Robust Nonparametric Regression via Sparsity Control With Application to Load Curve Data Cleansing

Gonzalo Mateos, *Student Member, IEEE*, and Georgios B. Giannakis, *Fellow, IEEE*

***Abstract*—Nonparametric methods are widely applicable to statistical inference problems, since they rely on a few modeling assumptions. In this context, the fresh look advocated here permeates benefits from variable selection and compressive sampling, to robustify nonparametric regression against outliers—that is, data markedly deviating from the postulated models. A variational counterpart to least-trimmed squares regression is shown closely related to an ℓ_0-(pseudo)norm-regularized estimator, that encourages *sparsity* in a vector explicitly modeling the outliers. This connection suggests efficient solvers based on convex relaxation, which lead naturally to a variational M-type estimator equivalent to the least-absolute shrinkage and selection operator (Lasso). Outliers are identified by judiciously tuning regularization parameters, which amounts to controlling the sparsity of the outlier vector along the whole *robustification* path of Lasso solutions. Reduced bias and enhanced generalization capability are attractive features of an improved estimator obtained after replacing the ℓ_0-(pseudo)norm with a nonconvex surrogate. The novel robust spline-based smoother is adopted to cleanse *load curve* data, a key task aiding operational decisions in the envisioned smart grid system. Computer simulations and tests on real load curve data corroborate the effectiveness of the novel sparsity-controlling robust estimators.**

***Index Terms*—Lasso, load curve cleansing, nonparametric regression, outlier rejection, sparsity, splines.**

I. Introduction

CONSIDER the classical problem of function estimation, in which an input vector $\mathbf{x} := [x_1, \ldots, x_p]' \in \mathbb{R}^p$ is given, and the goal is to predict the real-valued scalar response $y = f(\mathbf{x})$. Function f is unknown, to be estimated from a training data set $\mathcal{T} := \{y_i, \mathbf{x}_i\}_{i=1}^N$. When f is assumed to be a member of a finitely-parameterized family of functions, standard (non-)linear regression techniques can be adopted. If on the other hand, one is only willing to assume that f belongs to a (possibly infinite dimensional) space of "smooth" functions $\mathcal{H}$, then a *nonparametric* approach is in order, and this will be the focus of this work.

Without further constraints beyond $f \in \mathcal{H}$, functional estimation from finite data is an ill-posed problem. To bypass this challenge, the problem is typically solved by minimizing appropriately regularized criteria, allowing one to control model complexity; see, e.g., [14], [40]. It is then further assumed that $\mathcal{H}$ has the structure of a reproducing kernel Hilbert space (RKHS), with corresponding positive definite reproducing kernel function $K(\cdot,\cdot) : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$, and norm denoted by $\|\cdot\|_{\mathcal{H}}$. Under the formalism of *regularization networks*, one seeks $\hat{f}$ as the solution to the variational problem

$$\min_{f\in\mathcal{H}} \left[\sum_{i=1}^{N} V\left(y_i - f(\mathbf{x})\right) + \mu\|f\|_{\mathcal{H}}^2\right] \tag{1}$$

where $V(\cdot)$ is a convex loss function, and $\mu \geq 0$ controls complexity by weighting the effect of the smoothness functional $\|f\|_{\mathcal{H}}^2$. Interestingly, the Representer theorem asserts that the unique solution of (1) is finitely parametrized and has the form $\hat{f}(\mathbf{x}) = \sum_{i=1}^N \beta_i K(\mathbf{x}, \mathbf{x}_i)$, where $\{\beta_i\}_{i=1}^N$ can be obtained from $\mathcal{T}$; see e.g., [35], [44]. Further details on RKHS, and in particular on the evaluation of $\|f\|_{\mathcal{H}}$, can be found in, e.g., [44, Ch. 1]. A fundamental relationship between model complexity control and generalization capability, i.e., the predictive ability of $\hat{f}$ beyond the training set, was formalized in [43].

The generalization error performance of approaches that minimize the sum of squared model residuals [that is $V(u) = u^2$ in (1)] regularized by a term of the form $\|f\|_{\mathcal{H}}^2$, is degraded in the presence of outliers. This is because the least-squares (LS) part of the cost is not robust, and can result in severe overfitting of the (contaminated) training data [26]. Recent efforts have considered replacing the squared loss with a robust counterpart such as Huber's function, or its variants, but lack a data-driven means of selecting the proper threshold that determines which datum is considered an outlier [49]; see also [32]. Other approaches have instead relied on the so-termed ϵ-insensitive loss function, originally proposed to solve function approximation problems using support vector machines (SVMs) [43]. These family of estimators often referred to as support vector regression (SVR), have been shown to enjoy robustness properties; see, e.g., [31], [33], [38] and references therein. In [10], improved performance in the presence of outliers is achieved by refining the SVR solution through a subsequent robust learning phase.

The starting point here is a variational least-trimmed squares (VLTS) estimator, suitable for robust function approximation in $\mathcal{H}$ (Section II). It is established that VLTS is closely related to

Manuscript received April 03, 2011; revised August 10, 2011 and November 17, 2011; accepted December 20, 2011. Date of publication December 26, 2011; date of current version March 06, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Raviv Raich. This work was supported by MURI Grant (AFOSR FA9550-10-1-0567). This paper appeared in part in the *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, Prague, Czech Republic, May 22–27, 2011.

The authors are with the Dept. of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: mate0058@umn.edu; georgios@umn.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2011.2181837

an (NP-hard) ℓ_0-(pseudo)norm-regularized estimator, adopted to fit a regression model that explicitly incorporates an unknown *sparse* vector of outliers [19]. As in compressive sampling (CS) [41], efficient (approximate) solvers are obtained in Section III by replacing the outlier vector's ℓ_0-norm with its closest convex approximant, the ℓ_1-norm. This leads naturally to a variational M-type estimator of f, also shown equivalent to a least-absolute shrinkage and selection operator (Lasso) [39] on the vector of outliers (Section III-A). A tunable parameter in Lasso *controls* the *sparsity* of the estimated vector, and the number of outliers as a byproduct. Hence, effective methods to select this parameter are of paramount importance.

The link between ℓ_1-norm regularization and robustness was also exploited for parameter (but not function) estimation in [19] and [27]; see also [46] for related ideas in the context of face recognition, and error correction codes [5], [6]. In [19], however, the selection of Lasso's tuning parameter is only justified for Gaussian training data; whereas a fixed value motivated by CS error bounds is adopted in the Bayesian formulation of [27]. Here instead, a more general and systematic approach is pursued in Section III-B, building on contemporary algorithms that can efficiently compute all *robustifaction* paths of Lasso solutions (also known as homotopy paths) obtained for all values of the tuning parameter [13], [18], [20], [47]. In this sense, the method here capitalizes on but *is not limited to* sparse settings, since one can examine all possible sparsity levels along the robustification path. An estimator with reduced bias and improved generalization capability is obtained in Section IV, after replacing the ℓ_0-norm with a nonconvex surrogate, instead of the ℓ_1-norm that introduces bias [39], [50]. Simulated tests demonstrate the effectiveness of the novel approaches in robustifying thin-plate smoothing splines [12] (Section V-A), and in estimating the sinc function (Section V-B)—a paradigm typically adopted to assess performance of robust function approximation approaches [10], [49].

The motivating application behind the robust nonparametric methods of this paper is *load curve cleansing* [8]—a critical task in power systems engineering and management. Load curve data (also known as load profiles) refers to the electric energy consumption periodically recorded by meters at specific points across the power grid, e.g., end user-points and substations. Accurate load profiles are critical assets aiding operational decisions in the envisioned smart grid system [25]; see also [1], [2], [8]. However, in the process of acquiring and transmitting such massive volumes of information to a central processing unit, data is often noisy, corrupted, or lost altogether. This could be due to several reasons including meter misscalibration or outright failure, as well as communication errors due to noise, network congestion, and connectivity outages; see Fig. 1 for an example. In addition, data significantly deviating from nominal load models (outliers) are not uncommon, and could be attributed to unscheduled maintenance leading to shutdown of heavy industrial loads, weather constraints, holidays, strikes, and major sporting events, just to name a few.

In this context, it is critical to effectively reject outliers, and replace the contaminated data with "healthy" load predictions, i.e., to cleanse the load data. While most utilities carry out this task manually based on their own personnel's know-how, a first scalable and principled approach to load profile cleansing which is based on statistical learning methods was recently proposed in [8] and which also includes an extensive literature review on the related problem of outlier identification in time-series. After estimating the regression function f via either B-spline or Kernel smoothing, pointwise confidence intervals are constructed based on $\hat{f}$. A datum is deemed as an outlier whenever it falls outside its associated confidence interval. To control the degree of smoothing effected by the estimator, [8] requires the user to label the outliers present in a training subset of data, and in this sense the approach therein is not fully automatic. Here instead, a novel alternative to load curve cleansing is developed after specializing the robust estimators of Sections III and IV, to the case of cubic smoothing splines (Section V-C). The smoothness-and outlier sparsity-controlling parameters are selected according to the guidelines in Section III-B; hence, no input is required from the data analyst. The proposed spline-based method is tested on real load curve data from a government building.

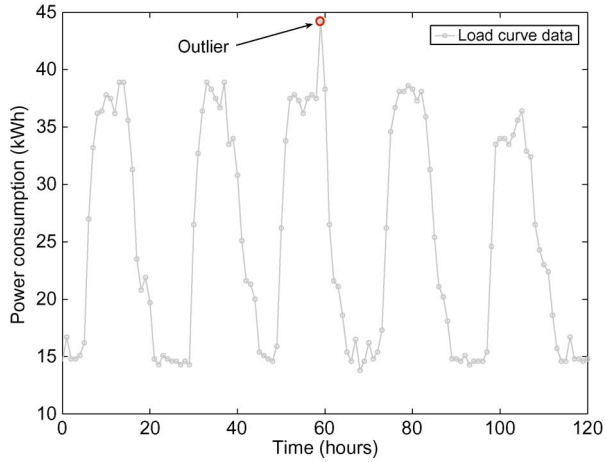

Fig. 1. Example of load curve data with outliers.

Concluding remarks are given in Section VI, while some technical details are deferred to the Appendix.

Notation: Bold uppercase letters will denote matrices, whereas bold lowercase letters will stand for column vectors. Operators $(\cdot)'$, $\mathrm{tr}(\cdot)$ and $E[\cdot]$ will denote transposition, matrix trace and expectation, respectively; $|\cdot|$ will be used for the cardinality of a set and the magnitude of a scalar. The ℓ_q norm of vector $\mathbf{x} \in \mathbb{R}^p$ is $\|\mathbf{x}\|_q := (\sum_{i=1}^p |x_i|^q)^{\frac{1}{q}}$ for $q \geq 1$; and $\|\mathbf{M}\|_F := \sqrt{\mathrm{tr}(\mathbf{M}\mathbf{M}')}$ is the matrix Frobenious norm. Positive definite matrices will be denoted by $\mathbf{M} \succ \mathbf{0}$. The $p \times p$ identity matrix will be represented by $\mathbf{I}_p$, while $\mathbf{0}_p$ will denote the $p \times 1$ vector of all zeros, and $\mathbf{0}_{p\times q} := \mathbf{0}_p\mathbf{0}_q'$.

II. Robust Estimation Problem

The training data comprises N *noisy* samples of f taken at the input points $\{\mathbf{x}_i\}_{i=1}^N$ (also known as knots in the splines parlance), and in the present context they can be possibly contaminated with outliers. Building on the parametric least-trimmed squares (LTS) approach [37], the desired robust estimate $\hat{f}$ can

be obtained as the solution of the following variational (V)LTS minimization problem:

$$\min_{f\in\mathcal{H}}\left[\sum_{i=1}^{s} r^2_{[i]}(f)+\mu\|f\|^2_{\mathcal{H}}\right] \tag{2}$$

where $r^2_{[i]}(f)$ is the ith-order statistic among the squared residuals $r^2_1(f),\ldots,r^2_N(f)$, and $r_i(f):=y_i-f(\mathbf{x}_i)$. In words, given a feasible $f\in\mathcal{H}$, to evaluate the sum of the cost in (2) one: i) computes all N squared residuals $\{r^2_i(f)\}_{i=1}^N$, ii) orders them to form the nondecreasing sequence $r^2_{[1]}(f)\leq\ldots\leq r^2_{[N]}(f)$; and iii) sums up the smallest s terms. As in the parametric LTS [37], the so-termed trimming constant s (also known as coverage) determines the breakdown point of the VLTS estimator, since the largest $N-s$ residuals do not participate in (2). Ideally, one would like to make $N-s$ equal to the (typically unknown) number of outliers N_o in the training data. For most pragmatic scenarios where N_o is unknown, the LTS estimator is an attractive option due to its high breakdown point and desirable theoretical properties, namely $\sqrt{N}$-consistency and asymptotic normality [37].

The tuning parameter $\mu\geq 0$ in (2) controls the tradeoff between fidelity to the (trimmed) data, and the degree of "smoothness" measured by $\|f\|^2_{\mathcal{H}}$. In particular, $\|f\|^2_{\mathcal{H}}$ can be interpreted as a generalized ridge regularization term penalizing more those functions with large coefficients in a basis expansion involving the eigenfunctions of the kernel K.

Given that the sum in (2) is a nonconvex functional, a nontrivial issue pertains to the existence of the proposed VLTS estimator, i.e., whether or not (2) attains a minimum in $\mathcal{H}$. Fortunately, a (conceptually) simple solution procedure suffices to show that a minimizer does indeed exist. Consider specifically a given subsample of s training data points, say $\{y_i,\mathbf{x}_i\}_{i=1}^s$, and solve

$$\min_{f\in\mathcal{H}}\left[\sum_{i=1}^{s} r^2_i(f)+\mu\|f\|^2_{\mathcal{H}}\right].$$

A unique minimizer of the form $\hat{f}^{(j)}(\mathbf{x})=\sum_{i=1}^s\beta_i^{(j)}K(\mathbf{x},\mathbf{x}_i)$ is guaranteed to exist, where j is used here to denote the chosen subsample, and the coefficients $\{\beta_i^{(j)}\}_{i=1}^s$ can be obtained by solving a particular linear system of equations [44, p. 11]. This procedure can be repeated for each subsample (there are $J:=\binom{N}{s}$ of these), to obtain a collection $\{\hat{f}^{(j)}(\mathbf{x})\}_{j=1}^J$ of candidate solutions of (2). The winner(s) $\hat{f}:=\hat{f}^{(j^*)}$ yielding the minimum cost, is the desired VLTS estimator.

Even though conceptually simple, the solution procedure just described guarantees existence of (at least) one solution, but entails a combinatorial search over all J subsamples which is intractable for moderate to large sample sizes N. In the context of linear regression, algorithms to obtain approximate LTS solutions are available; see e.g., [36].

A. Robust Function Approximation via ℓ_0-Norm Regularization

Instead of discarding large residuals, the alternative approach proposed here explicitly accounts for outliers in the regression model. To this end, consider the scalar variables $\{o_i\}_{i=1}^N$ one per training datum, taking the value $o_i=0$ whenever datum i adheres to the postulated nominal model, and $o_i\neq 0$ otherwise. A regression model naturally accounting for the presence of outliers is

$$y_i=f(\mathbf{x}_i)+o_i+\varepsilon_i,\quad i=1,\ldots,N \tag{3}$$

where $\{\varepsilon_i\}_{i=1}^N$ are zero-mean independent and identically distributed (i.i.d.) random variables modeling the observation errors. A similar model was advocated under different assumptions in [19] and [27], in the context of robust parametric regression; see also [5] and [46]. For an outlier-free datum i, (3) reduces to $y_i=f(\mathbf{x}_i)+\varepsilon_i$; hence, ε_i will be often referred to as the nominal noise. Note that in (3), both $f\in\mathcal{H}$ as well as the $N\times 1$ vector $\mathbf{o}:=[o_1,\ldots,o_N]'$ are unknown; thus, (3) is underdetermined. On the other hand, as outliers are expected to often comprise a small fraction of the training sample say, not exceeding 20%—vector $\mathbf{o}$ is typically *sparse*, i.e., most of its entries are zero; see also Remark 2. Sparsity compensates for underdeterminacy and provides valuable side-information when it comes to efficiently estimating $\mathbf{o}$, identifying outliers as a byproduct, and consequently performing *robust* estimation of the unknown function f.

A natural criterion for controlling outlier sparsity is to seek the desired estimate $\hat{f}$ as the solution of

$$\min_{\substack{f\in\mathcal{H}\\ \mathbf{o}\in\mathbb{R}^N}}\left[\sum_{i=1}^{N}\left(y_i-f(\mathbf{x}_i)-o_i\right)^2+\mu\|f\|^2_{\mathcal{H}}+\lambda_0\|\mathbf{o}\|_0\right] \tag{4}$$

where $\lambda_0\geq 0$ is a preselected sparsity controlling parameter, and $\|\mathbf{o}\|_0$ denotes the ℓ_0-norm of $\mathbf{o}$, which equals the number of nonzero entries of its vector argument. Unfortunately, analogously to related ℓ_0-norm regularized formulations in compressive sampling and sparse signal representations, problem (4) is NP-hard [34].

To further motivate model (3) and the proposed criterion (4) for robust nonparametric regression, it is worth checking the structure of the minimizers $\{\hat{f},\hat{\mathbf{o}}\}$ of the cost in (4). Consider for the sake of argument that λ_0 is given, and its value is such that $\|\hat{\mathbf{o}}\|_0=\nu$, for some $0\leq\nu\leq N$. The goal is to characterize $\hat{f}$, as well as the positions and values of the nonzero entries of $\hat{\mathbf{o}}$. Note that because $\|\hat{\mathbf{o}}\|_0=\nu$, the last term in (4) is constant, hence inconsequential to the minimization. Upon defining $\hat{r}_i:=y_i-\hat{f}(\mathbf{x}_i)$, it is not hard to see that the entries of $\hat{\mathbf{o}}$ satisfy

$$\hat{o}_i=\begin{cases}0, & |\hat{r}_i|\leq\sqrt{\lambda_0}\\ \hat{r}_i, & |\hat{r}_i|>\sqrt{\lambda_0}\end{cases},\quad i=1,\ldots,N \tag{5}$$

at the optimum. This is intuitive, since for those $\hat{o}_i\neq 0$ the best thing to do in terms of minimizing the overall cost is to set $\hat{o}_i=\hat{r}_i$, and thus null the corresponding squared-residual terms in (4). In conclusion, for the chosen value of λ_0 it holds that ν squared residuals effectively do not contribute to the cost in (4).

To determine the support of $\hat{\mathbf{o}}$ and $\hat{f}$, one alternative is to exhaustively test all $\binom{N}{\nu}$ admissible support combinations. For each one of these combinations (indexed by j), let $\mathcal{S}_j\subset\{1,\ldots,N\}$ be the index set describing the support of $\hat{\mathbf{o}}^{(j)}$, i.e., $\hat{o}_i^{(j)}\neq 0$ if and only if $i\in\mathcal{S}_j$; and $|\mathcal{S}_j|=\nu$. By virtue of (5), the corresponding candidate $\hat{f}^{(j)}$ minimizes

$$\min_{f\in\mathcal{H}}\left[\sum_{i\in\mathcal{S}_j} r^2_i(f)+\mu\|f\|^2_{\mathcal{H}}\right]$$

while $\hat{f}$ is the one among all $\{\hat{f}^{(j)}\}$ that yields the least cost. The previous discussion, in conjunction with the one preceding Section II-A completes the argument required to establish the following result.

Proposition 1: *If $\{\hat{f}, \hat{\mathbf{o}}\}$ minimizes (4) with λ_0 chosen such that $\|\hat{\mathbf{o}}\|_0 = N - s$, then $\hat{f}$ also solves the VLTS problem (2).*

The importance of Proposition 1 is threefold. First, it formally justifies model (3) and its estimator (4) for robust function approximation, in light of the well documented merits of LTS regression [36]. Second, it further solidifies the connection between sparse linear regression and robust estimation. Third, the ℓ_0-norm regularized formulation in (4) lends itself naturally to efficient solvers based on convex relaxation, the subject dealt with next.

III. Sparsity Controlling Outlier Rejection

To overcome the complexity hurdle in solving the robust regression problem in (4), one can resort to a suitable relaxation of the objective function. The goal is to formulate an optimization problem which is tractable, and whose solution yields a satisfactory approximation to the minimizer of the original hard problem. To this end, it is useful to recall that the ℓ_1-norm $\|\mathbf{x}\|_1$ of vector $\mathbf{x}$ is the closest convex approximation of $\|\mathbf{x}\|_0$. This property also utilized in the context of compressive sampling [41], provides the motivation to relax the NP-hard problem (4) to

$$\min_{\substack{f\in\mathcal{H}\\ \mathbf{o}\in\mathbb{R}^N}} \left[\sum_{i=1}^{N}\left(y_i - f(\mathbf{x}_i) - o_i\right)^2 + \mu\|f\|_{\mathcal{H}}^2 + \lambda_1\|\mathbf{o}\|_1\right]. \quad (6)$$

Being a convex optimization problem, (6) can be solved efficiently. The nondifferentiable ℓ_1-norm regularization term controls sparsity on the estimator of $\mathbf{o}$, a property that has been recently exploited in diverse problems in engineering, statistics and machine learning. A noteworthy representative is the least-absolute shrinkage and selection operator (Lasso) [39], a popular tool in statistics for joint estimation and continuous variable selection in linear regression problems. In its Lagrangian form, Lasso is also known as basis pursuit denoising in the signal processing literature, a term coined by [9] in the context of finding the best sparse signal expansion using an overcomplete basis.

It is pertinent to ponder on whether problem (6) has built-in ability to provide robust estimates $\hat{f}$ in the presence of outliers. The answer is in the affirmative, since a straightforward argument (details are deferred to the Appendix) shows that (6) is equivalent to a variational M-type estimator found by

$$\min_{f\in\mathcal{H}} \left[\sum_{i=1}^{N}\rho\left(y_i - f(\mathbf{x}_i)\right) + \mu\|f\|_{\mathcal{H}}^2\right] \quad (7)$$

where $\rho : \mathbb{R} \to \mathbb{R}$ is a scaled version of Huber's convex loss function [26]

$$\rho(u) := \begin{cases} u^2, & |u| \le \lambda_1/2 \\ \lambda_1|u| - \lambda_1^2/4, & |u| > \lambda_1/2. \end{cases} \quad (8)$$

Remark 1 (Regularized Regression and Robustness): Existing works on linear regression have pointed out the equivalence between ℓ_1-norm regularized regression and M-type estimators, under specific assumptions on the distribution of the outliers (ϵ-contamination) [19], [28]. However, they have not recognized the link with LTS through the convex relaxation of (4), and the connection asserted by Proposition 1. Here, the treatment goes beyond linear regression by considering nonparametric functional approximation in RKHS. Linear regression is subsumed as a special case, when the linear kernel $K(\mathbf{x}, \mathbf{y}) := \mathbf{x}'\mathbf{y}$ is adopted. In addition, no assumption is imposed on the outlier vector.

It is interesting to compare the ℓ_0- and ℓ_1-norm formulations [cf. (4) and (6), respectively] in terms of their equivalent purely variational counterparts in (2) and (7), that entail robust loss functions. While the VLTS estimator completely discards large residuals, ρ still retains them, but downweighs their effect through a linear penalty. Moreover, while (7) is convex, (2) is not and this has a direct impact on the complexity to obtain either estimator. Regarding the trimming constant s in (2), it controls the number of residuals retained and hence the breakdown point of VLTS. Considering instead the threshold $\frac{\lambda_1}{2}$ in Huber's function ρ, when the outliers' distribution is known *a priori*, its value is available in closed form so that the robust estimator is optimal in a well-defined sense [26]. Convergence in probability of M-type cubic smoothing splines estimators—a special problem subsumed by (7)—was studied in [11].

A. Solving the Convex Relaxation

Because (6) is jointly convex in f and $\mathbf{o}$, an alternating minimization (AM) algorithm can be adopted to solve (6), for fixed values of μ and λ_1. Selection of these parameters is a critical issue that will be discussed in Section III-B. AM solvers are iterative procedures that fix one of the variables to its most up to date value, and minimize the resulting cost with respect to the other one. Then the roles are reversed to complete one cycle, and the overall two-step minimization procedure is repeated for a prescribed number of iterations, or, until a convergence criterion is met. Letting $k = 0, 1, \ldots$ denote iterations, consider that $\mathbf{o} := \mathbf{o}^{(k-1)}$ is fixed in (6). The update for $f^{(k)}$ at the kth iteration is given by

$$f^{(k)} := \arg\min_{f\in\mathcal{H}} \left[\sum_{i=1}^{N}\left(\left(y_i - o_i^{(k-1)}\right) - f(\mathbf{x}_i)\right)^2 + \mu\|f\|_{\mathcal{H}}^2\right] \quad (9)$$

which corresponds to a standard regularization problem for functional approximation in $\mathcal{H}$ [14], but with *outlier-compensated* data $\{y_i - o_i^{(k-1)}, \mathbf{x}_i\}_{i=1}^N$. It is well known that the minimizer of the variational problem (9) is finitely parameterized, and given by the kernel expansion $f^{(k)}(\mathbf{x}) = \sum_{i=1}^N \beta_i^{(k)} K(\mathbf{x}, \mathbf{x}_i)$ [44]. The vector $\boldsymbol{\beta} := [\beta_1, \ldots, \beta_N]'$ is found by solving the linear system of equations

$$[\mathbf{K} + \mu\mathbf{I}_N]\boldsymbol{\beta}^{(k)} = \mathbf{y} - \mathbf{o}^{(k-1)} \quad (10)$$

where $\mathbf{y} := [y_1, \ldots, y_N]'$, and the $N \times N$ matrix $\mathbf{K} \succ \mathbf{0}$ has entries $[\mathbf{K}]_{ij} := K(\mathbf{x}_i, \mathbf{x}_j)$.

In a nutshell, updating $f^{(k)}$ is equivalent to updating vector $\boldsymbol{\beta}^{(k)}$ as per (10), where only the independent vector variable $\mathbf{y} - \mathbf{o}^{(k-1)}$ changes across iterations. Because the system matrix is positive definite, the per iteration systems of linear equations (10) can be efficiently solved after computing once, the Cholesky factorization of $\mathbf{K} + \mu\mathbf{I}_N$.

For fixed $f := f^{(k)}$ in (6), the outlier vector update $\mathbf{o}^{(k)}$ at iteration k is obtained as

$$\mathbf{o}^{(k)} := \arg\min_{\mathbf{o}\in\mathbb{R}^N} \left[\sum_{i=1}^{N}\left(r_i^{(k)} - o_i\right)^2 + \lambda_1\|\mathbf{o}\|_1\right] \tag{11}$$

Algorithm 1: AM Solver

Initialize $\mathbf{o}^{(-1)} = \mathbf{0}$, and run till convergence

for $k = 0, 1, \ldots$ **do**

Update $\boldsymbol{\beta}^{(k)}$ solving $[\mathbf{K} + \mu\mathbf{I}_N]\boldsymbol{\beta}^{(k)} = \mathbf{y} - \mathbf{o}^{(k-1)}$.

Update $\mathbf{o}^{(k)}$ via $o_i^{(k)} = \mathcal{S}\left(y_i - \sum_{j=1}^{N}\beta_j^{(k)}K(\mathbf{x}_i,\mathbf{x}_j), \frac{\lambda_1}{2}\right), i = 1, \ldots, N.$

end for

return $f(\mathbf{x}) = \sum_{i=1}^{N}\beta_i^{(\infty)}K(\mathbf{x},\mathbf{x}_i)$

where $r_i^{(k)} := y_i - \sum_{j=1}^{N}\beta_j^{(k)}K(\mathbf{x}_i,\mathbf{x}_j)$. Problem (11) can be recognized as an instance of Lasso for the so-termed orthonormal case, in particular for an identity regression matrix. The solution of such Lasso problems is readily obtained via soft-thresholding [17], in the form of

$$o_i^{(k)} := \mathcal{S}\left(r_i^{(k)}, \lambda_1/2\right), \quad i = 1, \ldots, N \tag{12}$$

where $\mathcal{S}(z,\gamma) := \text{sign}(z)(|z| - \gamma)_+$ is the soft-thresholding operator, and $(\cdot)_+ := \max(0,\cdot)$ denotes the projection onto the nonnegative reals. The coordinatewise updates in (12) are in par with the sparsifying property of the ℓ_1 norm, since for "small" residuals, i.e., $r_i^{(k)} \leq \frac{\lambda_1}{2}$, it follows that $o_i^{(k)} = 0$, and the ith training datum is deemed outlier free. Updates (10) and (12) comprise the iterative AM solver of the ℓ_1-norm regularized problem (6), which is tabulated as Algorithm 1. Convexity ensures convergence to the global optimum solution regardless of the initial condition; see e.g., [4].

Algorithm 1 is also conceptually interesting, since it explicitly reveals the intertwining between the outlier identification process, and the estimation of the regression function with the appropriate outlier-compensated data. An additional point is worth mentioning after inspection of (12) in the limit as $k \to \infty$. From the definition of the soft-thresholding operator $\mathcal{S}$, for those "large" residuals $\hat{r}_i := \lim_{k\to\infty} r_i^{(k)}$ exceeding $\frac{\lambda_1}{2}$ in magnitude, $\hat{o}_i = \hat{r}_i - \frac{\lambda_1}{2}$ when $\hat{r}_i > 0$, and $\hat{o}_i = \hat{r}_i + \frac{\lambda_1}{2}$ otherwise. In other words, larger residuals that the method identifies as corresponding to outlier-contaminated data are shrunk, but not completely discarded. By plugging $\hat{\mathbf{o}}$ back into (6), these "large" residuals cancel out in the squared error term, but still contribute linearly through the ℓ_1-norm regularizer. This is exactly what one would expect, in light of the equivalence established with the variational M-type estimator in (7).

Next, it is established that an alternative to solving a sequence of linear systems and scalar Lasso problems, is to solve a single instance of the Lasso with specific response vector and (nonorthonormal) regression matrix.

Proposition 2: *Consider* $\hat{\mathbf{o}}_{\text{Lasso}}$ *defined as*

$$\hat{\mathbf{o}}_{\text{Lasso}} := \arg\min_{\mathbf{o}\in\mathbb{R}^N} \|\mathbf{X}_\mu\mathbf{y} - \mathbf{X}_\mu\mathbf{o}\|_2^2 + \lambda_1\|\mathbf{o}\|_1 \tag{13}$$

where

$$\mathbf{X}_\mu := \begin{bmatrix}\mathbf{I}_N - \mathbf{K}(\mathbf{K}+\mu\mathbf{I}_N)^{-1} \\ (\mu\mathbf{K})^{1/2}(\mathbf{K}+\mu\mathbf{I}_N)^{-1}\end{bmatrix}. \tag{14}$$

Then, the minimizers $\{\hat{f}, \hat{\mathbf{o}}\}$ *of (6) are fully determined given* $\hat{\mathbf{o}}_{\text{Lasso}}$*, as* $\hat{\mathbf{o}} := \hat{\mathbf{o}}_{\text{Lasso}}$ *and* $\hat{f}(\mathbf{x}) = \sum_{i=1}^{N}\hat{\beta}_i K(\mathbf{x},\mathbf{x}_i)$*, with* $\hat{\boldsymbol{\beta}} = (\mathbf{K}+\mu\mathbf{I}_N)^{-1}(\mathbf{y} - \hat{\mathbf{o}}_{\text{Lasso}})$.

Proof: For notational convenience introduce the $N\times 1$ vectors $\mathbf{f} := [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]'$ and $\hat{\mathbf{f}} := [\hat{f}(\mathbf{x}_1), \ldots, \hat{f}(\mathbf{x}_N)]'$, where $\hat{f} \in \mathcal{H}$ is the minimizer of (6). Next, consider rewriting (6) as

$$\min_{\mathbf{o}\in\mathbb{R}^N}\left[\min_{f\in\mathcal{H}}\|(\mathbf{y}-\mathbf{o}) - \mathbf{f}\|_2^2 + \mu\|f\|_{\mathcal{H}}^2\right] + \lambda_1\|\mathbf{o}\|_1. \tag{15}$$

The quantity inside the square brackets is a function of $\mathbf{o}$, and can be written explicitly after carrying out the minimization with respect to $f \in \mathcal{H}$. From the results in [44], it follows that the vector of optimum predicted values at the points $\{\mathbf{x}_i\}_{i=1}^N$ is given by $\hat{\mathbf{f}} = \mathbf{K}\hat{\boldsymbol{\beta}} = \mathbf{K}(\mathbf{K}+\mu\mathbf{I}_N)^{-1}(\mathbf{y}-\mathbf{o})$; see also the discussion after (9). Similarly, one finds that $\|\hat{f}\|_{\mathcal{H}}^2 = \hat{\boldsymbol{\beta}}'\mathbf{K}\hat{\boldsymbol{\beta}} = (\mathbf{y}-\mathbf{o})'(\mathbf{K}+\mu\mathbf{I}_N)^{-1}\mathbf{K}(\mathbf{K}+\mu\mathbf{I}_N)^{-1}(\mathbf{y}-\mathbf{o})$. Having minimized (15) with respect to f, the quantity inside the square brackets is ($\boldsymbol{\Gamma}_\mu := (\mathbf{K}+\mu\mathbf{I}_N)^{-1}$)

$$\begin{aligned}
&\min_{f\in\mathcal{H}}\left[\|(\mathbf{y}-\mathbf{o}) - \mathbf{f}\|_2^2 + \mu\|f\|_{\mathcal{H}}^2\right] \\
&= \left\|(\mathbf{y}-\mathbf{o}) - \hat{\mathbf{f}}\right\|_2^2 + \mu\|\hat{f}\|_{\mathcal{H}}^2 \\
&= \|(\mathbf{y}-\mathbf{o}) - \mathbf{K}\boldsymbol{\Gamma}_\mu(\mathbf{y}-\mathbf{o})\|_2^2 + \mu(\mathbf{y}-\mathbf{o})'\boldsymbol{\Gamma}_\mu\mathbf{K}\boldsymbol{\Gamma}_\mu(\mathbf{y}-\mathbf{o}) \\
&= \|(\mathbf{I}_N - \mathbf{K}\boldsymbol{\Gamma}_\mu)\mathbf{y} - (\mathbf{I}_N - \mathbf{K}\boldsymbol{\Gamma}_\mu)\mathbf{o}\|_2^2 \\
&\quad + \mu(\mathbf{y}-\mathbf{o})'\boldsymbol{\Gamma}_\mu\mathbf{K}\boldsymbol{\Gamma}_\mu(\mathbf{y}-\mathbf{o}).
\end{aligned} \tag{16}$$

After expanding the quadratic form in the right-hand side of (16), and eliminating the term that does not depend on $\mathbf{o}$, problem (15) becomes

$$\min_{\mathbf{o}\in\mathbb{R}^N}\Big[\|(\mathbf{I}_N - \mathbf{K}\boldsymbol{\Gamma}_\mu)\mathbf{y} - (\mathbf{I}_N - \mathbf{K}\boldsymbol{\Gamma}_\mu)\mathbf{o}\|_2^2 - 2\mu\mathbf{y}'\boldsymbol{\Gamma}_\mu\mathbf{K}\boldsymbol{\Gamma}_\mu\mathbf{o} + \mu\mathbf{o}'\boldsymbol{\Gamma}_\mu\mathbf{K}\boldsymbol{\Gamma}_\mu\mathbf{o} + \lambda_1\|\mathbf{o}\|_1\Big].$$

Completing the squares one arrives at

$$\min_{\mathbf{o}\in\mathbb{R}^N}\left[\left\|\begin{bmatrix}\mathbf{I}_N - \mathbf{K}\boldsymbol{\Gamma}_\mu \\ (\mu\mathbf{K})^{1/2}\boldsymbol{\Gamma}_\mu\end{bmatrix}\mathbf{y} - \begin{bmatrix}\mathbf{I}_N - \mathbf{K}\boldsymbol{\Gamma}_\mu \\ (\mu\mathbf{K})^{1/2}\boldsymbol{\Gamma}_\mu\end{bmatrix}\mathbf{o}\right\|_2^2 + \lambda_1\|\mathbf{o}\|_1\right]$$

which completes the proof. ■

The result in Proposition 2 opens the possibility for effective methods to select λ_1. These methods to be described in detail in the ensuing section, capitalize on recent algorithmic advances on Lasso solvers, which allow one to efficiently compute $\hat{\mathbf{o}}_{\text{Lasso}}$ for all values of the tuning parameter λ_1. This is crucial for obtaining satisfactory robust estimates $\hat{f}$, since *controlling the sparsity* in $\mathbf{o}$ by tuning λ_1 is tantamount to controlling the number of outliers in model (3).

B. Selection of the Tuning Parameters: Robustification Paths

As argued before, the tuning parameters μ and λ_1 in (6) control the degree of smoothness in $\hat{f}$ and the number of

outliers (nonzero entries in $\hat{\mathbf{o}}_{\text{Lasso}}$), respectively. From a statistical learning theory standpoint, μ and λ_1 control the amount of regularization and model complexity, thus capturing the so-termed effective degrees of freedom [24]. Complex models tend to have worse generalization capability, even though the prediction error over the training set $\mathcal{T}$ may be small (overfitting). In the contexts of regularization networks [14] and Lasso estimation for regression [39], corresponding tuning parameters are typically selected via model selection techniques such as cross-validation, or, by minimizing the prediction error over an independent test set, if available [24]. However, these simple methods are severely challenged in the presence of multiple outliers. For example, the *swamping* effect refers to a very large value of the residual r_i corresponding to a left out clean datum $\{y_i, \mathbf{x}_i\}$, because of an unsatisfactory model estimation based on all data except i; data which contain outliers.

The idea here offers an alternative method to overcome the aforementioned challenges, and the possibility to efficiently compute $\hat{\mathbf{o}}_{\text{Lasso}}$ for all values of λ_1, given μ. A brief overview of the state-of-the-art in Lasso solvers is given first. Several methods for selecting μ and λ_1 are then described, which differ on the assumptions of what is known regarding the outlier model (3).

Lasso amounts to solving a quadratic programming (QP) problem [39]; hence, an iterative procedure is required to determine $\hat{\mathbf{o}}_{\text{Lasso}}$ in (13) for a given value of λ_1. While standard QP solvers can be certainly invoked to this end, an increasing amount of effort has been put recently toward developing fast algorithms that capitalize on the unique properties of Lasso. The Lasso variation of the LARS algorithm [13, Sec. 3.1] is an efficient scheme for computing the entire path of solutions (corresponding to all values of λ_1), elsewhere referred to as homotopy paths [13], [20], or, regularization paths [17]. LARS capitalizes on piecewise linearity of the Lasso path of solutions, while incurring the complexity of a single LS fit, i.e., when $\lambda_1 = 0$. Homotopy algorithms have been also developed to solve the Lasso online, when data pairs $\{y_i, \mathbf{x}_i\}$ are collected sequentially in time [3], [20]. Coordinate descent algorithms have been shown competitive, even outperforming LARS when p is large, as demonstrated in [18]; see also [17], [48], and the references therein. Coordinate descent solvers capitalize on the fact that Lasso can afford a very simple solution in the scalar case, which is given in closed form in terms of a soft-thresholding operation [cf. (12)]. Further computational savings are attained through the use of *warm starts* [17], when computing the Lasso path of solutions over a grid of decreasing values of λ_1. An efficient solver capitalizing on variable separability has been proposed in [47], while a semismooth Newton method was put forth in [22].

Consider then a grid of G_μ values of μ in the interval $[\mu_{\min}, \mu_{\max}]$, evenly spaced in a logarithmic scale. Likewise, for each μ consider a similar type of grid consisting of G_λ values of λ_1, where $\lambda_{\max} := 2\min_i |\mathbf{y}'\mathbf{X}'_\mu \mathbf{x}_{\mu,i}|$ is the minimum λ_1 value such that $\hat{\mathbf{o}}_{\text{Lasso}} \neq \mathbf{0}_N$ [18], and $\mathbf{X}_\mu := [\mathbf{x}_{\mu,1} \ldots \mathbf{x}_{\mu,N}]$ in (13). Typically, $\lambda_{\min} = \epsilon\lambda_{\max}$ with $\epsilon = 10^{-4}$, say. Note that each of the G_μ values of μ gives rise to a different λ grid, since $\lambda_{\max}$ depends on μ through $\mathbf{X}_\mu$. Given the previously surveyed algorithmic alternatives to tackle the Lasso, it is safe to assume that (13) can be efficiently solved over the (nonuniform) $G_\mu \times G_\lambda$ grid of values of the tuning parameters. This way, for each value of μ one obtains G_λ samples of the Lasso homotopy paths, henceforth referred to as *robustification paths* as a means of highlighting the connection between robustness and sparsity in the nonparametric context of the present work. As λ_1 decreases, more variables $\hat{o}_{\text{Lasso},i}$ enter the model signifying that more of the training data are deemed to contain outliers. An example of the robustification path is given in Fig. 3.

Based on the robustification paths and the prior knowledge available on the outlier model (3), several alternatives are given next to select the "best" pair $\{\mu, \lambda_1\}$ in the grid $G_\mu \times G_\lambda$.

Number of outliers is known: For each value of μ in the grid G_μ, by direct inspection of the robustification paths one can determine the range of values for λ_1, such that $\hat{\mathbf{o}}_{\text{Lasso}}$ has exactly N_o nonzero entries. This procedure yields a reduced grid $G_\mu \times \tilde{G}_\lambda$ of candidate tuning parameter pairs, which is again nonuniform since the obtained λ_1-intervals may differ per μ. Focusing on the reduced grid, and after discarding outliers which are now fixed and known, K-fold cross-validation can be applied to determine $\{\mu^*, \lambda_1^*\}$; see e.g., [24, Ch. 7].

Variance of the nominal noise is known: Supposing that the variance σ_ε^2 of the i.i.d. nominal noise variables ε_i in (3) is known, one can proceed as follows. Using the solution $\hat{f}$ obtained for each pair $\{\mu_i, \lambda_j\}$ on the grid, form the $G_\mu \times G_\lambda$ sample variance matrix $\bar{\mathbf{\Sigma}}$ with ijth entry

$$[\bar{\mathbf{\Sigma}}]_{ij} := \sum_{u|\hat{o}_{\text{Lasso},u}=0} \hat{r}_u^2/\hat{N}_o = \sum_{u|\hat{o}_{\text{Lasso},u}=0} \left(y_u - \hat{f}(\mathbf{x}_u)\right)^2/\hat{N}_o \tag{17}$$

where $\hat{N}_o$ stands for the number of nonzero entries in $\hat{\mathbf{o}}_{\text{Lasso}}$. Although not made explicit, the right-hand side of (17) depends on $\{\mu_i, \lambda_j\}$ through the estimate $\hat{f}$, $\hat{\mathbf{o}}_{\text{Lasso}}$ and $\hat{N}_o$. The entries $[\bar{\mathbf{\Sigma}}]_{ij}$ correspond to a sample estimate of σ_ε^2, without considering those training data $\{y_i, \mathbf{x}_i\}$ that the method determined to be contaminated with outliers, i.e., those indices i for which $\hat{o}_{\text{Lasso},i} \neq 0$. The "winner" tuning parameters $\{\mu^*, \lambda_1^*\} := \{\mu_{i^*}, \lambda_{j^*}\}$ are such that

$$[i^*, j^*] := \arg\min_{i,j} \left|[\bar{\mathbf{\Sigma}}]_{ij} - \sigma_\varepsilon^2\right| \tag{18}$$

which is an absolute variance deviation (AVD) criterion.

Variance of the nominal noise is unknown: If σ_ε^2 is unknown, one can still compute a robust estimate of the variance $\hat{\sigma}_\varepsilon^2$, and repeat the previous procedure (with known nominal noise variance) after replacing σ_ε^2 with $\hat{\sigma}_\varepsilon^2$ in (18). One option is based on the median absolute deviation (MAD) estimator, namely

$$\hat{\sigma}_\varepsilon := 1.4826 \times \text{median}_i \left(|\hat{r}_i - \text{median}_j(\hat{r}_j)|\right) \tag{19}$$

where the residuals $\hat{r}_i = y_i - \hat{f}(\mathbf{x}_i)$ are formed based on a nonrobust estimate of f, obtained e.g., after solving (6) with $\lambda_1 = 0$ and using a small subset of the training dataset $\mathcal{T}$. The factor 1.4826 provides an approximately unbiased estimate of the standard deviation when the nominal noise is Gaussian. Typically, $\hat{\sigma}_\varepsilon$ in (19) is used as an estimate for the scale of the errors in general M-type robust estimators; see, e.g., [11] and [32].

Remark 2 (How Sparse is Sparse): Even though the very nature of outliers dictates that N_o is typically a small fraction of N—and thus $\mathbf{o}$ in (3) is sparse—the method here capitalizes on, but *is not limited* to sparse settings. For instance, choosing $\lambda_1 \in [\lambda_{\min} \approx 0, \lambda_{\max}]$ along the robustification paths allows

one to continuously control the sparsity level, and potentially select the right value of λ_1 for any given $N_o \in \{1,\ldots,N\}$. Admittedly, if N_o is large relative to N, then even if it is possible to identify and discard the outliers, the estimate $\hat{f}$ may not be accurate due to the lack of outlier-free data. Interestingly, simulation results in [21] demonstrate that the performance of this paper's sparsity-controlling outlier rejection methods degrade gracefully, as $N_o \to N$.

IV. Refinement via Nonconvex Regularization

Instead of substituting $\|\mathbf{o}\|_0$ in (4) by its closest convex approximation, namely $\|\mathbf{o}\|_1$, letting the surrogate function to be nonconvex can yield tighter approximations. For example, the ℓ_0-norm of a vector $\mathbf{x} \in \mathbb{R}^n$ was surrogated in [7] by the logarithm of the geometric mean of its elements, or by $\sum_{i=1}^n \log|x_i|$. In rank minimization problems, apart from the nuclear norm relaxation, minimizing the logarithm of the determinant of the unknown matrix has been proposed as an alternative surrogate [16]. Adopting related ideas in the present nonparametric context, consider approximating (4) by

$$\min_{\substack{f\in\mathcal{H}\\ \mathbf{o}\in\mathbb{R}^N}} \left[\sum_{i=1}^N (y_i - f(\mathbf{x}_i) - o_i)^2 + \mu\|f\|_{\mathcal{H}}^2 + \lambda_0 \sum_{i=1}^N \log(|o_i|+\delta)\right] \tag{20}$$

where δ is a sufficiently small positive offset introduced to avoid numerical instability.

Since the surrogate term in (20) is concave, the overall problem is nonconvex. Still, local methods based on iterative linearization of $\log(|o_i| + \delta)$, around the current iterate $o_i^{(k)}$, can be adopted to minimize (20). From the concavity of the logarithm, its local linear approximation serves as a global overestimator. Standard majorization-minimization algorithms motivate minimizing the global linear overestimator instead. This leads to the following iteration for $k = 0, 1, \ldots$ (see, e.g., [30] for further details)

$$\left[f^{(k)}, \mathbf{o}^{(k)}\right] := \arg\min_{\substack{f\in\mathcal{H}\\ \mathbf{o}\in\mathbb{R}^N}} \left[\sum_{i=1}^N (y_i - f(\mathbf{x}_i) - o_i)^2 + \mu\|f\|_{\mathcal{H}}^2 + \lambda_0 \sum_{i=1}^N w_i^{(k)}|o_i|\right] \tag{21}$$

$$w_i^{(k)} := \left(\left|o_i^{(k-1)}\right| + \delta\right)^{-1}, \quad i = 1,\ldots,N. \tag{22}$$

It is possible to eliminate the optimization variable $f \in \mathcal{H}$ from (21), by direct application of the result in Proposition 2. The equivalent update for $\mathbf{o}$ at iteration k is then given by

$$\mathbf{o}^{(k)} := \arg\min_{\mathbf{o}\in\mathbb{R}^N} \left[\|\mathbf{X}_\mu\mathbf{y} - \mathbf{X}_\mu\mathbf{o}\|_2^2 + \lambda_0 \sum_{i=1}^N w_i^{(k)}|o_i|\right] \tag{23}$$

which amounts to an iteratively reweighted version of (13). If the value of $|o_i^{(k-1)}|$ is small, then in the next iteration the corresponding regularization term $\lambda_0 w_i^{(k)}|o_i|$ has a large weight, thus promoting shrinkage of that coordinate to zero. On the other hand when $|o_i^{(k-1)}|$ is significant, the cost in the next iteration downweighs the regularization, and places more importance to the LS component of the fit. For small δ, analysis of the limiting point $\mathbf{o}^*$ of (23) reveals that

$$\lambda_0 w_i^* |o_i^*| \approx \begin{cases} \lambda_0, & |o_i^*| \neq 0 \\ 0, & |o_i^*| = 0 \end{cases}$$

and hence, $\lambda_0 \sum_{i=1}^N w_i^*|o_i^*| \approx \lambda_0\|\mathbf{o}^*\|_0$.

A good initialization for the iteration in (23) and (22) is $\hat{\mathbf{o}}_{\text{Lasso}}$, which corresponds to the solution of (13) [and (6)] for $\lambda_0 = \lambda_1^*$ and $\mu = \mu^*$. This is equivalent to a single iteration of (23) with all weights equal to unity. The numerical tests in Section V will indicate that even a single iteration of (23) suffices to obtain improved estimates $\hat{f}$, in comparison to those obtained from (13). The following remark sheds further light towards understanding why this should be expected.

Remark 3 (Refinement Through Bias Reduction): Uniformly weighted ℓ_1-norm regularized estimators such as (6) are biased [50], due to the shrinkage effected on the estimated coefficients. It will be argued next that the improvements due to (23) can be leveraged to bias reduction. Several workarounds have been proposed to correct the bias in sparse regression, that could as well be applied here. A first possibility is to retain only the support of (13) and re-estimate the amplitudes via, e.g., the unbiased LS estimator [13]. An alternative approach to reducing bias is through nonconvex regularization using e.g., the smoothly clipped absolute deviation (SCAD) scheme [15]. The SCAD penalty could replace the sum of logarithms in (20), still leading to a nonconvex problem. To retain the efficiency of convex optimization solvers while simultaneously limiting the bias, suitably *weighted* ℓ_1-norm regularizers have been proposed instead [50]. The constant weights in [50] play a role similar to those in (22); hence, bias reduction is expected.

V. Numerical Experiments

A. Robust Thin-Plate Smoothing Splines

To validate the proposed approach to robust nonparametric regression, a simulated test is carried out here in the context of thin-plate smoothing spline approximation [12], [45]. Specializing (6) to this setup, the robust thin-plate splines estimator can be formulated as

$$\min_{\substack{f\in\mathcal{S}\\ \mathbf{o}\in\mathbb{R}^N}} \left[\sum_{i=1}^N (y_i - f(\mathbf{x}_i) - o_i)^2 + \mu\int_{\mathbb{R}^2} \|\nabla^2 f\|_F^2 d\mathbf{x} + \lambda_1\|\mathbf{o}\|_1\right] \tag{24}$$

where $\|\nabla^2 f\|_F$ denotes the Frobenius norm of the Hessian of $f : \mathbb{R}^2 \to \mathbb{R}$. The penalty functional

$$J[f] := \int_{\mathbb{R}^2} \|\nabla^2 f\|_F^2 d\mathbf{x} = \int_{\mathbb{R}^2} \left[\left(\frac{\partial^2 f}{\partial x_1^2}\right)^2 + 2\left(\frac{\partial^2 f}{\partial x_1 \partial x_2}\right)^2 + \left(\frac{\partial^2 f}{\partial x_2^2}\right)^2\right] d\mathbf{x} \tag{25}$$

extends to $\mathbb{R}^2$ the one-dimensional roughness regularization used in smoothing spline models. For $\mu = 0$, the (nonunique) estimate in (24) corresponds to a *rough* function interpolating the outlier compensated data; while as $\mu \to \infty$ the estimate is linear (cf. $\nabla^2\hat{f}(\mathbf{x}) \equiv \mathbf{0}_{2\times 2}$). The optimization is over $\mathcal{S}$,

the space of Sobolev functions, for which $J[f]$ is well defined [12, p. 85]. Reproducing kernel Hilbert spaces such as $\mathcal{S}$, with inner-products (and norms) involving derivatives are studied in detail in [44].

Different from the cases considered so far, the smoothing penalty in (25) is only a seminorm, since first-order polynomials vanish under $J[\cdot]$. Omitting details than can be found in [44, p. 30], a unique minimizer of (24) exists provided the input vectors $\{\mathbf{x}_i \in \mathbb{R}^2\}_{i=1}^N$ do not fall on a straight line. The solution admits the finitely parametrized form $\hat{f}(\mathbf{x}) = \sum_{i=1}^N \beta_i K(\mathbf{x}, \mathbf{x}_i) + \boldsymbol{\alpha}_1' \mathbf{x} + \alpha_0$, where in this case $K(\mathbf{x}, \mathbf{y}) := \|\mathbf{x} - \mathbf{y}\|^2 \log \|\mathbf{x} - \mathbf{y}\|$ is a radial basis function. In simple terms, the solution as a kernel expansion is augmented with a member of the null space of $J[\cdot]$. The unknown parameters $\{\boldsymbol{\beta}, \boldsymbol{\alpha}_1, \alpha_0\}$ are obtained in closed form, as solutions to a constrained, regularized LS problem; see [44, p. 33]. As a result, Proposition 2 still holds with minor modifications on the structure of $\mathbf{X}_\mu$.

Remark 4 (Bayesian Framework): Adopting a Bayesian perspective, one could model $f(\mathbf{x})$ in (3) as a sample function of a zero mean Gaussian stationary process, with covariance function $K(\mathbf{x}, \mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|^2 \log \|\mathbf{x}-\mathbf{y}\|$ [29]. Consider as well that $\{f(\mathbf{x}), \{o_i, \varepsilon_i\}_{i=1}^N\}$ are mutually independent, while $\varepsilon_i \sim \mathcal{N}\left(0, \frac{\mu^*}{2}\right)$ and $o_i \sim \mathcal{L}\left(0, \frac{\mu^*}{\lambda_1^*}\right)$ in (3) are i.i.d. Gaussian and Laplace distributed, respectively. From the results in [29] and a straightforward calculation, it follows that setting $\lambda_1 = \lambda_1^*$ and $\mu = \mu^*$ in (24) yields estimates $\hat{f}$ (and $\hat{\mathbf{o}}$) which are optimal in a maximum a posteriori sense. This provides yet another means of selecting the parameters μ and λ_1, further expanding the options presented in Section III-B.

The simulation setup is as follows. Noisy samples of the true function $f_o : \mathbb{R}^2 \to \mathbb{R}$ comprise the training set $\mathcal{T}$. Function f_o is generated as a Gaussian mixture with two components, with respective mean vectors and covariance matrices given by

$$\boldsymbol{\mu}_1 = \begin{bmatrix} 0.2295 \\ 0.4996 \end{bmatrix}, \quad \boldsymbol{\Sigma}_1 = \begin{bmatrix} 2.2431 & 0.4577 \\ 0.4577 & 1.0037 \end{bmatrix},$$
$$\boldsymbol{\mu}_2 = \begin{bmatrix} 2.4566 \\ 2.9461 \end{bmatrix}, \quad \boldsymbol{\Sigma}_2 = \begin{bmatrix} 2.9069 & 0.5236 \\ 0.5236 & 1.7299 \end{bmatrix}.$$

Function $f_o(\mathbf{x})$ is depicted in Fig. 4(a). The training data set comprises $N = 200$ examples, with inputs $\{\mathbf{x}_i\}_{i=1}^N$ drawn from a uniform distribution in the square $[0, 3] \times [0, 3]$. Several values ranging from 5% to 25% of the data are generated contaminated with outliers. Without loss of generality, the corrupted data correspond to the first N_o training samples with $N_o = \{10, 20, 30, 40, 50\}$, for which the response values $\{y_i\}_{i=1}^{N_o}$ are independently drawn from a uniform distribution over $[-3, 3]$. Outlier-free data are generated according to the model $y_i = f_o(\mathbf{x}_i) + \varepsilon_i$, where the independent additive noise terms $\varepsilon_i \sim \mathcal{N}(0, 10^{-1})$ are Gaussian distributed, for $i = N_o + 1, \ldots, 200$. For the case where $N_o = 20$, the data used in the experiment is shown in Fig. 2. Superimposed to the true function f_o are 180 black points corresponding to data drawn from the nominal model, as well as 20 red outlier points.

For this experiment, the nominal noise variance $\sigma_\varepsilon^2 = 10^{-1}$ is assumed known. A nonuniform grid of μ and λ_1 values is constructed, as described in Section III-B. The relevant parameters are $G_\mu = G_\lambda = 200$, $\mu_{\min} = 10^{-9}$, and $\mu_{\max} = 1$. For each value of μ, the λ_1 grid spans the interval defined by $\lambda_{\max} := 2 \min_i |\mathbf{y}' \mathbf{X}_\mu' \mathbf{x}_{\mu,i}|$ and $\lambda_{\min} = \epsilon \lambda_{\max}$, where $\epsilon = 10^{-4}$. Each of the G_μ robustification paths corresponding to the solution of (13) is obtained using the SpaRSA toolbox in [47], exploiting warm starts for faster convergence. Fig. 3 depicts an example with $N_o = 20$ and $\mu^* = 3.53 \times 10^{-1}$. With the robustification paths at hand, it is possible to form the sample variance matrix $\bar{\boldsymbol{\Sigma}}$ [cf. (17)], and select the optimum tuning parameters $\{\mu^*, \lambda_1^*\}$ based on the criterion (18). Finally, the robust estimates are refined by running a single iteration of (23) as described in Section IV. The value $\delta = 10^{-5}$ was utilized, and several experiments indicated that the results are quite insensitive to the selection of this parameter.

The same experiment was conducted for a variable number of outliers N_o, and the results are listed in Table I. In all cases, a 100% outlier identification success rate was obtained, for the chosen value of the tuning parameters. This even happened at the first stage of the method, i.e., $\hat{\mathbf{o}}_{\text{Lasso}}$ in (13) had the correct

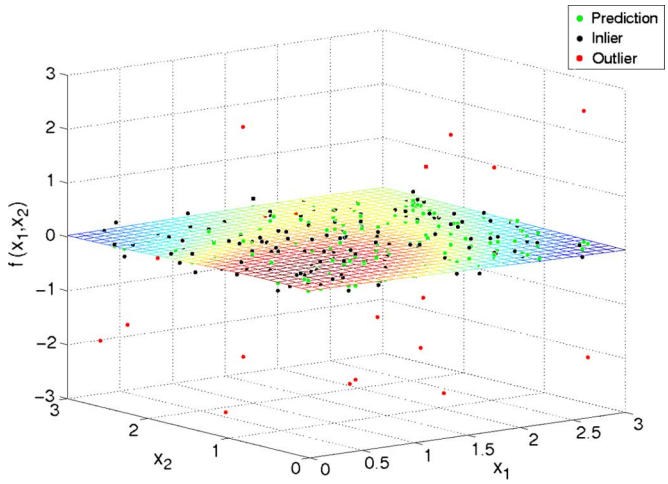

Fig. 2. True Gaussian mixture function $f_o(\mathbf{x})$, and its 180 noisy samples taken over $[0, 3] \times [0, 3]$ shown as black dots. The red dots indicate the $N_o = 20$ outliers in the training data set $\mathcal{T}$. The green points indicate the predicted responses $\hat{y}_i$ at the sampling points $\mathbf{x}_i$, from the estimate $\hat{f}$ obtained after solving (24). Note how all green points are close to the surface f_o.

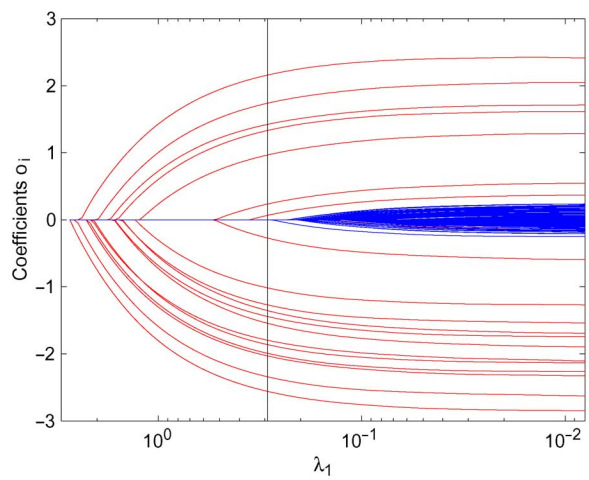

Fig. 3. Robustification path with optimum smoothing parameter $\mu^* = 3.53 \times 10^{-1}$. The data is corrupted with $N_o = 20$ outliers. The coefficients $\hat{o}_i$ corresponding to the outliers are shown in red, while the rest are shown in blue. The vertical line indicates the selection of $\lambda_1^* = 2.90 \times 10^{-1}$, and shows that the outliers were correctly identified.

TABLE I
RESULTS FOR THE THIN-PLATE SPLINES SIMULATED TEST

N_o	λ_1^*	μ^*	$\bar{\text{err}}$ **for** (6)	$\bar{\text{err}}$ **for** (20)	$\text{Err}_{\mathcal{T}}$ **for** (6)	$\text{Err}_{\mathcal{T}}$ **for** (20)
10	2.72×10^{-1}	9.72×10^{-2}	1.00×10^{-2}	9.92×10^{-3}	1.92×10^{-2}	1.47×10^{-2}
20	2.90×10^{-1}	3.53×10^{-1}	1.02×10^{-2}	1.03×10^{-2}	5.79×10^{-2}	4.86×10^{-2}
30	2.75×10^{-1}	4.33×10^{-2}	1.00×10^{-2}	9.80×10^{-3}	1.60×10^{-2}	1.32×10^{-2}
40	2.58×10^{-1}	9.90×10^{-1}	9.90×10^{-3}	1.07×10^{-2}	5.13×10^{-2}	2.90×10^{-2}
50	2.36×10^{-1}	5.34×10^{-1}	1.04×10^{-2}	1.03×10^{-2}	6.89×10^{-2}	4.53×10^{-2}

support in all cases. It has been observed in some other setups that (13) may select a larger support than $[1, N_o]$, but after running a few iterations of (23) the true support was typically identified. To assess quality of the estimated function $\hat{f}$, two figures of merit were considered. First, the *training error* $\bar{\text{err}}$ was evaluated as

$$\bar{\text{err}} = \frac{1}{N - N_o} \sum_{i=N_o}^{N} \left(y_i - \hat{f}(\mathbf{x}_i) \right)^2$$

i.e., the average loss over the training sample $\mathcal{T}$ after excluding outliers. Second, to assess the generalization capability of $\hat{f}$, an approximation to the *generalization error* $\text{Err}_{\mathcal{T}}$ was computed as

$$\text{Err}_{\mathcal{T}} = E\left[\left(y - \hat{f}(\mathbf{x}) \right)^2 | \mathcal{T} \right] \approx \frac{1}{\tilde{N}} \sum_{i=1}^{\tilde{N}} \left(\tilde{y}_i - \hat{f}(\tilde{\mathbf{x}}_i) \right)^2 \quad (26)$$

where $\{\tilde{y}_i, \tilde{\mathbf{x}}_i\}_{i=1}^{\tilde{N}}$ is an independent test set generated from the model $\tilde{y}_i = f_o(\tilde{\mathbf{x}}_i) + \varepsilon_i$. For the results in Table I, $\tilde{N} = 961$ was adopted corresponding to a uniform rectangular grid of 31 $\times$ 31 points $\tilde{\mathbf{x}}_i$ in $[0, 3] \times [0, 3]$. Inspection of Table I reveals that the training errors $\bar{\text{err}}$ are comparable for the function estimates obtained after solving (6) or its nonconvex refinement (20). Interestingly, when it comes to the more pragmatic generalization error $\text{Err}_{\mathcal{T}}$, the refined estimator (20) has an edge for all values of N_o. As expected, the bias reduction effected by the iteratively reweighting procedure of Section IV improves considerably the generalization capability of the method; see also Remark 3.

A pictorial summary of the results is given in Fig. 4, for $N_o = 20$ outliers. Fig. 4(a) depicts the true Gaussian mixture $f_o(\mathbf{x})$, whereas Fig. 4(b) shows the nonrobust thin-plate splines estimate obtained after solving

$$\min_{f \in \mathcal{S}} \left[\sum_{i=1}^{N} (y_i - f(\mathbf{x}_i))^2 + \mu \int_{\mathbb{R}^2} \|\nabla^2 f\|_F^2 d\mathbf{x} \right]. \quad (27)$$

Even though the thin-plate penalty enforces some degree of smoothness, the estimate is severely disrupted by the presence of outliers [cf. the difference on the z-axis ranges]. On the other hand, Fig. 4(c) and (d), respectively, shows the robust estimate $\hat{f}$ with $\lambda_1^* = 2.90 \times 10^{-1}$, and its bias reducing refinement. The improvement is apparent, corroborating the effectiveness of the proposed approach.

B. Sinc Function Estimation

The univariate function $\text{sinc}(x) := \frac{\sin(\pi x)}{(\pi x)}$ is commonly adopted to evaluate the performance of nonparametric regression methods [10], [49]. Given noisy training examples with a small fraction of outliers, approximating $\text{sinc}(x)$ over the interval $[-5, 5]$ is considered in the present simulated test. The sparsity-controlling robust nonparametric regression methods of this paper are compared with the SVR [43] and robust SVR in [10], for the case of the ϵ-insensitve loss function with values $\epsilon = 0.1$ and $\epsilon = 0.01$. In order to implement (R)SVR, routines from a publicly available SVM Matlab toolbox were utilized [23]. Results for the nonrobust regularization network approach in (1) (with $V(u) = u^2$) are reported as well, to assess the performance degradation incurred when compared to the aforementioned robust alternatives. Because the fraction of outliers $\left(\frac{N_o}{N}\right)$ in the training data is assumed known to the method of [10], the same will be assumed towards selecting the tuning parameters λ_1 and μ in (6), as described in Section III-B. The $\{\mu, \lambda_1\}$-grid parameters selected for the experiment in Section V-A were used here as well, except for $\mu_{\min} = 10^{-5}$. Space $\mathcal{H}$ is chosen to be the RKHS induced by the positive definite Gaussian kernel function $K(u, v) = \exp\left[-\frac{(u-v)^2}{(2\eta^2)}\right]$, with parameter $\eta = 0.1$ for all cases.

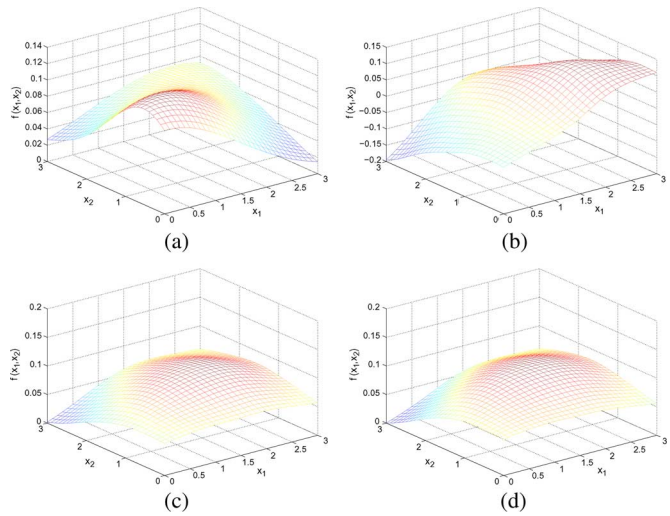

Fig. 4. Robust estimation of a Gaussian mixture using thin-plate splines. The data is corrupted with $N_o = 20$ outliers. (a) True function $f_o(\mathbf{x})$; (b) nonrobust predicted function obtained after solving (27); (c) predicted function after solving (24) with the optimum tuning parameters; (d) refined predicted function using the nonconvex regularization in (20).

The training set comprises $N = 50$ examples, with scalar inputs $\{x_i\}_{i=1}^{N}$ drawn from a uniform distribution over $[-5, 5]$. Uniformly distributed outliers $\{y_i\}_{i=1}^{N_o} \sim \mathcal{U}[-5, 5]$ are artificially added in $\mathcal{T}$, with $N_o = 3$ resulting in 6% contamination. Nominal data in $\mathcal{T}$ adheres to the model $y_i = \text{sinc}(x_i) + \varepsilon_i$ for $i = N_o + 1, \ldots, N$, where the independent additive noise

TABLE II
GENERALIZATION ERROR ($\mathrm{Err}_{\mathcal{T}}$) RESULTS FOR THE SINC FUNCTION ESTIMATION EXPERIMENT

Method	$\sigma_\varepsilon^2 = 1 \times 10^{-4}$	$\sigma_\varepsilon^2 = 1 \times 10^{-3}$	$\sigma_\varepsilon^2 = 1 \times 10^{-2}$
Nonrobust [(1) with $V(u) = u^2$]	5.67×10^{-2}	8.28×10^{-2}	1.13×10^{-1}
SVR with $\epsilon = 0.1$	5.00×10^{-3}	6.42×10^{-4}	6.15×10^{-3}
RSVR with $\epsilon = 0.1$	1.10×10^{-3}	5.10×10^{-4}	4.47×10^{-3}
SVR with $\epsilon = 0.01$	8.24×10^{-5}	4.79×10^{-4}	5.60×10^{-3}
RSVR with $\epsilon = 0.01$	7.75×10^{-5}	3.90×10^{-4}	3.32×10^{-3}
Sparsity-controlling in (6)	1.47×10^{-4}	6.56×10^{-4}	4.60×10^{-3}
Refinement in (20)	7.46×10^{-5}	3.59×10^{-4}	3.21×10^{-3}

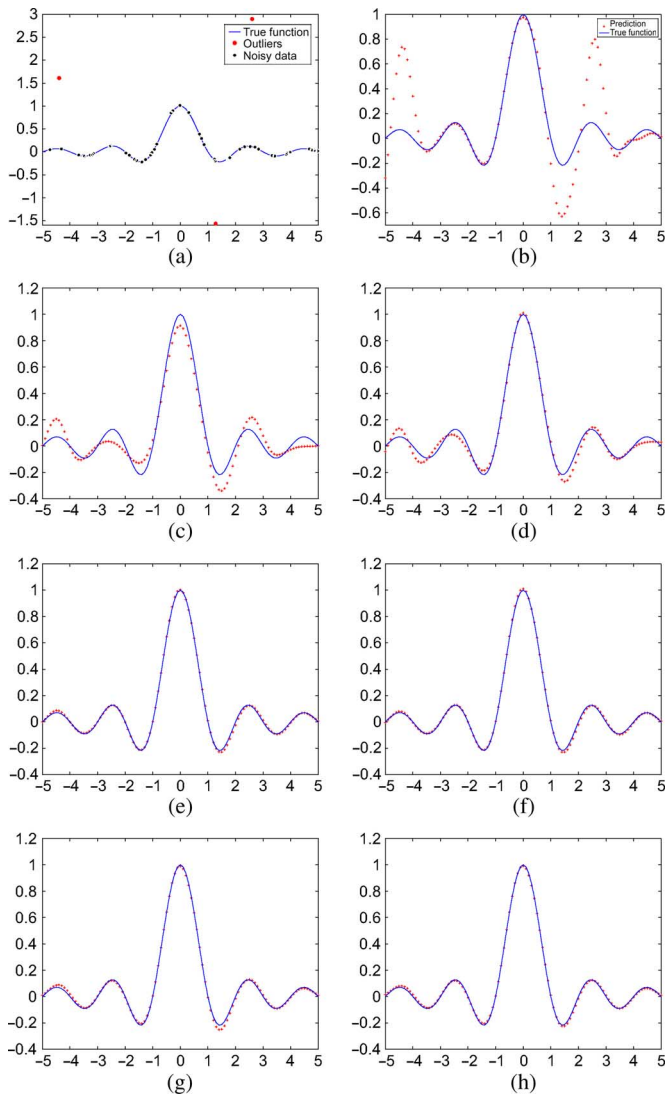

Fig. 5. Robust estimation of the sinc function. The data is corrupted with $N_o = 3$ outliers, and the nominal noise variance is $\sigma_\varepsilon^2 = 1 \times 10^{-4}$. (a) Noisy training data and outliers; (b) predicted values obtained after solving (1) with $V(u) = u^2$; (c) SVR predictions for $\epsilon = 0.1$; (d) RSVR predictions for $\epsilon = 0.1$; (e) SVR predictions for $\epsilon = 0.01$; (f) RSVR predictions for $\epsilon = 0.01$; (g) predicted values obtained after solving (6); (h) refined predictions using the nonconvex regularization in (20).

terms ε_i are zero-mean Gaussian distributed. Three different values are considered for the nominal noise variance, namely $\sigma_\varepsilon^2 = 1 \times 10^{-l}$ for $l = 2, 3, 4$. For the case where $\sigma_\varepsilon^2 = 1 \times 10^{-4}$, the data used in the experiment are shown in Fig. 5(a). Superimposed to the true function $\mathrm{sinc}(x)$ (shown in blue) are 47 black points corresponding to the noisy data obeying the nominal model, as well as 3 outliers depicted as red points.

The results are summarized in Table II, which lists the generalization errors $\mathrm{Err}_{\mathcal{T}}$ attained by the different methods tested, and for varying σ_ε^2. The independent test set $\{\tilde{y}_i, \tilde{x}_i\}_{i=1}^{\tilde{N}}$ used to evaluate (26) was generated from the model $\tilde{y}_i = \mathrm{sinc}(\tilde{x}_i) + \varepsilon_i$, where the $\tilde{x}_i$ define a $\tilde{N} = 101$-element uniform grid over $[-5, 5]$. A first (expected) observation is that all robust alternatives markedly outperform the nonrobust regularization network approach in (1), by an order of magnitude or even more, regardless of the value of σ_ε^2. As reported in [10], RSVR uniformly outperforms SVR. For the case $\epsilon = 0.01$, RSVR also uniformly outperforms the sparsity-controlling method in (6). Interestingly, after refining the estimate obtained via (6) through a couple iterations of (23) (cf. Section IV), the lowest generalization errors are obtained, uniformly across all simulated values of the nominal noise variance. Results for the RSVR with $\epsilon = 0.01$ come sufficiently close, and are equally satisfactory for all practical purposes; see also Fig. 5 for a pictorial summary of the results when $\sigma_\varepsilon^2 = 1 \times 10^{-4}$.

While specific error values or method rankings are arguably anecdotal, two conclusions stand out: i) model (3) and its sparsity-controlling estimators (6) and (20) are effective approaches to nonparametric regression in the presence of outliers; and ii) when initialized with $\hat{\mathbf{o}}_{\mathrm{Lasso}}$ the refined estimator (20) can considerably improve the performance of (6), at the price of a modest increase in computational complexity. While (6) endowed with the sparsity-controlling mechanisms of Section III-B tends to overestimate the "true" support of $\mathbf{o}$, numerical results have consistently shown that the refinement in Section IV is more effective when it comes to support recovery.

C. Load Curve Data Cleansing

In this section, the robust nonparametric methods described so far are applied to the problem of load curve cleansing outlined in Section I. Given load data $\mathcal{T} := \{y_i, t_i\}_{i=1}^N$ corresponding to a building's power consumption measurements y_i, acquired at time instants t_i, $i = 1, \ldots, N$, the proposed approach to load curve cleansing minimizes

$$\min_{\substack{f \in \mathcal{S} \\ \mathbf{o} \in \mathbb{R}^N}} \left[\sum_{i=1}^{N} (y_i - f(t_i) - o_i)^2 + \mu \int_{\mathbb{R}} f''(t) dt + \lambda_1 \|\mathbf{o}\|_1 \right] \tag{28}$$

where $f''(t)$ denotes the second-order derivative of $f : \mathbb{R} \to \mathbb{R}$. This way, the solution $\hat{f}$ provides a cleansed estimate of the load profile, and the support of $\hat{\mathbf{o}}$ indicates the instants where significant load deviations, or, meter failures occurred. Estimator (28) specializes (6) to the so-termed *cubic smoothing splines*; see, e.g., [24], [44]. It is also subsumed as a special case of the robust thin-plate splines estimator (24), when the target function f has domain in $\mathbb{R}$ [cf. how the smoothing penalty (25) simplifies to the one in (28) in the one-dimensional case].

In light of the aforementioned connection, it should not be surprising that $\hat{f}$ admits a unique, finite-dimensional minimizer, which corresponds to a *natural spline* with knots at $\{t_i\}_{i=1}^N$; see e.g., [24, p. 151]. Specifically, it follows that $\hat{f}(t) = \sum_{i=1}^N \hat{\theta}_i b_i(t)$, where $\{b_i(t)\}_{i=1}^N$ is the basis set of natural spline functions, and the vector of expansion coefficients $\hat{\boldsymbol{\theta}} := [\hat{\theta}_1, \ldots, \hat{\theta}_N]'$ is given by

$$\hat{\boldsymbol{\theta}} = (\mathbf{B}'\mathbf{B} + \mu\boldsymbol{\Psi})^{-1}\mathbf{B}'(\mathbf{y} - \hat{\mathbf{o}})$$

where matrix $\mathbf{B} \in \mathbb{R}^{N\times N}$ has ijth entry $[\mathbf{B}]_{ij} = b_j(t_i)$; while $\boldsymbol{\Psi} \in \mathbb{R}^{N\times N}$ has ijth entry $[\boldsymbol{\Psi}]_{ij} = \int b_i''(t)b_j''(t)dt$. Spline coefficients can be computed more efficiently if the basis of B-splines is adopted instead; details can be found in [24, p. 189] and [42].

Without considering the outlier variables in (28), a B-spline estimator for load curve cleansing was put forth in [8]. An alternative Nadaraya–Watson estimator from the Kernel smoothing family was considered as well. In any case, outliers are identified during a postprocessing stage, after the load curve has been estimated nonrobustly. Supposing for instance that the approach in [8] correctly identifies outliers most of the time, it still does not yield a cleansed estimate $\hat{f}$. This should be contrasted with the estimator (28), which accounts for the outlier compensated data to yield a cleansed estimate at once. Moreover, to select the "optimum" smoothing parameter μ, the approach of [8] requires the user to manually label the outliers present in a training subset of data, during a preprocessing stage. This subjective component makes it challenging to reproduce the results of [8], and for this reason comparisons with the aforementioned scheme are not included in the sequel.

Next, estimator (28) is tested on real load curve data provided by the NorthWrite Energy Group. The dataset consists of power consumption measurements (in kWh) for a government building, collected every fifteen minutes during a period of more than five years, ranging from July 2005 to October 2010. Data is downsampled by a factor of four, to yield one measurement per hour. For the present experiment, only a subset of the whole data is utilized for concreteness, where $N = 501$ was chosen corresponding to a 501-hour period. A snapshot of this training load curve data in $\mathcal{T}$, spanning a particular three-week period is shown in Fig. 6(a). Weekday activity patterns can be clearly discerned from those corresponding to weekends, as expected for most government buildings; but different, e.g., for the load profile of a grocery store. Fig. 6(b) shows the nonrobust smoothing spline fit to the training data in $\mathcal{T}$ (also shown for comparison purposes), obtained after solving

$$\min_{f\in\mathcal{S}} \left[\sum_{i=1}^N (y_i - f(t_i))^2 + \mu \int_{\mathbb{R}} f''(t)dt \right] \tag{29}$$

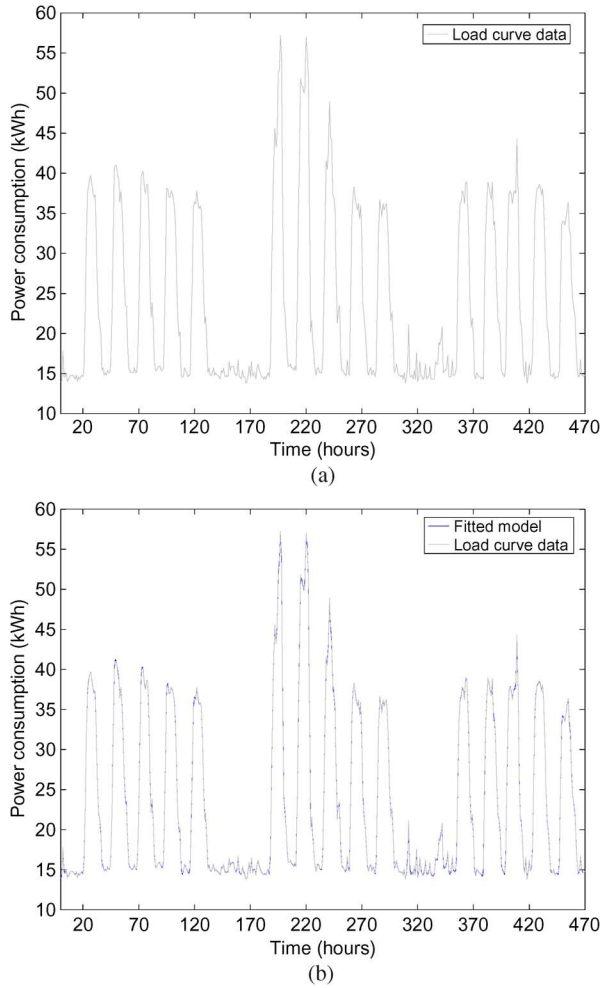

Fig. 6. Load curve data cleansing. (a) Noisy training data and outliers; (b) fitted load profile obtained after solving (29).

using Matlab's built-in spline toolbox. Parameter μ was chosen based on leave-one-out cross-validation, and it is apparent that no cleansing of the load profile takes place. Indeed, the resulting fitted function follows very closely the training data, even during the abnormal energy peaks observed on the so-termed "building operational transition shoulder periods."

Because with real load curve data the nominal noise variance σ_ε^2 in (3) is unknown, selection of the tuning parameters $\{\mu, \lambda_1\}$ in (28) requires a robust estimate of the variance $\hat{\sigma}_\varepsilon^2$ such as the MAD [cf. Section III-B]. Similar to [8], it is assumed that the nominal errors are zero mean Gaussian distributed, so that (19) can be applied yielding the value $\hat{\sigma}_\varepsilon^2 = 0.6964$. To form the residuals in (19), (29) is solved first using a small subset of $\mathcal{T}$ that comprises 126 measurements. A nonuniform grid of μ and λ_1 values is constructed, as described in Section III-B. Relevant parameters are $G_\mu = 100$, $G_\lambda = 200$, $\mu_{\min} = 10^{-3}$, $\mu_{\max} = 10$, and $\epsilon = 10^{-4}$. The robustification paths (one per μ value in the grid) were obtained using the SpaRSA toolbox in [47], with the sample variance matrix $\bar{\boldsymbol{\Sigma}}$ formed as in (17). The optimum tuning parameters $\mu^* = 1.637$ and $\lambda_1^* = 3.6841$ are finally determined based on the criterion (18), where the unknown σ_ε^2 is replaced with $\hat{\sigma}_\varepsilon^2$. Finally, the cleansed load curve is refined by running four iterations of (23) as described in Section IV, with a value of $\delta = 10^{-5}$. Results are depicted in Fig. 7, where

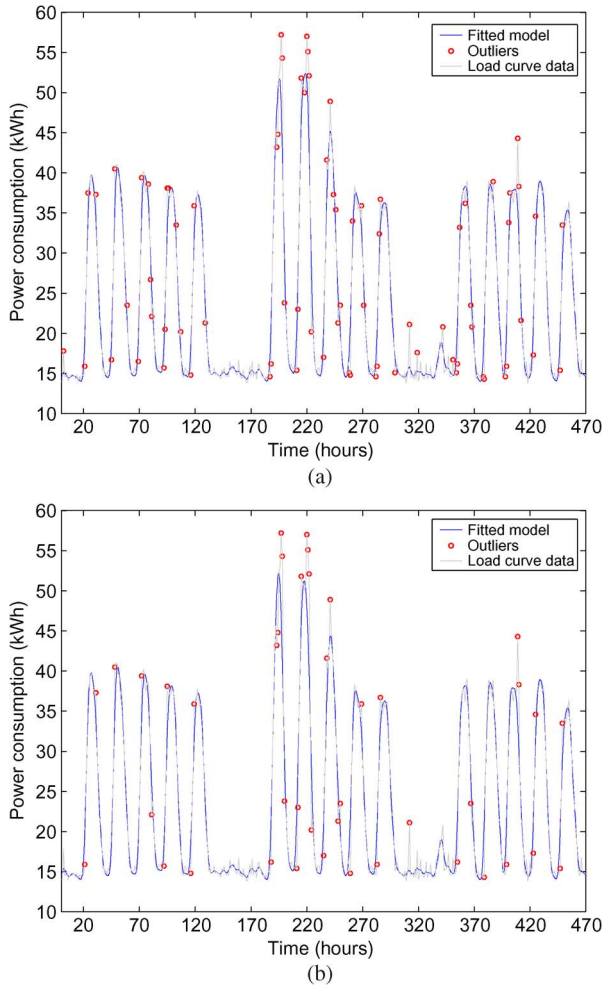

Fig. 7. Load curve data cleansing. (a) Cleansed load profile obtained after solving (28); (b) refined load profile obtained after using the nonconvex regularization in (20).

the cleansed load curves are superimposed to the training data in $\mathcal{T}$. Red circles indicate those data points deemed as outliers, information that is readily obtained from the support of $\hat{\mathbf{o}}$. By inspection of Fig. 7, it is apparent that the proposed sparsity-controlling estimator has the desired cleansing capability. The cleansed load curves closely follow the training data, but are smooth enough to avoid overfitting the abnormal energy peaks on the "shoulders." Indeed, these peaks are in most cases identified as outliers. As seen from Fig. 7(a), the solution of (28) tends to overestimate the support of $\mathbf{o}$, since one could argue that some of the red circles in Fig. 7(a) do not correspond to outliers. Again, the nonconvex regularization in Section IV prunes the outlier support obtained via (28), resulting in a more accurate result in terms of the residual fit to the data and reducing the number of outliers identified from 77 to 41.

VI. Concluding Summary

Outlier-robust nonparametric regression methods were developed in this paper for function approximation in RKHS. Building on a neat link between the seemingly unrelated fields of robust statistics and sparse regression, the novel estimators were found rooted at the crossroads of outlier-resilient estimation, the Lasso, and convex optimization. Estimators as fundamental as LS for linear regression, regularization networks, and (thin-plate) smoothing splines, can be robustified under the proposed framework.

Training samples from the (unknown) target function were assumed generated from a regression model, which explicitly incorporates an unknown sparse vector of outliers. To fit such a model, the proposed variational estimator minimizes a tradeoff between fidelity to the training data, the degree of "smoothness" of the regression function, and the sparsity level of the vector of outliers. While model complexity control effected through a smoothing penalty has quite well understood ramifications in terms of generalization capability, the major innovative claim here is that *sparsity control* is tantamount to robustness control. This is indeed the case since a tunable parameter in a Lasso reformulation of the variational estimator, controls the degree of sparsity in the estimated vector of model outliers. Selection of tuning parameters could be at first thought as a mundane task. However, arguing on the importance of such task in the context of robust nonparametric regression, as well as devising principled methods to effectively carry out smoothness and sparsity control, are at the heart of this paper's novelty. Sparsity control can be carried out at affordable complexity, by capitalizing on state-of-the-art algorithms that can efficiently compute the whole path of Lasso solutions. In this sense, the method here capitalizes on but is not limited to sparse settings where few outliers are present, since one can efficiently examine the gamut of sparsity levels along the robustification path. Computer simulations have shown that the novel methods of this paper outperform existing alternatives including SVR, and one if its robust variants.

As an application domain relevant to robust nonparametric regression, the problem of load curve cleansing for power systems engineering was also considered along with a solution proposed based on robust cubic spline smoothing. Numerical tests on real load curve data demonstrated that the smoothness and sparsity controlling methods of this paper are effective in cleansing load profiles, without user intervention to aid the learning process.

Appendix

Towards establishing the equivalence between problems (6) and (7), consider the pair $\{\hat{f}, \hat{\mathbf{o}}\}$ that solves (6). Assume that $\hat{f}$ is given, and the goal is to determine $\hat{\mathbf{o}}$. Upon defining the residuals $\hat{r}_i := y_i - \hat{f}(\mathbf{x}_i)$ and because $\|\mathbf{o}\|_1 = \sum_{i=1}^{N} |o_i|$, the entries of $\hat{\mathbf{o}}$ are separately given by

$$\hat{o}_i := \arg\min_{o_i \in \mathbb{R}} \left[(\hat{r}_i - o_i)^2 + \lambda_1 |o_i| \right], \quad i = 1, \ldots, N \tag{30}$$

where the term $\mu \|\hat{f}\|^2_{\mathcal{H}}$ in (6) has been omitted, since it is inconsequential for the minimization with respect to $\mathbf{o}$. For each $i = 1, \ldots, N$, because (30) is nondifferentiable at the origin one should consider three cases: i) if $\hat{o}_i = 0$, it follows that the minimum cost in (30) is $\hat{r}_i^2$; ii) if $\hat{o}_i > 0$, the first-order condition for optimality gives $\hat{o}_i = \hat{r}_i - \frac{\lambda_1}{2}$ provided $\hat{r}_i > \frac{\lambda_1}{2}$, and the minimum cost is $\lambda_1 \hat{r}_i - \frac{\lambda_1^2}{4}$; otherwise, iii) if $\hat{o}_i < 0$, it follows that $\hat{o}_i = \hat{r}_i + \frac{\lambda_1}{2}$ provided $\hat{r}_i < -\frac{\lambda_1}{2}$, and the minimum cost is $-\lambda_1 \hat{r}_i - \frac{\lambda_1^2}{4}$. In other words,

$$\hat{o}_i = \begin{cases} \hat{r}_i - \lambda_1/2, & \hat{r}_i > \lambda_1/2 \\ 0, & |\hat{r}_i| \leq \lambda_1/2 \\ \hat{r}_i + \lambda_1/2, & \hat{r}_i < -\lambda_1/2 \end{cases}, \quad i = 1, \ldots, N. \tag{31}$$

Upon plugging (31) into (30), the minimum cost in (30) after minimizing with respect to o_i is $\rho(\hat{r}_i)$ [cf. (8) and the argument preceding (31)]. All in all, the conclusion is that $\hat{f}$ is the minimizer of (7)—in addition to being the solution of (6) by definition—completing the proof. ■

ACKNOWLEDGMENT

The authors would like to thank the NorthWrite Energy Group and Prof. V. Cherkassky (Department of Electrical and Computer Engineering, University of Minnesota) for providing the load curve data analyzed in Section V-C.

REFERENCES

[1] U.S. Congress, Act of the Publ. L. No. 110-140, H.R. 6, Energy Independence and Security Act of 2007 Dec. 2007 [Online]. Available: http://www.gpo.gov/fdsys/pkg/PLAW-110publ140/content-detail.html

[2] The Smart Grid: An Introduction United States Department of Energy, Office of Electricity Delivery and Energy Reliability, Jan. 2010 [Online]. Available: http://www.oe.energy.gov/1165.htm

[3] M. S. Asif and J. Romberg, "Dynamic updating for l1 minimization," *IEEE Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 421–434, 2010.

[4] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA: Athena Scientific, 1999.

[5] E. J. Candes and P. A. Randall, "Highly robust error correction by convex programming," *IEEE Trans. Inf. Theory*, vol. 54, no. 7, pp. 2829–2840, 2008.

[6] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[7] E. J. Candes, M. B. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimzation," *J. Fourier Anal. Appl.*, vol. 14, pp. 877–905, Dec. 2008.

[8] J. Chen, W. Li, A. Lau, J. Cao, and K. Eang, "Automated load curve data cleansing in power systems," *IEEE Trans. Smart Grid*, vol. 1, pp. 213–221, Sep. 2010.

[9] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, pp. 33–61, 1998.

[10] C. C. Chuang, S. F. Fu, J. T. Jeng, and C. C. Hsiao, "Robust support vector regression networks for function approximation with outliers," *IEEE Trans. Neural Netw.*, vol. 13, pp. 1322–1330, Jun. 2002.

[11] D. D. Cox, "Asymptotics for M-type smoothing splines," *Ann. Stat.*, vol. 11, pp. 530–551, 1983.

[12] J. Duchon, *Splines Minimizing Rotation-Invariant Semi-Norms in Sobolev Spaces*. New York: Springer-Verlag, 1977.

[13] B. Efron, T. Hastie, I. M. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Stat.*, vol. 32, pp. 407–499, 2004.

[14] T. Evgeniou, M. Pontil, and T. Poggio, "Regularization networks and support vector machines," *Adv. Comput. Math.*, vol. 13, pp. 1–50, 2000.

[15] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *J. Amer. Stat. Assoc.*, vol. 96, pp. 1348–1360, 2001.

[16] M. Fazel, "Matrix rank minimization with applications," Ph.D. dissertation, Electr. Eng. Dept., Stanford Univ., Stanford, CA, 2002.

[17] J. Friedman, T. Hastie, H. Hofling, and R. Tibshirani, "Pathwise coordinate optimization," *Ann. Appl. Stat.*, vol. 1, pp. 302–332, 2007.

[18] J. Friedman, T. Hastie, and R. Tibshirani, "Regularized paths for generalized linear models via coordinate descent," *J. Stat. Softw.*, vol. 33, 2010 [Online]. Available: http://www.jstatsoft.org/v33/i01

[19] J. J. Fuchs, "An inverse problem approach to robust regression," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Phoenix, AZ, Mar. 1999, pp. 180–188.

[20] P. Garrigues and L. El Ghaoui, "Recursive Lasso: A homotopy algorithm for Lasso with online observations," presented at the Conf. Neural Inf. Process. Syst., Vancouver, Canada, Dec. 2008.

[21] G. B. Giannakis, G. Mateos, S. Farahmand, V. Kekatos, and H. Zhu, "USPACOR: Universal sparsity-controlling outlier rejection," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Prague, Czech Republic, May 2011, pp. 1952–1955.

[22] R. Griesse and D. A. Lorenz, "A semismooth Newton method for Tikhonov functionals with sparsity constraints," *Inverse Problems*, vol. 24, pp. 1–19, 2008.

[23] S. R. Gunn, Matlab SVM Toolbox, 1997 [Online]. Available: http://www.isis.ecs.soton.ac.uk/resources/svminfo/

[24] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed. New York: Springer, 2009.

[25] S. G. Hauser, "Vision for the smart grid," presented at the U.S. Department of Energy Smart Grid R&D Roundtable Meeting, Washington DC, Dec. 9, 2009.

[26] P. J. Huber and E. M. Ronchetti, *Robust Statistics*. New York: Wiley, 2009.

[27] Y. Jin and B. D. Rao, "Algorithms for robust linear regression by exploiting the connection to sparse signal recovery," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Dallas, TX, Mar. 2010, pp. 3830–3833.

[28] V. Kekatos and G. B. Giannakis, "From sparse signals to sparse residuals for robust sensing," *IEEE Trans. Signal Process.*, vol. 59, pp. 3355–3368, Jul. 2010.

[29] G. Kimbeldorf and G. Wahba, "A correspondence between bayesian estimation on stochastic processes and smoothing by splines," *Ann. Math. Stat.*, vol. 41, pp. 495–502, 1970.

[30] K. Lange, D. Hunter, and I. Yang, "Optimization transfer using surrogate objective functions (with discussion)," *J. Comput. Graph. Stat.*, vol. 9, pp. 1–59, 2000.

[31] Y. J. Lee, W. F. Heisch, and C. M. Huang, "ϵ-SSVR: A smooth support vector machine for ϵ-insensitive regression," *IEEE Trans. Knowl. Data Eng.*, vol. 17, pp. 678–685, 2005.

[32] O. L. Mangasarian and D. R. Musicant, "Robust linear and support vector regression," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, pp. 950–955, Sep. 2000.

[33] S. Mukherjee, E. Osuna, and F. Girosi, "Nonlinear prediction of chaotic time series using a support vector machine," in *Proc. Workshop Neural Netw. Signal Process.*, Amelia Island, FL, 1997, pp. 24–26.

[34] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, pp. 227–234, 1995.

[35] T. Poggio and F. Girosi, "A theory of networks for approximation and learning," Artificial Intelligence Lab., Massachusetts Inst. of Technology, Cambridge, A. I. Memo No. 1140, 1989, "A theory of networks for approximation and learning," .

[36] P. J. Rousseeuw and K. V. Driessen, "Computing LTS regression for large data sets," *Data Mining Knowl. Discovery*, vol. 12, pp. 29–45, 2006.

[37] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*. New York: Wiley, 1987.

[38] A. J. Smola and B. Scholkopf, "A tutorial on support vector regression," Royal Holloway College, London, Neuro COLT Tech. Rep. TR-1998-030, 1998.

[39] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Royal Stat. Soc B*, vol. 58, pp. 267–288, 1996.

[40] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*. Washington DC: W. H. Winston, 1977.

[41] J. Tropp, "Just relax: Convex programming methods for identifying sparse signals," *IEEE Trans. Inf. Theory*, vol. 51, pp. 1030–1051, Mar. 2006.

[42] M. Unser, "Splines: A perfect fit for signal and image processing," *IEEE Signal Process. Mag.*, vol. 16, pp. 22–38, Nov. 1999.

[43] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[44] G. Wahba, *Spline Models for Observational Data*. Philadelphia, PA: SIAM, 1990.

[45] G. Wahba and J. Wendelberger, "Some new mathematical methods for variational objective analysis using splines and cross validation," *Month. Weather Rev.*, vol. 108, pp. 1122–1145, 1980.

[46] J. Wright and Y. Ma, "Dense error correction via ℓ^1-minimization," *IEEE Trans. Inf. Theory*, vol. 56, no. 7, pp. 3540–3560, 2010.

[47] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. Signal Process.*, vol. 57, pp. 2479–2493, Jul. 2009.

[48] T. Wu and K. Lange, "Coordinate descent algorithms for lasso penalized regression," *Ann. Appl. Stat.*, vol. 2, pp. 224–244, 2008.

[49] J. Zhu, S. C. H. Hoi, and M. R. T. Lyu, "Robust regularized kernel regression," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, pp. 1639–1644, Dec. 2008.

[50] H. Zou, "The adaptive Lasso and its oracle properties," *J. Amer. Stat. Assoc.*, vol. 101, no. 476, pp. 1418–1429, 2006.

Gonzalo Mateos (S'07) received his B.Sc. degree in electrical engineering from the Universidad de la República (UdelaR), Montevideo, Uruguay, in 2005 and the M.Sc. degree in electrical and computer engineering from the University of Minnesota, Minneapolis, in 2009. Since August 2006, he has been working towards the Ph.D. degree as a Research Assistant with the Department of Electrical and Computer Engineering, University of Minnesota.

Since 2003, he is an Assistant with the Department of Electrical Engineering, UdelaR. From 2004 to 2006, he worked as a Systems Engineer at Asea Brown Boveri (ABB), Uruguay. His research interests lie in the areas of communication theory, signal processing and networking. His current research focuses on distributed signal processing, sparse linear regression, and statistical learning for social data analysis.

Georgios B. Giannakis (F'97) received the Diploma in electrical engineering from the National Technical University of Athens, Greece, in 1981. From 1982 to 1986, he was with the University of Southern California (USC), where he received the M.Sc. degree in electrical engineering in 1982, in mathematics in 1986, and the Ph.D. degree in electrical engineering in 1986.

Since 1999, he has been a Professor with the University of Minnesota, where he now holds an ADC Chair in Wireless Telecommunications in the Electrical and Computer Engineering Department and serves as Director of the Digital Technology Center. His general interests span the areas of communications, networking and statistical signal processing—subjects on which he has published more than 300 journal papers, 500 conference papers, two edited books, and two research monographs. Current research focuses on compressive sensing, cognitive radios, cross-layer designs, wireless sensors, and social and power grid networks.

Dr. Giannakis is the (co-)inventor of 21 patents issued, and the (co-)recipient of eight best paper awards from the IEEE Signal Processing (SP) and Communications Societies, including the G. Marconi Prize Paper Award in Wireless Communications. He also received Technical Achievement Awards from the SP Society (2000), from EURASIP (2005), a Young Faculty Teaching Award, and the G. W. Taylor Award for Distinguished Research from the University of Minnesota. He is a Fellow of EURASIP and has served the IEEE in a number of posts, including that of a Distinguished Lecturer for the IEEE-SP Society.