

# BASELINE DRIFT ESTIMATION FOR AIR QUALITY DATA USING QUANTILE TREND FILTERING

BY HALLEY L. BRANTLEY<sup>1,\*</sup>, JOSEPH GUINNESS<sup>2</sup> AND ERIC C. CHI<sup>1,†</sup>

<sup>1</sup>*Department of Statistics, North Carolina State University, \*[hlbrantl@ncsu.edu](mailto:hlbrantl@ncsu.edu); †[eric\\_chi@ncsu.edu](mailto:eric_chi@ncsu.edu)*

<sup>2</sup>*Department of Statistics and Data Science, Cornell University, [guinness@cornell.edu](mailto:guinness@cornell.edu)*

We address the problem of estimating smoothly varying baseline trends in time series data. This problem arises in a wide range of fields, including chemistry, macroeconomics and medicine; however, our study is motivated by the analysis of data from low cost air quality sensors. Our methods extend the quantile trend filtering framework to enable the estimation of multiple quantile trends simultaneously while ensuring that the quantiles do not cross. To handle the computational challenge posed by very long time series, we propose a parallelizable alternating direction method of multipliers (ADMM) algorithm. The ADMM algorithm enables the estimation of trends in a piecewise manner, both reducing the computation time and extending the limits of the method to larger data sizes. We also address smoothing parameter selection and propose a modified criterion based on the extended Bayesian information criterion. Through simulation studies and our motivating application to low cost air quality sensor data, we demonstrate that our model provides better quantile trend estimates than existing methods and improves signal classification of low-cost air quality sensor output.

**1. Introduction.** In the last decade, low cost and portable air quality sensors have enjoyed dramatically increased usage. These sensors can provide an uncalibrated measure of a variety of pollutants in near real time but deriving meaningful information from sensor data remains a challenge (Snyder et al. (2013)). For example, the “SPod” is a low-cost sensor currently being investigated by researchers at the U.S. Environmental Protection Agency to detect volatile organic compound (VOC) emissions from industrial facilities (Thoma et al. (2016)). Due to changes in temperature and relative humidity, the output signal exhibits a slowly varying baseline drift on the order of minutes to hours. Figure 1 provides an example of measurements from three SPod sensors colocated at the border of an industrial facility. All of the sensors respond to the pollutant signal which is illustrated by the three sharp transient spikes at 11:32, 14:10 and 16:03. However, the baseline drift varies from one sensor to another, obscuring the detection of the peaks that alert the intrusion of pollutants. We show later that by estimating the baseline drift in each sensor and removing it from the observed signals, peaks can be reliably detected from concordant residual signals from a collection of SPods using a simple data-driven thresholding strategy. Thus, accurately demixing a noisy observed time series into a slowly varying component and a transient component can lead to greatly improved and simplified downstream analysis.

While this work is motivated by the analysis of data from low-cost air quality sensors, the problem of demixing noisy time series into trends and transients is ubiquitous across many applications. This includes electrocardiogram data (Luo et al. (2013)), electrooculographic data (recording of eye movements) (Pettersson et al. (2013), Marandi and Sabzpoushan (2015)), mass spectrometry (Coombes, Baggerly and Morris (2007), Du, Kibbe and Lin (2006)), chromatography (Ning, Selesnick and Duval (2014)), macroeconomics (Yamada

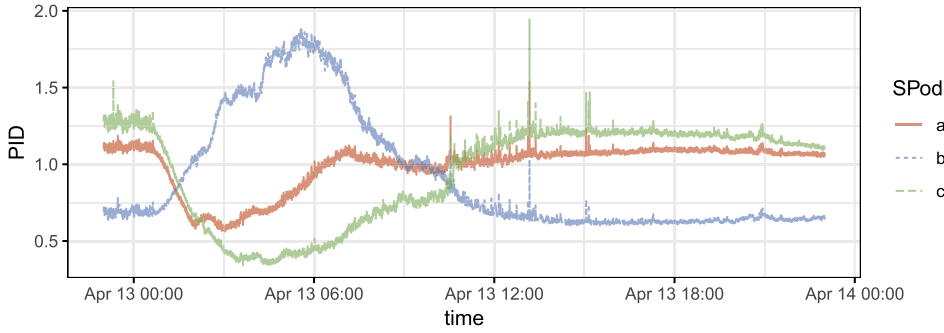


FIG. 1. Example of three colocated SPod PID sensor readings over a 24 hour period.

(2017)), environmental science (Brantley et al. (2014), Mecozzi (2014)), laser-induced breakdown spectroscopy (Franco, Milori and Boas (2018)) and astronomy (Oh et al. (2004)). In most instances, scalar functions of time  $y(t)$  are observed and assumed to be a superposition of: (i) an underlying slowly varying baseline trend  $\theta(t)$ , (ii) other more rapidly varying components  $s(t)$  and (iii) noise. For example, in electrocardiogram data one wishes to separate a train of transient waveforms  $s(t)$  reflecting the electrical activity of the heart from a nuisance “baseline wander”  $\theta(t)$  due to a patient’s respiration and other movements.

In practice,  $y(t)$  is observed at discrete time points  $t_1, \dots, t_n$ , and we model the vector of samples  $y_i = y(t_i)$  as

$$\mathbf{y} = \boldsymbol{\theta} + \mathbf{s} + \boldsymbol{\varepsilon},$$

where  $\theta_i = \theta(t_i)$ ,  $s_i = s(t_i)$  and  $\boldsymbol{\varepsilon} \in \mathbb{R}^n$  is a vector of uncorrelated noise. For notational simplicity, for the rest of the paper we assume that the time points take on the values  $t_i = i$ , but it is straightforward to generalize to an arbitrary grid of time points. In some applications the slowly varying component  $\boldsymbol{\theta}$  is the signal of interest, and the transient component  $\mathbf{s}$  is a vector of nuisance parameters. In our air quality application, the roles of  $\boldsymbol{\theta}$  and  $\mathbf{s}$  are reversed;  $\mathbf{s}$  represents the signal of interest, and  $\boldsymbol{\theta}$  represents a baseline drift that obscures the identification of the important transient events encoded in  $\mathbf{s}$ .

To tackle demixing problems, we introduce a scalable baseline estimation framework by building on  $\ell_1$ -trend filtering, a relatively new nonparametric estimation framework. Our contributions are three-fold:

- Kim et al. (2009) proposed using the check function as a possible extension of  $\ell_1$ -trend filtering but did not investigate it further. Here, we develop the basic  $\ell_1$ -quantile-trend-filtering framework and extend it to model *multiple quantiles simultaneously with non-crossing constraints to ensure validity and improve trend estimates*.
- To reduce computation time and extend the method to long time series, we develop a parallelizable ADMM algorithm. The algorithm proceeds by splitting the time domain into overlapping windows, fitting the model separately for each of the windows and reconciling estimates from the overlapping intervals.
- Finally, we propose a modified criterion for performing model selection.

In the rest of the paper, we detail our quantile trend filtering algorithms (Section 2) and investigate how to choose the smoothing parameter (Section 3). We demonstrate through simulation studies that our proposed methods provide better or comparable estimates of non-parametric quantile trends than existing methods (Section 4). We further show that quantile trend filtering is a more effective method of drift removal for low-cost air quality sensors and results in improved signal classification compared to quantile smoothing splines (Section 5). Finally, we discuss potential extensions of quantile trend filtering (Section 6).

## 2. Baseline trend estimation.

2.1. *Background.* Kim et al. (2009) originally proposed  $\ell_1$ -trend filtering to estimate trends with piecewise polynomial functions, assuming that the observed time series  $\mathbf{y}$  consists of a trend  $\boldsymbol{\theta}$  plus uncorrelated noise  $\boldsymbol{\varepsilon}$ , namely,  $\mathbf{y} = \boldsymbol{\theta} + \boldsymbol{\varepsilon}$ . The estimated trend is the solution to the following convex optimization problem:

$$\underset{\boldsymbol{\theta}}{\text{minimize}} \frac{1}{2} \|\mathbf{y} - \boldsymbol{\theta}\|_2^2 + \lambda \|\mathbf{D}^{(k+1)} \boldsymbol{\theta}\|_1,$$

where  $\lambda$  is a nonnegative regularization parameter and the matrix  $\mathbf{D}^{(k+1)} \in \mathbb{R}^{(n-k-1) \times n}$  is the discrete difference operator of order  $k+1$ . To understand the purpose of penalizing the 1-norm of  $\mathbf{D}^{(k+1)} \boldsymbol{\theta}$ , consider the difference operator when  $k=0$ .

$$\mathbf{D}^{(1)} = \begin{pmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{pmatrix}.$$

Thus,  $\|\mathbf{D}^{(1)} \boldsymbol{\theta}\|_1 = \sum_{i=1}^{n-1} |\theta_i - \theta_{i+1}|$ , which is known as the total variation denoising penalty in one dimension in the signal processing literature (Rudin, Osher and Fatemi (1992)) or the fused lasso penalty in the statistics literature (Tibshirani et al. (2005)). The penalty term incentivizes solutions which are piecewise constant. For  $k \geq 1$ , the difference operator  $\mathbf{D}^{(k+1)} \in \mathbb{R}^{(n-k-1) \times n}$  is defined recursively as follows:

$$\mathbf{D}^{(k+1)} = \mathbf{D}^{(1)} \mathbf{D}^{(k)}.$$

Penalizing the 1-norm of the vector  $\mathbf{D}^{(k+1)} \boldsymbol{\theta}$  produces estimates of  $\boldsymbol{\theta}$  that are piecewise polynomials of order  $k$ .

Tibshirani (2014) proved that with a judicious choice of  $\lambda$  the trend filtering estimate converges to the true underlying function at the minimax rate for functions whose  $k$ th derivative is of bounded variation and showed that trend filtering is locally adaptive when the time series consists of only the trend and random noise, which is illustrated in Figure 2(a). As noted earlier, in some applications, such as the air quality monitoring problem considered in this paper, the data contain a rapidly varying signal in addition to the slowly varying trend and noise. Figure 2(b) shows that standard trend filtering is not designed to distinguish between the slowly varying trend and the rapidly varying signal, as the smooth component estimate  $\boldsymbol{\theta}$  is biased toward the peaks of the transient components.

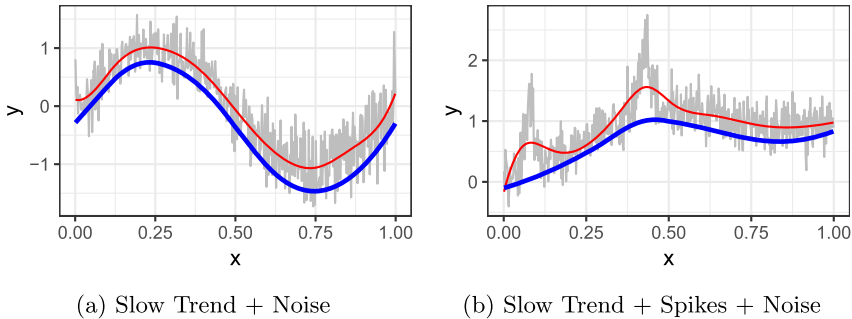


FIG. 2. Examples of trend filtering solutions (thick line) and 15th quantile trending filtering solution (thin line). Standard trend filtering performs well in the no-signal case (a) but struggles to distinguish between the slowly varying trend and the rapidly varying signal (b). The quantile trend is not affected by the signal and provides an estimate of the baseline.

To account for the presence of transient components in the observed time series  $\mathbf{y}$ , we propose quantile trend filtering Figure 2(b). To estimate the trend in the  $\tau$ th quantile, we solve the convex optimization problem

$$(1) \quad \underset{\boldsymbol{\theta}}{\text{minimize}} \rho_{\tau}(\mathbf{y} - \boldsymbol{\theta}) + \lambda \|\mathbf{D}^{(k+1)}\boldsymbol{\theta}\|_1,$$

where  $\rho_{\tau}(\mathbf{r})$  is the check function

$$(2) \quad \rho_{\tau}(\mathbf{r}) = \sum_{i=1}^n r_i (\tau - \mathbb{1}(r_i < 0))$$

and  $\mathbb{1}(A)$  is 1 if its input  $A$  is true and 0 otherwise. Note that we do not explicitly model  $\mathbf{s}$ . Rather, we focus on estimating  $\boldsymbol{\theta}$ . We then estimate  $\mathbf{s} + \boldsymbol{\varepsilon}$  as the difference  $\mathbf{y} - \boldsymbol{\theta}$ .

Before elaborating on how we compute our proposed  $\ell_1$ -quantile trend filtering estimator, we discuss similarities and differences between our proposed estimator and existing quantile trend estimators.

**2.1.1. Relationship to prior work.** In this application, as well as those described in Ning, Selesnick and Duval (2014), Marandi and Sabzpooshan (2015) and Pettersson et al. (2013), the goal is to estimate the trend in the baseline, not the mean. We can define the trend in the baseline as the trend in a low quantile of the data. A variety of methods for estimating quantile trends have already been proposed. Koenker and Bassett (1978) were the first to propose substituting the sum-of-squares term with the check function (2) to estimate a conditional quantile instead of the conditional mean. Later, Koenker, Ng and Portnoy (1994) proposed quantile trend filtering with  $k = 1$  producing quantile trends that are piecewise linear, but they did not consider extensions to higher order differences. Rather than using the  $\ell_1$ -norm to penalize the discrete differences, Nychka et al. (1995) used the smoothing spline penalty based on the square of the  $\ell_2$ -norm

$$\sum_{i=1}^n \rho_{\tau}(y(t_i) - \theta(t_i)) + \lambda \int \theta''(t)^2 dt,$$

where  $\theta(t)$  is a smooth function of time and  $\lambda$  is a tuning parameter that controls the degree of smoothing. Oh, Lee and Nychka (2011) proposed an algorithm for solving the quantile smoothing spline problem by approximating the check function with a differentiable function. Racine and Li (2017) propose a method for estimating quantile trends that does not employ the check function. In their method the response is constrained to follow a location scale model and the conditional quantiles are estimated by combining Gaussian quantile functions with a kernel smoother and solving a local-linear least squares problem.

**2.2. Quantile trend filtering.** We combine the ideas of quantile regression and trend filtering. For a single quantile level  $\tau$ , the quantile trend filtering problem is given in (1). As with classic quantile regression, the quantile trend filtering problem is a linear program which can be solved by a number of methods. We want to estimate multiple quantiles simultaneously and to ensure that our quantile estimates are valid by enforcing the constraint that if  $\tau_2 > \tau_1$ , then  $Q(\tau_2) \geq Q(\tau_1)$ , where  $Q$  is the quantile function of  $\mathbf{y}$ . Even if a single quantile is ultimately desired, ensuring noncrossing allows information from nearby quantiles to be used to improve the estimates, as we will see in the peak detection experiments in Section 4.2. Given quantiles  $\tau_1 < \tau_2 < \dots < \tau_J$ , the optimization problem becomes

$$(3) \quad \underset{\boldsymbol{\theta} \in \mathcal{C}}{\text{minimize}} \sum_{j=1}^J [\rho_{\tau_j}(\mathbf{y} - \boldsymbol{\theta}_j) + \lambda_j \|\mathbf{D}^{(k+1)}\boldsymbol{\theta}_j\|_1],$$

where  $\Theta = (\theta_1 \ \theta_2 \ \cdots \ \theta_J) \in \mathbb{R}^{n \times J}$  is a matrix whose  $j$ th column corresponds to the  $j$ th quantile signal  $\theta_j$  and the set  $\mathcal{C} = \{\Theta \in \mathbb{R}^{n \times J} : \theta_{ij} \leq \theta_{ij'} \text{ for } j \leq j'\}$  encodes the noncrossing quantile constraints. The additional noncrossing constraints are linear inequalities involving the parameters; so the noncrossing quantile trends can still be estimated by a number of available linear programming solvers. We allow for the possibility that the degree of smoothness in the trends varies by quantile by allowing the smoothing parameter to vary with quantile. In the rest of this paper, we use  $k = 2$  to produce piecewise quadratic polynomials and report numerical results using the commercial solver Gurobi (Gurobi Optimization (2018)) and its R package implementation. However, we could easily substitute a free solver such as the Rglpk package by Theussl and Hornik (2017).

**2.3. ADMM for big data.** As the size of the data increases, computation time becomes prohibitive. In our application to air quality sensor data, measurements are recorded every second resulting in 86,400 observations per day. This number of observations is already too large to use with currently available R packages used for estimating quantile trends (Nychka et al. (2017), Koenker (2018)). To our knowledge, no one has addressed the problem of finding smooth quantile trends of series that are too large to be processed simultaneously. We propose a divide-and-conquer approach via an ADMM algorithm for solving large problems in a piecewise fashion.

**2.3.1. Formulation.** To decrease computation time and extend our method to larger problems, we divide our observed series  $y_i$  with  $i = 1, \dots, n$  into  $W$  overlapping windows of observations, defining the vector of sequential elements indexed from  $l_w$  to  $u_w$  as  $\mathbf{y}^{(w)} = \{y_{l_w}, y_{l_w+1}, \dots, y_{u_w-1}, y_{u_w}\}$ , with

$$1 = l_1 < l_2 < u_1 < l_3 < u_2 < l_4 < u_3 < \cdots < u_W = n.$$

We define  $n_w = u_w - l_w + 1$  so that  $\mathbf{y}^{(w)} \in \mathbb{R}^{n_w}$ . Figure 3 shows an example of 1200 observations being mapped into three equally sized overlapping windows of observations. While the overlapping trend estimates between  $l_2$  and  $u_2$  do not vary dramatically, the difference is more pronounced in the trend in the 5th quantile between  $l_3$  and  $u_2$ . Thus, we need a way of enforcing estimates to be identical in the overlapping regions.

Given quantiles  $\tau_1 < \cdots < \tau_J$ , we introduce dummy variables  $\theta_j^{(w)} \in \mathbb{R}^{n_w}$  as the value of the  $\tau_j$ th quantile trend in window  $w$ . We then “stitch” together the  $W$  quantile trend estimates into consensus over the overlapping regions by introducing the constraint  $\theta_{ij}^{(w)} = \theta_{i+l_w-1,j}$  for  $i = 1, \dots, n_w$  and for all  $j$ . Let  $\Theta^{(w)}$  be the matrix whose  $j$ th column is  $\theta_j^{(w)}$ . Then, we

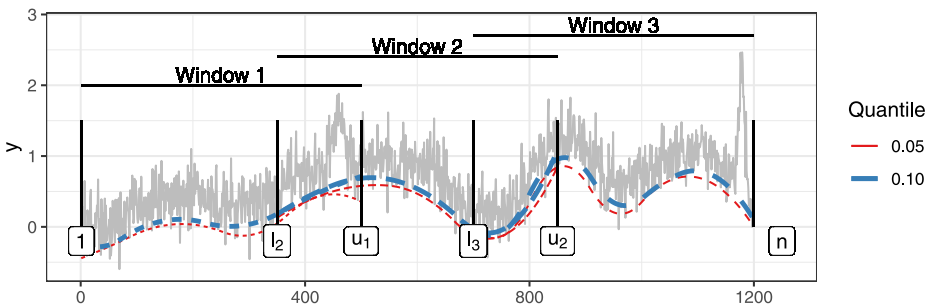


FIG. 3. Window boundaries and trends fit separately in each window. Each window’s trend estimate is plotted in a different line type.

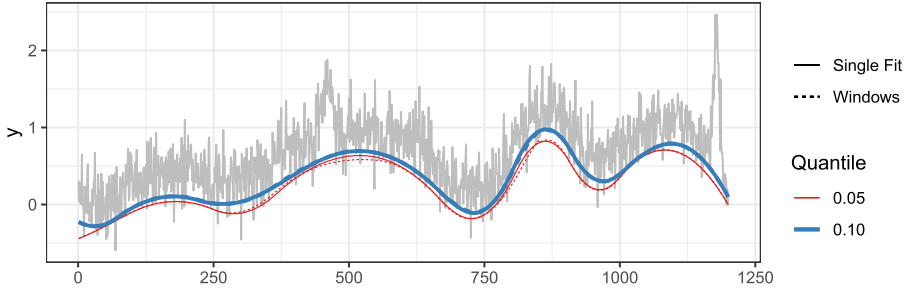


FIG. 4. *Trend fit with our ADMM algorithm with three windows (converged in seven iterations), compared to trend from simultaneous fit.*

can write these constraints more concisely as  $\Theta^{(w)} = \mathbf{U}^{(w)} \Theta$ , where  $\mathbf{U}^{(w)} \in \{0, 1\}^{n_w \times n}$  is a matrix that selects rows of  $\Theta$  corresponding to the  $w$ th window, namely,

$$\mathbf{U}^{(w)} = \begin{pmatrix} \mathbf{e}_{l_w}^\top \\ \vdots \\ \mathbf{e}_{u_w}^\top \end{pmatrix},$$

where  $\mathbf{e}_i \in \mathbb{R}^n$  denotes the  $i$ th standard basis vector. Furthermore, let  $\iota_{\mathcal{C}}$  denote the indicator function of the noncrossing quantile constraint, namely,  $\iota_{\mathcal{C}}(\Theta)$  is zero if  $\Theta \in \mathcal{C}$  and infinity otherwise. Our windowed quantile trend optimization problem can then be written as

(4) 
$$\begin{aligned} &\text{minimize} && \sum_{w=1}^W \left\{ \sum_{j=1}^J [\rho_{\tau_j}(\mathbf{y}^{(w)} - \boldsymbol{\theta}_j^{(w)}) + \lambda_j \|\mathbf{D}^{(k+1)} \boldsymbol{\theta}_j^{(w)}\|_1] + \iota_{\mathcal{C}}(\Theta^{(w)}) \right\} \\ &\text{subject to} && \Theta^{(w)} = \mathbf{U}^{(w)} \Theta \quad \text{for } w = 1, \dots, W. \end{aligned}$$

The solution to (4) is *not* identical to the solution to (3) because of double counting of the overlapping sections. The solutions are very close, however, and the differences are essentially immaterial concerning downstream analysis. Figure 4 provides an illustration of the trends estimated using multiple windows compared with the trends estimated using a single window; estimates using multiple and single windows are nearly indistinguishable.

2.3.2. *Algorithm.* The ADMM algorithm (Gabay and Mercier (1975), Glowinski and Marrocco (1975)) is described in greater detail by Boyd et al. (2011), but we briefly review how it can be used to iteratively solve the following equality constrained optimization problem which is a more general form of (4):

(5) 
$$\begin{aligned} &\text{minimize} && f(\boldsymbol{\phi}) + g(\tilde{\boldsymbol{\phi}}) \\ &\text{subject to} && \mathbf{A}\boldsymbol{\phi} + \mathbf{B}\tilde{\boldsymbol{\phi}} = \mathbf{c}. \end{aligned}$$

Recall that finding the minimizer to an equality constrained optimization problem is equivalent to the identifying the saddle point of the Lagrangian function associated with the problem (5). ADMM seeks the saddle point of a related function called the augmented Lagrangian,

$$\mathcal{L}_\gamma(\boldsymbol{\phi}, \tilde{\boldsymbol{\phi}}, \boldsymbol{\omega}) = f(\boldsymbol{\phi}) + g(\tilde{\boldsymbol{\phi}}) + \langle \boldsymbol{\omega}, \mathbf{c} - \mathbf{A}\boldsymbol{\phi} - \mathbf{B}\tilde{\boldsymbol{\phi}} \rangle + \frac{\gamma}{2} \|\mathbf{c} - \mathbf{A}\boldsymbol{\phi} - \mathbf{B}\tilde{\boldsymbol{\phi}}\|_2^2,$$

where the dual variable  $\boldsymbol{\omega}$  is a vector of Lagrange multipliers and  $\gamma$  is a nonnegative tuning parameter. When  $\gamma = 0$ , the augmented Lagrangian coincides with the ordinary Lagrangian.

ADMM minimizes the augmented Lagrangian one block of variables at a time before updating the dual variable  $\omega$ . This yields the following sequence of updates at the  $(m + 1)$ th ADMM iteration:

$$\begin{aligned}
 (6) \quad \phi_{m+1} &= \arg \min_{\phi} \mathcal{L}_{\gamma}(\phi, \tilde{\phi}_m, \omega_m), \\
 \tilde{\phi}_{m+1} &= \arg \min_{\tilde{\phi}} \mathcal{L}_{\gamma}(\phi_{m+1}, \tilde{\phi}, \omega_m), \\
 \omega_{m+1} &= \omega_m + \gamma(\mathbf{c} - \mathbf{A}\phi_{m+1} - \mathbf{B}\tilde{\phi}_{m+1}).
 \end{aligned}$$

Returning to our constrained windows problem given in (4), let  $\Omega^{(w)}$  denote the Lagrange multiplier matrix for the  $w$ th consensus constraint, namely,  $\Theta^{(w)} = \mathbf{U}^{(w)}\Theta$ , and let  $\omega_j^{(w)}$  denote its  $j$ th column.

The augmented Lagrangian is given by

$$\mathcal{L}(\Theta, \{\Theta^{(w)}\}, \{\Omega^{(w)}\}) = \sum_{w=1}^W \mathcal{L}_w(\Theta, \Theta^{(w)}, \Omega^{(w)}),$$

where

$$\begin{aligned}
 \mathcal{L}_w(\Theta, \Theta^{(w)}, \Omega^{(w)}) &= \sum_{j=1}^J \left[ \rho_{\tau_j}(\mathbf{y}^{(w)} - \theta_j^{(w)}) + \lambda_j \|\mathbf{D}^{(k+1)}\theta_j^{(w)}\|_1 \right. \\
 &\quad \left. + (\theta_j^{(w)} - \mathbf{U}^{(w)}\theta_j)^{\top} \omega_j^{(w)} + \frac{\gamma}{2} \|\theta_j^{(w)} - \mathbf{U}^{(w)}\theta_j\|_2^2 \right] + \iota_{\mathcal{C}}(\Theta^{(w)}),
 \end{aligned}$$

and where  $\gamma$  is a positive tuning parameter.

The ADMM algorithm alternates between updating the consensus variable  $\Theta$ , the window variables  $\{\Theta^{(w)}\}$  and the Lagrange multipliers  $\{\Omega^{(w)}\}$ . At the  $(m + 1)$ th iteration, we perform the following sequence of updates:

$$\begin{aligned}
 \Theta_{m+1} &= \arg \min_{\Theta} \mathcal{L}(\Theta, \{\Theta_m^{(w)}\}, \{\Omega_m^{(w)}\}), \\
 \Theta_{m+1}^{(w)} &= \arg \min_{\{\Theta^{(w)}\}} \mathcal{L}(\Theta_{m+1}, \{\Theta^{(w)}\}, \{\Omega_m^{(w)}\}).
 \end{aligned}$$

*Updating  $\Theta$ .* Some algebra shows that, defining  $i_w = i - l_w + 1$ , updating the consensus variable step is computed as follows:

$$(7) \quad \theta_{ij} = \begin{cases} \frac{1}{2}(\theta_{i_{w-1}j}^{(w-1)} + \theta_{i_{w}j}^{(w)}) - \frac{1}{2\gamma}(\omega_{i_{w-1}j}^{(w-1)} + \omega_{i_{w}j}^{(w)}) & \text{if } l_w \leq i \leq u_{w-1}, \\ \theta_{i_{w}j}^{(w)} & \text{if } u_{w-1} < i \leq l_{w+1}, \\ \frac{1}{2}(\theta_{i_{w}j}^{(w)} + \theta_{i_{w+1}j}^{(w+1)}) - \frac{1}{2\gamma}(\omega_{i_{w}j}^{(w)} + \omega_{i_{w+1}j}^{(w+1)}) & \text{if } l_{w+1} < i \leq u_w. \end{cases}$$

The consensus update (7) is rather intuitive. We essentially average the trend estimates in overlapping sections of the windows, subject to some adjustment by the Lagrange multipliers, and leave the trend estimates in nonoverlapping sections of the windows untouched. For notational ease, we write the consensus update (7) compactly as  $\Theta = \psi(\{\Theta^{(w)}\}, \{\Omega^{(w)}\})$ .



**Algorithm 1** ADMM algorithm for quantile trend filtering with windows

---

```

Define  $\mathbf{D} = \mathbf{D}^{(k+1)}$ .
initialize:
for  $w = 1, \dots, W$  do
   $\Theta_0^{(w)} \leftarrow \arg \min_{\Theta^{(w)} \in \mathcal{C}} \sum_{j=1}^J \rho_{\tau_j}(\mathbf{y}^{(w)} - \theta_j^{(w)}) + \lambda \|\mathbf{D}\theta_j^{(w)}\|_1$ 
   $\Omega_0^{(w)} \leftarrow \mathbf{0}$ 
end for
 $m \leftarrow 0$ 
repeat
   $\Theta_{m+1} \leftarrow \psi(\{\Theta_m^{(w)}\}, \{\Omega_m^{(w)}\})$ 
  for  $w = 1, \dots, W$  do
     $\Theta_{m+1}^{(w)} \leftarrow \arg \min_{\Theta^{(w)}} \mathcal{L}_w(\Theta_{m+1}, \Theta^{(w)}, \Omega_m^{(w)})$ 
     $\Omega_{m+1}^{(w)} \leftarrow \Omega_m^{(w)} + \gamma(\Theta_{m+1}^{(w)} - \mathbf{U}^{(w)}\Theta_{m+1})$ 
  end for
   $m \leftarrow m + 1$ 
until convergence
return  $\Theta_m$ 

```

---

*Updating  $\{\Theta^{(w)}\}$ .* We then estimate the trend separately in each window, which can be done in parallel, while penalizing the differences in the overlapping pieces of the trends as outlined in Algorithm 1. The use of the augmented Lagrangian converts the problem of solving a potentially large linear program into solving a collection of smaller quadratic programs. The `gurobi` R package ([Gurobi Optimization \(2018\)](#)) can solve quadratic programs in addition to linear programs, but we can also use the free R package `quadprog` ([Weingessel and Turlach \(2013\)](#)).

The windowed quantile trend filtering optimization problem (4) opens the door to significant computational gains over solving the original quantile trend filtering problem (3). The quantile trend filtering problem for a single window is a linear program with  $N \times J$  parameters (number of observations by number of quantiles) which can be solved in computational time proportional to  $(NJ)^3$ . Consequently, solving a large problem using Algorithm 1 should require less computational time than solving (3), even if the subproblems are solved sequentially. We demonstrate the advantages of Algorithm 1 through timing experiments (Figure 5). For each data size, 25 datasets were simulated using the peaks simulation design described below. Trends for the fifth, tenth and fifteenth quantiles were fit simultaneously using  $\lambda_j = n/5$  for all  $j$ . We use from one to four windows for each data size with an overlap of 500. Figure 5 shows that using four windows instead of one on data sizes of 55,000 provides a factor

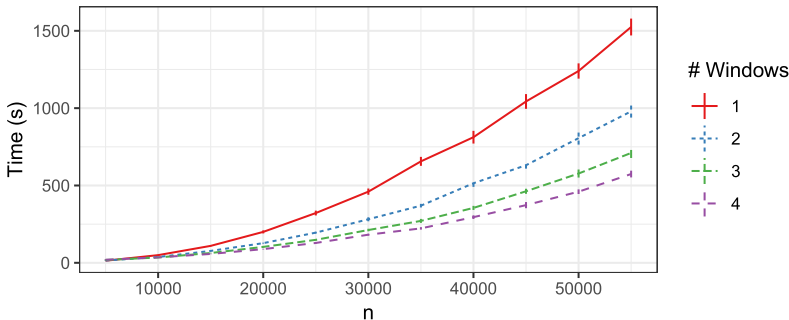


FIG. 5. Timing experiments comparing quantile trend filtering with varying numbers of windows by data size.



of 3 decrease in computation time. The timing experiments were conducted on an Intel Xeon based Linux cluster using two processor cores.

We close the discussion of Algorithm 1 by addressing how to decide the number of windows and their degree of overlap. As the windowed and original quantile trend filtering problems provide nearly indistinguishable estimates, a simple rule of thumb for choosing  $W$  is to choose as few windows as one's computational budget permits. Regarding the overlap between adjacent windows, the overlap is application dependent and should be larger than the expected time duration of transient events.

Additional details regarding the convergence properties and stopping criteria for Algorithm 1 are in Section 1 of the Supplementary Material (Brantley, Guinness and Chi (2020)).

**3. Model selection.** An important practical issue in baseline estimation is the choice of the regularization parameter  $\lambda$  which controls the degree of smoothness in  $\theta$ . In this section we introduce four methods for choosing  $\lambda$ . The first is a validation based approach; the latter three are based on information criteria. Each of the criteria we compare is calculated for a single quantile ( $\tau_j$ ). Rather than combine results over quantiles, we allow the value of  $\lambda$  to vary by quantile, resulting in  $\lambda = \{\lambda_1, \dots, \lambda_J\}$ . To choose the best value for each  $\lambda_j$ , we first estimate all of the quantile trends using  $\lambda_j = \lambda$  for all  $j$  over a grid of values for  $\lambda$ . We then determine the  $\lambda_j$  that maximizes the criteria chosen evaluated using  $\tau_j$ . Finally, we reestimate the noncrossing trends with the optimal values for  $\lambda_j$ . A more thorough approach would involve fitting the model on a  $J$  dimensional grid of values for  $\lambda$ , but this is computationally infeasible.

**3.1. Validation.** Our method can easily handle missing data by defining the check loss function to output 0 for missing values. Specifically, we use the following modified function  $\tilde{\rho}_\tau$  in place of the  $\rho_\tau$  function given in (2)

$$(8) \quad \tilde{\rho}_\tau(\mathbf{r}) = \sum_{i \notin \mathcal{V}} r_i (\tau - \mathbb{1}(r_i < 0)),$$

where  $\mathcal{V}$  is a held-out validation subset of  $\{1, \dots, n\}$  and solve the problem

$$(9) \quad \underset{\theta \in \mathcal{C}}{\text{minimize}} \sum_{j=1}^J [\tilde{\rho}_{\tau_j}(\mathbf{y} - \theta_j) + \lambda_j \|\mathbf{D}^{(k+1)} \theta_j\|_1].$$

The modified penalized negative log-likelihood in (9) corresponds to making a missing completely at random (MCAR) assumption and can be solved via Algorithm 1 with a trivial modification to the quadratic program subproblems. Although it is normally considered a strong assumption in general, the MCAR assumption does not present any disadvantages for model selection purposes as the validation subset can be selected by taking a completely random sample from among the observed entries  $y_i$ . For each quantile level  $j$ , we select the  $\lambda_j$  that minimizes the hold-out prediction error quantified by  $\tilde{\rho}_{\tau_j}(\mathbf{y} - \hat{\theta}_j(\lambda_j))$ , where  $\hat{\theta}_j(\lambda_j)$  is the solution to (9) and

$$\check{\rho}_\tau(\mathbf{r}) = \sum_{i \in \mathcal{V}} r_i (\tau - \mathbb{1}(r_i < 0)).$$

**3.2. Information criteria.** Koenker, Ng and Portnoy (1994) addressed the choice of regularization parameter by proposing the Schwarz criterion for the selection of  $\lambda$ ,

$$\text{SIC}(p_\lambda, \tau_j) = \log \left[ \frac{1}{n} \rho_{\tau_j}(\mathbf{y} - \theta_j) \right] + \frac{1}{2n} p_\lambda \log n,$$

where  $p_\lambda = \sum_i \mathbb{1}(y_i = \hat{\theta}_i)$  is the number of noninterpolated points which can be thought of as active knots. Equivalently,  $p_\lambda$  can be substituted with the number of nonzero components in  $\mathbf{D}^{(k+1)}\hat{\boldsymbol{\theta}}_j$ , which we denote  $\nu$  and have found to be more numerically stable. The SIC is based on the traditional Bayesian information criterion (BIC) which is given by

$$(10) \quad \text{BIC}(\nu) = -2\log(L(\hat{\boldsymbol{\theta}})) + \nu \log n,$$

where  $L$  is the likelihood function. If we take the approach used in Bayesian quantile regression (Yu and Moyeed (2001)) and view minimizing the check function as maximizing the asymmetric Laplace likelihood

$$L(\mathbf{y} \mid \boldsymbol{\theta}) = \left[ \frac{\tau^n(1-\tau)}{\sigma} \right]^n \exp \left\{ -\rho_\tau \left( \frac{\mathbf{y} - \boldsymbol{\theta}}{\sigma} \right) \right\},$$

we can compute the BIC as

$$\text{BIC}(\nu, \tau_j) = \frac{2}{\sigma} \rho_{\tau_j}(\mathbf{y} - \hat{\boldsymbol{\theta}}_j) + \nu \log n,$$

where  $\hat{\boldsymbol{\theta}}$  is the estimated trend and  $\nu$  is the number of nonzero elements of  $\mathbf{D}^{(k+1)}\hat{\boldsymbol{\theta}}$ . We can choose any  $\sigma > 0$  and have found empirically that  $\sigma = \frac{1-|1-2\tau|}{2}$  produces stable estimates.

A limitation of the BIC, however, is that it may be inconsistent in the high-dimensional setting. This issue arises as a consequence of an implicit assumption in deriving the BIC that all models under consideration are equally likely. Consequently, in the context of variable selection methods, such as the lasso, models that include more covariates (up to half of all possible covariates) will have a larger prior probability of being selected under the BIC criterion. In the context of this work, using the BIC may have a tendency to favor the selection of undersmoothed signals, as signals  $\boldsymbol{\theta}$  with more nonzero entries in the vector  $\mathbf{D}^{(k+1)}\boldsymbol{\theta}$  will be assigned higher prior probabilities under the BIC.

To address this issue, Chen and Chen (2008) modified the prior probabilities to dampen the prior weight on larger models, or in the context of this work, undersmoothed signals, and proposed the extended Bayesian information criteria (eBIC), specifically designed for large parameter spaces.

$$\text{eBIC}_\gamma(\nu) = -2\log(L(\hat{\boldsymbol{\theta}})) + \nu \log n + 2\gamma \log \left( \frac{P}{\nu} \right), \quad \gamma \in [0, 1],$$

where  $P$  is the total number of possible parameters and  $\nu$  is the number of nonzero parameters included in a given model. Chen and Chen (2008) prove that the eBIC is model selection consistent under mild regularity conditions. We used the eBIC criterion with  $\gamma = 1$  and  $P = n - k - 1$ . We note that the eBIC could be used to select not only  $\lambda$  but also  $k$  by plugging in for  $\nu$  the expected degrees of freedom in the regular trend filtering model (Tibshirani (2014)).

We compare the performance of the SIC, scaled eBIC (with  $\sigma$  defined above) and validation methods in our simulation studies below.

**4. Simulation studies.** We conduct two simulation studies to compare the performance of our quantile trend filtering method and regularization parameter selection criteria with previously published methods. The first study compares the method's ability to estimate quantiles when the observed series consists of a smooth trend plus independent error but does not contain transient components. The second study is based on our application and compares the method's ability to estimate baseline trends and enable peak detection when the time series contains a nonnegative, transient signal in addition to the trend and random component.

We compare three criteria for choosing the smoothing parameter for quantile trend filtering:  $\lambda$  chosen using SIC (3.2) (`detrendr_SIC`),  $\lambda$  chosen using the validation method

with the validation set consisting of every 5th observation (`detrendr_valid`) and  $\lambda$  chosen using the proposed eBIC criterion (3.2) (`detrendr_eBIC`). For the second study we also examine the effect of the noncrossing quantile constraint by estimating the quantile trends separately and choosing  $\lambda$  using eBIC (`detrendr_Xing`). We do not include `detrendr_Xing` in the first study because the difference in quantiles is large enough that we would not expect the noncrossing constraint to make a difference.

We also compare the performance of our quantile trend filtering method with three previously published methods, none of which guarantee non-crossing quantiles:

- `npqw`: The local linear quantile method (quantile-ll) described in Racine and Li (2017). Code was obtained from the author.
- `qsreg`: Quantile smoothing splines described in Oh et al. (2004) and Nychka et al. (1995) and implemented in the `fields` R package (Nychka et al. (2017)). The regularization parameter was chosen using generalized cross-validation.
- `rqss`: Quantile trend filtering with  $k = 1$  available in the `quantreg` package and described in Koenker, Ng and Portnoy (1994). The regularization parameter is chosen using a grid search and minimizing the SIC (3.2), as described in Koenker, Ng and Portnoy (1994).

4.1. *Estimating quantiles.* To compare performance in estimating quantile trends in the absence of a signal component, three simulation designs from Racine and Li (2017) were considered. For all designs  $t = 1, \dots, n$ ,  $x(t) = t/n$  and the response  $y$  was generated as

$$y(t) = \sin(2\pi x(t)) + \epsilon(x(t)).$$

The errors were simulated as independent draws from the following distributions:

- Gaussian:  $\epsilon(x(t)) \sim N(0, (\frac{1+x(t)^2}{4})^2)$ .
- Beta:  $\epsilon(x(t)) \sim \text{Beta}(1, 11 - 10x(t))$ .
- Mixed normal:  $\epsilon(x(t))$  is simulated from a mixture of  $N(-1, 1)$  and  $N(1, 1)$  with mixing probability  $x(t)$ .

The true quantile trends and example simulated data sets are shown in Figure 6. One hundred datasets were generated for each design and size (300, 500 and 1000) considered for a total of 900 datasets.

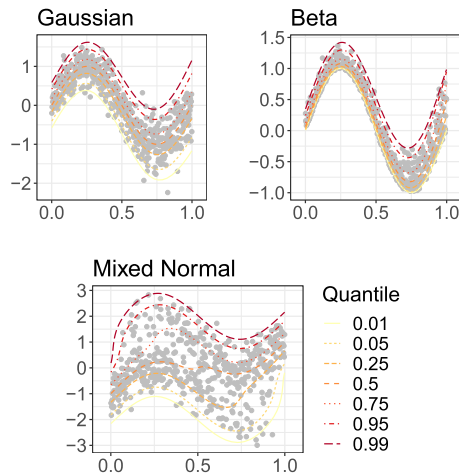


FIG. 6. Simulated data with true quantile trends.

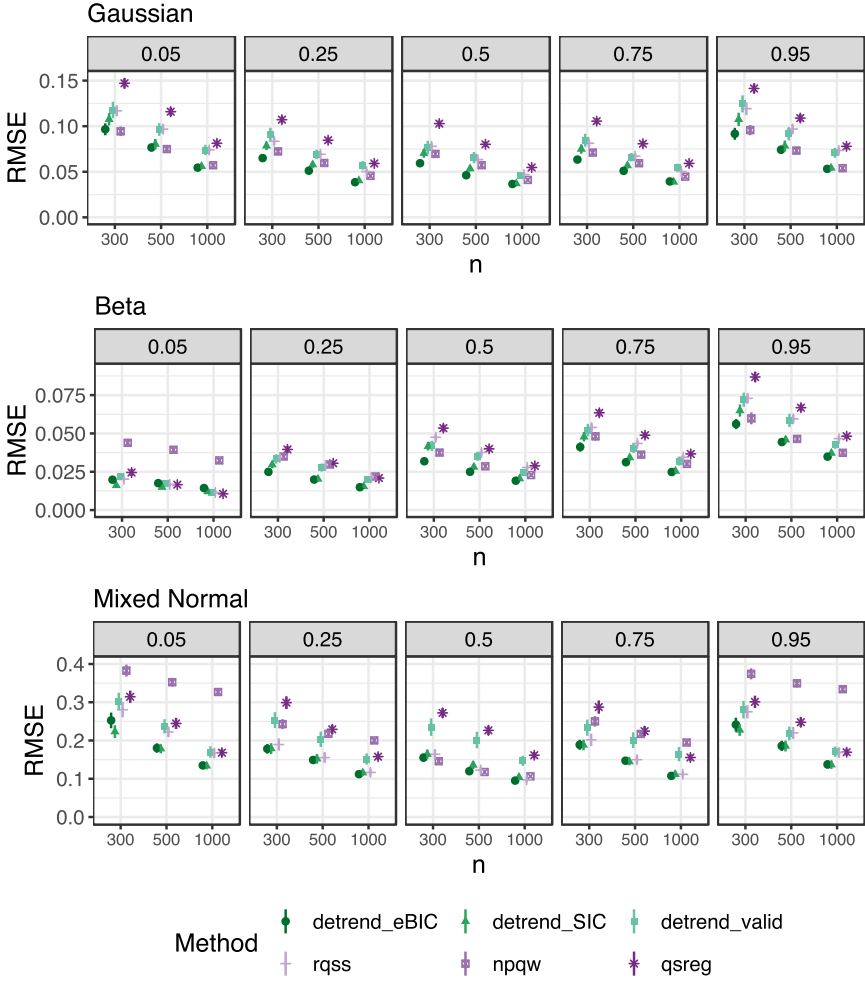


FIG. 7. RMSE by design, method, quantile and data size for designs without a transient component. Points and error bars represent mean RMSE  $\pm$  twice the standard error.

Quantile trends were estimated for  $\tau = \{0.05, 0.25, 0.5, 0.75, 0.95\}$ , and the root mean squared error was calculated as  $\text{RMSE}(\tau_j) = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{\theta}_{ij} - \theta_{ij})^2}$ , where  $\theta_{ij}$  is the true value of the  $\tau_j$ th quantile of  $y$  at  $t = i$ . Figure 7 shows the mean RMSE plus or minus twice the standard error for each method, quantile level and sample size (table available in Section 3 of the Supplementary Material (Brantley, Guinness and Chi (2020))). In all three designs the proposed `detrend` methods are either better than or comparable to existing methods. Overall, the `detrend_eBIC` performs best. In the mixed normal design, specifically, our methods have lower RMSEs for the 5th and 95th quantiles. The `npqw` method performs particularly poorly on the mixed normal design due to the violation of the assumption that the data come from a scale-location model.

**4.2. Peak detection.** The second simulation design is closely motivated by our air quality analysis problem. We assume that the measured data can be represented by

$$y(t_i) = \theta(t_i) + s(t_i) + \varepsilon_i,$$

where  $t_i = i$  for  $i = 1, \dots, n$ ,  $\theta(t)$  is the drift component that varies smoothly over time,  $s(t)$  is the true signal at time  $t$  and  $\varepsilon_i$  are i.i.d. errors distributed as  $N(0, 0.25^2)$ . We generate  $\theta(t)$  using cubic natural spline basis functions with degrees of freedom sampled from

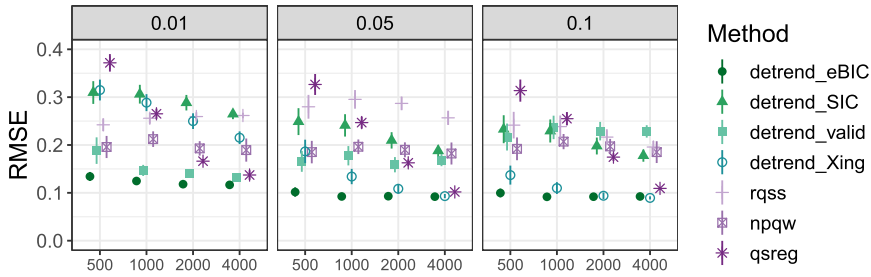


FIG. 8. *RMSE by method, quantile and data size for peaks design. Points and error bars represent mean RMSE  $\pm$  twice the standard error.*

a Poisson distribution with mean parameter equal to  $n/100$  and coefficients drawn from an exponential distribution with rate 1. The true signal function  $s(t)$  is assumed to be zero with peaks generated using the Gaussian density function. The number of peaks is sampled from a binomial distribution with size equal to  $n$  and probability equal to 0.005 with location parameters uniformly distributed between 1 and  $n - 1$  and bandwidths uniformly distributed between 2 and 12. The simulated peaks were multiplied by a factor that was randomly drawn from a normal distribution with mean 20 and standard deviation of 4. An example dataset with four signal peaks is shown in Figure 9. One hundred datasets were generated for each  $n \in \{500, 1000, 2000, 4000\}$ .

We compare the ability of the methods to estimate the true quantiles of  $y(t_i) - s(t_i)$  for  $\tau \in \{0.01, 0.05, 0.1\}$  and calculate the RMSE (Figure 8, table available in Section 3 of the Supplementary Material (Brantley, Guinness and Chi (2020))). In this simulation study our proposed method `detrend_eBIC` method substantially outperforms the others. The `detrend_Xing` method, which is the `detrend_eBIC` method fit without the non-crossing constraints, performs similarly for larger quantiles and larger datasets. However, `detrend_Xing` produces significantly worse estimates for more extreme quantiles ( $\tau = 0.01$ ) and smaller data sets ( $\tau = 0.05$  and  $n = 500$ ). These results indicate that, even when a single quantile is of interest, simultaneously fitting nearby quantiles and utilizing the noncrossing constraint can improve estimates when data is sparse by using information from nearby quantiles. The `qsreg` method is comparable to the `detrend_eBIC` method on the larger datasets, but its performance deteriorates as the data size shrinks. The `npqw` and `rqss` methods both perform poorly on this design.

While minimizing RMSE is desirable in general, in our application the primary metric of success is accurately classifying observations  $y_i$  into signal present or absent. To evaluate the accuracy of our method compared to other methods, we define true signal as any time point when the simulated peak value is greater than 0.5. We compare three different quantiles for the baseline estimation and four different thresholds for classifying the signal after

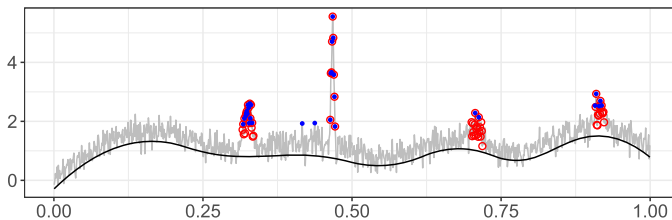


FIG. 9. *Example signal classification using threshold. Large hollow points indicate true signal ( $y_i - \theta_i > 0.5$ ); small filled points indicate observations classified as signal, that is, values greater than 1.2 after baseline removal using `detrend_eBIC`.*

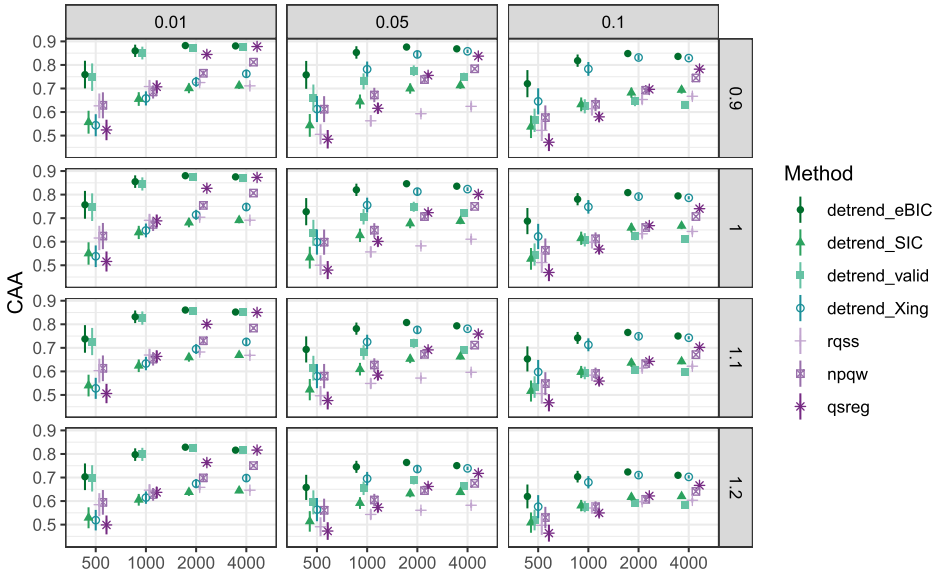


FIG. 10. Class averaged accuracy by threshold, data size and method (1 is best 0.5 is worst).

subtracting the estimated baseline from the observations. Figure 9 illustrates the observations classified as signal after subtracting the baseline trend compared to the “true signal.”

To compare the resulting signal classifications, we calculate the class averaged accuracy (CAA), which is defined as

$$CAA = \frac{1}{2} \left[ \frac{\sum_{i=1}^n \mathbb{1}[\delta_i = 1 \cap \hat{\delta}_i = 1]}{\sum_{i=1}^n \mathbb{1}[\delta_i = 1]} + \frac{\sum_{i=1}^n \mathbb{1}[\delta_i = 0 \cap \hat{\delta}_i = 0]}{\sum_{i=1}^n \mathbb{1}[\delta_i = 0]} \right],$$

where  $\delta_i \in \{0, 1\}$  is the vector of true signal classifications and  $\hat{\delta}_i \in \{0, 1\}$  is the vector of estimated signal classifications, namely,  $\hat{\delta}_i = \mathbb{1}(y_i - \hat{\theta}_i > h)$ , where  $h$  is the chosen threshold. We use this metric because our classes tend to be very imbalanced with many more zeros than ones. The CAA metric should give a score close to 0.5 both for random guessing and also for trivial classifiers such as  $\hat{\delta}_i = 0$  for all  $i$ .

Our `detrend_eBIC` method results in the largest CAA values (Figure 10) in addition to the smallest RMSE values (Figure 8). Tables corresponding to Figure 10 and Figure 8 are available in Section 3 of the Supplementary Material (Brantley, Guinness and Chi (2020)). While `qsreg` was competitive with our method in some cases, in the majority of cases the largest CAA values for each threshold were produced using the `detrend_eBIC` method with the 1st or 5th quantiles.

**5. Analysis of air quality data.** The low-cost “SPod” air quality sensors output a time series that includes a slowly varying baseline, the sensor response to pollutants and high frequency random noise. These sensors record measurements every second and are used to monitor pollutant concentrations at the perimeter of industrial facilities. Time points with high concentrations are identified and compared with concurrent wind direction and speed. Ideally, three colocated and time aligned sensors (as shown in Figure 1) responding to a pollutant plume would result in the same signal classification after baseline trend removal and proper threshold choice. We first illustrate the difference between our `detrend_eBIC` method, hereafter referred to as `detrendr`, and `qsreg` using data from 13:10 to 15:10 from Figure 1 (Section 5.1). We then compare the methods on the complete day shown in Figure 1, estimating trends by applying the `qsreg` method to two hour nonoverlapping windows of the data and Algorithm 1 to the entire day. We focus on this day because data from

three sensors was available. Finally, we examine an entire week of measurements from two colocated sensors (Section 5.2).

**5.1. Short series of air quality measurements.** We compare our `detrendr` method with the `qsreg` method on a two-hour subset of one-second SPod data ( $n = 7200$ ) both to facilitate visualization and because the `qsreg` method cannot handle all 24 hours simultaneously. We estimate the baseline trend using  $\tau = \{0.01, 0.05, 0.1\}$  and compare three thresholds for classifying the signal. The thresholds are calculated as the 90th, 95th and 99th quantiles of the detrended series for each SPod. If there is signal present in the dataset, values above these thresholds should occur simultaneously on all three SPods. We do not use class-averaged accuracy to compare the signal classifications because we do not have a reference value to define as the “true” signal. Instead, we compute the variation of information (VI) which compares the similarity between two classifications. Given the signal classifications for SPods  $a$  and  $b$ ,  $\delta_i^{(a)} \in \{0, 1\}$  and  $\delta_i^{(b)} \in \{0, 1\}$ , for  $i \in \{1, \dots, n\}$  the VI is defined as

$$r_{jk} = \frac{1}{n} \sum_i \mathbb{1}(\delta_i^{(a)} = j \cap \delta_i^{(b)} = k),$$

$$\text{VI}(a, b) = - \sum_{j,k} r_{jk} \left[ \log \left( \frac{r_{jk}}{\frac{1}{n} \sum_i \mathbb{1}(\delta_i^{(a)} = j)} \right) + \log \left( \frac{r_{jk}}{\frac{1}{n} \sum_i \mathbb{1}(\delta_i^{(b)} = k)} \right) \right],$$

where  $(j, k) \in \{(0, 0), (0, 1), (1, 0), (1, 1)\}$ . The VI is a distance metric for measuring similarity of classifications and will be 0 if the classifications are identical and increase as the classifications become more different.

Figure 11 shows the estimated 5th quantile trends from each method for each SPod. The `detrendr` method results in a smoother baseline estimate while the `qsreg` method absorbs more of the peaks obscuring some of the signal. Figure 12 shows the series after subtracting the `detrendr` estimate of the 5th quantile and classifying the signal using the 95th quantile of the detrended data. The 90th and 99th thresholds are also shown for comparison as dotted and dashed lines, respectively. The largest peaks at 13:10 are easily identified as signal, but good baseline estimates also enable proper classification of the smaller peaks like the one at

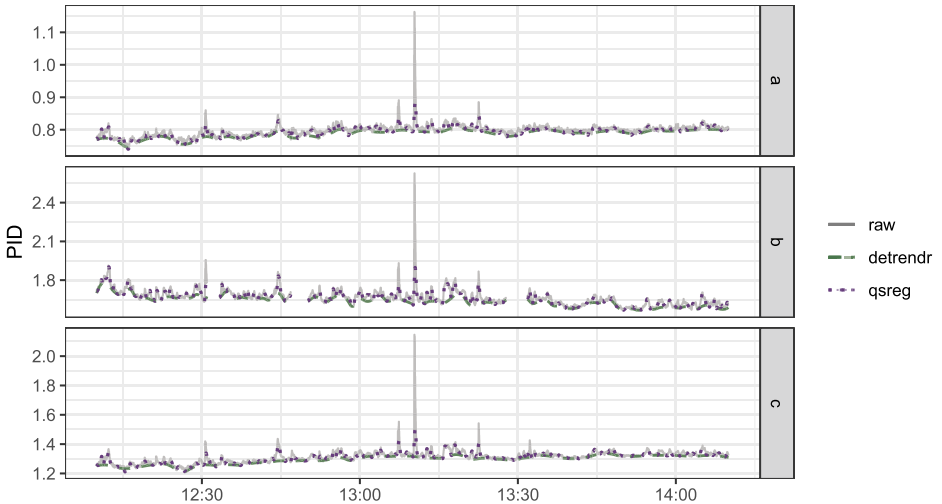


FIG. 11. Estimated 5th quantile trends for SPods  $a$ ,  $b$  and  $c$ , using `qsreg` and `detrendr`. SPod  $c$  contains some missing values that were interpolated before the trends were estimated. `qsreg` is more influenced by the signal component apparent on all nodes resulting in under-smoothing.



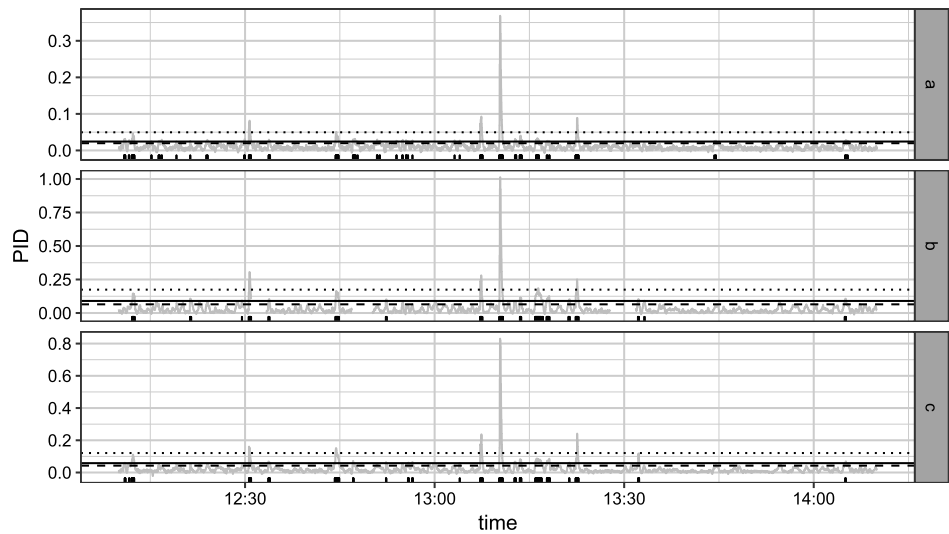


FIG. 12. Rugplot showing locations of signal after baseline removal using the `detrendr` estimate of 5th quantile. Horizontal dashed lines represent the thresholds calculated using the 90th, 95th and 99th quantiles of the detrended data. The 95th quantile (solid line) was used to classify the signal shown as vertical lines at the bottom of the plot.

12:30. The under-smoothing of the `qsreg` method results in less similar signal classifications and higher VI values for the 90th and 95th quantile thresholds (Figure 13). However, when the 99th threshold is used, only the highest observations are classified as signal, and the baseline estimation method is not as important (Figure 13).

5.2. Long series of air quality measurements. Algorithm 1 was used to remove the baseline drift from the full day of data (Figure 1) consisting of 86,400 observations per SPod and compared to the series detrended using the `qsreg` trends estimated using nonoverlapping two hour windows. As in the shorter illustration, the `detrendr` method results in generally lower VI scores than the `qsreg` method (Figure 14). The `detrendr` method also results in better correlation in the detrended series, as is illustrated in (Figure 15). The Spearman correlation coefficients for SPods a and b, SPods a and c and SPods b and c after removing

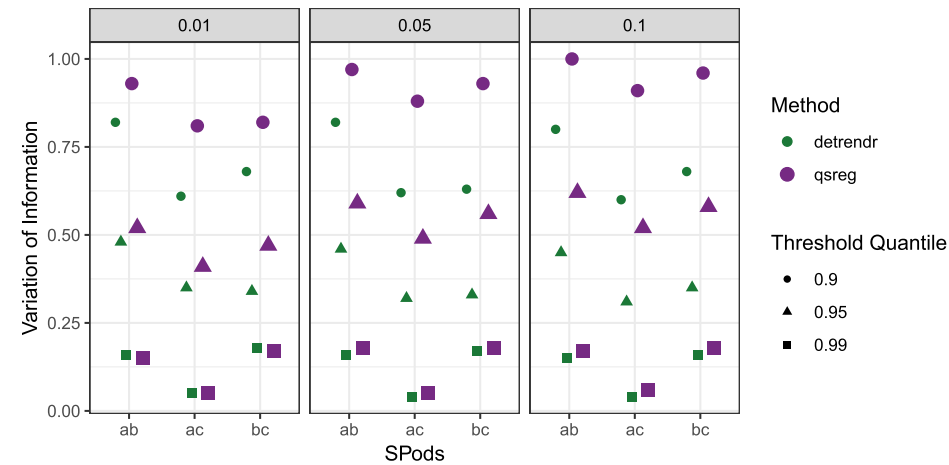


FIG. 13. Variation of information between sensors by method (color and size), quantile (columns) and threshold (shape) for two hour time period.

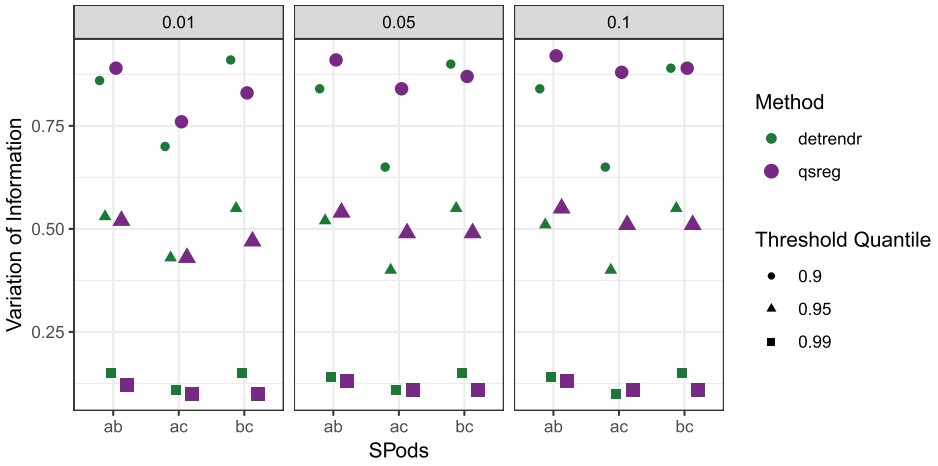


FIG. 14. Variation of information between sensors by method (color and size), quantile (columns) and threshold (shape) for full day.

the 5th quantile trend using `detrendr` were 0.37, 0.75 and 0.43, compared with 0.07, 0.24 and 0.16 using `qsreg`. The noise variance was higher for SPod b than for SPods a and c, resulting in lower correlation and higher VI values for the ab and bc metrics compared with ac.

Finally, we estimated the quantile trends for seven days of measurements of two colocated SPods. Figure 16 demonstrates the improvement in classification similarity when using `detrendr`. Each point represents a day of measurements and all points that fall below the dashed line have more similar classifications using `detrendr` compared to `qsreg`. The improvement of `detrendr` over `qsreg` is more severe at lower thresholds. This indicates that `detrendr` gives greater agreement on signal classification when the methods are tuned to deliver positive classifications more frequently.

**6. Conclusion and discussion.** We have expanded the quantile trend filtering method by implementing a noncrossing constraint, introducing a new algorithm for processing big time series data and proposing a modified criteria for smoothing parameter selection. Furthermore, we have demonstrated the utility of quantile trend filtering in both simulations and applied settings. Our ADMM algorithm for large series both reduces the computing time and allows trends to be estimated on series that cannot be estimated simultaneously, while our scaled extended BIC criterion was shown to provide better estimates of quantile trends in series with and without a signal component.

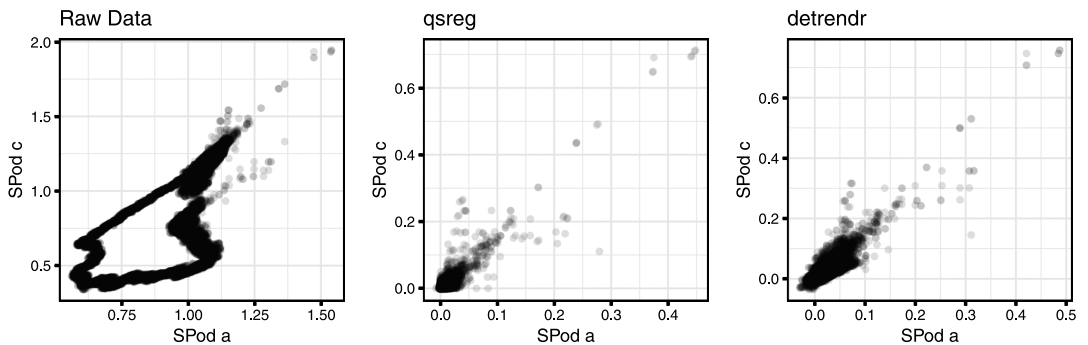


FIG. 15. SPod a versus SPod c before and after detrending with `qsreg` and `detrendr`.

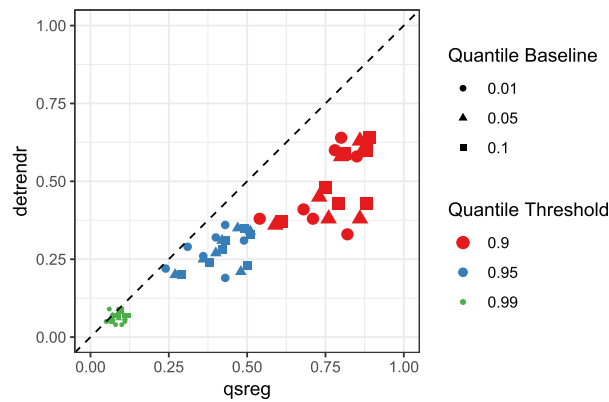


FIG. 16. Variation of information (VI) for *detrendr* and *qsreg* by quantile trend and threshold. Each point represents a full day of data. The dashed line represents  $y = x$ . In most cases *detrendr* results in a lower VI than *qsreg*.

In our application to low-cost air quality sensor data, we have shown that the baseline drift in low-cost air quality sensors can be removed through estimating quantile trends, but the data size was too large for existing methods to be computationally feasible. While *qsreg* cannot feasibly handle more than a few hours data, our new methods were able to process 24 hours simultaneously and deliver signal classifications that were more consistent between the two sensors for a week of data (168 hours). In the future, quantile trend filtering could be extended to observations measured at nonuniform spacing by incorporating the distance in covariate spacing into the differencing matrix. It could also be extended to estimate smooth spatial trends by a similar adjustment to the differencing matrix based on spatial distances between observations.

While we focus on obtaining a quantile trend filtering estimator by solving an optimization problem in this work, another potential extension would be to investigate a fully Bayesian formulation of quantile trend filtering. [Faulkner and Minin \(2018\)](#) proposed several fully Bayesian variations of the trend filter by utilizing Laplace and Horseshoe shrinkage priors on the  $k$ th-order differences in values of the unknown target function. They used a Gaussian likelihood for the data corresponding to the squared loss used by [Kim et al. \(2009\)](#). A fully Bayesian variation of the quantile trend filter could be implemented by building on the work of [Faulkner and Minin \(2018\)](#) by substituting an asymmetric Laplace likelihood for the Gaussian likelihood.

Finally, we note that quantile regression has often been used to perform robust estimation in the presence of outliers. Indeed, one could pose our peak detection problem as an outlier detection problem where outliers could only occur as extreme positive and sudden shifts from a smooth and slowly varying baseline. Consequently, an important potential use case of our quantile trend filtering method is in robust smoothing of time series data. We leave this and the other extensions discussed above for future work.

An R-package *detrendr* containing code to perform the methods described in the article is available in the Supplementary Material ([Brantley, Guinness and Chi \(2020\)](#)) and at <https://github.com/halleybrantley/detrendr>.

**Acknowledgments.** The authors greatly appreciate Eben Thoma at the U.S. Environmental Protection Agency for providing the SPod datasets. The authors also thank Chandni Malhotra for help with preliminary numerical experiments. This work was supported by the National Science Foundation under grants No. 1613219 and No. 1752692 and the National Institutes of Health under grant No. R01ES027892. The authors are also thankful for the

Oak Ridge Institute of Science Fellowship that supported this work. Plots were created using ggplot2 (Wickham (2016)).

*Disclaimer* The views expressed in this publication are those of the authors and do not represent the views or policies of the U.S. Environmental Protection Agency.

## SUPPLEMENTARY MATERIAL

**Supplement A: Algorithm details and simulation tables** (DOI: [10.1214/19-AOAS1318SUPPA](https://doi.org/10.1214/19-AOAS1318SUPPA); .pdf). The supplementary materials contain additional details on the ADMM algorithm as well as tables summarizing the results plotted in Figures 7, 8, and 10.

**Supplement B: detrendr R package** (DOI: [10.1214/19-AOAS1318SUPPB](https://doi.org/10.1214/19-AOAS1318SUPPB); .zip). R-package detrendr containing code to perform the methods described in the article (GNU zipped tar file).

## REFERENCES

- BOYD, S., PARIKH, N., CHU, E., PELEATO, B., ECKSTEIN, J. et al. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **3** 1–122.
- BRANTLEY, H. L., GUINNESS, J. and CHI, E. C. (2020). Supplement to “Baseline drift estimation for air quality data using quantile trend filtering.” <https://doi.org/10.1214/19-AOAS1318SUPPA>, <https://doi.org/10.1214/19-AOAS1318SUPPB>.
- BRANTLEY, H., HAGLER, G., KIMBROUGH, E., WILLIAMS, R., MUKERJEE, S. and NEAS, L. (2014). Mobile air monitoring data-processing strategies and effects on spatial air pollution trends. *Atmos. Meas. Tech.* **7** 2169–2183.
- CHEN, J. and CHEN, Z. (2008). Extended Bayesian information criteria for model selection with large model spaces. *Biometrika* **95** 759–771. [MR2443189 https://doi.org/10.1093/biomet/asn034](https://doi.org/10.1093/biomet/asn034)
- COOMBES, K. R., BAGGERLY, K. A. and MORRIS, J. S. (2007). Pre-processing mass spectrometry data. In *Fundamentals of Data Mining in Genomics and Proteomics* 79–102. Springer, Berlin.
- DU, P., KIBBE, W. A. and LIN, S. M. (2006). Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching. *Bioinformatics* **22** 2059–2065.
- FAULKNER, J. R. and MININ, V. N. (2018). Locally adaptive smoothing with Markov random fields and shrinkage priors. *Bayesian Anal.* **13** 225–252. [MR3737950 https://doi.org/10.1214/17-BA1050](https://doi.org/10.1214/17-BA1050)
- FRANCO, M. A. D. M., MILORI, D. M. B. P. and BOAS, P. R. V. (2018). Comparison of algorithms for baseline correction of LIBS spectra for quantifying total carbon in Brazilian soils. ArXiv Preprint. Available at [arXiv:1805.03695](https://arxiv.org/abs/1805.03695).
- GABAY, D. and MERCIER, B. (1975). A Dual Algorithm for the Solution of Non Linear Variational Problems via Finite Element Approximation. Institut de recherche d’informatique et d’automatique.
- GLOWINSKI, R. and MARROCCO, A. (1975). Sur l’approximation, par éléments finis d’ordre un, et la résolution, par pénalisation-dualité, d’une classe de problèmes de Dirichlet non linéaires. *Rev. Française Automat. Informat. Recherche Opérationnelle Sér. Rouge Anal. Numér.* **9** 41–76. [MR0388811](https://doi.org/10.1016/S0378-4222(75)90001-1)
- GUROBI OPTIMIZATION, LLC (2018). Gurobi Optimizer Reference Manual.
- KIM, S.-J., KOH, K., BOYD, S. and GORINEVSKY, D. (2009).  $l_1$  trend filtering. *SIAM Rev.* **51** 339–360. [MR2505584 https://doi.org/10.1137/070690274](https://doi.org/10.1137/070690274)
- KOENKER, R. (2018). quantreg: Quantile Regression. R package version 5.36.
- KOENKER, R. and BASSETT, G. JR. (1978). Regression quantiles. *Econometrica* **46** 33–50. [MR0474644 https://doi.org/10.2307/1913643](https://doi.org/10.2307/1913643)
- KOENKER, R., NG, P. and PORTNOY, S. (1994). Quantile smoothing splines. *Biometrika* **81** 673–680. [MR1326417 https://doi.org/10.1093/biomet/81.4.673](https://doi.org/10.1093/biomet/81.4.673)
- LUO, Y., HARGRAVES, R. H., BELLE, A., BAI, O., QI, X., WARD, K. R., PFAFFENBERGER, M. P. and NAJARIAN, K. (2013). A hierarchical method for removal of baseline drift from biomedical signals: Application in ECG analysis. *Sci. World J.* **2013**.
- MARANDI, R. Z. and SABZPOUSHAN, S. (2015). Qualitative modeling of the decision-making process using electrooculography. *Behav. Res. Methods* **47** 1404–1412.
- MECOZZI, M. (2014). A polynomial curve fitting method for baseline drift correction in the chromatographic analysis of hydrocarbons in environmental samples. *APCBEE Proc.* **10** 2–6.
- NING, X., SELESNICK, I. W. and DUVAL, L. (2014). Chromatogram baseline estimation and denoising using sparsity (BEADS). *Chemom. Intell. Lab. Syst.* **139** 156–167.

- NYCHKA, D., GRAY, G., HAALAND, P., MARTIN, D. and O'CONNELL, M. (1995). A nonparametric regression approach to syringe grading for quality improvement. *J. Amer. Statist. Assoc.* **90** 1171–1178.
- NYCHKA, D., FURRER, R., PAIGE, J. and SAIN, S. (2017). *fields: Tools for Spatial Data*. R package version 9.6.
- OH, H.-S., LEE, T. C. M. and NYCHKA, D. W. (2011). Fast nonparametric quantile regression with arbitrary smoothing methods. *J. Comput. Graph. Statist.* **20** 510–526. [MR2847806](#) <https://doi.org/10.1198/jcgs.2010.10063>
- OH, H.-S., NYCHKA, D., BROWN, T. and CHARBONNEAU, P. (2004). Period analysis of variable stars by robust smoothing. *J. Roy. Statist. Soc. Ser. C* **53** 15–30. [MR2037881](#) <https://doi.org/10.1111/j.1467-9876.2004.00423.x>
- PETTERSSON, K., JAGADEESAN, S., LUKANDER, K., HENELIUS, A., HÆGGSTRÖM, E. and MÜLLER, K. (2013). Algorithm for automatic analysis of electro-oculographic data. *Biomed. Eng. Online* **12** 110.
- RACINE, J. S. and LI, K. (2017). Nonparametric conditional quantile estimation: A locally weighted quantile kernel approach. *J. Econometrics* **201** 72–94. [MR3704677](#) <https://doi.org/10.1016/j.jeconom.2017.06.020>
- RUDIN, L. I., OSHER, S. and FATEMI, E. (1992). Nonlinear total variation based noise removal algorithms *Phys. D, Nonlinear Phenom.* **60** 259–268.
- SNYDER, E., WATKINS, T., SOLOMON, P., THOMA, E., WILLIAMS, R., HAGLER, G., SHELOW, D., HINDIN, D., KILARU, V. et al. (2013). The changing paradigm of air pollution monitoring *Environ. Sci. Technol.* **47** 11369.
- THEUSSL, S. and HORNIK, K. (2017). *Rglpk: R/GNU Linear Programming Kit Interface*. R package version 0.6-3.
- THOMA, E. D., BRANTLEY, H. L., OLIVER, K. D., WHITAKER, D. A., MUKERJEE, S., MITCHELL, B., WU, T., SQUIER, B., ESCOBAR, E. et al. (2016). South Philadelphia passive sampler and sensor study. *J. Air Waste Manage. Assoc.* **66** 959–970.
- TIBSHIRANI, R. J. (2014). Adaptive piecewise polynomial estimation via trend filtering. *Ann. Statist.* **42** 285–323. [MR3189487](#) <https://doi.org/10.1214/13-AOS1189>
- TIBSHIRANI, R., SAUNDERS, M., ROSSET, S., ZHU, J. and KNIGHT, K. (2005). Sparsity and smoothness via the fused lasso. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **67** 91–108. [MR2136641](#) <https://doi.org/10.1111/j.1467-9868.2005.00490.x>
- WEINGESSEL, A. and TURLACH, B. A. (2013). *quadprog: Functions to solve quadratic programming problems*. R package version 1.5-5.
- WICKHAM, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer, New York.
- YAMADA, H. (2017). Estimating the trend in US real GDP using the  $\ell_1$  trend filtering. *Appl. Econ. Lett.* **24** 713–716.
- YU, K. and MOYEED, R. A. (2001). Bayesian quantile regression. *Statist. Probab. Lett.* **54** 437–447. [MR1861390](#) [https://doi.org/10.1016/S0167-7152\(01\)00124-9](https://doi.org/10.1016/S0167-7152(01)00124-9)