bounded total variation (i.e., C_n is a constant), then the trend filtering estimate converges to f_0 in probability at the minimax rate.

Remark. The conditions in Theorem 3 should all look familiar to those in Theorems 1 and 2 from Section 5, except for the design conditions (42), (43). The first condition (42) is needed by Mammen & van de Geer (1997) for their convergence result on locally adaptive splines for unevenly spaced inputs,

$$\frac{1}{n}\sum_{i=1}^{n}\big(\hat{f}(x_i) - f_0(x_i)\big)^2 = O_{\mathbb{P}}(n^{-(2k+2)/(2k+3)}C_n^{2/(2k+3)}).$$

In words, condition (42) maintains that no pair of adjacent inputs should be too far apart (and is clearly satisfied for evenly spaced inputs over $[0,1]$). The second condition (43) is sufficient to imply

$$\|G_2^{(x)} - H_2^{(x)}\|_\infty = O(1/n),$$

where $G_2^{(x)}, H_2^{(x)}$ are the last $n-k-1$ columns of $G^{(x)}, H^{(x)}$ (the locally adaptive regression splines and trend filtering basis matrices, respectively). This enables us to apply Corollary 4 in Appendix B to bound the difference between the trend filtering and locally adaptive regression spline estimates. (Recall that Lemma 10 in Appendix C establishes the above condition for evenly spaced inputs.) At this point, assuming the condition (43) in Theorem 3 seems to be the most explicit way of ensuring the bound between $G^{(x)}, H^{(x)}$ that is needed for Corollary 4, but we feel that this condition can be made even more explicit, and likely, simplified. This is left for future work.

7 Astrophysics data example

We examine data from an astrophysics simulation model for quasar spectra, provided by Yu Feng, with help from Mattia Ciollaro and Jessi Cisewski. Quasars are among the most luminous objects in the universe. Because of this, we can observe them at great distances, and features in their spectra reveal information about the universe along the line-of-sight of a given quasar. A quasar spectrum drawn from this model is displayed in the top left panel of Figure 9. The spectrum, in black, shows the flux (y-axis) as a function of log wavelength (x-axis). Noisy realizations of this true curve are plotted as gray points, measured at $n = 1172$ points, (approximately) equally spaced on the log wavelength scale. We see that the true function has a dramatically different level of smoothness as it traverses the wavelength scale, being exceptionally wiggly on the left side of the domain, and much smoother on the right. (The error variance is also seen to be itself inhomogeneous, with larger errors around the wiggly portions of the curve, but for simplicity we do not account for this.) The wiggly left side of the spectrum are absorption features called the Lyman-alpha forest. For more information about the Lyman-alpha forest, including a recent Sloan Digital Sky Survey data release of over 50,000 quasar spectra, see Lee et al. (2012).

To estimate the underlying function, we applied trend filtering, smoothing splines, and wavelet smoothing, each over 146 values of degrees of freedom (from 4 to 150 degrees to freedom). Locally adaptive regression splines were not compared because of their extreme proximity to trend filtering. The `smooth.spline` function in R was used to fit the smoothing spline estimates, and because it produces cubic order smoothing splines, we considered cubic order trend filtering and wavelets with 4 vanishing moments to put all of the methods on more or less equal footing. Wavelet smoothing was fit using the `wavethresh` package in R, and the "wavelets on the interval" option was chosen to handle the boundary conditions (as periodicity and symmetry are not appropriate assumptions for the boundary behavior in this example), which uses an algorithm of Cohen et al. (1993). Wavelet transforms generally require the number of observations to be a power of 2 (this is at least true in the `wavethresh` implementation), and so we restricted the wavelet smoothing estimate to use the first 1024 points with the smallest log wavelengths.

Figure 9 demonstrates the function estimates from these methods, run on the single data set shown in the top left panel (the observations are not drawn in the remaining panels so as not to cloud the plots). Each estimate was tuned to have 81 degrees of freedom. We can see that trend filtering (top right panel) captures many features of the true function, picking up the large spike just before $x = 3.6$, but missing some of the action on the left side. The smoothing spline estimate (bottom left) appears fairly similar, but it does not fit the magnitudes of the wiggly components as well. Wavelet smoothing (bottom right) detects the large spike, but badly overfits the true function to the left of this spike, and even misses gross smoothness features to the right.

We further compared the three contending methods by computing their average squared error loss to the true function, over 20 draws from the simulated model. This is shown in the left panel of Figure 10. Trend filtering outperforms smoothing splines for lower values of model complexity (degrees of freedom); this can be attributed to its superior capability for local adaptivity, a claim both empirically supported by the simulations in Section 2.2, and formally explained by the theory in Section 5. Wavelet smoothing is not competitive in terms of squared error loss. Although in theory it achieves the same (minimax) rate of convergence as trend filtering, it seems in the current setting to be hurt by the high noise level at the left side of the domain; wavelet smoothing overfits in this region, which inflates the estimation variance.

Finally, we compared trend filtering to a smoothing spline estimator whose tuning parameter varies over the input domain (to yield a finer level of local adaptivity). For an example of a recent proposal of such an estimator, see Wang et al. (2013) (see also the references therein). Methods that fit a flexibly varying tuning parameter over the domain can become very complicated, and so to simplify matters for the quasar spectrum data, we allowed the smoothing spline estimator two different tuning parameters λ_1, λ_2 to the left and right of $x = 3.6$. Note that this represents somewhat of an ideal scenario for variable parameter smoothing splines, as we fixed an appropriate division of the domain based on knowledge of the true function. It should also be noted that we fit the split smoothing spline estimator over a total of $146 \cdot 146 = 23,1316$ values of degrees of freedom (146 in each half of the domain), which puts it at an advantage over the other methods. See the right panel of Figure 10 for the results. Over 20 simulated data sets, we fit split smoothing splines whose degrees of freedom d_1, d_2 on the left and right sides of $x = 3.6$ ranged from 4 to 150. For each value of degrees of freedom d, the plotted curve shows the minimum squared error loss over all models with $d_1 + d_2 = d$. Interestingly, despite all of the advantages imparted by its setup, the split smoothing spline estimator performs basically on par with trend filtering.

8 Extensions and discussion

We have shown that trend filtering, a newly proposed method for nonparametric regression of Kim et al. (2009), is both fast and locally adaptive. Two of the major tools for adaptive spline estimation are smoothing splines and locally adaptive regression splines; in short, the former estimators are fast but not locally adaptive, and the latter are locally adaptive but not fast. Trend filtering lies in a comparatively favorable position: its estimates can be computed in $O(n^{3/2})$ worst-case complexity (at a fixed value of the tuning parameter λ, using a primal-dual interior point algorithm), which is slower than the $O(n)$ complexity of smoothing splines, but not by a big margin; its estimates also achieve the same convergence rate as locally adaptive regression splines over a broad class of underlying functions (which is, in fact, the minimax rate over this class).

One way to construct trend filtering estimates, conceptually, is to start with the lasso form for locally adaptive regression splines (20), but then replace the matrix G in (22), which is generated

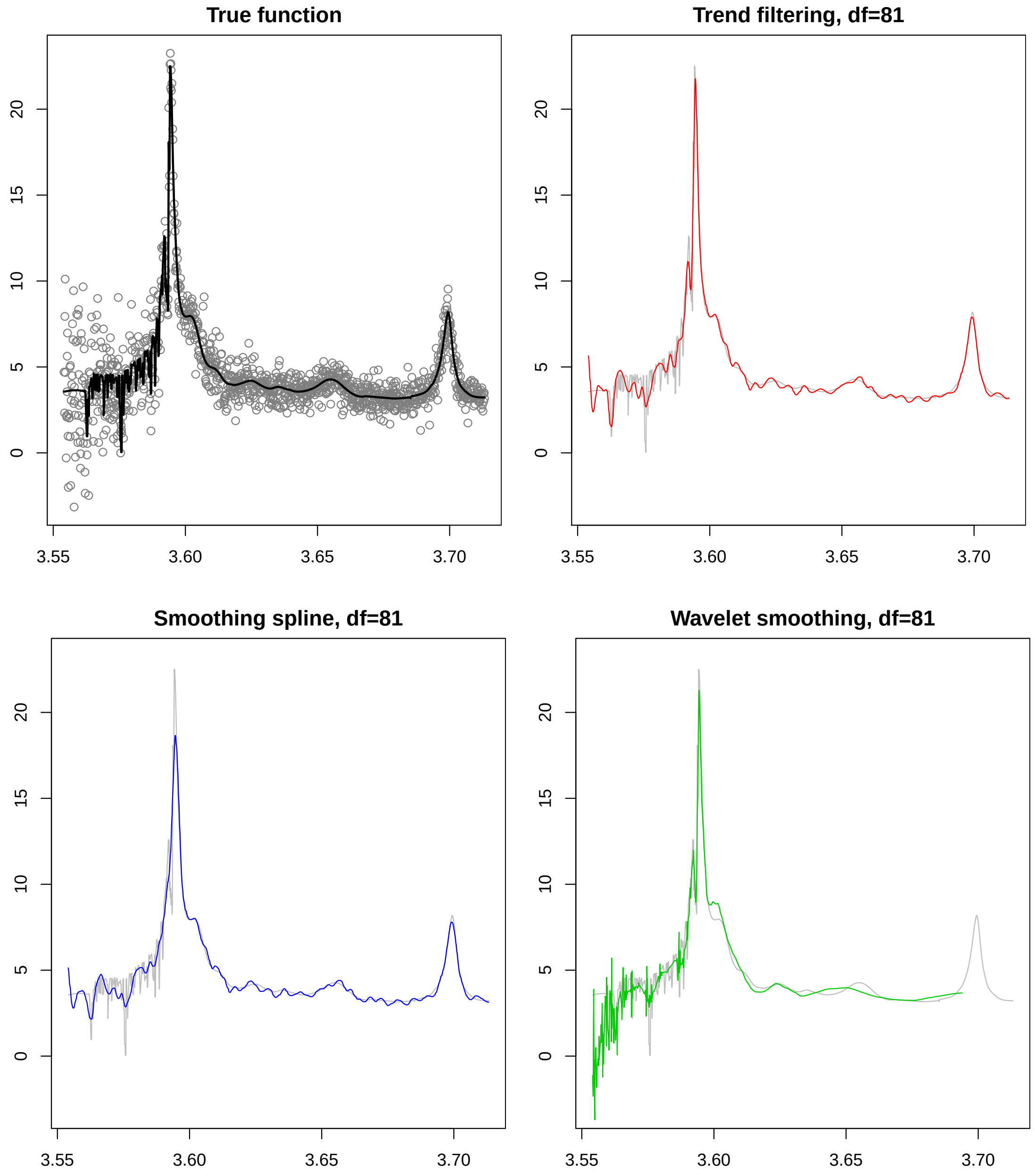

Figure 9: *The top left panel shows data simulated from a model for quasar spectrum. The true curve, in black, displays flux as a function of log wavelength. The gray points are noisy observations at* $n = 1172$ *wavelength values. We fit trend filtering, smoothing splines, and wavelets to these points, and tuned each to have 81 degrees of freedom. (This value was chosen because it corresponded to the trend filtering model with the minimum squared error loss, averaged 20 simulations—see Figure 10.) The resulting estimates are displayed in the top right, bottom left, and bottom right panels, respectively (with the true function plotted in the background, in gray). Trend filtering and smoothing splines give similar fits, except that trend filtering does a better job of estimating the large peak at around* $x = 3.6$*, as well as some of the finer features of the true function to the left of this. Wavelet smoothing also does well in detecting the extreme peak, but then overfits the true function on the left side, and underfits on the right.*

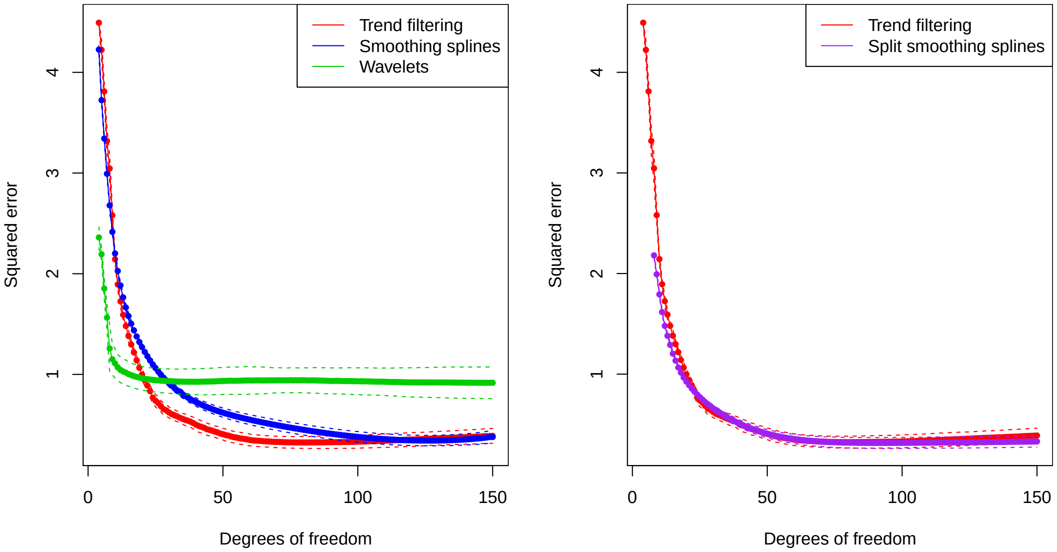

Figure 10: *Left plot: the squared error loss in estimating the true light curve for the quasar spectrum data, using trend filtering, smoothing splines, and wavelets, fit over a range of model complexities (degrees of freedom values). The results were averaged over 20 simulated data sets, with standard errors drawn as dotted lines. Trend filtering achieves a significantly lower squared error loss than smoothing splines for models of low complexity; both perform much better than wavelets. Right plot: trend filtering versus a smoothing spline estimator that fits a different smoothing parameter on two halves of the domains (on either side of the peak around* $x = 3.6$*); the methods perform comparably.*

by the truncated power series, with the matrix H in (27), generated by something like their falling factorial counterparts. This precisely defines trend filtering, and it has the distinct computational advantage that H has a sparse banded inverse (whereas the inverse of G is dense). Moreover, the matrix H is close enough to G that trend filtering estimates retain some of the desirable theoretical properties of locally adaptive regression splines, i.e., their minimax rate of convergence. Although this change-of-basis perspective is helpful for the purposes of mathematical analysis, the original representation for trend filtering (2) is certainly more natural, and also perhaps more useful for constructing related estimators whose characteristics go beyond (piecewise) polynomial smoothness of a given order. We finish by discussing this, in Section 8.2. First, we briefly discuss an extension to multivariate inputs.

8.1 Multivariate trend filtering

An important extension concerns the case of multivariate inputs $x_1, \ldots x_n \in \mathbb{R}^p$. In this case, there are two strategies for extending trend filtering that one might consider. The first is to extend the definition of the discrete difference operators to cover multivariate inputs—the analogous extension here for smoothing splines are thin plate splines (Wahba 1990, Green & Silverman 1994). An extension such as this is "truly" multivariate, and is an ambitious undertaking; even just the construction of an appropriate multivariate discrete difference operator is a topic deserving its own study.

A second, more modest approach for multivariate input points is to fit an additive model whose individual component functions are fit by (univariate) trend filtering. Hastie & Tibshirani (1990) introduced additive models, of the form

$$y_i = \sum_{j=1}^{p} f_j(x_{ij}) + \epsilon_i, \quad i = 1, \dots n. \tag{44}$$

The model (44) considers the contributions from each variable in the input space marginally. Its estimates will often scale better with the underlying dimension p, both in terms of computational and statistical efficiency, when compared to those from a "true" multivariate extension that considers variables jointly. Fitting the component functions $\hat{f}_1, \dots \hat{f}_p$ is most often done with a backfitting (blockwise coordinate descent) procedure, where we cycle through estimating each $\hat{f}_j$ by fitting the current residual to the jth variable, using a univariate nonparametric regression estimator. Common practice is to use smoothing splines for these individual univariate regressions, but given their improved adaptivity properties and comparable computational efficiency, using trend filtering estimates for these inner regressions is an idea worth investigating.

8.2 Synthesis versus analysis

Synthesis and analysis are concepts from signal processing that, roughly speaking, describe the acts of building up an estimator by adding together a number of fundamental components, respectively, whittling down an estimator by removing certain undesirable components. The same terms are also used to convey related concepts in many scientific fields. In this section, we compare synthesis and analysis in the context of ℓ_1 penalized estimation. Suppose that we want to construct an estimator of $y \in \mathbb{R}^n$ with some particular set of desired characteristics, and consider the following two general problem formulations:

$$\min_{\theta \in \mathbb{R}^p} \frac{1}{2}\|y - X\theta\|_2^2 + \lambda\|\theta\|_1 \tag{45}$$

$$\min_{\beta \in \mathbb{R}^n} \frac{1}{2}\|y - \beta\|_2^2 + \lambda\|D\beta\|_1. \tag{46}$$

The first form (45) is the synthesis approach: here we choose a matrix $X \in \mathbb{R}^{n \times p}$ whose columns are atoms or building blocks for the characteristics that we seek, and in solving the synthesis problem (45), we are adaptively selecting a number of these atoms to form our estimate of y. Problem (46), on the other hand, is the analysis approach: instead of enumerating an atom set via X, we choose a penalty matrix $D \in \mathbb{R}^{m \times n}$ whose rows represent uncharacteristic behavior. In solving the problem (46), we are essentially orthogonalizing our estimate with respect to some adaptively chosen rows of D, therefore directing it away from uncharacteristic behavior.

The original representation of trend filtering in (2) falls into the analysis framework, with $D = D^{(k+1)}$, the $(k+1)$st order discrete difference operator; its basis representation in (23) falls into the synthesis framework, with $X = H$, the falling factorial basis matrix (an unimportant difference is that the ℓ_1 penalty only applies to part of the coefficient vector). In the former, we shape the trend filtering estimate by penalizing jumps in its $(k+1)$st discrete derivative across the input points; in the latter, we build it from a set of basis functions, each of which is nonsmooth at only one different input point. Generally, problems (45) and (46) can be equated if D has full row rank (as it does with trend filtering), but not if D is row rank deficient [see Tibshirani & Taylor (2011), Elad et al. (2007)].

Here we argue that it can actually be easier to work from the analysis perspective instead of the synthesis perspective for the design of nonparametric regression estimators. (The reverse can also be true in other situations, though that is not our focus.) For example, suppose that we wanted to construct an estimator that displays piecewise polynomial smoothness across the input points, but additionally, is identically zero over some appropriately chosen subintervals in its domain. It helps to see an example: see the left panel in Figure 11. Working from the analysis point of view, such an estimate is easily achieved by adding a pure ℓ_1 penalty to the usual trend filtering criterion, as in

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^n}{\operatorname{argmin}} \ \frac{1}{2}\|y - \beta\|_2^2 + \lambda_1 \|D^{(k+1)}\beta\|_1 + \lambda_2\|\beta\|_1. \tag{47}$$

We call (47) the sparse trend filtering estimate. This could be of interest if, e.g., $y \in \mathbb{R}^n$ is taken to be the pairwise differences between two sequences of observations, e.g., between two response curves over time; in this case, the zeros of $\hat{\beta}$ indicate regions in time over which the two responses are deemed to be more or less the same. It is important to note that an estimate with these properties seems difficult to construct from the synthesis perspective—it is unclear what basis elements, when added together, would generically yield an estimate like that in (47).

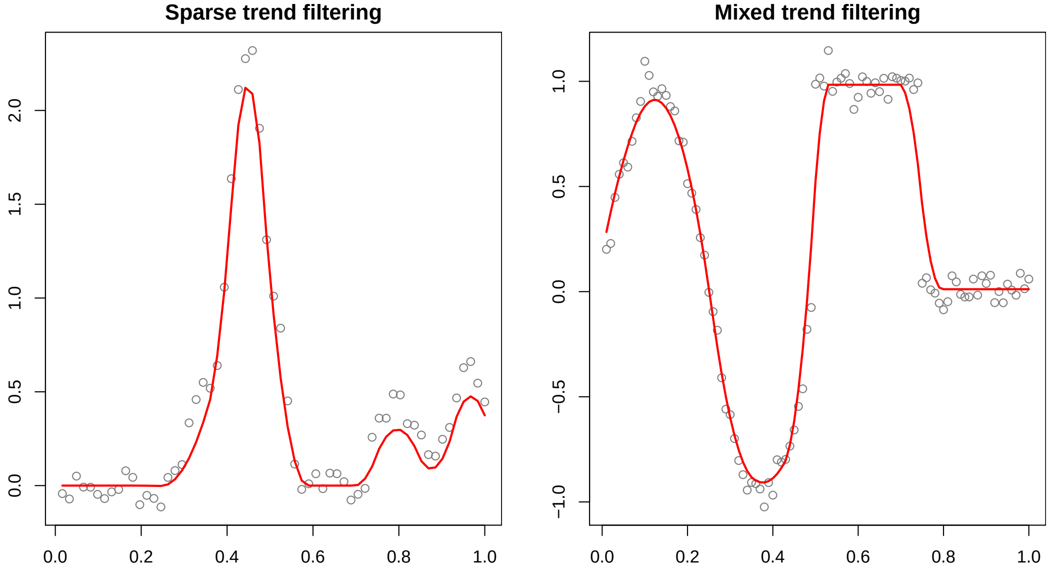

Figure 11: *Left panel: a small example of sparse quadratic trend filtering ($k = 2$). The estimate $\hat{\beta}$ in* (47) *is identically zero for inputs approximately between 0 and 0.25, and 0.6 and 0.75. Right panel: an example of constant/quadratic mixed trend filtering ($k_1 = 0$ and $k_2 = 2$). The estimate defined in* (48) *is first piecewise quadratic over the first half of its domain, but then is flat in two stretches over the second half.*

As another example, suppose that we had prior belief that the observations $y \in \mathbb{R}^n$ were drawn from an underlying function that possesses different orders of piecewise polynomial smoothness, k_1 and k_2, at different parts of its domain. We could then solve the mixed trend filtering problem,

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^n}{\operatorname{argmin}} \ \frac{1}{2}\|y - \beta\|_2^2 + \lambda_1 \|D^{(k_1+1)}\beta\|_1 + \lambda_2\|D^{(k_2+1)}\beta\|_1. \tag{48}$$

The right panel of Figure 11 shows an example, with an underlying function that is mixed piecewise quadratic and piecewise constant. Again it seems much more difficult to construct an estimate like (48), i.e., one that can flexibly adapt to the appropriate order of smoothness at different parts of its domain, using the synthesis framework. Further study of the synthesis versus analysis perspectives for estimator construction will be pursued in future work.

Acknowledgements

We thank the reviewing editors and referees for their comments, which lead to an improved paper. We thank Alessandro Rinaldo and Larry Wasserman for many helpful and encouraging conversations. We thank Iain Johnstone for his generous help regarding Besov spaces and the corresponding minimax rates, in Section 5.1. We also thank Mattia Ciollaro and Jessi Cisewski for their help with the astrophysics example in Section 7. Finally, we thank Trevor Hastie and Jonathan Taylor for early stimulating conversations that provided the inspiration to work on this project in the first place. This work was supported by NSF grant DMS-1309174.

A Lasso and continuous-time representations

A.1 Proof of Lemma 2

Consider making the variable transformation $\alpha = n^k/k! \cdot D\beta$ in (2), with $D \in \mathbb{R}^{n\times n}$ defined as

$$D = \begin{bmatrix} D_1^{(0)} \\ \vdots \\ D_1^{(k)} \\ D^{(k+1)} \end{bmatrix},$$

where $D_1^{(i)} \in \mathbb{R}^{1\times n}$ denotes the first row of the ith discrete difference operator $D^{(i)}$, for $i = 0, \ldots k$ (and $D^{(0)} = I$ by convention). We first show that $D^{-1} = M$, where $M = M^{(0)} \cdot \ldots \cdot M^{(k)}$ and

$$M^{(i)} = \begin{bmatrix} I_{i\times i} & 0 \\ 0 & L_{(n-i)\times(n-i)} \end{bmatrix} \quad \text{for } i = 0, \ldots k.$$

Here $I_{i\times i}$ is the $i \times i$ identity matrix, and $L_{(n-i)\times(n-i)}$ is the $(n-i)\times(n-i)$ lower triangular matrix of 1s. In particular, we prove that $M^{-1} = D$ by induction on k. When $k = 0$, that

$$\begin{bmatrix} 1 & 0 & 0 & \ldots & 0 \\ 1 & 1 & 0 & \ldots & 0 \\ 1 & 1 & 1 & \ldots & 0 \\ \vdots & & & & \\ 1 & 1 & 1 & \ldots & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 & 0 & \ldots & 0 & 0 \\ -1 & 1 & 0 & \ldots & 0 & 0 \\ 0 & -1 & 1 & \ldots & 0 & 0 \\ \vdots & & & & & \\ 0 & 0 & 0 & \ldots & -1 & 1 \end{bmatrix}$$

can be seen directly by inspection. Assume now that the statement holds for $k-1$. Then

$$\begin{aligned} \Big(M^{(0)} \cdot \ldots \cdot M^{(k)}\Big)^{-1} &= (M^{(k)})^{-1}\Big(M^{(0)} \cdot \ldots \cdot M^{(k-1)}\Big)^{-1} \\ &= \begin{bmatrix} I & 0 \\ 0 & L^{-1} \end{bmatrix} \begin{bmatrix} D_1^{(0)} \\ \vdots \\ D_1^{(k-1)} \\ D^{(k)} \end{bmatrix} \\ &= \begin{bmatrix} D_1^{(0)} \\ \vdots \\ D_1^{(k-1)} \\ L^{-1}D^{(k)} \end{bmatrix}, \end{aligned}$$

where we have abbreviated $I = I_{k\times k}$ and $L = L_{(n-k)\times(n-k)}$, and in the second equality we used the inductive hypothesis. Moreover,

$$L^{-1}D^{(k)} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & \dots & 0 & 0 \\ \vdots & & & & & \\ 0 & 0 & 0 & \dots & -1 & 1 \end{bmatrix} D^{(k)} = \begin{bmatrix} D_1^{(k)} \\ D^{(k+1)} \end{bmatrix},$$

completing the inductive proof. Therefore, substituting $\alpha = n^k/k! \cdot D\beta$ in (2) yields the problem

$$\hat{\alpha} = \operatorname*{argmin}_{\alpha\in\mathbb{R}^n} \frac{1}{2}\Big\|y - \frac{k!}{n^k}M\alpha\Big\|_2^2 + \lambda \sum_{j=k+2}^{n} |\alpha_j|.$$

It is now straightforward to check that the last $n-k-1$ columns of $(k!/n^k)M$ match those of H, as defined in (24) in the lemma. Further, the first $k+1$ columns of $(k!/n^k)M$ have the same linear span as the first $k+1$ columns of H, which is sufficient because the ℓ_1 penalty above [and in (23)] only applies to the last $n-k-1$ coefficients. □

A.2 Proof of Lemma 4

Define $s_i^{(0)} = 1$ for all i, and

$$s_i^{(k)} = \sum_{j=1}^{i-1} s_j^{(k-1)} \quad \text{for } k = 1,2,3,\dots,$$

i.e., $s_i^{(k)}$ is the kth order cumulative sum of $(1,1,\dots 1) \in \mathbb{R}^i$, with lag 1. We will prove that

$$\frac{(x-k)(x-k+1)\cdot\ldots\cdot(x-1)}{k!} = s_x^{(k)} \quad \text{for all } x = 1,2,3,\dots \text{ and } k = 1,2,3,\dots, \tag{49}$$

by induction on k. Note that this would be sufficient to prove the result in the lemma, as it would show that the bottom right $(n-k-1)\times(n-k-1)$ sub-block of H in (24), which can be expressed as

$$\frac{k!}{n^k} \cdot \begin{bmatrix} s_{k+1}^{(k)} & 0 & \dots & 0 \\ s_{k+2}^{(k)} & s_{k+1}^{(k)} & \dots & 0 \\ \vdots & & & \\ s_{n-1}^{(k)} & s_{n-2}^{(k)} & \dots & s_{k+1}^{(k)} \end{bmatrix},$$

is equal to that in (27). We now give the inductive argument for (49). As for the base case, $k=1$: clearly $x - 1 = s_x^{(1)}$ for all x. Assume that the inductive hypothesis holds for k. Then for any x,

$$\begin{aligned} s_x^{(k+1)} &= \sum_{i=1}^{x-1} \frac{(i-k)(i-k+1)\cdot\ldots\cdot(i-1)}{k!} \\ &= \sum_{i=1}^{x-1}\sum_{j=1}^{i-1} \frac{(j-k+1)(j-k+2)\cdot\ldots\cdot(j-1)}{(k-1)!}, \end{aligned}$$

by the inductive hypothesis. Switching the order of the summations,

$$\begin{aligned} s_x^{(k+1)} &= \sum_{j=1}^{x-2} \sum_{i=j+1}^{x-1} \frac{(j-k+1)(j-k+2)\cdot \ldots \cdot (j-1)}{(k-1)!} \\ &= \sum_{j=1}^{x-2} \frac{(j-k+1)(j-k+2)\cdot \ldots \cdot (j-1)}{(k-1)!} \cdot (x-j-1) \\ &= \sum_{j=1}^{x-2} \frac{(j-k+1)(j-k+2)\cdot \ldots \cdot (j-1)}{(k-1)!} \cdot (x-k-1-j+k). \end{aligned}$$

Grouping terms and again applying the inductive hypothesis,

$$s_x^{(k+1)} = \frac{(x-k-1)(x-k-2)\cdot \ldots \cdot (x-2)}{k!} \cdot (x-k-1) - s_{x-1}^{(k+1)} \cdot k.$$

Noting that $s_x^{(k+1)} = s_{x-1}^{(k+1)} + (x-k-1)\cdot \ldots \cdot (x-2)/k!$, and rearranging terms finally gives

$$s_{x-1}^{(k+1)} = \frac{(x-k-2)(x-k-1)\cdot \ldots \cdot (x-2)}{(k+1)!}.$$

Since x was arbitrary, this completes the inductive step, and hence the proof. □

B Bounding the difference in lasso fitted values

B.1 Lasso problems in standard form

Consider two lasso problems sharing the same outcome vector $y \in \mathbb{R}^n$,

$$\min_{\theta \in \mathbb{R}^p} \frac{1}{2}\|y - X\theta\|_2^2 + \lambda \|\theta\|_1, \tag{50}$$

$$\min_{\alpha \in \mathbb{R}^p} \frac{1}{2}\|y - Z\alpha\|_2^2 + \lambda' \|\alpha\|_1, \tag{51}$$

where $X, Z \in \mathbb{R}^{n\times p}$, and $\lambda, \lambda' \geq 0$. One might ask: if λ, λ' are chosen appropriately, can we bound the difference in the fitted values $X\hat{\theta}$ and $Z\hat{\alpha}$ of (50) and (51), respectively, in terms of the difference between X and Z? This question can be answered by first deriving a "basic inequality", much like the one used for bounding lasso prediction error.

Lemma 6. *For fixed λ, λ', solutions $\hat{\theta}$ and $\hat{\alpha}$ of lasso problems* (50) *and* (51) *satisfy*

$$\frac{1}{2}\|X\hat{\theta} - Z\hat{\alpha}\|_2^2 \leq \langle y - X\hat{\theta}, Z\hat{\alpha}\rangle + (\lambda' - \lambda)\|\hat{\theta}\|_1 - \lambda'\|\hat{\alpha}\|_1 + R, \tag{52}$$

where $R = \frac{1}{2}\|(X-Z)\hat{\theta}\|_2^2 + \langle y - X\hat{\theta}, (X-Z)\hat{\theta}\rangle$.

Proof. Note that by optimality,

$$\begin{aligned} \frac{1}{2}\|y - Z\hat{\alpha}\|_2^2 + \lambda'\|\hat{\alpha}\|_1 &\leq \frac{1}{2}\|y - Z\hat{\theta}\|_2^2 + \lambda'\|\hat{\theta}\|_1 \\ &= \frac{1}{2}\|y - X\hat{\theta}\|_2^2 + \lambda'\|\hat{\theta}\|_1 + R, \end{aligned}$$

where $R = \frac{1}{2}\|y - Z\hat{\theta}\|_2^2 - \frac{1}{2}\|y - X\hat{\theta}\|_2^2$. We can rearrange the above inequality as

$$\frac{1}{2}\|Z\hat{\alpha}\|_2^2 - \frac{1}{2}\|X\hat{\theta}\|_2^2 \le \langle y, Z\hat{\alpha} - X\hat{\theta}\rangle + \lambda'\|\hat{\theta}\|_1 - \lambda'\|\hat{\alpha}\|_1 + R.$$

Writing $y = X\hat{\theta} + (y - X\hat{\theta})$ within the inner product on the right-hand side above, and bringing the term involving $X\hat{\theta}$ over to the left-hand side, we have

$$\begin{aligned}\frac{1}{2}\|X\hat{\theta} - Z\hat{\alpha}\|_2^2 &\le \langle y - X\hat{\theta}, Z\hat{\alpha} - X\hat{\theta}\rangle + \lambda'\|\hat{\theta}\|_1 - \lambda'\|\hat{\alpha}\|_1 + R, \\ &= \langle y - X\hat{\theta}, Z\hat{\alpha}\rangle + (\lambda' - \lambda)\|\hat{\theta}\|_1 - \lambda'\|\hat{\alpha}\|_1 + R,\end{aligned}$$

where in the last line we used the fact that $\langle y - X\hat{\theta}, X\hat{\theta}\rangle = \lambda\|\hat{\theta}\|_1$, from the KKT conditions for problem (50). Lastly, we rewrite

$$\begin{aligned}R &= \frac{1}{2}\|Z\hat{\theta}\|_2^2 - \frac{1}{2}\|X\hat{\theta}\|_2^2 + \langle y, X\hat{\theta} - Z\hat{\theta}\rangle \\ &= \frac{1}{2}\|X\hat{\theta} - Z\hat{\theta}\|_2^2 + \langle y - X\hat{\theta}, X\hat{\theta} - Z\hat{\theta}\rangle,\end{aligned}$$

which completes the proof. □

In order to bound $\|X\hat{\theta} - Z\hat{\alpha}\|_2$, the goal now is to determine conditions under which the right-hand side in (52) is small. Note that both terms in R involves the difference $X - Z$, which will have small entries if X and Z are close. The second term in (52) can be controlled by taking λ' and λ to be close. As for the first term in (52), we can rewrite

$$\langle y - X\hat{\theta}, Z\hat{\alpha}\rangle = \langle y - X\hat{\theta}, (Z - X)\hat{\alpha}\rangle + \langle X^T(y - X\hat{\theta}), \hat{\alpha}\rangle;$$

above, the first term again involves the difference $X - Z$, and the second term can be balanced by the term $-\lambda'\|\hat{\alpha}\|_1$ appearing in (52) if λ and λ' are chosen carefully. These ideas are all made precise in the next lemma.

Lemma 7. *Consider a sequence of lasso problems* (50), (51) *(all quantities $p, y, X, Z, \lambda, \lambda'$ considered as functions of n), such that $\lambda' = (1 + \delta)\lambda$ for some fixed $\delta > 0$. Assume that*

$$\sqrt{p}\|X - Z\|_\infty\|\hat{\theta}\|_1 = O\Big(\sqrt{\lambda\|\hat{\theta}\|_1}\Big) \quad \textit{and} \quad \frac{\sqrt{p}\|X - Z\|_\infty\|y - X\hat{\theta}\|_2}{\lambda} \to 0 \quad \textit{as } n \to \infty. \tag{53}$$

Then any solutions $\hat{\theta}, \hat{\alpha}$ of (50), (51) *satisfy*

$$\|X\hat{\theta} - Z\hat{\alpha}\|_2 = O\Big(\sqrt{\lambda\|\hat{\theta}\|_1}\Big). \tag{54}$$

Proof. As suggested in the discussion before the lemma, we rewrite the term $\langle y - X\hat{\theta}, Z\hat{\alpha}\rangle$ in the right-hand side of (52) as

$$\begin{aligned}\langle y - X\hat{\theta}, Z\hat{\alpha}\rangle &= \langle y - X\hat{\theta}, (Z - X)\hat{\alpha}\rangle + \langle X^T(y - X\hat{\theta}), \hat{\alpha}\rangle \\ &\le \|y - X\hat{\theta}\|_2\|(X - Z)\hat{\alpha}\|_2 + \lambda\|\hat{\alpha}\|_1 \\ &\le \sqrt{p}\|y - X\hat{\theta}\|_2\|X - Z\|_\infty\|\hat{\alpha}\|_1 + \lambda\|\hat{\alpha}\|_1,\end{aligned}$$

where in the second line we used Hölder's inequality and the fact that $\|X^T(y - X\hat{\theta})\|_\infty \le \lambda$ from the KKT conditions for (50), and in the third line we used the bound $\|Ax\|_2 \le \sqrt{p}\|A\|_\infty\|x\|_1$ for a matrix

$A \in \mathbb{R}^{n \times p}$, where $\|A\|_\infty$ denotes the maximum element of A in absolute value. The assumption that $\sqrt{p}\|X - Z\|_\infty \|y - X\hat{\theta}\|_2/\lambda \to 0$ now implies that, for large enough n,

$$\langle y - X\hat{\theta}, Z\hat{\alpha}\rangle \leq \delta\lambda\|\hat{\alpha}\|_1 + \lambda\|\hat{\alpha}\|_1 = (1+\delta)\lambda\|\hat{\alpha}\|_1.$$

Plugging this into the right-hand side of (52), and using $\lambda' = (1+\delta)\lambda$, we see that

$$\frac{1}{2}\|X\hat{\theta} - Z\hat{\alpha}\|_2^2 \leq \delta\lambda\|\hat{\theta}\|_1 + R.$$

Finally,

$$R \leq \frac{1}{2}\Big(\sqrt{p}\|X - Z\|_\infty \|\hat{\theta}\|_1\Big)^2 + \sqrt{p}\|X - Z\|_\infty \|y - X\hat{\theta}\|_2 \|\hat{\theta}\|_1,$$

and using both conditions in (53), we have $R = O(\lambda\|\hat{\theta}\|_1)$, completing the proof. □

Remark. Had we instead chosen $\lambda' = \lambda$, which may seem like more of a natural choice for pairing the two problems (50), (51), the same arguments would have yielded the final bound

$$\|X\hat{\theta} - Z\hat{\alpha}\|_2 = O\Big(\sqrt{\lambda \max\{\|\hat{\theta}\|_1, \|\hat{\alpha}\|_1\}}\Big).$$

For some purposes, this may be just fine. However, the envisioned main use case of this lemma is one in which some (desirable) theoretical properties are known for the lasso problem (50) with a particular predictor matrix X, and analogous results for a lasso problem (51) with similar predictor matrix Z are sought (e.g., this is the usage for locally adaptive regression splines and trend filtering); in such a case, a bound of the form (54) is preferred as it does not depend at all on the output of problem (51).

The second condition in (53) involves the quantity $y - X\hat{\theta}$, and so may appear more complicated than necessary. Indeed, under weak assumptions on y and $X\hat{\theta}$, this condition can be simplified.

Corollary 3. *Consider again a sequence of lasso problems* (50), (51) *such that* $\lambda' = (1+\delta)\lambda$ *for some fixed* $\delta > 0$. *Assume that the outcome vector* y *is drawn from the regression model*

$$y = \mu + \epsilon,$$

where $\epsilon_1, \ldots \epsilon_n$ *are i.i.d. with* $\mathbb{E}[\epsilon_i^4] < \infty$, *and assume that* $\|\mu - X\hat{\theta}\|_2 = O_{\mathbb{P}}(\sqrt{n})$. *Further assume*

$$\sqrt{p}\|X - Z\|_\infty \|\hat{\theta}\|_1 = O_{\mathbb{P}}\Big(\sqrt{\lambda\|\hat{\theta}\|_1}\Big) \quad \textit{and} \quad \sqrt{np}\|X - Z\|_\infty/\lambda \to 0 \quad \textit{as } n \to \infty.$$

Then any solutions $\hat{\theta}, \hat{\alpha}$ *of* (50), (51) *satisfy*

$$\|X\hat{\theta} - Z\hat{\alpha}\|_2 = O_{\mathbb{P}}\Big(\sqrt{\lambda\|\hat{\theta}\|_1}\Big).$$

Proof. Note that

$$\|y - X\hat{\theta}\|_2 \leq \|\epsilon\|_2 + \|\mu - X\hat{\theta}\|_2.$$

Both terms on the right-hand side above are $O_{\mathbb{P}}(\sqrt{n})$; for the second term, this is true by assumption, and for the first term, we can use the fact that $\epsilon_1, \ldots \epsilon_n$ are i.i.d. with finite fourth moment to argue

$$\mathbb{P}\left(\frac{\sum_{i=1}^n \epsilon_i^2}{n} - \mathbb{E}[\epsilon_i^2] > 1\right) \leq \frac{\mathrm{Var}(\epsilon_i^2)}{n} \to 0,$$

where we used Chebyshev's inequality. Therefore $\|y - X\hat{\theta}\|_2 = O_{\mathbb{P}}(\sqrt{n})$, and to show that $\sqrt{p}\|X - Z\|_\infty \|y - X\hat{\theta}\|_2/\lambda \to 0$ in probability, it suffices to show $\sqrt{np}\|X - Z\|_\infty/\lambda \to 0$. □

Remark. The fourth moment condition on the errors, $\mathbb{E}[\epsilon_i^4] < \infty$, is not a strong one. E.g., any sub-Gaussian distribution [which was our distributional assumption in (31) for the theoretical work in Section 5] has finite moments of all orders. Moreover, the assumption $\|\mu - X\hat{\theta}\|_2 = O_{\mathbb{P}}(\sqrt{n})$ is also quite weak; we only maintain that the average prediction error $\|\mu - X\hat{\theta}\|_2/\sqrt{n}$ is bounded in probability, and not even that it converges to zero.

B.2 Lasso problems in nonstandard form

Now consider two lasso problems

$$\min_{\theta \in \mathbb{R}^p} \frac{1}{2}\|y - X\theta\|_2^2 + \lambda\|\theta_2\|_1, \tag{55}$$

$$\min_{\alpha \in \mathbb{R}^p} \frac{1}{2}\|y - Z\alpha\|_2^2 + \lambda'\|\alpha_2\|_1, \tag{56}$$

where the coefficient vectors decompose as $\theta = (\theta_1, \theta_2)$, $\alpha = (\alpha_1, \alpha_2) \in \mathbb{R}^{p_1+p_2}$, and we partition the columns of the predictor matrices accordingly,

$$X = [X_1 \; X_2], \; Z = [Z_1 \; Z_2] \in \mathbb{R}^{n \times (p_1+p_2)}.$$

In words, the ℓ_1 penalties in (55), (56) only apply to part of the coefficient vectors. We have the same goal of the last section: to bound $\|X\hat{\theta} - Z\hat{\alpha}\|_2$ in terms of the differences between X and Z.

Simply transforming (55), (56) into standard form lasso problems (by partially solving for θ_1, α_1) will not generically provide a tight bound, because the predictor matrices of the resulting lasso problems have restricted images.[8] Therefore, we must rederive the analogues of Lemmas 6 and 7, and Corollary 3. For the upcoming bounds in Lemma 9 and Corollary 4, it is critical that $X_1 = Z_1$, i.e., the predictor variables corresponding to unpenalized coefficients in (55), (56) are identical. We state the results below, but omit the proofs because they follow essentially the same arguments as those in the last section.

Lemma 8. *For fixed λ, λ', solutions $\hat{\theta}$ and $\hat{\alpha}$ of lasso problems* (55) *and* (56) *satisfy*

$$\frac{1}{2}\|X\hat{\theta} - Z\hat{\alpha}\|_2^2 \le \langle y - X\hat{\theta}, Z\hat{\alpha}\rangle + (\lambda' - \lambda)\|\hat{\theta}_2\|_1 - \lambda'\|\hat{\alpha}_2\|_1 + R,$$

where $R = \frac{1}{2}\|(X - Z)\hat{\theta}\|_2^2 + \langle y - X\hat{\theta}, (X - Z)\hat{\theta}\rangle$.

Lemma 9. *Consider a sequence of lasso problems* (55), (56) *(where all quantities $p_1, p_2, y, X, Z, \lambda, \lambda'$ are considered as functions of n), such that $X_1 = Z_1$ and $\lambda' = (1+\delta)\lambda$ for some fixed $\delta > 0$. Assume that*

$$\sqrt{p_2}\|X_2 - Z_2\|_\infty \|\hat{\theta}_2\|_1 = O\Big(\sqrt{\lambda\|\hat{\theta}_2\|_1}\Big) \quad \text{and} \quad \frac{\sqrt{p_2}\|X_2 - Z_2\|_\infty \|y - X\hat{\theta}\|_2}{\lambda} \to 0 \quad \text{as } n \to \infty.$$

Then any solutions $\hat{\theta}, \hat{\alpha}$ of (50), (51) *satisfy*

$$\|X\hat{\theta} - Z\hat{\alpha}\|_2 = O\Big(\sqrt{\lambda\|\hat{\theta}_2\|_1}\Big).$$

[8] In more detail, solving for $X_1\hat{\theta}_1 = P_{X_1}(y - X_2\hat{\theta}_2)$ and $Z_1\hat{\theta}_1 = P_{Z_1}(y - Z_2\hat{\alpha}_2)$ [where $P_A = A(A^TA)^+A^T$ denotes the projection matrix onto col(A), the column space of a matrix A], yields "new" standard form lasso problems with predictor matrices $(I - P_{X_1})X_2$ and $(I - P_{Z_1})Z_2$. This is problematic because we require a bound in X_2 and Z_2.

Corollary 4. *Consider again a sequence of lasso problems* (55), (56) *with* $X_1 = Z_1$ *and* $\lambda' = (1+\delta)\lambda$ *for some fixed* $\delta > 0$. *Assume that the outcome vector* y *is drawn from the model*

$$y = \mu + \epsilon,$$

where $\epsilon_1, \dots \epsilon_n$ *are i.i.d. with* $\mathbb{E}[\epsilon_i^4] < \infty$, *and assume that* $\|\mu - X\hat{\theta}\|_2 = O_{\mathbb{P}}(\sqrt{n})$. *Further assume*

$$\sqrt{p_2}\|X_2 - Z_2\|_\infty \|\hat{\theta}_2\|_1 = O_{\mathbb{P}}\Big(\sqrt{\lambda\|\hat{\theta}_2\|_1}\Big) \quad \text{and} \quad \sqrt{np_2}\|X_2 - Z_2\|_\infty/\lambda \to 0 \quad \text{as } n \to \infty. \tag{57}$$

Then any solutions $\hat{\theta}, \hat{\alpha}$ *of* (55), (56) *satisfy*

$$\|X\hat{\theta} - Z\hat{\alpha}\|_2 = O_{\mathbb{P}}\Big(\sqrt{\lambda\|\hat{\theta}_2\|_1}\Big).$$

C Convergence of trend filtering and locally adaptive regression spline basis matrices

Lemma 10. *For any integer* $k \geq 0$, *consider the matrices* $G_2, H_2 \in \mathbb{R}^{n \times (n-k-1)}$, *the last* $n - k - 1$ *columns of the trend filtering and locally adaptive regression spline basis matrices in* (22), (27). *With evenly spaced inputs on* $[0, 1]$, $\{x_1, x_2, \dots x_n\} = \{1/n, 2/n, \dots 1\}$, *we have*

$$\|G_2 - H_2\|_\infty = O(1/n).$$

Proof. For $k = 0, 1$ the result is vacuous, as $G_2 = H_2$ according to Lemma 3 in Section 3. Hence we assume $k \geq 2$, and without a loss of generality, we assume that k is even, since the case for k odd follows from essentially the same arguments. By Lemma 11, for large enough n,

$$\|G_2 - H_2\|_\infty = \frac{1}{n^k}\left(\prod_{i=1}^k (n-1-i) - \left(n - 1 - \frac{k+2}{2}\right)^k\right),$$

i.e., for large enough n,

$$n\|G_2 - H_2\|_\infty = \underbrace{\frac{\left(n-1-\frac{k+2}{2}\right)^k}{n^k}}_{a_n} \cdot \underbrace{n\left(\frac{\prod_{i=1}^k (n-1-i)}{\left(n-1-\frac{k+2}{2}\right)^k} - 1\right)}_{b_n},$$

We investigate the convergence of the sequence $a_n \cdot b_n$ as defined above. It is clear that $a_n \to 1$ as $n \to \infty$. Hence it remains to bound b_n.

To this end, consider the term

$$\prod_{i=1}^k (n-1-i) = \frac{(n-1)!}{(n-k-1)!}.$$

We use Stirling's approximation to both the numerator and denominator, writing

$$\frac{(n-1)!}{(n-k-1)!} = \underbrace{\frac{(n-1)! \Big/ \left((n-1)^{n-1/2} e^{-n+1}\sqrt{2\pi}\right)}{(n-k-1)! \Big/ \left((n-k-1)^{n-k-1/2} e^{-n+k+1}\sqrt{2\pi}\right)}}_{c_n} \cdot \frac{(n-1)^{n-1/2}}{(n-k-1)^{n-k-1/2}} \cdot e^{-k}.$$

Therefore

$$\begin{aligned}\frac{\prod_{i=1}^k (n-1-i)}{\left(n-1-\frac{k+2}{2}\right)^k} &= c_n \cdot \frac{\left((n-1) \,/\, \left(n-1-\frac{k+2}{2}\right)\right)^{n-1/2}}{\left((n-k-1) \,/\, \left(n-1-\frac{k+2}{2}\right)\right)^{n-k-1/2}} \cdot e^{-k} \\ &= c_n \cdot \underbrace{\frac{\left(1+\frac{(k-2)/2}{n-k-1}\right)^{n-k-1/2} e^{-(k-2)/2}}{\left(1-\frac{(k+2)/2}{n-1}\right)^{n-1/2} e^{(k+2)/2}}}_{d_n}.\end{aligned}$$

At this point, we have expressed $b_n = n(c_n d_n - 1)$. Note that $c_n \to 1$ by Stirling's formula, and $d_n \to 1$ by the well-known limit for e^x,

$$e^x = \lim_{t\to\infty} \left(1+\frac{x}{t}\right)^t. \tag{58}$$

The question is of course how fast these two sequences c_n, d_n converge; if the remainder $c_n d_n - 1$ is $O(1/n)$, then $b_n = O(1)$ and indeed $\|G_2 - H_2\|_\infty = O(1/n)$.

First we address c_n. It is known that Stirling's approximation satisfies [e.g., see Nemes (2010)]

$$\frac{n!}{n^{n+1/2} e^{-n}\sqrt{2\pi}} = e^{\gamma_n}, \quad \text{where } \frac{1}{12n+1} \le \gamma_n \le \frac{1}{12n}.$$

Hence

$$c_n = \exp(\gamma_{n-1} - \gamma_{n-k-1}) \le \exp\left(\frac{1}{12(n-1)}\right).$$

Next we address d_n. Lemma 12 derives the following error bound for the exponential limit in (58):

$$\left(1+\frac{x}{n}\right)^n e^{-x} = e^{\delta_{x,n}}, \quad \text{where } \frac{-x^2}{n+x} \le \delta_{x,n} \le 0,$$

for sufficiently large n. Therefore

$$\begin{aligned} d_n &= \exp\left(\delta_{(k-2)/2,n-k-1} - \delta_{-(k+2)/2,n-1}\right) \cdot \left(\frac{1+\frac{(k-2)/2}{n-k-1}}{1-\frac{(k+2)/2}{n-1}}\right)^{1/2} \\ &= \exp\left(\delta_{(k-2)/2,n-k-1} - \delta_{-(k+2)/2,n-1}\right) \cdot \left(\frac{n-1}{n-k-1}\right)^{1/2} \\ &\le \exp\left(\frac{(k+2)^2}{4n-2(k+4)}\right) \cdot \left(\frac{n-1}{n-k-1}\right)^{1/2}. \end{aligned}$$

We can simplify, for large enough n,

$$\frac{1}{12(n-1)} + \frac{(k+2)^2}{4n-2(k+4)} \le \frac{(k+2)^2}{n},$$

and putting this all together, we have

$$b_n = n(c_n d_n - 1) \le n\left(\exp\left(\frac{(k+2)^2}{n}\right) \cdot \left(\frac{n-1}{n-k-1}\right)^{1/2} - 1\right).$$

An application of l'Hôpital's rule shows that the bound on the right-hand hand side above converges to a positive constant. This completes the proof. □

Lemma 11. *Let $k \geq 2$. If k is even, then*

$$\|G_2 - H_2\|_\infty = \frac{1}{n^k}\left(\prod_{i=1}^{k}(n-1-i) - \left(n-1-\frac{k+2}{2}\right)^k\right),$$

for sufficiently large n; if k is odd, then

$$\|G_2 - H_2\|_\infty = \frac{1}{n^k}\left(\prod_{i=1}^{k}(n-1-i) - \left(n-1-\frac{k+1}{2}\right)^k\right),$$

for sufficiently large n.

Proof. We prove the result for k even; the proof for k odd is similar. Consider first the sequence

$$a_n = \prod_{i=1}^{k}(n-i) - \left(n - \frac{(k+2)}{2}\right)^k.$$

Note that

$$a_n = \left(\frac{k(k+2)}{2} - \frac{k(k+1)}{2}\right) \cdot n^{k-1} + O(n^{k-2}).$$

Because the coefficient of the leading term is positive, we know that $a_n \to \infty$ as $n \to \infty$; further, for large enough n, this convergence is monotone (since a_n is polynomial in n). Now recall that G_2, H_2 are the last $n-k-1$ columns of G, H, in (22), (27). Hence

$$(H_2 - G_2)_{ij} = \frac{1}{n^k} \cdot \begin{cases} 0 & \text{if } i \leq j + (k+2)/2 \\ -(i-j-(k+2)/2)^k & \text{if } j + (k+2)/2 < i \leq j+k \\ a_{i-j} & \text{if } i > j+k, \end{cases}$$

given by taking $j+k+1$ in place of j in (22), (27). For $i - j \leq k$, the term $(i-j-(k+2)/2)^k$ is bounded by k^k. And since $a_n \uparrow \infty$, as argued above, we conclude that $\|G_2 - H_2\|_\infty = a_{n-1}/n^k$ for sufficiently large n. □

Lemma 12. *For any $x \in \mathbb{R}$, and for sufficiently large $t > 0$,*

$$\left(1+\frac{x}{t}\right)^t e^{-x} = e^{\delta_{x,t}}, \quad \textit{where } \frac{-x^2}{t+x} \leq \delta_{x,t} \leq 0.$$

Proof. Define $f(t) = (1+x/t)^t$. Consider

$$\log f(t) = t \cdot (\log(t+x) - \log t),$$

which is well-defined as long as $t \geq \max\{-x, 0\}$. Because log is a concave function, its tangent line is a global overestimate of the function, that is,

$$\log(t+x) \leq \log t + \frac{1}{t} \cdot x,$$

which means that $t \cdot (\log(t+x) - \log t) \leq x$, i.e., $f(t) \leq e^x$. Hence $f(t)e^{-x} = e^{\delta_{x,t}}$ where $\delta_{x,t} \leq 0$. The lower bound on $\delta_{x,t}$ follows similarly. Again by concavity,

$$\log t \leq \log(t+x) + \frac{1}{t+x} \cdot (-x),$$

so $\log(t+x) - \log t \geq x/(t+x)$, and $f(t) \geq \exp(tx/(t+x))$. Therefore $f(t)e^{-x} \geq \exp(tx/(t+x) - x) = \exp(-x^2/(t+x))$. □

References

Cohen, A., Daubechies, I. & Vial, P. (1993), 'Wavelets on the interval and fast wavelet transforms', *Applied and Computational Harmonic Analysis* **1**, 54–81.

de Boor, C. (1978), *A Practical Guide to Splines*, Springer, New York.

DeVore, R. & Lorentz, G. (1993), *Constructive Approximation*, Springer, Berlin.

Donoho, D. L. & Johnstone, I. (1995), 'Adapting to unknown smoothness via wavelet shrinkage', *Journal of the American Statistical Association* **90**(432), 1200–1224.

Donoho, D. L. & Johnstone, I. (1998), 'Minimax estimation via wavelet shrinkage', *Annals of Statistics* **26**(8), 879–921.

Efron, B., Hastie, T., Johnstone, I. & Tibshirani, R. (2004), 'Least angle regression', *Annals of Statistics* **32**(2), 407–499.

Elad, M., Milanfar, P. & Rubinstein, R. (2007), 'Analysis versus synthesis in signal priors', *Inverse problems* **23**(3), 947–968.

Green, P. & Silverman, B. (1994), *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*, Chapman & Hall/CRC Press, Boca Raton.

Green, P. & Yandell, B. (1985), 'Semi-parametric generalized linear models', *Proceedings of the International Conference on Generalized Linear Models* **32**, 44–55.

Hastie, T. & Tibshirani, R. (1990), *Generalized additive models*, Chapman and Hall, London.

Hastie, T., Tibshirani, R. & Friedman, J. (2008), *The Elements of Statistical Learning; Data Mining, Inference and Prediction*, Springer, New York. Second edition.

Johnstone, I. (2011), *Gaussian estimation: Sequence and wavelet models*, Under contract to Cambridge University Press. Online version at `http://www-stat.stanford.edu/~imj`.

Kim, S.-J., Koh, K., Boyd, S. & Gorinevsky, D. (2009), 'ℓ_1 trend filtering', *SIAM Review* **51**(2), 339–360.

Lee, K.-G., Bailey, S., Bartsch, L. E., Carithers, W., Dawson, K. S., Kirkby, D., Lundgren, B., Margala, D., Palanque-Delabrouille, N., Pieri, M. M., Schlegel, D. J., Weinberg, D. H., Yeche, C., Aubourg, E., Bautista, J., Bizyaev, D., Blomqvist, M., Bolton, A. S., Borde, A., Brewington, H., Busca, N. G., Croft, R. A. C., Delubac, T., Ebelke, G., Eisenstein, D. J., Font-Ribera, A., Ge, J., Hamilton, J.-C., Hennawi, J. F., Ho, S., Honscheid, K., Le Goff, J.-M., Malanushenko, E., Malanushenko, V., Miralda-Escude, J., Myers, A. D., Noterdaeme, P., Oravetz, D., Pan, K., Paris, I., Petitjean, P., Rich, J., Rollinde, E., Ross, N. P., Rossi, G., Schneider, D. P., Simmons, A., Snedden, S., Slosar, A., Spergel, D. N., Suzuki, N., Viel, M. & Weaver, B. A. (2012), The BOSS Lyman-alpha forest sample from SDSS data release 9. arXiv: 1211.5146.

Mallat, S. (2008), *A wavelet tour of signal processing*, Academic Press, San Diego. Third edition.

Mammen, E. & van de Geer, S. (1997), 'Locally apadtive regression splines', *Annals of Statistics* **25**(1), 387–413.

Nemes, G. (2010), 'On the coefficients of the asymptotic expansion of $n!$', *Journal of Integer Sequences* **13**(6), 5.

Nussbaum, M. (1985), 'Spline smoothing in regression models and asymptotic efficiency in L_2', *Annals of Statistics* **13**(3), 984–997.

Osborne, M., Presnell, B. & Turlach, B. (1998), 'Knot selection for regression splines via the lasso', *Dimension Reduction, Computational Complexity, and Information* **30**, 44–49.

Rosset, S. & Zhu, J. (2007), 'Piecewise linear regularized solution paths', *Annals of Statistics* **35**(3), 1012–1030.

Rudin, L. I., Osher, S. & Faterni, E. (1992), 'Nonlinear total variation based noise removal algorithms', *Physica D: Nonlinear Phenomena* **60**, 259–268.

Tibshirani, R. J. & Arnold, T. (2013), Efficient implementations of the generalized lasso dual path algorithm. In preparation.

Tibshirani, R. J. & Taylor, J. (2011), 'The solution path of the generalized lasso', *Annals of Statistics* **39**(3), 1335–1371.

Tibshirani, R. J. & Taylor, J. (2012), 'Degrees of freedom in lasso problems', *Annals of Statistics* **40**(2), 1198–1232.

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J. & Knight, K. (2005), 'Sparsity and smoothness via the fused lasso', *Journal of the Royal Statistical Society: Series B* **67**(1), 91–108.

Wahba, G. (1990), *Spline Models for Observational Data*, Society for Industrial and Applied Mathematics, Philadelphia.

Wang, X., Du, P. & Shen, J. (2013), 'Smoothing splines with varying smoothing parameter', *Biometrika* **100**(4), 955–970.