ROBUST REGRESSION IN RKHS - AN OVERVIEW

George Papageorgiou, Pantelis Bouboulis, Sergios Theodoridis

Department of Informatics
and Telecommunications
University of Athens
Athens, Greece, 157 84
Emails: geopapag, stheodor@di.uoa.gr, panbouboulis@gmail.com

ABSTRACT

The paper deals with the task of robust nonlinear regression in the presence of outliers. The problem is dealt in the context of reproducing kernel Hilbert spaces (RKHS). In contrast to more classical approaches, a recent trend is to model the outliers as a sparse vector noise component and mobilize tools from the sparsity-aware/compressed sensing theory to impose sparsity on it. In this paper, three of the most popular approaches are considered and compared. These represent three major directions in sparsity-aware learning context; that is, a) a greedy approach b) a convex relaxation of the sparsity-promoting task via the ℓ_1 norm-based regularization of the least-squares cost and c) a Bayesian approach making use of appropriate priors, associated with the involved parameters.

Index Terms— Robust regression in RKHS, learning with kernels, kernel greedy algorithm for robust denoising - (KGARD), robust non-linear regression

1. INTRODUCTION

Classification and regression have always been two major tasks in the field of machine learning, and signal processing. The task of "learning" in the presence of outliers and the urge to develop robust parameter estimation techniques, is not new. However, it has been drawing attention again, recently, in almost every field. The current paper deals with the non-parametric non-linear regression task. The non-linearity is modelled via the assumption that the unknown non-linear function, that quantifies the input-output dependence, lies in a RKHS. Moreover, our focus is entirely set in the context of robust estimation, i.e., performing the estimation, disregarding the outliers that our data is contaminated with. The purpose of this work is to analyse and compare related competitive state-of-the-art methods.

This research has been co-financed by the European Union (European Social Fund - ESF) and Greek National funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Aristeia I - 621.

In the classic problem of non-linear regression, where outliers are not present, each measurement y_i is assumed to be generated via the non-linear model:

$$y_i = f(\boldsymbol{x}_i) + v_i,\ i = 1, ..., N, \tag{1}$$

where v_i are random noise variables. Given the training set/sample $\mathcal{D} = \{(y_i, \boldsymbol{x}_i)\}_{i=1}^N$, our goal is to learn the input-output relation $\hat{y}_i = \hat{f}(\boldsymbol{x}_i)$. All kernel methods, assume that f belongs to a space of "smooth" functions $\mathcal{H}$, which is assumed to have a structure of a reproducing kernel Hilbert space (RKHS). These are inner product function spaces, in which every function is reproduced by an associated (space defining) kernel, i.e., $f(\boldsymbol{x}) = \langle f, \kappa(., \boldsymbol{x})\rangle_{\mathcal{H}}$. RKHS are very important in the field of machine learning, due to the celebrated Representer theorem, which states that the solution of any regularized risk functional, i.e.,

$$\min_f \left\{ \sum_{i=1}^N \big(y_i - f(\boldsymbol{x}_i)\big)^2 + \lambda ||f||_{\mathcal{H}}^2 \right\},\ \lambda \geq 0, \tag{2}$$

over the training set $\mathcal{D}$, where $\|\cdot\|_{\mathcal{H}} = \sqrt{\langle \cdot, \cdot \rangle_{\mathcal{H}}}$, admits a representation of the form $f(\boldsymbol{x}) = \sum_{i=1}^N \alpha_i \kappa(\boldsymbol{x}, \boldsymbol{x}_i)$. The regularization term, is used in order to guard the solution against overfitting (see, e.g., [1], [2, 3]). It can be shown that this method is optimal only for the case where the noise variables are independent zero-mean, e.g., i.i.d Gaussian. Hence, although the previous formulation has been successfully applied to remove Gaussian noise [4–6], it has been established that the presence of outliers renders the solution sensitive to overfitting. [7].

For the case where both outlier and inlier noise are present (e.g., the random variable v_i in (1) originates from a heavy tailed distribution instead), the Least-Squares approach fails to provide adequate estimation results, as demonstrated in Figure 1. To deal with the overfitting phenomenon, the noise scalar v_i is decomposed into two parts; that is, a sequence, u_i, associated with the outliers and a sequence η_i, associated with the inlier noise; hence $v_i = u_i + \eta_i$ and the input-output

978-0-9928626-3-3/15/$31.00 ©2015 IEEE

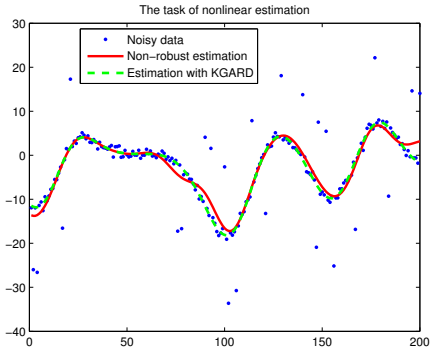

Fig. 1. The original data are corrupted with 20 dB Gaussian noise plus 12% outliers. The estimation performed with a non-robust cost function (least-squares loss), is greatly affected by the presence of outliers, attaining a $MSE_r = 1.81$ (red line). The robust approach (KGARD), improves the estimation significantly, attaining a $MSE_g = 0.05$ (green dashed line).

relation converts to:

$$y_i = f(\boldsymbol{x}_i) + u_i + \eta_i,\ i = 1, 2, ..., N, \tag{3}$$

as proposed in [8] and also in [9]. Thus, instead of attempting to estimate f, one should also provide estimates to u_i's, which, in the sequel, have to be extracted from the corrupted measurements. This family of methods falls into the category of robust estimation techniques (together with other robust methods, e.g., weighted least squares).

2. SPARSITY-AWARE LEARNING TECHNIQUES

Sparsity-aware learning techniques have dominated the scientific research for the past decade. In our context, sparsity constraints are imposed on the outlier vector. This is due to the fact that the outliers are expected to often comprise only a small fraction of the training sample. Thus, most of the values of u_i's are zeros. In general, a percentage of less than $15 - 20\%$ of non zero values is expected. Higher percentages are not very appealing, even though a few methods have been proposed to perform well in such cases. However such results should not be considered reliable (read [10–12]). In light of the previous discussion, the problem in (2), is reformulated accordingly.

Let $\boldsymbol{u} := (u_1, u_2, ..., u_N)^T$, be modelled as a *sparse* vector. Thus, by definition of sparsity, i.e., employing a sparsity constraint on $\boldsymbol{u}$ via the ℓ_0 norm, the optimization task can be cast as:

$$\begin{aligned} &\min_{\boldsymbol{u}, f \in \mathcal{H}} ||\boldsymbol{u}||_0 \\ &\text{s.t.} \sum_{i=1}^{N} \left(y_i - f(\boldsymbol{x}_i) - u_i\right)^2 + \lambda ||f||^2_{\mathcal{H}} \leq \varepsilon, \end{aligned} \tag{4}$$

for fixed threshold parameters $\varepsilon > 0$ and $\lambda > 0$. Clearly, the goal is to minimize the number of outliers, while preserving a low training error and simultaneously keeping the function smooth.

A major drawback is that the previous task (4), is of combinatorial nature, i.e., is by definition NP-hard. To overcome this obstacle, a number of techniques have been proposed.

2.1. Kernel Greedy Algorithm for Robust Denoising (KGARD)

The most recent approach, is attempting to solve (4) via *greedy* selection methods. In spite of their simplicity, such methods manage to perform the sparse approximation successfully, under certain assumptions. The proposed scheme is based on the popular *Orthogonal Matching Pursuit* (OMP) algorithm, see [3, 13–15], which is the core of a number of schemes that belong to the greedy family of methods.

Inspired by the Representer theorem, we assume that $f = \sum_{i=1}^{N} \alpha_i \kappa(\cdot, \boldsymbol{x}_i) + c$, i.e, a bias term, c, is also included in f. Hence, the set of functions is enlarged (see also the Semi-parametric Representer theorem in [1]). To this end, instead of solving problem (4), we target our efforts at estimating the solution of:

$$\begin{aligned} &\min_{\boldsymbol{u}, \boldsymbol{\alpha}, c} ||\boldsymbol{u}||_0 \\ &\text{s.t.} \quad ||\boldsymbol{y} - K\boldsymbol{\alpha} - c\mathbf{1} - \boldsymbol{u}||_2^2 + \lambda ||\boldsymbol{\alpha}||_2^2 + \lambda c^2 \leq \varepsilon, \end{aligned} \tag{5}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^N$ are the kernel expansion coefficients, $c \in \mathbb{R}$ is the bias term, $\mathbf{1} \in \mathbb{R}^N$ is the vector of ones and $\boldsymbol{y}, \boldsymbol{u} \in \mathbb{R}^N$ are the measurement and outlier vectors, respectively.

The main concept of the KGARD scheme is to perform an estimation for the sparse outlier vector via greedy selection, as summarized in Algorithm 1. In particular, after performing a least squares step on the subspace defined only by the columns vectors of $[K\mathbf{1}]$, a minimum angle selection step (outlier detection) follows, among vectors that belong to the second part of matrix $X = [K\ \mathbf{1}\ I_N]$, i.e., the identity matrix I_N. In other words, a search is performed at each step, and the column vector $\boldsymbol{e}_{j_k}$ of I_N ($j_k \in J := \{1, ..., n\}$) which maximizes the absolute inner product with the current residual, as defined in Algorithm 1, is selected. Then, the subspace is augmented by the selected column (regarded as an outlier) and a new least squares task is performed. After the method's termination, the estimate to our uncorrupted data is computed as

$$\hat{\boldsymbol{y}} = K\hat{\boldsymbol{\alpha}} + \hat{c}\mathbf{1}.$$

Furthermore, efficient implementations, such as Cholesky decomposition, QR factorization and the matrix inversion lemma (MIL), are applied and found to greatly reduce the complexity for the method. In particular, instead of a matrix inversion, which generally requires $\mathcal{O}(N^3)$ flops at each step, the use of the Cholesky decomposition reduces the complexity to $\mathcal{O}(N^2)$, due to the fact, that the inversion is avoided

Algorithm 1 Kernel Greedy Algorithm for Robust Denoising: KGARD

procedure KGARD($K,\ \boldsymbol{y},\ \lambda,\ \varepsilon$)
  $k \leftarrow 0,\ X = [K\ \mathbf{1}\ I_N]$
  $\mathcal{S}_{ac} \leftarrow \{1, ..., N+1\},\ \mathcal{S}_{inac} \leftarrow \{N+2, ..., 2N+1\}$
  $\hat{\boldsymbol{z}} := \left(X_{\mathcal{S}_{ac}}^T X_{\mathcal{S}_{ac}} + \lambda^2 B_{\mathcal{S}_{ac}}\right)^{-1} X_{\mathcal{S}_{ac}}^T \boldsymbol{y}$
  $\boldsymbol{r} \leftarrow \boldsymbol{y} - X_{\mathcal{S}_{ac}} \hat{\boldsymbol{z}}$
  while $\|\boldsymbol{r}\|_2 > \varepsilon$ **do**
    $k \leftarrow k + 1$
    $j_k := \arg\max_{j \in J} |r_j|$ ▷ Selection step.
    $\mathcal{S}_{ac} \leftarrow \mathcal{S}_{ac} \cup \{j_k + N + 1\},\ \mathcal{S}_{inac} \leftarrow \mathcal{S}_{inac} - \{j_k + N + 1\}$
    $\hat{\boldsymbol{z}} := \left(X_{\mathcal{S}_{ac}}^T X_{\mathcal{S}_{ac}} + \lambda^2 B_{\mathcal{S}_{ac}}\right)^{-1} X_{\mathcal{S}_{ac}}^T \boldsymbol{y}$
    $\boldsymbol{r} \leftarrow \boldsymbol{y} - X_{\mathcal{S}_{ac}} \hat{\boldsymbol{z}}$
  Return vector $\hat{\boldsymbol{z}} = \left(\hat{\boldsymbol{\alpha}}^T, \hat{c}, \hat{\boldsymbol{u}}^T\right)^T$ after k iterations.

by updating the Cholesky matrix instead. The scheme[1] was introduced in [16, 17] and has been found to be very efficient.

2.2. Convex relaxation - ℓ_1 minimization with ADMM

An alternative path, in order to achieve sparse optimization, while avoiding the non-convex formulation of (4), is to relax the ℓ_0 norm of the sparse outlier vector $\boldsymbol{u}$, with its closest convex one, i.e., the ℓ_1 norm. Thus, using the linear expression $f = \sum_{i=1}^{N} \alpha_i \kappa(\cdot, \boldsymbol{x}_i)$, we have the following Lagrangian formulation:

$$\min_{\boldsymbol{\alpha}, \boldsymbol{u}} \left\{ \|\boldsymbol{y} - K\boldsymbol{\alpha} - \boldsymbol{u}\|_2^2 + \lambda \boldsymbol{\alpha}^T K \boldsymbol{\alpha} + \mu ||\boldsymbol{u}||_1 \right\}, \quad (6)$$

for $\lambda > 0$ and $\mu > 0$.

The equation model in (6), was proposed in [9] and the method that was used is the alternating direction method of multipliers (ADMM), as demonstrated in algorithm 2 for $\boldsymbol{w} = \mathbf{1}$ and S the soft-thresholding operator, i.e., $S(z, \gamma) := \text{sign}(z) \max(0, |z| - \gamma)$.

Algorithm 2 (Weighted) Alternating directions solver: WAM

procedure WAM($K,\ \boldsymbol{y},\ \mu,\ \lambda,\ \boldsymbol{w}$)
  $\boldsymbol{u}^{(0)} \leftarrow \mathbf{0}$
  for $k = 1, 2, ...$ **do**
    $\boldsymbol{\alpha}^{(k)} \leftarrow [K + \lambda I_N]^{-1} \left(\boldsymbol{y} - \boldsymbol{u}^{(k-1)}\right)$
    $\boldsymbol{r} = \boldsymbol{y} - K\boldsymbol{\alpha}^{(k)},\ \boldsymbol{u}^{(k)} \leftarrow S\left(r_i, \frac{w_i \mu}{2}\right), i = 1, ...N$
  Return $\boldsymbol{\alpha}^{(k)}$ and $\boldsymbol{u}^{(k)}$ after k iterations.

The proposed scheme is more economic than the original ADMM method and could be further optimized by applying a Cholesky factorization (with cost $O(N^2)$ after the factorization) instead of an inversion, since matrix $[K + \lambda I_N]$ remains unchanged. Although the minimization task is now convex, the performance towards error reduction is limited, due to the relaxation from the ℓ_0 to the ℓ_1 norm. However, the authors have resorted to a refined and more efficient method, as originally proposed in [18], that attempts to solve

$$\min_{\boldsymbol{\alpha}, \boldsymbol{u}} \left\{ \|\boldsymbol{y} - K\boldsymbol{\alpha} - \boldsymbol{u}\|_2^2 + \lambda \boldsymbol{\alpha}^T K \boldsymbol{\alpha} + \mu \sum_{i=1}^{N} \log\left(|u_i| + \delta\right) \right\},$$

for $\delta > 0$ (in order to avoid numerical instability), using the linear approximation of the concave logarithmic function, via the use of the reweighted ℓ_1-norm technique. The scheme is summarized in algorithm 3. The refinement step of AM solver

Algorithm 3 Refined AM solver: RAM

procedure RAM($K,\ \boldsymbol{y},\ \mu,\ \lambda,\ \delta$)
  $\left[\boldsymbol{\alpha}^{(0)}, \boldsymbol{u}^{(0)}\right] \leftarrow \text{WAM}(K, \boldsymbol{y}, \mu, \lambda, \mathbf{1})$
  for $k = 1, 2$ **do**
    $w_i^{(k)} = (|u_i^{(k-1)}| + \delta)^{-1},\ i = 1, ..., N,$
    $\left[\boldsymbol{\alpha}^{(k)}, \boldsymbol{u}^{(k)}\right] \leftarrow \text{WAM}(K, \boldsymbol{y}, \mu, \lambda, \boldsymbol{w}^{(k)})$
  Return $\boldsymbol{\alpha}^{(2)}$ after 2 iterations.

(WAM solver with weights equal to 1) greatly improves the performance of the original ADMM. Moreover, it should be noted, that more than 2 iterations do not offer significant improvements on the performance of the method, since the initialization is already optimum. Furthermore, we should emphasize that the optimum parameters (λ_*, μ_*) to be used with RAM, are not identical to those of WAM with $\boldsymbol{w} = \mathbf{1}$. Thus, the convergence speed of the RAM scheme, may be greater than that of the simple AM. Finally, theoretical properties of the method prove that for small values of $\delta > 0$ method attempts to approximate the ℓ_0 norm of the sparse outlier vector $\boldsymbol{u}$.

2.3. Sparse Bayesian learning approach - RB-RVM

The Sparse Bayesian learning scheme, called Robust Bayesian-RVM (RB-RVM), was introduced in [8] and is a modification of the Relevance Vector Machine that has been presented in [19].

Here, assuming a linear representation including the bias term, the authors suggest the reformulation of (3), to $\boldsymbol{y} = \Psi \boldsymbol{\alpha_u} + \boldsymbol{\eta}$, where $\Psi = [\mathbf{1}\ K I_N]$, $\boldsymbol{\alpha_u} = [\boldsymbol{\alpha}_b^T, \boldsymbol{u}^T]^T$ and $\boldsymbol{\alpha}_b = [c, \alpha_1, ..., \alpha_N]^T$. Then, the joint posterior distribution of $\boldsymbol{\alpha}_b$ and $\boldsymbol{u}$ (assumed independent) is given by:

$$p(\boldsymbol{\alpha}_b, \boldsymbol{u}|\boldsymbol{y}) = \frac{p(\boldsymbol{\alpha}_b) p(\boldsymbol{u}) p(\boldsymbol{y}|\boldsymbol{\alpha}_b, \boldsymbol{u})}{p(\boldsymbol{y})},$$

given the observations $\boldsymbol{y}$ and the prior distributions on $\boldsymbol{\alpha}_b$ and $\boldsymbol{u}$. The inference procedure follows the steps of the classical RVM [19], taking into account that $p(\boldsymbol{y}|\boldsymbol{\alpha}_b, \boldsymbol{u}) = \mathcal{N}(\Psi \boldsymbol{\alpha_u}, \sigma^2 I_N)$, where σ^2 is the inlier Gaussian noise variance, and adopting some 'sparsity promoting' priors i.e.,

$$p(\boldsymbol{v}|\boldsymbol{h}) = \prod_{i=0}^{N} \mathcal{N}(v_i|0, h^{-1}), \quad (7)$$

for vectors $\boldsymbol{\alpha}_b$ and $\boldsymbol{u}$, with hyper-parameters $\boldsymbol{\beta} = [\beta_0, \beta_1, ..., \beta_N]^T$ and $\boldsymbol{\delta} = [\delta_0, \delta_1, ..., \delta_N]^T$, respectively. Since the maximization of $p(\boldsymbol{y}|\boldsymbol{\beta}, \boldsymbol{\delta}, \sigma^2)$ is performed by an EM algorithm, the parameters $\boldsymbol{\beta}_{MP}, \boldsymbol{\delta}_{MP}$ and σ_{MP}^2 are estimated and then used for computing the posterior covariance and mean given by

$$\Sigma = \left(\sigma^{-2} \Psi^T \Psi + A_{MP}\right)^{-1} \text{ and } m = \sigma^{-2} \Sigma \Psi^T \boldsymbol{y}, \quad (8)$$

[1] The Matlab code can be found at http://bouboulis.mysch.gr/kernels.html.

where $A_{MP} = diag(\beta_{MP0}, ..., \beta_{MPN}, \delta_{MP1}, ..., \delta_{MPN})$. Finally, prediction is accomplished, using the covariance and mean of the posterior distribution for the parameter part $\boldsymbol{\alpha}_b$ of $\boldsymbol{\alpha}_{\boldsymbol{u}}$, i.e., $\Sigma_{\boldsymbol{\alpha}_b} = \Sigma(1 : N+1, 1 : N+1)$ and $m_{\boldsymbol{\alpha}_b} = m(1 : N+1)$. As noted, the RB-RVM rationale is closely relate to the RVM formulation. The difference is that instead of inferring just the parameter vector $\boldsymbol{\alpha}_b$, the method infers the joint parameter-outlier vector $\boldsymbol{\alpha}_{\boldsymbol{u}}$ by replacing the matrix $[\mathbf{1}\ K]$ with matrix $\Psi = [\mathbf{1}\ K\ I_N]$ and the use of only the parameter part of the estimated $\boldsymbol{\alpha}_{\boldsymbol{u}}$ for prediction.

3. THEORETICAL GUARANTEES

The WAM and RAM methods are guaranteed to converge, since they both deal with a convex task. On the other hand, no theoretical results regarding the convergence have been established for the Bayesian approach. For both methods, no theoretical results concerning the performance have been proved. However, for the greedy method, properties regarding the convergence of the method as well as the recovery of the support pattern for the sparse outlier vector (for the case where only outliers exist in the noise), have been studied.

The following proposition guarantees that KGARD, will converge to a solution of minimum error.

Proposition 1 *The residual obtained at each iteration cycle of KGARD is strictly decreasing. Moreover, the residual eventually will drop below the predefined threshold ε.*

We should emphasize here, that cautious selection should be made on ε. If this threshold is predefined extremely small, the proposed procedure will continue and model extra noise samples as impulses (those originating from an inlier source), filling up the vector $\boldsymbol{u}$, which will no longer be sparse. Hence, sensible tuning of the parameter ε is of great importance for the method.

The second theorem, provides the conditions, under which, KGARD succeeds in recovering the support of the sparse outlier vector. The theorem has been derived for the case of outlier noise only. Results for the case where both inlier and outlier noise are present have also been derived, yet the associated conditions turn out not to be realistic and are not reported.

Theorem 1 *Let K be a full rank, square, real valued matrix. Suppose, that $\boldsymbol{y} = K\boldsymbol{\alpha}_0 + c_0\mathbf{1} + \boldsymbol{u}_0$, where $\boldsymbol{u}_0$ is a sparse outlier vector. It is guaranteed that the algorithm will recover the support of the sparse outlier vector, if the maximum singular value $\sigma_M(X_0)$, of matrix $X_0 = [K\ \mathbf{1}]$, satisfies:*

$$\sigma_M(X_0) < \lambda \sqrt{\frac{\min|u_0| - \lambda\sqrt{2}||\boldsymbol{\theta}_0||_2}{2||\boldsymbol{u}_0||_2 - \min|u_0| + \lambda\sqrt{2}||\boldsymbol{\theta}_0||_2}}, \quad (9)$$

where $\min|u_0|$ is the smallest absolute value of the outlier vector over the nonzero coordinates, $\lambda > 0$ is the regularization parameter for KGARD and $\boldsymbol{\theta}_0 = \begin{pmatrix} \boldsymbol{\alpha}_0 \\ c_0 \end{pmatrix}$.

4. COMPARISON OF THE METHODS

In the experimental section, we have tested and compared all methods towards estimation (MSE) and two sets of experiments have been performed. For the first test, the fraction of outliers is left to vary while keeping the variance of the Gaussian noise fixed. In the second test, we have reversed the process, i.e., we have varied the σ of the Gaussian noise keeping fixed the fraction of outliers. For all tests, the values of required parameters, were set after performing cross-validation steps and optimized to reach the best possible performance for each method (in terms of MSE).

The uncorrupted measurements were generated via $\boldsymbol{y}_0 = f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i \kappa(\boldsymbol{x}_i, \boldsymbol{x})$, for $N = 200$ points over the interval $[0, 1]$, using the Gaussian kernel with $\sigma = 0.1$. The coefficient vector $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_N]^T$ is a sparse vector with non-zeros at a percentage of $7.5\% - 12.5\%$ and values drawn from the Gaussian distribution $\mathcal{N}(0, 20^2)$. That way, only a few kernels participate in the representation of $\boldsymbol{y}_0$, which is often the case. For both tests, the (corrupted) data were generated via

$$\boldsymbol{y} = \boldsymbol{y}_0 + \boldsymbol{u_0} + \boldsymbol{\eta},$$

where $\boldsymbol{u}_0$ is the sparse outlier vector with values ± 40 and the inlier noise vector $\boldsymbol{\eta} \sim \mathcal{N}(0, \sigma^2)$. It should also be noted, that experiments have been performed with the use of various non-linear functions and results were very similar to the ones we present.

On the first set of experiments, we have tested the MSE obtained between the uncorrupted ($\boldsymbol{y}_0$) and estimated data ($\hat{\boldsymbol{y}}$) and averaged over 100 independent runs, with $\sigma = 4$ for the inlier noise and for various fractions of outliers. For the KGARD the parameters were set at $\lambda = 0.3$ and $\varepsilon = 15$. For the RAM, $\lambda = 0.1$ and the values of μ are given in table 1, along with the performance for each method.

On the second set of experiments, we have tested the MSE obtained between the uncorrupted ($\boldsymbol{y}_0$) and estimated data ($\hat{\boldsymbol{y}}$) and averaged over 100 independent runs, with a fixed fraction of outliers at 10% and for various values of the σ of the inlier noise, as demonstrated at table 2. Both experiments, lead to the conclusion that KGARD is more robust as it preserves the lowest MSE for all ranges of outlier fraction or variance of the inlier noise. Moreover, the RAM scheme also attains a notably low and improved MSE (compared to the simple AM). In fact, this actually indicates that the greedy method, i.e., KGARD, performs a better sparse approximation to the ℓ_0 norm of the sparse outlier vector, which cannot be outperformed by any relaxation-based method.

5. CONCLUSIONS

This work, addresses the task of non-linear regression in the context of RKHS modeling and the presence of outliers. The motivation of this work, is to compare the cutting edge

Outliers %	RB-RVM	KGARD	RAM
5%	3.17	1.16	1.25 ($\mu = 31$)
10%	3.73	1.21	1.34 ($\mu = 33$)
15%	3.97	1.25	1.35 ($\mu = 32$)
20%	4.08	1.31	1.46 ($\mu = 28$)
25%	4.35	1.49	1.65 ($\mu = 28$)

Table 1. Mean square error (MSE) for various fractions of outliers and σ of the inlier noise $\sigma = 4$.

σ	RB-RVM	KGARD	RAM
0	$9.21 \cdot 10^{-5}$	$2.91 \cdot 10^{-13}$	$1.05 \cdot 10^{-10}$
		($\varepsilon = 0.01, \lambda = 10^{-12}$)	($\mu = 0.005,\ \lambda = 10^{-6}$)
1	0.34	$9.87 \cdot 10^{-2}$	0.10
		($\varepsilon = 5, \lambda = 0.001$)	($\mu = 14,\ \lambda = 0.01$)
2	1.09	0.33	0.41
		($\varepsilon = 10, \lambda = 0.1$)	($\mu = 20,\ \lambda = 0.01$)
4	3.73	1.21	1.34
		($\varepsilon = 15, \lambda = 0.3$)	($\mu = 33,\ \lambda = 0.1$)
6	7.47	2.61	3.07
		($\varepsilon = 20, \lambda = 0.8$)	($\mu = 31,\ \lambda = 0.1$)
8	12.12	4.79	6.18
		($\varepsilon = 20, \lambda = 0.8$)	($\mu = 30,\ \lambda = 0.1$)

Table 2. Mean square error (MSE) for various values of the σ of the inlier Gaussian noise and a fraction of outliers at 10%.

methods available and measure the gains towards approximation/error reduction. We conclude, that the greedy method performs a better sparse approximation to the ℓ_0 norm of the sparse outlier vector, which cannot be outperformed by any relaxation-based method. Moreover, theoretical results, under certain assumptions, concerning the performance of the greedy-based approach, are reported.

REFERENCES

[1] Alex J. Smola and Bernhard Schölkopf, *Learning with Kernels*, The MIT Press, 2002.

[2] Sergios Theodoridis and Konstantinos Koutroumbas, *Pattern Recognition, 4th Edition*, Academic press, 2008.

[3] Sergios Theodoridis, *Machine Learning: A Bayesian and Optimization Perspective*, Academic Press, 2015.

[4] Hiroyuki Takeda, Sina Farsiu, and Peyman Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, 2007.

[5] Sylvain Durand and Jacques Froment, "Reconstruction of wavelet coefficients using total variation minimization," *SIAM Journal on Scientific computing*, vol. 24, no. 5, pp. 1754–1767, 2003.

[6] Patrick L Combettes and J-C Pesquet, "Image restoration subject to a total variation constraint," *IEEE Transactions on Image Processing*, vol. 13, no. 9, pp. 1213–1222, 2004.

[7] Pantelis Bouboulis, Konstantinos Slavakis, and Sergios Theodoridis, "Adaptive kernel-based image denoising employing semi-parametric regularization," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1465–1479, 2010.

[8] Kaushik Mitra, Ashok Veeraraghavan, and Rama Chellappa, "Robust RVM regression using sparse outlier model," in *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 1887–1894.

[9] Gonzalo Mateos and Georgios B. Giannakis, "Robust nonparametric regression via sparsity control with application to load curve data cleansing," *IEEE Transactions on Signal Processing*, vol. 60, no. 4, pp. 1571–1584, 2012.

[10] Peter J Huber, *Wiley Series in Probability and Mathematics Statistics*, Wiley Online Library, 1981.

[11] Ricardo A Maronna, R Douglas Martin, and Victor J Yohai, *Robust statistics*, J. Wiley, 2006.

[12] Peter J Rousseeuw and Annick M Leroy, *Robust regression and outlier detection*, vol. 589, John Wiley & Sons, 2005.

[13] Yagyensh Chandra Pati, Ramin Rezaiifar, and PS Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Signals, Systems and Computers, Conference Record of The Twenty-Seventh Asilomar Conference on*. IEEE, 1993, pp. 40–44.

[14] Joel A Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.

[15] Joel A Tropp and Anna C Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[16] George Papageorgiou, Pantelis Bouboulis, and Sergios Theodoridis, "Robust kernel-based regression using orthogonal matching pursuit," in *Machine Learning for Signal Processing (MLSP), 2013 IEEE International Workshop on*. IEEE, 2013, pp. 1–6.

[17] Pantelis Bouboulis, George Papageorgiou, and Sergios Theodoridis, "Robust image denoising in RKHS via orthogonal matching pursuit," in *Cognitive Information Processing (CIP), 2014 4th International Workshop on*. IEEE, 2014, pp. 1–6.

[18] Emmanuel J Candes, Michael B Wakin, and Stephen P Boyd, "Enhancing sparsity by reweighted 1 minimization," *Journal of Fourier analysis and applications*, vol. 14, no. 5-6, pp. 877–905, 2008.

[19] Michael E Tipping, "Sparse Bayesian learning and the relevance vector machine," *The journal of machine learning research*, vol. 1, pp. 211–244, 2001.