CSIAM Trans. Appl. Math.
doi: 10.4208/csiam-am.2020-0212

Vol. **2**, No. 1, pp. 81-107
March 2021

Identification of Corrupted Data via k-Means Clustering for Function Approximation

Jun Hou[1], Yeonjong Shin[2] and Dongbin Xiu[1,*]

[1] *Department of Mathematics, The Ohio State University, Columbus, OH 43210, USA.*
[2] *Division of Applied Mathematics, Brown University, Providence, RI 02912, USA.*

Received 2 September 2020; Accepted 24 November 2020

Abstract. In addition to measurement noises, real world data are often corrupted by unexpected internal or external errors. Corruption errors can be much larger than the standard noises and negatively affect data processing results. In this paper, we propose a method of identifying corrupted data in the context of function approximation. The method is a two-step procedure consisting of approximation stage and identification stage. In the approximation stage, we conduct straightforward function approximation to the entire data set for preliminary processing. In the identification stage, a clustering algorithm is applied to the processed data to identify the potentially corrupted data entries. In particular, we found k-means clustering algorithm to be highly effective. Our theoretical analysis reveal that under sufficient conditions the proposed method can exactly identify all corrupted data entries. Numerous examples are provided to verify our theoretical findings and demonstrate the effectiveness of the method.

AMS subject classifications: 42C05, 41A10, 65D15

Key words: Data corruption, function approximation, sparse approximation, k-means clustering.

1 Introduction

Real world data are never perfect—in addition to standard measurement noises, they are often corrupted by unexpected and uncontrollable internal or external errors. The causes of corruption include human processing errors, data transmission or storage errors, machine malfunction during data collection, etc. The resulting corrupted errors can be large in magnitude and do not follow certain statistical laws. The presence of data corruptions thus can significantly impact data analysis results in a negative manner.

*Corresponding author. *Email addresses:* hou.345@osu.edu (J. Hou), yeonjong_shin@brown.edu (Y. Shin), xiu.16@osu.edu (D. Xiu)

In this paper, we consider the problem of identifying data corruptions in the context of regression modeling (supervised learning). Our approach is motivated by *function approximation with corruptions* [8,30]. Let $f(x)$ be an unknown function defined in a bounded domain D, $x_i \in D$ be an input data and $f(x_i)$ be its corresponding clean output value for $i=1,\ldots,m$. We are interested in the case where the data vector is corrupted by unexpected external errors that may be caused by aforementioned reasons. That is, the available data vector is given by

$$y = f + e_s,$$

where $f = (f(x_1),\cdots,f(x_m))^T$ is the clean output vector (which may contain the standard noises), and $e_s \in \mathbb{R}^m$ is the corruption vector with sparsity s, which stands for the number of corrupted data entries. While the vector y is the available data vector, no information on e_s is available.

A general procedure of approximating functions can be described as a class of minimization problem. Given a set of basis $\{\phi_j\}_{j=1}^n$ in D, we consider an approximation in the form of $\widetilde{f}(x) = \sum_{j=1}^n c_j \phi_j(x)$. We are interested in the oversampled case, $m > n$. The standard approach seeks to find the coefficients $c = (c_1,\cdots,c_n)^T$ that minimize the errors, i.e.,

$$\min_c \|y - Ac\|, \quad \text{where } A = \big(a_{ij}\big) = \big(\phi_j(x_i)\big) \text{ and } y = (y_i).$$

We note that the available data y is contaminated by e_s and the clean output f is not available to us. The use of the vector 2-norm yields the well-known least squares (LSQ) method, whose literature is too large to mention here. In general, LSQ method is known to be robust when the corruption errors are relatively small (e.g. Gaussian noise). The use of the vector 1-norm yields the ℓ_1 minimization, which is called least absolute deviations (LAD) is also studied extensively in [1, 4, 6, 26, 27, 30, 31]. The LAD method is known to be robust against outliers and sparse corruptions [30]. In the spirit of seeking sparsity, one can also employ any sparse approximation techniques that include ℓ_{1-2} minimization [23, 33–35] (the difference between the 1-norm and the 2-norm), or ℓ_p minimization [10, 11, 32] for $0 < p < 1$. Although these methods are capable of producing accurate function approximation, *they can not detect corrupted data*, especially when the number s of corrupted data is unknown.

In this paper, we present an approach for identifying the corrupted data entries in a given measurement data vector without the knowledge of the number (s) of the corrupted data entries. We propose a two-step procedure that consists of approximation and identification stages. At the approximation stage, we conduct function approximation with the corrupted data and obtain a residual vector. At the identification stage, we apply a clustering algorithm to the residual vector to separate the residues into corrupted entries and clean entries. Specifically, we employ k-means clustering [3, 17, 24], a well-established clustering algorithm with a wide range of applications [5, 13, 21, 25]. We then provide theoretical results on the sufficient conditions under which the proposed approach can detect the corrupted data exactly (Theorem 4.1).

There exist other data cleaning techniques to detect (mostly) outliers in data. See, for example, overviews in [2, 18] and the references therein. In particular, the clustering based approach [20, 36] is the most relevant one to the present work. However, most of the approaches are not designed for regression-type data (supervised learning) and often require probabilistic assumptions on the corruption. A distinct and novel feature of our method is the explicit use of function approximation algorithm in conjunction with the clustering algorithm. This allows us to obtain accurate regression results in the presence of data corruption, and more importantly, without making any probabilistic assumptions on the corruption.

This rest of this paper is organized as follows. After the basic problem setup and preliminary in Section 2, we present the proposed method in Section 3. Theoretical guarantees are presented in Section 4. To demonstrate the effectiveness of our proposed method, a set of numerical examples are presented in Section 5.

2 Setup and preliminary

Let $f(x)$ be an unknown target function defined in a domain D in $\mathbb{R}^d$. Let us denote $\{1,2,\cdots,m\}$ by $[m]$ for any $m\in\mathbb{N}$. Given a set $\{\phi_i(x)\}_{i=1}^n$ of basis functions, we write an approximation to $f(x)$ as

$$f(x)\approx\tilde{f}(x)=\sum_{i=1}^{n}c_i\phi_i(x), \tag{2.1}$$

where $c=(c_1,\cdots,c_n)^T$ is the coefficient vector. Let

$$A=(a_{ij}),\qquad a_{ij}=\phi_j(x_i),\quad i\in[m],j\in[n], \tag{2.2}$$

be the model matrix and $f=(f(x_1),\cdots,f(x_m))^T$ be the function value vector of $f(x)$ at locations $x_1,\ldots,x_m$. Let

$$y=f+e_s \tag{2.3}$$

be the actual data vector, where e_s is external corruption error vector with sparsity $s\geq 0$. The sparsity s is defined as the cardinality of the support of e_s. That is, let Λ be the support of e_s,

$$\Lambda=\{i\,|\,(e_s)_i\neq 0\}=\text{supp}(e_s),\quad \text{such that}\quad |\Lambda|=s. \tag{2.4}$$

Throughout this paper we consider the overdetermined case, i.e., $m>n$, and assume the model matrix A is full rank. Our goal is to identify the locations of the corruptions, i.e., Λ, without the prior knowledge of $s=|\Lambda|$.

We remark that the function value vector f may contain standard (small) random noises (e.g. Gaussian). These standard noises are typically filtered out by a chosen regression algorithm. We do not aim to detect such small noises, and their presence in f does not change our results.

2.1 Function approximation with corrupted data

We briefly review function approximation techniques under corrupted data. We emphasize that only the data vector y is available and the clean data f is not available.

We provide a unified approach that describes many approximation methods including LSQ, LAD, ℓ_{1-2} and ℓ_p minimization. We seek to find a coefficient vector that minimizes the error:

$$\min_c \|y - Ac\|. \tag{2.5}$$

The underlying principle of the unified framework lies on the following equivalence. Let $F \in \mathbb{R}^{(m-n)\times m}$ be the kernel of the model matrix A, which can be explicitly constructed via QR factorization, see [30]. Then, the problem of (2.5) is equivalent to

$$\min_g \|g\| \quad \text{subject to} \quad Fg = Fy. \tag{2.6}$$

When the vector 1-norm is used, the equivalence has been established in [8]. Since $g = y - Ac$ is equivalent to $Fg = Fy$, the equivalence remains true in more general choices of metric, e.g., ℓ_{1-2}, ℓ_p where $0 < p < 1$. Also, given a solution g^* of (2.6), since $y - g^*$ is in the range of A, one can obtain its corresponding solution c^* of (2.5) by solving $Ac = y - g^*$. When vector 1-norm is employed, this becomes the least absolve deviation (LAD) method, also known as ℓ_1 minimization. Its solver is well established, cf., [7].

If the error is measured by the vector 2-norm, this becomes the well known least squares (LSQ) problem, and the solution to (2.5) is

$$c^* = A^\dagger y,$$

where $A^\dagger$ is the Moore-Penrose pseudoinverse of A.

If the error is measured by the difference between the vector 1-norm and 2-norm, (2.6) becomes ℓ_{1-2} minimization, which has been studied in [14,23,34,35]. It can also be solved efficiently using methods such as difference of convex functions (DCA) [35].

If the error is measured by vector p-norm with $0 < p < 1$, the problems becomes the ℓ_p minimization. See, for example, [9,28,29].

The work of [30] rigorously proved that the ℓ_1-minimization solution with corrupted data are close to the regression results with uncorrupted data f, thus effectively eliminating the corruption errors. Further details can be found in [30]. The ℓ_{1-2} and ℓ_p minimization show good performance of eliminating the effect of corruption errors in function approximation in the underdetermined system in a number of works [28,34,35], although their proofs remain lacking at the moment. We conduct some experiments to evaluate its performance in the overdetermined system.

2.2 k-means clustering

The k-means clustering [24] is one of the most common methods for clustering problems. Given a data vector $r = (r_1, \ldots, r_m)$, the k-means clustering method classifies r into $k > 1$

groups according to

$$\min_{\mathcal{I}_i, i=1,\cdots,k} \frac{1}{|\mathcal{I}_i|} \sum_{j\in\mathcal{I}_i} (r_i - \mu_{\mathcal{I}_i})^2, \tag{2.7}$$

where

$$\mu_{\mathcal{I}_i} = \frac{1}{|\mathcal{I}_i|} \sum_{j\in\mathcal{I}_i} r_i, \qquad [m] = \bigcup_{i=1}^{k} \mathcal{I}_i, \qquad \mathcal{I}_i \cap \mathcal{I}_j = \emptyset, \quad \forall i \neq j. \tag{2.8}$$

Since solving the k-means clustering problem (3.1) is NP-hard, a heuristic local search method, termed Lloyd's algorithm [22], is commonly used in practice.

3 Proposed method

We present a two-step procedure to identify the corrupted data entries in a given data vector. We first apply a straightforward function approximation scheme, as described in Section 2.1, to the data vector and obtain not only the coefficient vector c^* but also a residual vector $r = |y - Ac^*|$. We then apply the k-means clustering algorithm to the residual vector r to identify the index set of the corruption vector e_s. Here is an outline of the method.

- Step 1: Approximation. Given a set of data $\{(x_i, y_i)\}_{i=1}^m$ that contain corrupted entries, select a set of basis $\{\phi_i(x)\}_{i=1}^n$ and compute the solution c^* to

  $$\min_c \|y - Ac\|,$$

  where the norm is determined by the selected approximation method. Evaluate the residual vector $r = |y - Ac^*|$.

- Step 2: Identification. Apply k-means clustering to the residual vector r with $k = 2$ and obtain

  $$\mathcal{I}^* = \underset{\mathcal{I}\subset[m], |\mathcal{I}|\leq\lfloor\frac{m}{2}\rfloor}{\operatorname{argmin}} \frac{1}{|\mathcal{I}|} \sum_{i\in\mathcal{I}} (r_i - \mu_{\mathcal{I}})^2 + \frac{1}{|\mathcal{I}^{\complement}|} \sum_{i\in\mathcal{I}^{\complement}} (r_i - \mu_{\mathcal{I}^{\complement}})^2, \tag{3.1}$$

  where $\mathcal{I}^{\complement}$ is the complement of $\mathcal{I}$. The set $\mathcal{I}^*$ is the index set of corrupted data in y, i.e., $\mathcal{I}^* = \Lambda$ where Λ is defined in (2.4).

Once the corrupted data entries are identified, one can remove them from the data set and re-run the function approximation method for improved results.

The choice of the approximation method in Step 1 shall have significant impact on the overall performance of the proposed method. Based on the results of [30], we advocate the use of sparsity promoting methods such as LAD, ℓ_{1-2} or ℓ_p with $0 < p < 1$.

4 Theoretical results

In this section we provide theoretical analysis of the proposed method. Let us first define the following quantities that measure the magnitude of the corruptions and the approximation errors from the first stage:

$$\begin{aligned} a_{\min} &= \min_{1\le i\le m} |(f-Ac^*)_i|, \qquad |e|_{\min} = \min_{i\in\Lambda} |(e_s)_i|, \\ a_{\max} &= \max_{1\le i\le m} |(f-Ac^*)_i|, \qquad |e|_{\max} = \max_{i\in\Lambda} |(e_s)_i|, \end{aligned} \tag{4.1}$$

where Λ is the support of e_s in (2.4). We make the following assumptions.

Assumption 4.1. Suppose $a_{\max} < |e|_{\min}$, i.e., the external corruption errors are larger than the approximation errors. And the number of corrupted data is smaller than the half of the total number of data, i.e., $0 < s \le \lfloor \frac{m}{2} \rfloor$.

We note that under Assumption 4.1, if the number s of corrupted data were known, one could identify the corrupted data entries by simply choosing the first s largest components from the residual vector. In this paper, however, we do not know the sparsity s a priori.

We are now in a position to present our main result that provides general sufficient conditions that guarantee the exact identification of the corrupted entries.

Theorem 4.1. *Suppose Assumption* 4.1 *holds and the external errors satisfy*

$$\frac{\|e\|_1}{s} > \gamma a_{\max}, \qquad \text{where} \quad \gamma = 2 + \frac{m}{2\sqrt{s(m-s)}}. \tag{4.2}$$

For any subset $\Omega \subset [m]$ of $u = t+l$ elements, let $\Omega_t = \Lambda \cap \Omega$, where $|\Omega_t| = t$, and $\Omega_l = \Lambda^{\complement} \cap \Omega$, where $|\Omega_l| = l$. Suppose the followings are satisfied: for all $0 < u = l+t \le \lfloor \frac{m}{2} \rfloor$, and $t \neq 0$,

$$\begin{aligned} \sum_{i\in\Omega_t^{\complement}} |(e_s)_i| &\ge \alpha_u \sum_{i\in\Omega_t} |(e_s)_i| + \mathcal{C}_{t,l}(c_a) a_{\max}, \qquad \text{if} \quad \mu_\Omega < \mu_{\Lambda^{\complement}}, \\ \sum_{i\in\Omega_t^{\complement}} |(e_s)_i| &\ge -\tilde{\alpha}_u \sum_{i\in\Omega_t} |(e_s)_i| + \mathcal{H}_{t,l}(c_a) a_{\max}, \quad \text{if} \quad \mu_\Omega \ge \mu_{\Lambda^{\complement}}, \end{aligned} \tag{4.3}$$

where $\frac{a_{\max}-a_{\min}}{a_{\max}} \le c_a \le 1$, μ_I is defined on (2.8) *for some index set $I \subset [m]$,*

$$\alpha_x = \frac{m-x}{m-s-x}\left(-1+\sqrt{\frac{s(m-s)}{x(m-x)}}\right), \quad \tilde{\alpha}_x = \frac{m-x}{m-s-x}\left(1+\sqrt{\frac{s(m-s)}{x(m-x)}}\right), \tag{4.4}$$

and

$$\mathcal{C}_{t,l}(c_a) = \begin{cases} 2(s-t), & \text{if } 0 < u < s, \\ 2(s-t) + 2t|\alpha_u| + \left[l\sqrt{\frac{s(m-u)}{u(m-s)}} - (s-t) \right] c_a, & \text{if } s \le u \le \lfloor \frac{m}{2} \rfloor, \end{cases}$$

$$\mathcal{H}_{t,l}(c_a) = 2(s-t+t\tilde{\alpha}_u) + l\sqrt{\frac{s(m-u)}{u(m-s)}} c_a.$$

Then the solution set $\mathcal{I}^*$ of the k-means clustering algorithm (3.1) *is equal to* Λ. *That is, the k-means clustering algorithm identifies the corrupted data entries exactly.*

Proof. The proof can be found in Appendix A. □

The condition of (4.2) can be understood that the external errors are non-trivial. The non-triviality can be viewed in two ways. One way is that the corruptions are larger than the approximation errors. The other is that the approximation performed in the first stage produces an accurate solution that has sufficiently small errors compared to the corruptions.

Theorem 4.1 provides general sufficient conditions for the perfect identification, however, the conditions of (4.3) are not easy to be checked. Thereby, we provide a simplified version as follows. For notational convenience, we set $0\cdot\alpha_0 := 0$.

Theorem 4.2. *Under the same conditions of* (4.2), *suppose the external errors satisfy*

$$|e|_{\min} \geq 2a_{\max} + \mathcal{M}, \tag{4.5}$$

where

$$\mathcal{M} = \max\left\{ (s-1)\alpha_{s-1}|e|_{\max},\ g\Big(\lfloor \frac{m}{2} \rfloor\Big) c_a \cdot a_{\max},\ \frac{m}{2\sqrt{s(m-s)}} c_a \cdot a_{\max} \right\},$$

c_a, α_u *are defined in* (4.4), *and*

$$g(u) = \frac{(u-s)(m-s-u)}{\sqrt{s(m-s)u(m-u)} - s(m-s)}. \tag{4.6}$$

Then the k-means clustering exactly identifies the corrupted data.

Proof. The proof can be found in Appendix B. □

It is worth noting that when $s = 1$, since $0\cdot\alpha_0 = 0$, $\mathcal{M}$ does not depend on $|e|_{\max}$. Therefore, the single location can be exactly identified as long as the external error is sufficiently larger than the approximation errors. Also, when the approximation scheme interpolates one of the uncorrupted data $y_{[m]\setminus\Lambda}$, since $a_{\min} = \min_{1\leq i\leq m} |(f - Ac^*)_i| = 0$, we have $c_a = 1$.

The condition of (4.5) implies $|e|_{\min} \geq 2a_{\max} + (s-1)\alpha_{s-1}|e|_{\max}$. This shows that the smallest magnitude of the corruptions should not be too small compared to the largest one. Roughly speaking, as long as the corruption vector e_s has a large (sample) mean and a small (sample) variance, relative to the approximation errors, the proposed method can successfully identify the corrupted data out of y.

The following result is a special case when all corruptions have the same magnitude.

Corollary 4.1. *Suppose the external errors have a constant magnitude, i.e.,* $|(e_s)_i| = |e|$ *for all* $i \in \Lambda$*, and the condition of* (4.2) *is satisfied. If the corruption errors satisfy* $|e| \geq \tilde{\gamma}|a|_{\max}$*, where*

$$\tilde{\gamma} = \max\left\{2 + c_a \max\left\{g\left(\lfloor \frac{m}{2} \rfloor\right), \frac{m}{2\sqrt{s(m-s)}}\right\}, \frac{2}{1-(s-1)\alpha_{s-1}}\right\}, \tag{4.7}$$

and $g(u)$ *is defined in* (4.6)*, the k-means clustering exactly identifies the corrupted data.*

Proof. It can be checked that $(s-1)\alpha_{s-1} < 1$ as follow:

$$\begin{aligned}
& (s-1)\alpha_{s-1} < 1 \\
\iff & \frac{(s-1)(m-s+1)}{m-2s+1}\sqrt{\frac{s(m-s)}{(s-1)(m-s+1)}} < 1 + \frac{(s-1)(m-s+1)}{m-2s+1} \\
\iff & \sqrt{s(s-1)(m-s+1)(m-s)} < s(m-s) \\
\iff & (s-1)(m-s+1) < s(m-s) \\
\iff & 2s < m+1.
\end{aligned}$$

Then the proof immediately follows from Theorem 4.2 as $e_{\max} = e_{\min} = e$. □

5 Numerical examples

We provide several numerical examples to demonstrate the performance of the proposed method and verify our theoretical findings.

For the approximation stage of our two-stage method, we employ four approximation schemes to demonstrate the performances. These include ℓ_{1-2}, ℓ_p with $p = 0.5$, LAD and LSQ minimization. To solve the ℓ_1-minimization problem (LAD), we employ ℓ_1-Magic package [7]. To solve the ℓ_{1-2} minimization problem, difference-of-convex algorithm (DCA) [35] is adopted. To solve the ℓ_p minimization with $p = 0.5$. we utilize iteratively reweighed least squares [28]. Since LAD method can effectively eliminate the effect of the corrupted data [30], we conduct experiments to validate ℓ_{1-2} and ℓ_p $(0 < p < 1)$ minimizations can also perform similarly. For the identification stage of our method, we apply the *k*-means clustering (3.1). Specifically, we employ `k-means++` [3], a widely used algorithm in many applications [5, 13, 21, 25].

5.1 Setup of the numerical tests

We consider several test functions from [16], which have been widely used for multidimensional function integration and approximation tests. More specifically, we choose the

following functions:

$$\begin{aligned}
f_1 &= \exp\left(-\sum_{i=1}^{d} c_i^2\left(\frac{x_i+1}{2}-w_i\right)^2\right); \quad \text{(GAUSSIAN)};\\
f_2 &= \exp\left(-\sum_{i=1}^{d} c_i\left|\frac{x_i+1}{2}-w_i\right|\right); \quad \text{(CONTINUOUS)};\\
f_3 &= \left(1+\sum_{i=1}^{d} c_i\frac{(x_i+1)}{2}\right)^{-(d+1)}, \quad c_i=\frac{1}{i^2}; \quad \text{(CORNER PEAK)};\\
f_4 &= \prod_{i=1}^{d}\left(c_i^{-2}+\left(\frac{x_i+1}{2}-w_i\right)^2\right)^{-1}; \quad \text{(PRODUCT PEAK)},
\end{aligned} \tag{5.1}$$

where $c=(c_1,\cdots,c_d)$ are parameters controlling the difficulty of the functions, and $w=(w_1,\cdots,w_d)$ are shifting parameters. Also, for $d=2$, we consider the Bird function [19], Franke function [15] and Matlab Peaks function.

$$\begin{aligned}
f_5 &= (x_1-x_2)^2+\sin(x_1)e^{(1-\cos(x_2))^2}+\cos(x_2)e^{(1-\sin(x_1))^2}; \quad \text{(BIRD)};\\
f_6 &= 0.75\exp\left((9x_1-2)^2/4-(9x_2-2)^2/4\right)\\
&\quad +0.75\exp\left((9x_1+1)^2/49-(9x_2+1)/10\right)\\
&\quad +0.5\exp\left((9x_1-7)^2/4-(9x_2-3)^2/4\right)\\
&\quad -0.2\exp\left(-(9x_1-4)^2-(9x_2-7)^2\right); \quad \text{(FRANKE)};\\
f_7 &= 3(1-x)^2\exp\left(-x^2-(y+1)^2\right)\\
&\quad -10\left(x/5-x^3-y^5\right)\exp\left(-x^2-y^2\right)\\
&\quad -1/3\exp\left(-(x+1)^2-y^2\right); \quad \text{(PEAKS)}.
\end{aligned} \tag{5.2}$$

We generate the corrupted data y by adding external errors e_s to the samples of the test functions f, i.e., $y=f+e_s$. The sample points are uniformly drawn from $[-1,1]^d$. The corruptions are drawn from a Gaussian distribution $\mathcal{N}(\mu,\sigma^2)$.

To measure how successfully the proposed method identifies the corrupted data, we introduce identification success rate (ISR), which is defined by the ratio of the number of the correctly identified corrupted data to the total number of corrupted data. More precisely,

$$\text{ISR} := \frac{|\mathcal{I}^*\cap\Lambda|}{|\Lambda|},$$

where $\Lambda=\text{supp}(e_s)$ and $\mathcal{I}^*$ is the solution to (3.1). We report the results by different choices of methods, in conjunction with ℓ_{1-2}, ℓ_p, LAD and LSQ minimization in approximation stage, and label them as 'k-ℓ_{1-2}', 'k-ℓ_p', 'k-LAD' and 'k-LSQ', respectively.

Once some (potentially corrupted) data $y_{\mathcal{I}^*}$ are identified by the proposed method, one can exclude these from the original data y and re-apply the approximation scheme on the filtered data set $y_{[m]\setminus\mathcal{I}^*}$ to obtain improved approximation results. These improved results will be compared against the results obtained by naive use of the ℓ_{1-2}, ℓ_p, LAD and LSQ to the original corrupted data y, labelled as 'ℓ_{1-2}', 'ℓ_p', 'LAD' and 'LSQ', respectively. For reference, we also present the results by LSQ on uncorrupted perfect data f, denoted as 'p-LSQ' with "p" stands for *perfect* data. The numerical approximation errors are computed in vector 2-norm on a fixed set of points, different from the sampling points used in the approximation. In all tests, we report results averaged over 50 independent simulations.

Different basis functions in the approximations are considered. We present the results by Legendre polynomials for polynomial regression and by radial basis functions for non-polynomial regression. Different dimensions d are also considered. All the methods behave quite similarly in different dimensions. Therefore, we present a set of selected results to cover all combinations of the tests.

5.2 Four-dimensional examples: Legendre polynomial basis

First, the results in four dimensions $d=4$ using normalized Legendre polynomials are shown. We choose the approximation space to be the total degree polynomial space $\mathbb{P}_k$ of degree up to k, whose dimensionality is

$$n=n_k=\dim\mathbb{P}_k=\binom{d+k}{d}. \tag{5.3}$$

The number of samples m and the number of corruptions s are set to be $m\sim n\log n$ and $s=\alpha\cdot m$, respectively, where α is the corruption rate. It is known in [12] that it is necessary to choose the number of samples by $m\sim n\log n$ in order to obtain stable LSQ polynomial regression results. In our tests, we set $m=2\lceil n\log n\rceil$ and $s=\max\{1,\lfloor\alpha m\rfloor\}$.

In Fig. 1, we show the ISR results by 'k-ℓ_{1-2}', 'k-ℓ_p', 'k-LAD' and 'k-LSQ' for f_1, with respect to the increasing polynomial degree from 1 to 8. The rate of corruption is $\alpha=0.05$, i.e., 5% of the samples are corrupted. At the polynomial degree $k=1$, since $s=1$, the IRS is either 0 or 1. The left of Fig. 1 shows the results obtained with corruption errors setting as a constant 10, representing rather large non-probabilistic errors. The minimum and the maximum ISRs among 50 independent simulations are also presented as dashed lines. We observe that the 'k-ℓ_{1-2}', 'k-ℓ_p' and 'k-LAD' perfectly identify the corrupted data at all polynomial degrees in all 50 independent simulations. The 'k-LSQ' identifies (on average) more than 95% of the corrupted data at all degrees. Even in the worst case, the minimum ISRs of 'k-LSQ' are over 85%, except for the polynomial degrees of 2. This indicates that ℓ_{1-2}, ℓ_p and LAD are preferred choices in the approximation stage over LSQ in this case. On the right of Fig. 1, the results obtained with corruption errors generated as a constant of 0.1 are shown. These are relatively small corruption errors that may blend into the approximation errors. We observe that the ISRs by all methods notably drop,

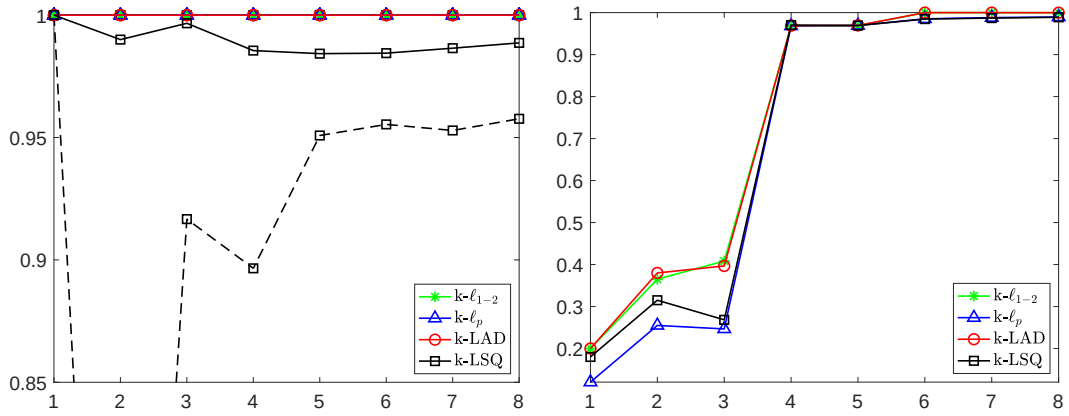

Figure 1: Identification success rates for f_1 with $c_i=2$ and $w_i=0.5$ are shown with respect to the polynomial degree from 1 to 8 at $d=4$. In the approximation stage, the approximation coefficients are obtained by ℓ_{1-2}, ℓ_p, LAD and LSQ on $m=2n\log n$ sample points. The corruption rate is set to be $\alpha=0.05$. The corruption errors are (left) $(e_s)_\Lambda=10$ and (right) $(e_s)_\Lambda=0.1$.

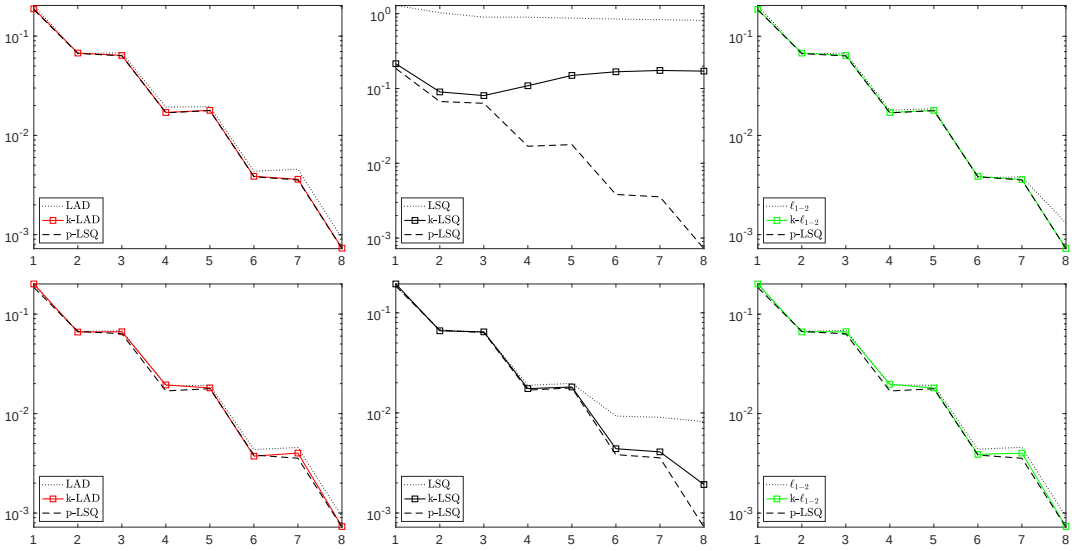

Figure 2: Approximation errors for f_1 with $c_i=2$ and $w_i=0.5$ are shown with respect to the polynomial degree from 1 to 8 at $d=4$. The corruption errors are (top) $(e_s)_\Lambda=10$ and (bottom) $(e_s)_\Lambda=0.1$. On the left, the results of k-LAD are shown. On the middle and the right, the results of k-LSQ and k-ℓ_{1-2} are shown, respectively.

especially at the lower degrees of $k=1,2,3$. However, at the higher polynomial degrees when numerical approximation errors become smaller, the averaged ISRs by all methods are above 95%. More importantly, perfect identifications are achieved by 'k-ℓ_{1-2}' and 'k-LAD' at $k=6,7,8$.

In Fig. 2, we compare the approximation errors for f_1 by k-LAD k-LSQ and k-ℓ_{1-2}, where p-LSQ is used for reference. The corruption errors are $(e_s)_\Lambda=10$ on the top and $(e_s)_\Lambda=0.1$ on the bottom. On the left, the results by LAD and k-LAD methods are shown.

Table 1: The cardinality of the polynomial space n at dimension $d=4$ along with the number of samples m and the number of corruptions s at the corruption rate of $\alpha=0.05$. The constant $\tilde{\gamma}$ of (4.7) is also shown along with the maximum of the maximum approximation errors by LAD of 50 independent simulations.

$d=4$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$	$k=8$
n	5	15	35	70	126	210	330	495
m	18	82	250	596	1,220	2,246	3,828	6,144
s; ($\alpha=0.05$)	1	4	12	29	61	112	191	307
$(s-1)\alpha_{s-1}$	0	0.47	0.49	0.50	0.50	0.50	0.50	0.50
$\tilde{\gamma}$ of (4.7)	5.18	5.32	5.33	5.32	5.29	5.30	5.30	5.30
$a_{\max}$ of LAD	0.97	0.35	0.41	0.26	0.65	0.16	0.16	0.07

In the middle and right, the results by LSQ methods and ℓ_{1-2} are shown, respectively. Since the results by ℓ_p and k-ℓ_p are almost identical to those by LAD and k-LAD, we decide not to present. As expected from [30], we clearly observe that LADs produce a lot smaller approximation errors than those by LSQs. Also, it can be seen that all results produced by using LAD methods are almost identical with those by p-LSQ and compared with LAD, k-LAD is always slightly better or equally good. On the right, we compared ℓ_{1-2} and k-ℓ_{1-2}, the performance of ℓ_{1-2} is similar to that of LAD. In the middle, we observe that k-LSQ performs better than or at least similarly to the standard LSQ using the corrupted data. It can be seen that on the right where k-LSQ outperforms LSQ, the ISRs by 'k-LSQ' are at least 90% on average. On the bottom where the performances of 'k-LSQ' and LSQ are similar, all methods produce very similar results. This is mainly due to the small magnitude of the external errors.

On Table 1, we compute the constant $\tilde{\gamma}$ of (4.7) to justify the results of 100% ISRs of 'k-LAD' at $(e_s)_\Lambda=10$. It can be checked that the assumption of Corollary 4.1 is

$$10=|e|_{\min}\geq 5.35|a|_{\max},$$

and it is satisfied in all cases. Therefore, by Corollary 4.1, the k-LAD should exactly identify the corrupted data at all 50 independent simulations and the left of Fig. 1 verifies it.

Similar behaviors are observed in other test functions of (5.1). In Fig. 3, we present the ISRs for f_2, f_3 and f_4 with respect to the increasing polynomial degree from 1 to 8. Here, the external errors are chosen to be rather large; $(e_s)_\Lambda\sim\mathcal{N}(10,1)$ for both f_2 and f_3, and $(e_s)_\Lambda\sim\mathcal{N}(10^5,10^4)$ for f_4. We can observe that the k-ℓ_{1-2}, k-ℓ_p and k-LAD perfectly identify all locations of corrupted data at all polynomial degrees, in all 50 independent simulations, and at all test functions. Although the k-LSQ can not achieve the perfect identifications, it identifies (on average) more than 97% of the corrupted data.

On the top of Fig. 4, we show the ISRs for f_2,f_3,f_4 where the external errors are chosen to be $(e_s)_\Lambda\sim\mathcal{N}(10,20^2)$ for f_2, $(e_s)_\Lambda\sim\mathcal{N}(0,10^2)$ for f_3, and $(e_s)_\Lambda\sim\mathcal{N}(10,1)$ for f_4. We observe that the ISRs by all methods notably drop. Our theoretical analysis indicates that the perfect identification can be achieved as long as the condition of Theorem 4.2 is

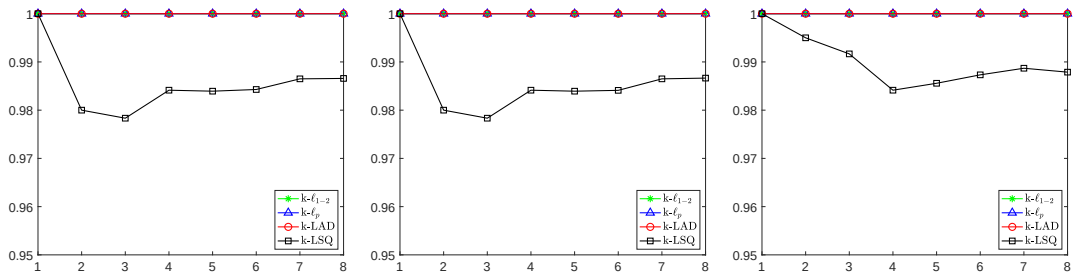

Figure 3: The identification success rates (ISRs) with respect to the polynomial degree from 1 to 8 at $d=4$ are shown. On the left, the results for f_2 with $c_i=i$, $w_i=0.5$ at $(e_s)_\Lambda \sim \mathcal{N}(10,1)$ are shown. On the middle, the results for f_3 at $(e_s)_\Lambda \sim \mathcal{N}(10,1)$ are shown. On the right, the results for f_4 with $c_i=d$, $w_i=0$ at $(e_s)_\Lambda \sim \mathcal{N}(10^5,10^4)$ are shown.

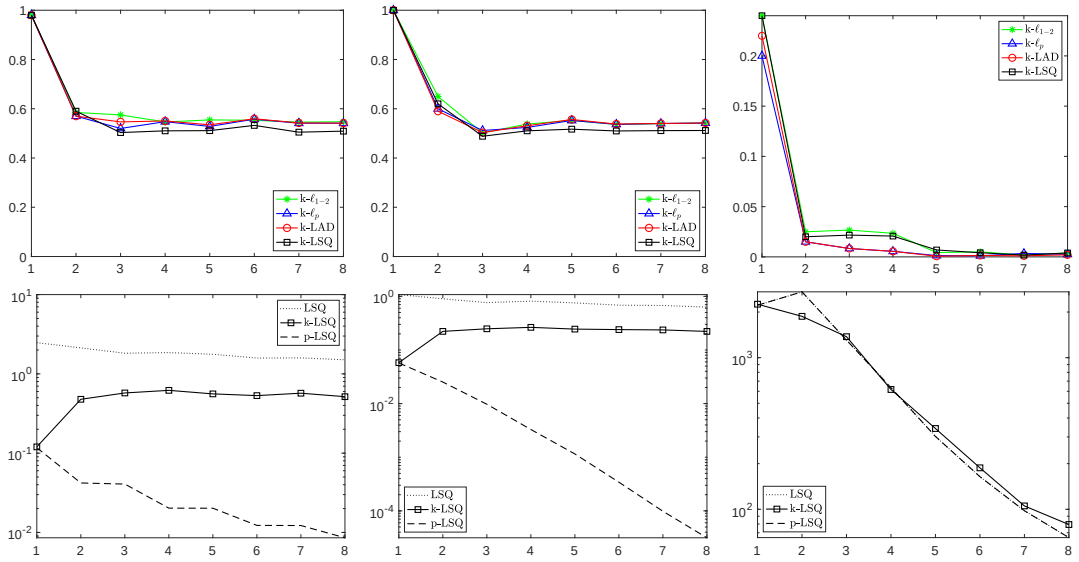

Figure 4: Identification success rates and approximation errors are shown with respect to the polynomial degree from 1 to 8 at $d=4$ on the top and on the bottom, respectively. On the bottom, we compare LSQ, k-LSQ, and p-LSQ. On the left, the results for f_2 with $c_i=i$, $w_i=0.5$ at $(e_s)_\Lambda \sim \mathcal{N}(10,20^2)$ are shown. On the middle, the results for f_3 at $(e_s)_\Lambda \sim \mathcal{N}(0,10^2)$ are shown. On the right, the results for f_4 with $c_i=d$, $w_i=0$ at $(e_s)_\Lambda \sim \mathcal{N}(10,1)$ are shown.

satisfied. Especially,

$$
\begin{aligned}
&|e|_{\min} \geq 2a_{\max} + (s-1)\alpha_{s-1}|e|_{\max}, \qquad a_{\max} = \max_{1\leq i\leq m} |(f-Ac^*)_i|, \\
&|e|_{\min} \geq 2a_{\max} + \max\left\{ g\left(\frac{m}{2}\right), \frac{m}{2\sqrt{s(m-s)}} \right\} a_{\max},
\end{aligned}
\tag{5.4}
$$

where $g(x)$ is defined in Theorem 4.2. These conditions indicate: (a) the minimum magnitude of the external errors should not be too small compared to the maximum magnitude

of the external errors; and, (b) the minimum magnitude of the external errors should be sufficiently larger than the approximation errors. In the view of (a) and (b), Fig. 4 can well be explained. On the top left and middle, it can be seen that k-ℓ_{1-2}, k-ℓ_p, k-LAD and k-LSQ achieve the perfect identifications at the polynomial degree 1 which results in $s=1$ at $\alpha=0.05$. In this case, condition (a) does not apply, and only condition (b) is needed for the perfect identifications. At the polynomial degrees $k>1$, since $(e_s)_\Lambda \sim \mathcal{N}(10,20^2)$ for f_2, and $(e_s)_\Lambda \sim \mathcal{N}(0,10^2)$ for f_3, (a) is very unlikely to be satisfied. In the right of Fig. 4, the proposed methods barely identify the corrupted data at all polynomial degrees. In this case, since $(e_s) \sim \mathcal{N}(10,1)$, (a) is very likely to be satisfied. However, since the target function f_4 here has the magnitude of $\mathcal{O}(10^5)$, with a peak value $2^{16}=65{,}536$, the corruption errors are smaller than the approximation errors. Thus (b) is not satisfied.

On the bottom of Fig. 4, the approximation results for (left) f_2, (middle) f_3, (right) f_4 by all LSQs are shown, respectively. We can observe that although k-LSQ solutions for f_2 and f_3 are still contaminated by the external errors, k-LSQ produces smaller approximation errors than those by the standard LSQ with the corrupted data. On the right, all LSQ methods produce almost identical results. Again, this is because the external errors are too small to be detected in this case.

5.3 Two-dimensional examples: Radial basis functions

Here we present the results by radial basis functions. In particular, we consider the Gaussian radial basis function,

$$\phi_i(x)=\exp(-\epsilon^2\|x-z_i\|_2^2), \tag{5.5}$$

centered at the points z_i with parameter $\epsilon>0$. Then center z_i are chosen to be tensor equidistance points.

In Fig. 5, the results using $\epsilon=8.6$ for the Bird function (5.2) are shown with respect to the number of basis functions. The number of data is set to be 1800 and the corruption rates are set to be $\alpha=0.1$. That is, 10% of data is corrupted, i.e., $s=180$. The ISRs and approximation errors for the bird function are shown on the top and bottom in Fig. 5, respectively. On the left, the approximation errors and ISRs on $e_s \sim \mathcal{N}(10,1)$, rather big errors, are shown with respect to increasing number of radial basis functions. All methods identify over 80% corrupted data at all 50 independent simulations when the number of basis is 49, 64, 81 and 100. It can be seen that as the number of basis increases the average ISRs increase and the approximation errors decrease. On the right of Fig. 5, the approximation errors and the ISRs on $e_s \sim \mathcal{N}(1,1)$, rather small errors, are shown. Similar to the previous examples, the ISRs by both methods notably drop. This is due to the relatively smaller magnitude of the corruption errors. Note that the approximation errors decrease with respect to the increasing number of basis functions for all methods except LSQ, even though the corruptions are partially identified.

In Fig. 6, we show the ISR results by 'k-ℓ_{1-2}', 'k-ℓ_p', 'k-LAD' and 'k-LSQ' for f_6 and f_7, with respect to the increasing number of basis functions. The rate of corruption is $\alpha=0.05$. On the left, the results of Franke function are shown with corruption errors drawn from

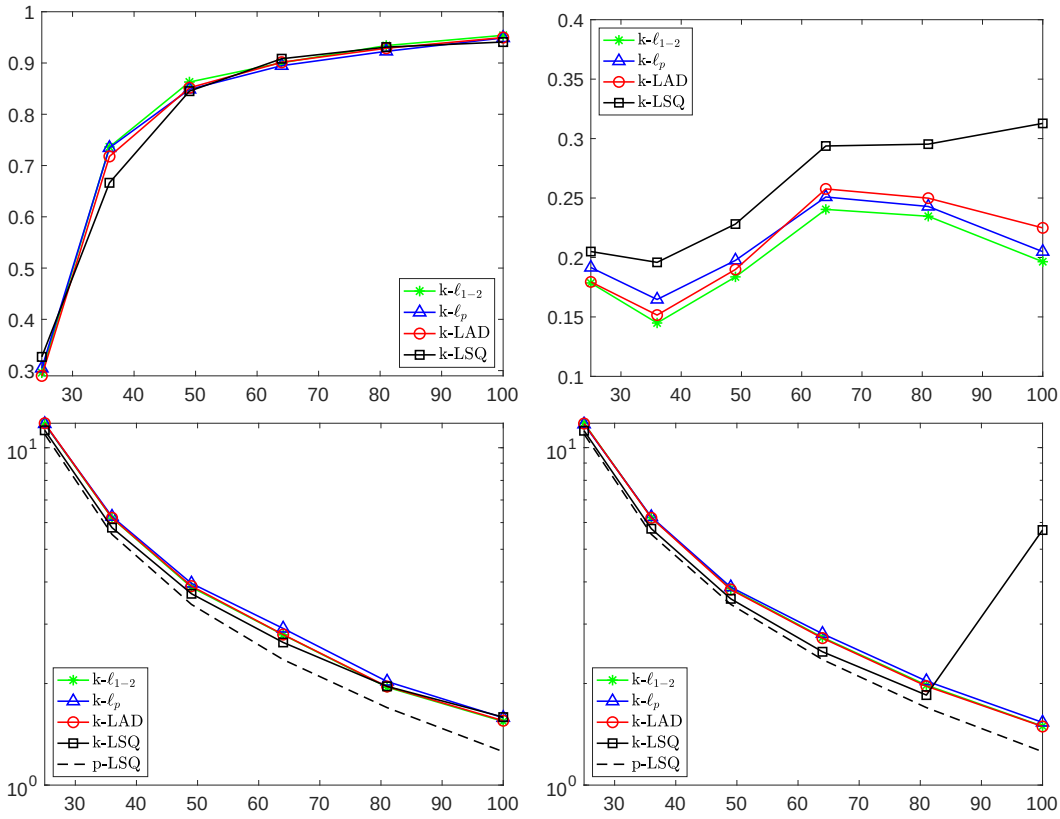

Figure 5: Identification success rates (top) and approximation errors (bottom) for the Bird function (5.2) are shown with respect to the number of radial basis functions at $d=2$. In the approximation stage, the approximation coefficients are obtained by ℓ_{1-2}, ℓ_p, LAD and LSQ on $m=1800$ sample points. The corruption rate is set to be $\alpha=0.1$. The external errors are generated from (left) $(e_s)_\Lambda \sim \mathcal{N}(10,1)$ and (right) $(e_s)_\Lambda \sim \mathcal{N}(1,1)$.

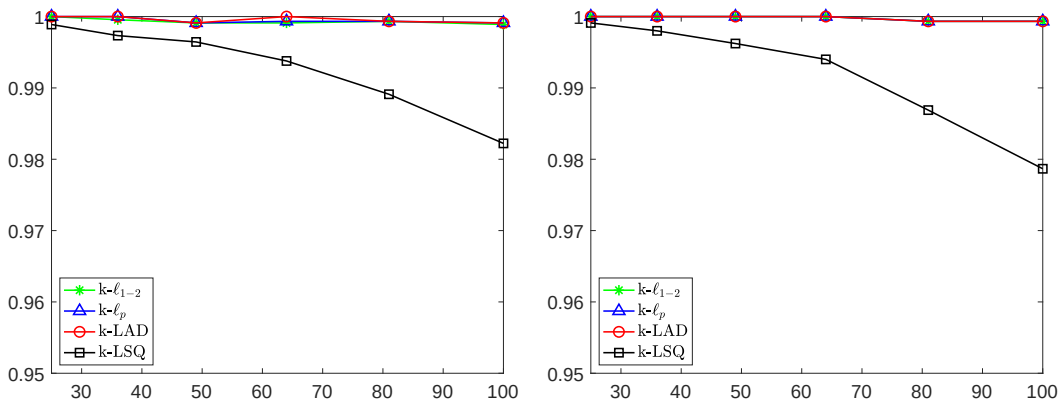

Figure 6: Identification success rates for f_6 are shown respect to the number of radial basis functions at $d=2$. In the approximation stage, the approximation coefficients are obtained by ℓ_{1-2}, ℓ_p, LAD and LSQ on 1800 sample points. The corruption rate is set to be $\alpha=0.05$. The corruption errors are drawn from (left) $\mathcal{N}(10,1)$ and (right) $\mathcal{N}(-10,1)$.

$\mathcal{N}(10,1)$. On the right, the results of Peaks function are shown with corruption errors drawn from $\mathcal{N}(-10,1)$. We observe that the 'k-ℓ_{1-2}', 'k-ℓ_p', and 'k-LAD' identify over 99% corruptions and 'k-LSQ' identifies over 97% corruptions.

6 Summary

We proposed a two-stage approach to identify corrupted data from a given data set. In the first stage, an approximation method is applied to the original data set to create a function approximation model. In particular, sparsity promoting methods such as ℓ_1 minimization, $\ell_1-\ell_2$ minimization are suitable. In the second stage, k-means clustering algorithm is applied to the approximation model to identify the corrupted data entries. We demonstrated that, both theoretically and numerically, the proposed approach is highly effective and, under mild conditions, can exactly identify all the corrupted entries.

A Proof of Theorem 4.1

Proof. Let $\mathcal{F}(\mathcal{I})=\sum_{i\in\mathcal{I}}|r_i-\mu_{\mathcal{I}}|^2$, where $\mu_{\mathcal{I}}=\frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}|r_i|$. Then it can be checked that $\mathcal{F}(\mathcal{I})=\sum_{i\in\mathcal{I}}r_i^2-\frac{1}{|\mathcal{I}|}\left(\sum_{i\in\mathcal{I}}|r_i|\right)^2$. For the true corruption index Λ, we have

$$\mathcal{F}(\Lambda)=\sum_{i\in\Lambda}(e_i+a_i)^2-\frac{1}{s}\left(\sum_{i\in\Lambda}|e_i+a_i|\right)^2,$$
$$\mathcal{F}(\Lambda^{\complement})=\sum_{i\in\Lambda^{\complement}}a_i^2-\frac{1}{m-s}\left(\sum_{i\in\Lambda^{\complement}}|a_i|\right)^2,$$

and for any index $\mathcal{I}$ such that $|\mathcal{I}|=u=t+l$, we have

$$\mathcal{F}(\mathcal{I})=\sum_{i\in\Omega_t}(e_i+a_i)^2+\sum_{i\in\Omega_l}a_i^2-\frac{1}{u}\left(\sum_{i\in\Omega_t}|e_i+a_i|+\sum_{i\in\Omega_l}|a_i|\right)^2,$$
$$\mathcal{F}(\mathcal{I})=\sum_{i\in\Omega_t^{\complement}}(e_i+a_i)^2+\sum_{i\in\Omega_l^{\complement}}a_i^2-\frac{1}{m-u}\left(\sum_{i\in\Omega_t^{\complement}}|e_i+a_i|+\sum_{i\in\Omega_l^{\complement}}|a_i|\right)^2.$$

For notational convenience, let

$$A_1=\sum_{i\in\Omega_t}|a_i+e_i|,\qquad A_2=\sum_{i\in\Omega_t^{\complement}}|a_i+e_i|,\qquad B_1=\sum_{i\in\Omega_l}|a_i|,\qquad B_2=\sum_{i\in\Omega_l^{\complement}}|a_i|.$$

The goal is to show the following inequality

$$\mathcal{O}(\Lambda):=\mathcal{F}(\Lambda)+\mathcal{F}(\Lambda^{\complement})\le\mathcal{O}(\mathcal{I}):=\mathcal{F}(\mathcal{I})+\mathcal{F}(\mathcal{I}^{\complement}).$$

Observe that

$$\mathcal{O}(\Lambda)=\sum_{i\in\Lambda}(e_i+a_i)^2+\sum_{i\in\Lambda^{\complement}}a_i^2-\frac{1}{s}(A_1+A_2)^2-\frac{1}{m-s}(B_1+B_2)^2,$$
$$\mathcal{O}(\mathcal{I})=\sum_{i\in\Lambda}(e_i+a_i)^2+\sum_{i\in\Lambda^{\complement}}a_i^2-\frac{1}{u}(A_1+B_1)^2-\frac{1}{m-u}(A_2+B_2)^2.$$

Hence, it suffices to show that

$$\frac{1}{u}(A_1+B_1)^2+\frac{1}{m-u}(A_2+B_2)^2\leq\frac{1}{s}(A_1+A_2)^2+\frac{1}{m-s}(B_1+B_2)^2,$$

or equivalently,

$$\begin{aligned}0\leq&\left(\frac{1}{s}-\frac{1}{m-u}\right)A_2^2+2\left(\frac{A_1}{s}-\frac{B_2}{m-u}\right)A_2+\left(\frac{1}{s}-\frac{1}{u}\right)A_1^2\\&+\left(\frac{1}{m-s}-\frac{1}{u}\right)B_1^2+\left(\frac{1}{m-s}-\frac{1}{m-u}\right)B_2^2+\frac{2}{m-s}B_1B_2-\frac{2}{u}A_1B_1.\end{aligned}$$

Note that $u=t+l$ is the number of elements in any subset of Ω such that $|\Omega|\leq|\Omega^{\complement}|$. Also $2s<m$. Due to its symmetry, without loss of generality, let $0<u=t+l<m-s$. Then the coefficient of A_2 is positive and the quadratic formula with respect to A_2 gives

$$A_2\geq\frac{-\left(\frac{A_1}{s}-\frac{B_2}{m-u}\right)+\sqrt{\mathcal{D}}}{\frac{1}{s}-\frac{1}{m-u}},\quad\text{or}\quad A_2\leq\frac{-\left(\frac{A_1}{s}-\frac{B_2}{m-u}\right)-\sqrt{\mathcal{D}}}{\frac{1}{s}-\frac{1}{m-u}},$$

where

$$\begin{aligned}\mathcal{D}&=\left(\frac{A_1}{s}-\frac{B_2}{m-u}\right)^2-\left(\frac{1}{s}-\frac{1}{m-u}\right)\mathcal{J},\\\mathcal{J}&=\frac{u-s}{su}A_1^2+\left(\frac{1}{m-s}-\frac{1}{u}\right)B_1^2-\frac{u-s}{(m-s)(m-u)}B_2^2+\frac{2}{m-s}B_1B_2-\frac{2}{u}A_1B_1.\end{aligned}$$

After some calculations, we have

$$\mathcal{D}=\frac{((m-s)A_1+(m-s-u)B_1-uB_2)^2}{s(m-s)u(m-u)}.$$

Since

$$\sqrt{\mathcal{D}}=\frac{|(m-s)A_1+(m-s-u)B_1-uB_2|}{\sqrt{s(m-s)u(m-u)}}=\sqrt{\frac{u(m-s)}{s(m-u)}}|\mu_{\mathcal{I}}-\mu_{\Lambda^{\complement}}|,$$

where $\mu_{\Lambda^c}=\frac{1}{m-s}\sum_{i\in\Lambda^c}|r_i|$, and $\mu_{\mathcal{I}}=\frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}|r_i|$, we have

$$\begin{aligned} A_2 &\geq \frac{-(m-u)A_1+sB_2+\sqrt{s(m-s)u(m-u)}|\mu_{\mathcal{I}}-\mu_{\Lambda^c}|}{m-s-u} \\ \text{or} \quad A_2 &\leq \frac{-(m-u)A_1+sB_2-\sqrt{s(m-s)u(m-u)}|\mu_{\mathcal{I}}-\mu_{\Lambda^c}|}{m-s-u}. \end{aligned}$$

Furthermore, one can equivalently rewrite the above as

$$\begin{aligned} \mu_\Lambda &\geq \frac{-A_1+B_2+\sqrt{\frac{(m-s)u(m-u)}{s}}|\mu_{\mathcal{I}}-\mu_{\Lambda^c}|}{m-s-u}, \\ \text{or} \quad \mu_\Lambda &\leq \frac{-A_1+B_2-\sqrt{\frac{(m-s)u(m-u)}{s}}|\mu_{\mathcal{I}}-\mu_{\Lambda^c}|}{m-s-u}. \end{aligned}$$

Suppose the external errors are sufficiently large enough to satisfy

$$\frac{\|e\|_1}{s} > \left(2-\frac{\lfloor\frac{m}{2}\rfloor}{m-s-\lfloor\frac{m}{2}\rfloor}\right)a_{\max}, \tag{A.1}$$

where $m/2>s>0$. Then we can exclude the case of

$$\mu_\Lambda \leq \frac{-A_1+B_2-\sqrt{\frac{(m-s)u(m-u)}{s}}|\mu_{\mathcal{I}}-\mu_{\Lambda^c}|}{m-s-u} \tag{A.2}$$

as follow. The assumption (A.1) implies that

$$\begin{aligned} & \frac{m-s-\lfloor\frac{m}{2}\rfloor}{s(m-s)}\sum_{i\in\Lambda}|e_i| > \mu_{\Lambda^c}+\frac{m-s-\lfloor\frac{m}{2}\rfloor}{s(m-s)}\sum_{i\in\Lambda}|a_i| \\ \Longleftrightarrow\ & \frac{m-s-\lfloor\frac{m}{2}\rfloor}{s(m-s)}\sum_{i\in\Lambda}(|e_i|-|a_i|) > \mu_{\Lambda^c} \\ \Longrightarrow\ & \frac{m-s-\lfloor\frac{m}{2}\rfloor}{s(m-s)}\sum_{i\in\Lambda}|e_i-a_i| > \mu_{\Lambda^c} \\ \Longrightarrow\ & \frac{m-s-u}{m-s}\mu_\Lambda > \mu_{\Lambda^c}. \end{aligned} \tag{A.3}$$

And (A.2) is equivalent to

$$\begin{aligned}
&(m-s-u)\mu_\Lambda+\sqrt{\frac{(m-s)u(m-u)}{s}}|\mu_I-\mu_{\Lambda^\complement}|+A_1\le B_2\\
\iff\ &(m-s-u)\mu_\Lambda+\sqrt{\frac{(m-s)u(m-u)}{s}}|\mu_I-\mu_{\Lambda^\complement}|+A_1+B_1\le B_1+B_2\\
\iff\ &\mu_\Lambda+\sqrt{\frac{u(m-u)}{s(m-s)}}|\mu_I-\mu_{\Lambda^\complement}|+\frac{u}{m-s}(\mu_I-\mu_\Lambda)\le\mu_{\Lambda^\complement}\\
\iff\ &\frac{m-s-u}{m-s}\mu_\Lambda-\mu_{\Lambda^\complement}+\sqrt{\frac{u(m-u)}{s(m-s)}}|\mu_I-\mu_{\Lambda^\complement}|+\frac{u}{m-s}\mu_I\le 0.
\end{aligned}$$

Since (A.3), the above gives a contradiction.

We thus focus on the case where

$$A_2\ge\frac{-(m-u)A_1+sB_2+\sqrt{s(m-s)u(m-u)}|\mu_{\mathcal{I}}-\mu_{\Lambda^\complement}|}{m-s-u}.$$

Depending on $|\mu_{\mathcal{I}}-\mu_{\Lambda^\complement}|$, we obtain the following two cases: if $\mu_{\mathcal{I}}\ge\mu_{\Lambda^\complement}$,

$$A_2\ge\alpha_u A_1+\beta_u B_1+\gamma_u B_2, \tag{A.4}$$

where

$$\alpha_x=\frac{m-x}{m-s-x}\left(-1+\sqrt{\frac{s(m-s)}{x(m-x)}}\right),\qquad \beta_x=\sqrt{\frac{s(m-x)}{x(m-s)}},\qquad \gamma_x=\frac{s}{m-s}\frac{\alpha_x}{\beta_x},$$

and if $\mu_{\mathcal{I}}<\mu_{\Lambda^\complement}$,

$$A_2\ge-\tilde{\alpha}_u A_1-\beta_u B_1+\tilde{\gamma}_u B_2, \tag{A.5}$$

where

$$\tilde{\alpha}_x=\frac{m-x}{m-s-x}\left(1+\sqrt{\frac{s(m-s)}{x(m-x)}}\right),\qquad \tilde{\gamma}_x=\frac{s}{m-s}\frac{\tilde{\alpha}_x}{\beta_x}.$$

Let $\eta=\{i\in\Lambda\,|\,\mathrm{sign}(\hat{e}_i)\neq\mathrm{sign}(a_i)\}$ and $\eta_t=\{i\in\Lambda\,|\,\mathrm{sign}(\hat{e}_i)\neq\mathrm{sign}(a_i)\}\cap\Omega_t$. Then $\eta=\eta_t\cup\eta_t^\complement$. Let assume $|e_i|>|a_i|$ for $i\in\Lambda$. Then

$$\begin{aligned}
A_1&=\sum_{i\in\Omega_t}|e_i+a_i|=\sum_{i\in\Omega_t}|e_i|-\sum_{i\in\eta_t}|a_i|+\sum_{i\in\Omega_t\setminus\eta_t}|a_i|,\\
A_2&=\sum_{i\in\Omega_t^\complement}|e_i+a_i|=\sum_{i\in\Omega_t^\complement}|e_i|-\sum_{i\in\eta_t^\complement}|a_i|+\sum_{i\in\Omega_t^\complement\setminus\eta_t^\complement}|a_i|.
\end{aligned}$$

Thus (A.4) becomes

$$\sum_{i\in\Omega_t^\complement}|e_i|\ge\alpha_u\sum_{i\in\Omega_t}|e_i|+\alpha_u\left(-\sum_{i\in\eta_t}|a_i|+\sum_{i\in\Omega_t\setminus\eta_t}|a_i|\right)+\sum_{i\in\eta_t^\complement}|a_i|-\sum_{i\in\Omega_t^\complement\setminus\eta_t^\complement}|a_i|+\beta_u B_1+\gamma_u B_2,$$

and (A.5) becomes

$$\sum_{i\in\Omega_t^{\complement}}|e_i|\geq-\tilde{\alpha}_u\sum_{i\in\Omega_t}|e_i|-\tilde{\alpha}_u\left(-\sum_{i\in\eta_t}|a_i|+\sum_{i\in\Omega_t\setminus\eta_t}|a_i|\right)+\sum_{i\in\eta_t^{\complement}}|a_i|-\sum_{i\in\Omega_t^{\complement}\setminus\eta_t^{\complement}}|a_i|-\beta_u B_1+\tilde{\gamma}_u B_2.$$

Let $a_{\min}=\min_{1\leq i\leq m}|(f-Ac^*)_i|$ and $a_{\max}=\max_{1\leq i\leq m}|(f-Ac^*)_i|$. A sufficient condition for (A.4) is

$$\begin{aligned}\sum_{i\in\Omega_t^{\complement}}|e_i|\geq\alpha_u\sum_{i\in\Omega_t}|e_i|+(l\beta_u+|\eta_t^{\complement}|)a_{\max}-(s-t-|\eta_t^{\complement}|)a_{\min}\\+\alpha_u\left(-\sum_{i\in\eta_t}|a_i|+\sum_{i\in\Omega_t\setminus\eta_t}|a_i|\right)+\gamma_u B_2,\end{aligned}$$

and a sufficient condition for (A.5) is

$$\begin{aligned}\sum_{i\in\Omega_t^{\complement}}|e_i|\geq-\tilde{\alpha}_u\sum_{i\in\Omega_t}|e_i|+|\eta_t^{\complement}|a_{\max}-(s-t-|\eta_t^{\complement}|+l\beta_u)a_{\min}\\-\tilde{\alpha}_u\left(-\sum_{i\in\eta_t}|a_i|+\sum_{i\in\Omega_t\setminus\eta_t}|a_i|\right)+\tilde{\gamma}_u B_2.\end{aligned}$$

Case 1: $\mu_{\mathcal{I}}<\mu_{\Lambda^{\complement}}$. Since $\tilde{\alpha}_u>0$,

$$\begin{aligned}\sum_{i\in\Omega_t^{\complement}}|e_i|\geq-\tilde{\alpha}_u\sum_{i\in\Omega_t}|e_i|+(|\eta_t^{\complement}|+|\eta_t|\tilde{\alpha}_u+(m-s-l)\tilde{\gamma}_u)a_{\max}\\-\left[(s-t-|\eta_t^{\complement}|)+(t-|\eta_t|)\tilde{\alpha}_u+l\beta_u\right]a_{\min}.\end{aligned}$$

Note that

$$\begin{aligned}-l\beta_u+(m-s-l)\tilde{\gamma}_u&=s\left[1+\frac{t}{m-s-u}\left(1+\sqrt{\frac{(m-s)(m-u)}{su}}\right)\right]\\&=s+t(\tilde{\alpha}_u-1).\end{aligned}\tag{A.6}$$

Let $0\leq c_a<1$ such that $(1-c_a)a_{\max}\leq|a_i|$ for all $1\leq i\leq m$. Then the above can be written as

$$\sum_{i\in\Omega_t^{\complement}}|e_i|\geq-\tilde{\alpha}_u\sum_{i\in\Omega_t}|e_i|+\tilde{\mathcal{H}}_{t,l}(c_a)a_{\max},\tag{A.7}$$

where

$$\begin{aligned}\tilde{\mathcal{H}}_{t,l}(c_a)=&-(s-t-2|\eta_t^{\complement}|)-(t-2|\eta_t|)\tilde{\alpha}_u+s+t(\tilde{\alpha}_u-1)\\&+\left[(s-t-|\eta_t^{\complement}|)+(t-|\eta_t|)\tilde{\alpha}_u+l\beta_u\right]c_a.\end{aligned}$$

By maximizing $\tilde{\mathcal{H}}_{t,l}(c_a)$ with respect to $|\eta_t$ and $|\eta_t^{\complement}$, we have

$$\mathcal{H}_{t,l}(c_a) := \max_{|\eta_t|,|\eta_t^{\complement}|} \tilde{\mathcal{H}}_{t,l}(c_a) = 2(s-t+t\tilde{\alpha}_u)+l\beta_u c_a,$$

where the maximum occurs at $|\eta_t|=t$ and $|\eta_t^{\complement}|=s-t$. Then a sufficient condition for (A.7) is

$$\sum_{i\in\Omega_t^{\complement}} |e_i| \geq -\tilde{\alpha}_u \sum_{i\in\Omega_t} |e_i| + \mathcal{H}_{t,l}(c_a) a_{\max}. \tag{A.8}$$

Suppose

$$\frac{\|e_s\|_1}{s} > \left(2+\frac{m}{2\sqrt{s(m-s)}}\right) a_{\max}.$$

Then (A.8) is automatically satisfied at $t=0$:

$$\begin{aligned}
&\text{Eq. (A.8) at } t=0\\
\iff\ & \|e_s\|_1 \geq \mathcal{H}_{0,l}(c_a) a_{\max} = s\left(2+\frac{\sqrt{l(m-l)}}{\sqrt{s(m-s)}}c_a\right) a_{\max}, \quad \forall 1\leq l\leq \lfloor \frac{m}{2} \rfloor\\
\impliedby\ & \frac{\|e_s\|_1}{s} > \left(2+\frac{m}{2\sqrt{s(m-s)}}\right) a_{\max}.
\end{aligned}$$

Case 2: $\mu_{\mathcal{I}} \geq \mu_{\Lambda^{\complement}}$. Note that we have, for $s\leq u=t+l<m-s$,

$$\alpha_{t+l}\leq\alpha_s=0, \qquad 0<\beta_{t+l}\leq\beta_s=1, \qquad \gamma_{t+l}\leq\gamma_s=0,$$

and for $0<u=t+l<s$,

$$\alpha_{t+l}>\alpha_s=0, \qquad \beta_{t+l}>\beta_s=1, \qquad \gamma_{t+l}>\gamma_s=0.$$

A sufficient condition for (A.4) is

$$\begin{aligned}
\sum_{i\in\Omega_t^{\complement}} |e_i| \geq\ & \alpha_u \sum_{i\in\Omega_t} |e_i| + (l\beta_u + |\eta_t^{\complement}|) a_{\max} - (s-t-|\eta_t^{\complement}|) a_{\min}\\
& + \alpha_u \left(-\sum_{i\in\eta_t} |a_i| + \sum_{i\in\Omega_t\setminus\eta_t} |a_i| \right) + \gamma_u B_2.
\end{aligned}$$

For $s\leq u<m-s$, since $\alpha_u\leq 0$,

$$\begin{aligned}
\sum_{i\in\Omega_t^{\complement}} |e_i| \geq\ & \alpha_u \sum_{i\in\Omega_t} |e_i| + (l\beta_u + |\eta_t^{\complement}| - |\eta_t|\alpha_u) a_{\max}\\
& - \Big[(s-t-|\eta_t^{\complement}|) - (t-|\eta_t|)\alpha_u \Big] a_{\min} + (m-s-l)\gamma_u a_{\min}
\end{aligned}$$

and for $0<u<s$, since $\alpha_u>0$,

$$\sum_{i\in\Omega_t^{\complement}}|e_i|\geq\alpha_u\sum_{i\in\Omega_t}|e_i|+\Big(l\beta_u+|\eta_t^{\complement}|+(t-|\eta_t|)\alpha_u\Big)a_{\max}$$
$$-\Big[(s-t-|\eta_t^{\complement}|)+|\eta_t|\alpha_u\Big]a_{\min}+(m-s-l)\gamma_u a_{\max}.$$

Let $0\leq c_a<1$ such that $(1-c_a)a_{\max}\leq|a_i|$ for all $1\leq i\leq m$. By combining the above cases, we could write

$$\sum_{i\in\Omega_t^{\complement}}|e_i|\geq\alpha_u\sum_{i\in\Omega_t}|e_i|+\tilde{\mathcal{C}}_{t,l}(c_a)a_{\max},$$

where

$$\tilde{\mathcal{C}}_{t,l}(c_a)=(t-2|\eta_t|)\alpha_u+l\beta_u+(m-s-l)\gamma_u-(s-t-2|\eta_t^{\complement}|)$$
$$+c_a\Big\{(s-t-|\eta_t^{\complement}|)+\alpha_u|\eta_t|-[(m-s-l)\gamma_u+t\alpha_u]\mathbb{1}_{\{s\leq u<m-s\}}\Big\}.$$

For any $0\leq|\eta_t|\leq t$ and $0\leq|\eta_t^{\complement}|\leq s-t$,

$$\tilde{\mathcal{C}}_{t,l}(c_a)\leq t|\alpha_u|+(l\beta_u+(m-s-l)\gamma_u-s)+2s-t-c_a[(m-s-l)\gamma_u+t\alpha_u]\mathbb{1}_{\{s\leq u<m-s\}}.$$

Note that

$$l\beta_u+(m-s-l)\gamma_u=s\left[1+\frac{t}{m-s-u}\left(1-\sqrt{\frac{(m-s)(m-u)}{su}}\right)\right].$$

Thus we have

$$\mathcal{C}_{t,l}(c_a):=\max_{|\eta_t|,|\eta_t^{\complement}|}\mathcal{C}_{t,l}(c_a)$$
$$=2s+t\left\{|\alpha_u|-1+\frac{s}{m-s-u}\left(1-\sqrt{\frac{(m-s)(m-u)}{su}}\right)\right\}$$
$$+c_a\left[l\sqrt{\frac{s(m-u)}{u(m-s)}}-(s-t)\right]\mathbb{1}_{\{s\leq u<m-s\}}. \tag{A.9}$$

The second term in (A.9) can be simplified as follow: For $0<u<s$,

$$\alpha_u-1+\frac{s}{m-s-u}\left(1-\sqrt{\frac{(m-s)(m-u)}{su}}\right)$$
$$=\frac{m-u}{m-s-u}\left(-1+\sqrt{\frac{s(m-s)}{u(m-u)}}\right)-1+\frac{s}{m-s-u}\left(1-\sqrt{\frac{(m-s)(m-u)}{su}}\right)$$
$$=-2+\frac{1}{m-s-u}\left(\sqrt{\frac{s(m-s)(m-u)}{u}}-\sqrt{\frac{s(m-s)(m-u)}{u}}\right)$$
$$=-2$$

and for $s \le u < m-s$,

$$\begin{aligned}
&-\alpha_u - 1 + \frac{s}{m-s-u}\left(1-\sqrt{\frac{(m-s)(m-u)}{su}}\right) \\
&= \frac{m-u}{m-s-u}\left(1-\sqrt{\frac{s(m-s)}{u(m-u)}}\right) - 1 + \frac{s}{m-s-u}\left(1-\sqrt{\frac{(m-s)(m-u)}{su}}\right) \\
&= \frac{2s}{m-s-u}\left[1-\sqrt{\frac{(m-s)(m-u)}{su}}\right].
\end{aligned}$$

Therefore,

$$\mathcal{C}_{t,l}(c_a) = \begin{cases} 2(s-t), & \text{if } 0 < u < s, \\ 2(s-t) + 2t|\alpha_u| + c_a\left[l\sqrt{\frac{s(m-u)}{u(m-s)}} - (s-t)\right], & \text{if } s \le u \le \lfloor \frac{m}{2} \rfloor. \end{cases}$$

Hence, a sufficient condition for (A.4) is

$$\sum_{i \in \Omega_t^{\complement}} |e_i| \ge \alpha_u \sum_{i \in \Omega_t} |e_i| + \mathcal{C}_{t,l}(c_a) a_{\max}, \tag{A.10}$$

which completes the proof. □

B Proof of Theorem 4.2

Proof. A sufficient condition for (A.8) is

$$|e|_{\min} \ge 2a_{\max} + \frac{l\beta_u}{s-t+t\tilde{\alpha}_u} c_a a_{\max} \tag{B.1}$$

for all t,l such that $1 \le t+l \le \frac{m}{2}$. It follows from (A.6) that

$$\begin{aligned}
\frac{l\beta_u}{s-t+t\tilde{\alpha}_u} &= \frac{l\sqrt{\frac{s(m-u)}{u(m-s)}}}{s-t+t\tilde{\alpha}_u} = \frac{\frac{ls}{m-s}\sqrt{\frac{(m-s)(m-u)}{su}}}{s + \frac{st}{m-s-u}\left(1+\sqrt{\frac{(m-s)(m-u)}{su}}\right)} \\
&= \frac{\frac{l}{m-s}\sqrt{\frac{(m-s)(m-u)}{su}}}{\frac{m-s-l}{m-s-u} + \frac{t}{m-s-u}\sqrt{\frac{(m-s)(m-u)}{su}}} \\
&= \frac{l}{(m-s)\left[\frac{t}{m-s-u} + \frac{m-s-l}{m-s-u}\sqrt{\frac{su}{(m-s)(m-u)}}\right]} := W(t,l).
\end{aligned}$$

For a fixed u,

$$\max_{t+l=u} W(t,l) = W(0,u),$$

which implies

$$\max_{t+l=u,1\le u\le \frac{m}{2}} W(t,l) = \max_{1\le u\le \frac{m}{2}} W(0,u) = \max_{1\le u\le \frac{m}{2}} \sqrt{\frac{u(m-u)}{s(m-s)}} = \frac{m}{2\sqrt{s(m-s)}}.$$

Therefore, we have

$$|e|_{\min} \ge 2a_{\max} + \frac{m}{2\sqrt{s(m-s)}} c_a a_{\max} \tag{B.2}$$

as a sufficient condition for (A.5).

Let us first recall that

$$\alpha_u = \frac{m-u}{m-s-u}\left(-1+\sqrt{\frac{s(m-s)}{u(m-s)}}\right).$$

For $0<u<s$, it follows from $\alpha_u>0$ that a sufficient condition for (A.10) is

$$(s-t)|e|_{\min} \ge \alpha_u t|e|_{\max} + 2(s-t)a_{\max},$$

which leads

$$|e|_{\min} \ge 2a_{\max} + \max_{0<u<s}\left(\frac{t|\alpha_u|}{s-t}\right)|e|_{\max}.$$

Let $h(t,l) = \frac{t|\alpha_u|}{s-t}$ where $u = t+l$. One can check that α_u is decreasing on $[0,s]$. Thus for fixed $0<t<s$,

$$\max_{0\le l<s-t} h(t,l) = h(t,0),$$

and one can also check that $h(t,0)$ is increasing on $[1,s)$. Therefore,

$$\max_{0<t+l<s} h(t,l) = \max_{0<t<s}\left\{\max_{0\le l<s-t} h(t,l)\right\} = \max_{0<t<s} h(t,0) = h(s-1,0).$$

For $s\le u\le \lfloor \frac{m}{2}\rfloor$, since $\alpha_u \le 0$, a sufficient condition for (A.10) is

$$(s-t)|e|_{\min} \ge \alpha_u t|e|_{\min} + 2(s-t+t|\alpha_u|)a_{\max} + c_a\left[l\sqrt{\frac{s(m-u)}{u(m-s)}} - (s-t)\right]a_{\max},$$

which gives

$$|e|_{\min} \ge 2a_{\max} + \max_{s\le u\le \lfloor \frac{m}{2}\rfloor,(t,l)\ne(s,0)} \frac{\left[l\sqrt{\frac{s(m-u)}{u(m-s)}} - (s-t)\right]}{s-t+t|\alpha_u|} c_a a_{\max}.$$

Note that when $(t,l)=(s,0)$, we obtain a trivial inequality of $0\geq 0$. It can be checked that for $s\leq u\leq \lfloor \frac{m}{2} \rfloor$, $|\alpha_u|<1$ as follow:

$$\begin{aligned}
&|\alpha_u| = \frac{m-s}{m-s-u}\left(1-\sqrt{\frac{s(m-s)}{u(m-u)}}\right)<1\\
\iff\ & 1-\sqrt{\frac{s(m-s)}{u(m-u)}}<\frac{m-s-u}{m-s}\\
\iff\ & 1-\frac{m-s-u}{m-s}<\sqrt{\frac{s(m-s)}{u(m-u)}}\\
\iff\ & \frac{s}{m-u}<\sqrt{\frac{s(m-s)}{u(m-u)}}\\
\iff\ & su<(m-s)(m-u).
\end{aligned}$$

Let $\tilde{h}(t,l)=\frac{l\sqrt{\frac{s(m-u)}{u(m-s)}}-(s-t)}{s-t+t|\alpha_u|}$. When $u=s$, since $\alpha_s=0$ and $l+t=s$, $h(t,s-t)=0$ for all $0\leq t<s$. Since $\sqrt{\frac{s(m-u)}{u(m-s)}}<1$ and $|\alpha_u|-1<0$, for fixed u such that $s<u\leq \lfloor \frac{m}{2} \rfloor$, we have

$$\max_{t+l=u}\tilde{h}(t,l)=\tilde{h}(s,u-s):=g(u).$$

Lemma B.1. *$g(u)$ is an increasing function.*

*Proof of Lemma B.*1. It follows from

$$\begin{aligned}
g(u)=\tilde{h}(s,u-s)&=\frac{1}{\sqrt{s(m-s)}}\frac{u-s}{|\alpha_u|}\sqrt{\frac{m-u}{u}}\\
&=\frac{1}{\sqrt{s(m-s)}}\frac{(u-s)(m-s-u)}{\sqrt{u(m-u)}-\sqrt{s(m-s)}}
\end{aligned}$$

that we show that $g'(u)\geq 0$ as follow:

$$\begin{aligned}
&\sqrt{s(m-s)}\frac{dg}{du}\\
=&\frac{(m-2u)(\sqrt{u(m-u)}-\sqrt{s(m-s)})-(u-s)(m-s-u)\frac{m-2u}{2\sqrt{u(m-u)}}}{(\sqrt{u(m-u)}-\sqrt{s(m-s)})^2}\\
=&\frac{(m-2u)\left((\sqrt{u(m-u)})^2-2\sqrt{u(m-u)}\sqrt{s(m-s)}+(\sqrt{s(m-s)})^2\right)}{2\sqrt{u(m-u)}(\sqrt{u(m-u)}-\sqrt{s(m-s)})^2}\\
=&\frac{m-2u}{2\sqrt{u(m-u)}}\geq 0.
\end{aligned}$$

□

Therefore, it follows from Lemma B.1 that

$$\max_{s\le u\le\lfloor\frac{m}{2}\rfloor,(t,l)\neq(s,0)}\tilde{h}(t,l)=\max_{s<u\le\lfloor\frac{m}{2}\rfloor}\tilde{h}(s,u-s)=\max_{s<u\le\lfloor\frac{m}{2}\rfloor}g(u)=g\Big(\lfloor\frac{m}{2}\rfloor\Big).$$

By combining the above with (B.2), the proof is completed. □

References

[1] N. ABDELMALEK, *On the discrete linear l_1 approximation and l_1 solutions of overdetermined linear equations*, J. Approx. Theory, 11 (1974), pp. 38–53.

[2] C. C. AGGARWAL, *Outlier analysis*, in Data mining, Springer, 2015, pp. 237–263.

[3] D. ARTHUR AND S. VASSILVITSKII, *k-means++: The advantages of careful seeding*, in Proceedings of the 18th Annual ACM-SIAM symposium on Discrete algorithms, SIAM, Philadelphia, 2007, pp. 1027–1035.

[4] I. BARRODALE AND F. ROBERTS, *An improved algorithm for discrete l_1 linear approximation*, SIAM J. Numer. Anal., 10 (1973), pp. 839–848.

[5] P. BELLOT AND M. EL-BÈZE, *A clustering method for information retrieval*, Technical ReportIR-0199, Laboratoire dInformatique dAvignon, France, (1999).

[6] P. BLOOMFIELD AND W. STEIGER, *Least absolute deviations curve-titting*, SIAM J. Sci. Comput., 1 (1980), pp. 290–301.

[7] E. CANDÈS AND J. ROMBERG, *l1-magic: Recovery of sparse signals via convex programming*, URL: www.acm.caltech.edu/l1magic/downloads/l1magic.pdf, 4 (2005), p. 14.

[8] E. CANDÈS AND T. TAO, *Decoding by linear programming*, Information Theory, IEEE Transactions on, 51 (2005), pp. 4203–4215.

[9] E. J. CANDES, M. B. WAKIN, AND S. P. BOYD, *Enhancing sparsity by reweighted ℓ_1 minimization*, Journal of Fourier analysis and applications, 14 (2008), pp. 877–905.

[10] R. CHARTRAND, *Exact reconstruction of sparse signals via nonconvex minimization*, IEEE Signal Proc. Lett., 14 (2007), pp. 707–710.

[11] R. CHARTRAND AND W. YIN, *Iteratively reweighted algorithms for compressive sensing*, in IEEE International Conference on Acoustics, Speech and Signal Processing, Las Vegas, NV, USA, March 31-April 4, 2008, IEEE, 2008, pp. 3869–3872.

[12] A. COHEN, M. A. DAVENPORT, AND D. LEVIATAN, *On the stability and accuracy of least squares approximations*, Found. Comput. Math., 13 (2013), pp. 819–834.

[13] M. B. COHEN, S. ELDER, C. MUSCO, C. MUSCO, AND M. PERSU, *Dimensionality reduction for k-means clustering and low rank approximation*, in Proceedings of the forty-seventh annual ACM symposium on Theory of computing, 2015, pp. 163–172.

[14] E. ESSER, Y. LOU, AND J. XIN, *A method for finding structured sparse solutions to nonnegative least squares problems with applications*, SIAM J. Imaging Sci., 6 (2013), pp. 2010–2046.

[15] R. FRANKE, *A critical comparison of some methods for interpolation of scattered data*, tech. report, NAVAL POSTGRADUATE SCHOOL MONTEREY CA, 1979.

[16] A. GENZ, *Testing multidimensional integration routines*, in Tools, Methods, and Languages for Scientific and Engineering Computation, B. Ford, J. Rault, and F. Thomasset, eds., North-Holland, 1984, pp. 81–94.

[17] J. A. HARTIGAN AND M. A. WONG, *Algorithm as 136: A k-means clustering algorithm*, Journal of the royal statistical society. series c (applied statistics), 28 (1979), pp. 100–108.

[18] Z. HE, S. DENG, AND X. XU, *An optimization model for outlier detection in categorical data*, in International Conference on Intelligent Computing, Springer, 2005, pp. 400–409.

[19] M. JAMIL AND X.-S. YANG, *A literature survey of benchmark functions for global optimisation problems*, Intl J Math Model Num Opt., 4 (2013), pp. 150–194.
[20] M. JIANG, S. TSENG, AND C. SU, *Two-phase clustering process for outliers detection*, Pattern recognition letters, 22 (2001), pp. 691–700.
[21] M. J. LI, M. K. NG, Y.-M. CHEUNG, AND J. Z. HUANG, *Agglomerative fuzzy k-means clustering algorithm with selection of number of clusters*, IEEE transactions on knowledge and data engineering, 20 (2008), pp. 1519–1534.
[22] S. LLOYD, *Least squares quantization in pcm*, IEEE Trans. Inf. Theory, 28 (1982), pp. 129–137.
[23] Y. LOU, P. YIN, Q. HE, AND J. XIN, *Computing sparse representation in a highly coherent dictionary based on difference of l_1 and l_2*, J. Sci. Comput., 64 (2015), pp. 178–196.
[24] J. MACQUEEN, *Some methods for classification and analysis of multivariate observations*, in Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, Berkeley, 1967, pp. 281–297.
[25] J. L. MARROQUIN AND F. GIROSI, *Some extensions of the k-means algorithm for image segmentation and pattern classification*, tech. report, Massachusetts Institute of Technology, 1993.
[26] S. NARULA AND J. WELLINGTON, *The minimum sum of absolute errors regression: A state of the art survey*, Inter. Stat. Rev., 50 (1982), p. 317326.
[27] M. OSBORNE AND G. WATSON, *On an algorithm for discrete nonlinear l_1 approximation*, Computer J., 14 (1971), pp. 184–188.
[28] B. D. RAO AND K. KREUTZ-DELGADO, *An affine scaling methodology for best basis selection*, IEEE Transactions on signal processing, 47 (1999), pp. 187–200.
[29] Y. SHE ET AL., *Thresholding-based iterative selection procedures for model selection and shrinkage*, Electronic Journal of statistics, 3 (2009), pp. 384–415.
[30] Y. SHIN AND D. XIU, *Correcting data corruption errors for multivariate function approximation*, SIAM J. Sci. Comput., 38 (2016), pp. A2492–A2511.
[31] K. SPYROPOULOS, E. KIOUNTOUZIS, AND A. YOUNG, *Discrete approximation in the l_1 norm*, Computer J., 16 (1972), pp. 180–186.
[32] Z. XU, X. CHANG, F. XU, AND H. ZHANG, *$l_{1/2}$ regularization: A thresholding representation theory and a fast solver*, IEEE Trans. Neural Netw. Learn. Syst., 23 (2012), pp. 1013–1027.
[33] L. YAN, L. GUO, AND D. XIU, *Stochastic collocation algorithms using ℓ_1-minimization*, Int. J. UQ, 2 (2012), pp. 279–293.
[34] L. YAN, Y. SHIN, AND D. XIU, *Sparse approximation using ℓ_1-ℓ_2 minimization and its application to stochastic collocation*, SIAM J. Sci. Comput., 39 (2017), pp. A229–A254.
[35] P. YIN, Y. LOU, Q. HE, AND J. XIN, *Minimization of ℓ_{1-2} for compressed sensing*, SIAM J. Sci. Comput., 37 (2015), pp. A536–A563.
[36] A. ZHANG, S. SONG, AND J. WANG, *Sequential data cleaning: A statistical approach*, in Proceedings of the 2016 International Conference on Management of Data, 2016, pp. 909–924.