Global Linear and Local Superlinear Convergence of IRLS for Non-Smooth Robust Regression

Liangzu Peng
Mathematical Institute for Data Science
Johns Hopkins University
lpeng25@jhu.edu

Christian Kümmerle
Department of Computer Science
University of North Carolina at Charlotte
kuemmerle@uncc.edu

René Vidal
Mathematical Institute for Data Science
Johns Hopkins University
rvidal@jhu.edu

Abstract

We advance both the theory and practice of robust ℓ_p-quasinorm regression for $p \in (0, 1]$ by using novel variants of *iteratively reweighted least-squares* (IRLS) to solve the underlying non-smooth problem. In the convex case, $p = 1$, we prove that this IRLS variant converges globally at a linear rate under a mild, deterministic condition on the feature matrix called the *stable range space property*. In the non-convex case, $p \in (0, 1)$, we prove that under a similar condition, IRLS converges locally to the global minimizer at a superlinear rate of order $2 - p$; the rate becomes quadratic as $p \to 0$. We showcase the proposed methods in three applications: real phase retrieval, regression without correspondences, and robust face restoration. The results show that (1) IRLS can handle a larger number of outliers than other methods, (2) it is faster than competing methods at the same level of accuracy, (3) it restores a sparsely corrupted face image with satisfactory visual quality.

1 Introduction

Given a feature matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ with $m \gg n$ and a response vector $\boldsymbol{y} \in \mathbb{R}^m$, the problem

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} \left\| \boldsymbol{A}\boldsymbol{x} - \boldsymbol{y} \right\|_p \tag{1}$$

of ℓ_p-regression has a variety of different applications depending on the choice of $\|\boldsymbol{v}\|_p = \left(\sum_i |\boldsymbol{v}_i|^p\right)^{1/p}$. While $p = 2$ corresponds to standard linear regression, the choice of $p > 2$ arises naturally in semi-supervised learning on graphs [1, 2, 3], and a lot of activity has been dedicated recently to the computational complexity analysis for the case $p > 1$ [4, 5, 6].

In this paper, we assume there is a coefficient vector $\boldsymbol{x}^* \in \mathbb{R}^n$ such that the residual $\boldsymbol{r}^* := \boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$ is k-sparse, in which case a choice of $p \in (0, 1]$ is of interest. Indeed, for the convex and non-smooth case $p = 1$, (1) is known as *least absolute deviation*, which dates back to the time of Boscovich around in the middle of the 18th century [7, 8]. Since then, it has been known intuitively that (1) ($p = 1$) is robust to *large but few measurement errors* (quoting [9]), i.e., that (1) is robust to outliers.

The algorithmic and theoretical understanding of (1) for $p = 1$ has been of long-standing interest to statisticians, which has led to a vast literature spanning from the classic 1964 paper of Huber [10] to recent contributions [11, 12, 13, 14, 15, 16, 17, 18, 19, 20]. That being said, for $p = 1$, there is a close, but somewhat underexplored connection between (1) and the *basis pursuit* problem [21],

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

Algorithm 1: IRLS for ℓ_p-Regression ($\texttt{IRLS}_p$)

Input: $\boldsymbol{A} = [\boldsymbol{a}_1, \dots, \boldsymbol{a}_m]^\top \in \mathbb{R}^{m\times n}$, $\boldsymbol{y} = [y_1, \dots, y_m]^\top \in \mathbb{R}^m$, $p \in (0,1]$;

Weight initialization $\boldsymbol{w}^{(0)} \leftarrow [w_1^{(0)}; \dots; w_m^{(0)}] \in \mathbb{R}^m$; // Instead, one can initialize some vector $\boldsymbol{x}^{(1)}$.

For $t \leftarrow 0, 1, \dots$:

$$\boldsymbol{x}^{(t+1)} \leftarrow \operatorname*{argmin}_{\boldsymbol{x}\in\mathbb{R}^n} \sum\nolimits_{i=1}^{m} w_i^{(t)} (\boldsymbol{a}_i^\top \boldsymbol{x} - y_i)^2 \tag{2}$$

Update $\epsilon^{(t+1)}$ suitably based on $\boldsymbol{x}^{(t+1)}$, $\boldsymbol{A}$, and $\boldsymbol{y}$; // See Section 2.2 for details.

$$w_i^{(t+1)} \leftarrow \max\left\{|\boldsymbol{a}_i^\top \boldsymbol{x}^{(t+1)} - y_i|, \epsilon^{(t+1)}\right\}^{p-2} \quad \forall i = 1, \dots, m \tag{3}$$

and more specifically, the theoretical [22, 23, 24] and algorithmic aspects [25, 26, 27] of compressed sensing [28] (see also Section 2.1). For $p \in (0,1)$, problem (1) is non-smooth and non-convex, and generally less well understood. To the best of our knowledge, the only paper that considered (1) with $p \in (0,1)$ is [29], where the authors presented a condition (based on *restricted isometry constants*) that guarantees exact recovery of $\boldsymbol{x}^*$ from (1). Other related works come either from the compressed sensing literature, where an ℓ_p (and noisy) version of basis pursuit has been considered as a sparsity-promoting formulation [30, 31, 32, 26, 33, 34, 35, 36], or from the literature of matrix recovery/completion, where the *Schatten-p* norm has come into play as a non-convex surrogate for rank minimization [37, 38, 39, 40, 41, 42]. A key message from these works is that ℓ_p or Schatten-p minimization with $p \in (0,1)$ offers better information-theoretic properties (e.g., requires fewer samples for exact recovery) than minimization with $p = 1$.

Here, we study an *iteratively reweighted least-squares* method ($\texttt{IRLS}_p$) to solve (1) with $p \in (0,1]$. As listed in Algorithm 1, $\texttt{IRLS}_p$ alternates between solving a weighted least-squares problem (2) and updating the weights (3); see Section 2.2 for more elaboration on IRLS. The simplicity of this idea (with its impressive performance) justifies its popularity in many machine learning [43, 44, 45, 46] and computer vision [47, 48, 49, 50, 51] applications. Based on recent advances on IRLS [26, 27], we make several contributions for understanding (1) and $\texttt{IRLS}_p$. We state our contributions next.

The Stable Range Space Property (Section 3.1). We put forward the use of the (stable) *range space property* (RSP) for studying the robust regression problem (1). The stable RSP was proposed by [52] in a different context to analyze a compressed sensing algorithm for solving a weighted basis pursuit problem at each iteration. In analogy to the *nullspace property* in compressed sensing [24, 28], we show that the RSP is a necessary and sufficient condition for guaranteeing that the ℓ_p-regression problem in (1) admits $\boldsymbol{x}^*$ as its unique solution (Proposition 2). Moreover, we show that if $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ has i.i.d. $\mathcal{N}(0,1)$ entries and if m is large enough, then the (stable) RSP holds with high probability (Proposition 3). This justifies its use as the core assumption in our analysis.

Global Linear Convergence (Section 3.2). We prove in Theorem 1 that, under a stable RSP assumption and with $\epsilon^{(t)}$ suitably updated, the $\texttt{IRLS}_p$ Algorithm 1 with $p = 1$ converges linearly to the ground-truth $\boldsymbol{x}^*$ from any initial weight $\boldsymbol{w}^{(0)}$ (or equivalently from any initial point $\boldsymbol{x}^{(1)}$). Note that while (accelerated) first-order methods (e.g., (sub-)gradient descent and proximal algorithms) can also solve the convex and non-smooth problem (1) with $p = 1$, they exhibit, in general, (global) sub-linear rates at best [53, 54]. To our knowledge, [17] is the only paper that claims global linear convergence of a different IRLS variant for (1) with $p = 1$; we compare our results with the ones of [17] in Section 3.2. On the other hand, Theorem 1 is inspired by the IRLS method of [27] for basis pursuit; based on [27], we suitably modify their proof strategy, and thus obtain a faster linear rate.

Local Superlinear Convergence (Section 3.3). We prove in Theorem 2 that, under a stable RSP assumption and with $\epsilon^{(t)}$ suitably updated, the $\texttt{IRLS}_p$ Algorithm 1 with $p \in (0,1)$ converges to $\boldsymbol{x}^*$ superlinearly, provided that $\boldsymbol{x}^{(1)}$ falls into a certain neighborhood of $\boldsymbol{x}^*$. To the best of our knowledge, no similar result exists for ℓ_p-regression (1). While Theorem 2 is inspired by the IRLS algorithm of [26] for ℓ_p-basis pursuit, their IRLS method does not work well for small p (Section 2.2), and unlike in our work, their radius of local convergence diminishes greatly as $p \to 0$ or $m \to \infty$.

Applications (Section 4). We illustrate the performance of $\texttt{IRLS}_p$ for *real phase retrieval* [55, 56], *linear regression without correspondences* [57, 58, 59, 60], and *face restoration from sparsely*

corrupted measurements [61, 62, 63]. For real phase retrieval (Section 4.1), we show that $\texttt{IRLS}_{0.1}$ needs only $m = 2n - 1$ measurements to recover $\boldsymbol{x}^*$ up to sign, with an additional assumption. Theoretically, even brute-force can fail to identify $\pm\boldsymbol{x}^*$ for fewer than $2n - 1$ measurements [64]. Empirically, many methods, including *Kaczmarz* [65, 66, 56], *PhaseLamp* [67], *truncated Wirtinger flow* [68], and *coordinate descent* [69], fail with $m = 2n - 1$ Gaussian measurements (Figure 2a). For linear regression without correspondences (Section 4.2), we show in Figures 2b-2c that, $\texttt{IRLS}_{0.1}$ is uniformly faster (20-100x) and more accurate than *PDLP* [70] (merged into Google or-tools) and the commercial solver *Gurobi* [71], both of which solve (1) with $p = 1$ as a linear program, and also than *subgradient descent* implemented by Beck & Guttmann-Beck [72]. For face restoration (Section 4.3), we present both quantitative and qualitative results on the Extended Yale B dataset [73].

2 Background

2.1 Connection of Robust Regression and Compressed Sensing

The ℓ_p-regression problem (1) has a natural correspondence to the sparse recovery problem

$$\min_{\boldsymbol{r}\in\mathbb{R}^m} \|\boldsymbol{r}\|_p \qquad \text{s.t.} \qquad \boldsymbol{D}\boldsymbol{r} = \boldsymbol{z} \tag{4}$$

where $\boldsymbol{D}$ is a $(m-n)\times m$ matrix, $\boldsymbol{z}\in\mathbb{R}^{m-n}$ a vector and the objective $\|\boldsymbol{r}\|_p$ penalizes coefficient vectors with too many non-zero coordinates. For $p = 1$, (4) is also called *basis pursuit* [74]. Problems (1) and (4) are known to be related in the following sense (as implicitly stated in [21, 29]):

Proposition 1. *Suppose that the range space of $\boldsymbol{A}$ is equal to the nullspace of $\boldsymbol{D}$, and that $\boldsymbol{z} = -\boldsymbol{D}\boldsymbol{y}$. If $\boldsymbol{x}_1$ globally minimizes (1), then $\boldsymbol{r}_1 := \boldsymbol{A}\boldsymbol{x}_1 - \boldsymbol{y}$ globally minimizes (4). On the other hand, if $\boldsymbol{r}_2$ globally minimizes (4), then there exists some $\boldsymbol{x}_2$ with $\boldsymbol{r}_2 = \boldsymbol{A}\boldsymbol{x}_2 - \boldsymbol{y}$ that globally minimizes (1).*

Proposition 1 sheds light on how we "transfer", with new insights, from the analysis of [26, 27] for (4) to results for (1). In what follows, we treat [26, 27] in the context of robust regression and highlight the contribution of our work relative to [26, 27] and other existing results, whenever possible.

2.2 Iteratively Reweighted Least-Squares: The Basics and New Insights

We first discuss two different updating rules for the *smoothing parameter* $\epsilon^{(t)}$ of Algorithm 1: the *fixed* rule and the *dynamic* rule. We then consider the weight updating strategy (3). Along the way we use synthetic experiments to illustrate ideas (for synthetic data generation, see Appendix D.2). In doing so, we intend to provide a review of the state-of-the-art on the variants of IRLS.

Fixed Smoothing Parameter. Instead of updating $\epsilon^{(t)}$ at each iteration, most works on IRLS use a fixed and small positive number, e.g., $\epsilon := \epsilon^{(t)} = 0.001$ for each t [43, 75, 44, 46, 48, 50, 49, 76]. The intuition is to avoid division by the potentially very small residual $|\boldsymbol{a}_i^\top \boldsymbol{x}^{(t)} - y_i|$. But this common practice of fixing ϵ comes with at least three issues. First, IRLS with a fixed $\epsilon > 0$ converges only to an "ϵ-approximate" point, not exactly the global minimizer ([77, 78, 43, 75], Figure 1a). Second, even if setting ϵ small (e.g., $\epsilon = 10^{-15}$) could lead to an accurate enough solution for $p = 1$ (Figure 1a), it can fail for $p < 1$ (Figure 1b). Finally, it makes obtaining a global linear convergence rate guarantee difficult: We are not aware of any theory about a global linear rate of IRLS with fixed ϵ.

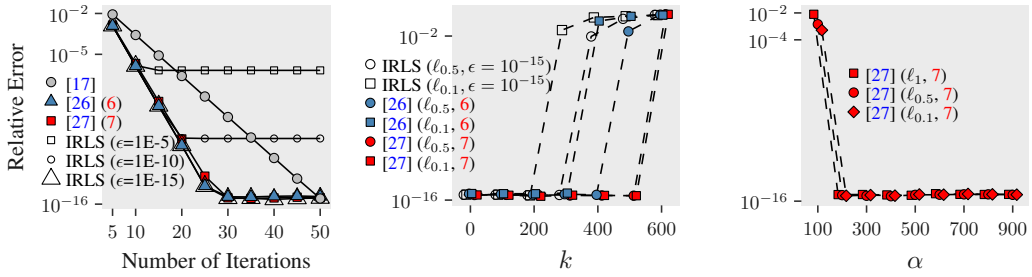
Dynamic Smoothing Parameter. Researchers have used different insights to reach the consensus of dynamically updating $\epsilon^{(t)}$ [26, 17, 79, 27, 80]. The first insight is that convergence to global minimizers ensues if $\epsilon^{(t)}$ is suitably decreased to 0 [26]. With $\beta \in (0, 1)$, one such decreasing rule is

$$\epsilon^{(t+1)} \leftarrow \beta\epsilon^{(t)} \text{ if certain conditions are satisfied, or keep } \epsilon^{(t+1)} \leftarrow \epsilon^{(t)} \text{ otherwise [17, 80].} \tag{5}$$

Figure 1a depicts the performance of [17] with an arbitrary choice $\beta = 0.5$ and $\epsilon^{(0)} = 1$. Also shown in Figure 1a are the IRLS methods with update rules of [26] and [27], which we discuss next.

The basis pursuit paper [26] also proposed a dynamic updating rule for $\epsilon^{(t)}$. In our context (1), it sets

$$\epsilon^{(0)} \leftarrow \infty, \quad \epsilon^{(t+1)} \leftarrow \min\left\{\epsilon^{(t)}, [\boldsymbol{r}^{(t+1)}]_{\alpha+1}/m\right\}, \qquad \boldsymbol{r}^{(t+1)} := \boldsymbol{A}\boldsymbol{x}^{(t+1)} - \boldsymbol{y} \in \mathbb{R}^m, \tag{6}$$



(a) $p = 1$ (b) $p < 1$ (c) Sensitivity to α ($k = 200$)

Figure 1: Figure 1a: The relative error $\|\boldsymbol{x}^{(t)} - \boldsymbol{x}^*\|_2 / \|\boldsymbol{x}^*\|_2$ at each iteration t of different IRLS variants for ℓ_1-regression ($k = 200$). Figure 1b: The relative error of $\texttt{IRLS}_p$ Algorithm 1 and that of [26] for ℓ_p-regression (50 iterations). Figure 1c: The sensitivity of (6) and (7) to mis-specification of α (50 iterations). In Figure 1, we set $m = 1000, n = 10$, and results are averaged over 20 trials.

where the hyper-parameter[1] α is a non-negative integer, and $[\boldsymbol{r}^{(t+1)}]_{\alpha+1}$ is the $(\alpha+1)$-th largest element of the residual $\boldsymbol{r}^{(t+1)}$ in absolute values. Clearly, (6) creates a non-increasing sequence of smoothing parameters $\epsilon^{(t)}$, and the decay rate of $\epsilon^{(t)}$ is adaptive to the data (in this case to the residual $\boldsymbol{r}^{(t+1)}$). While at first glance it is not clear how fast $\epsilon^{(t)}$ decays, [26] showed that the decay rate of $\epsilon^{(t)}$ in (6) is locally linear for $p = 1, \alpha = k$ under mild conditions, in accordance with the decay of the objective values.

The final dynamic update rule for the parameter $\epsilon^{(t)}$ is from [27][1] and improves upon (6) [26] via:

$$\begin{aligned} \epsilon^{(0)} \leftarrow \infty, \qquad \epsilon^{(t+1)} \leftarrow \min\left\{\epsilon^{(t)}, \sigma^{(t+1)}/m\right\} \\ \text{where } \sigma^{(t+1)} \leftarrow \min\left\{\left\|\boldsymbol{r}^{(t+1)} - \boldsymbol{z}\right\|_1 : \boldsymbol{z} \in \mathbb{R}^m \text{ is } \alpha\text{-sparse}\right\} \end{aligned} \tag{7}$$

Observe that (7) computes the ℓ_1-norm $\sigma^{(t+1)}$ of the best α-term approximation of the residual $\boldsymbol{r}^{(t+1)}$, which is in general larger than $[\boldsymbol{r}^{(t+1)}]_{\alpha+1}$ of (6), while both update rules (6) and (7) rely on the hyper-parameter α. Ideally, if $\alpha = k$, and if IRLS with (6) or (7) converges to $\boldsymbol{x}^*$, then both $[\boldsymbol{r}^{(t+1)}]_{\alpha+1}$ and $\sigma^{(t+1)}$ will approach 0 for large enough t. While update rules (6) and (7) perform similarly for ℓ_1-regression (Figure 1a), (6) fails more easily for small p than (7) (Figure 1b). Finally, we note that overestimating the sparsity level k by setting $\alpha > k$ deteriorates the performance of (7) only slightly (Figure 1c).

Summary. In Section 2.2 we delivered two take-away messages: (1) IRLS with a fixed $\epsilon^{(t)}$ finds some approximate solution, sometimes good enough, (2) dynamically updating $\epsilon^{(t)}$ as per (5)-(7) leads to local [26] or global [17, 27] linear convergence guarantees. In view of Figure 1, we will next consider the $\texttt{IRLS}_p$ Algorithm 1 with update rule (7) for $\epsilon^{(t)}$, and analyze its convergence rates.

3 Convergence Theory of IRLS for Robust Regression

In Section 3.1 we introduce the stable range space property and justify it as our core assumption. Under this assumption, we prove the global linear convergence (Theorem 1) and local superlinear convergence (Theorem 2) of the $\texttt{IRLS}_p$ Algorithm 1 in Sections 3.2 and 3.3, respectively.

3.1 The Stable Range Space Property

Definition 1 (Stable Range Space Property)**.** A matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ is said to satisfy the *range space property (RSP)* of order k, or the k-RSP for short, if the following holds for any vector $\boldsymbol{d}$ in the range

[1] In [26] and [27], α is the sparsity level k, but k might be unknown in practice, so we treat it as a hyper-parameter. That being said, in the experiments we set $\alpha = k$ by default, for simplicity and in light of Figure 1c.

space of $\boldsymbol{A}$ and any set $S \subset \{1, \ldots, m\}$ of cardinality at most k:

$$\sum_{i \in S} |d_i| < \sum_{i \in S^c} |d_i| \tag{8}$$

The *stable RSP* of $\boldsymbol{A}$ is defined in the same way as the RSP except that we now require

$$\sum_{i \in S} |d_i| \leq \eta \sum_{i \in S^c} |d_i| \tag{9}$$

for some $\eta \in (0, 1)$. We write "(k, η)-stable RSP" to emphasize the parameters k and η.

By definition, the stable RSP implies the RSP. Note that the (k, η)-stable RSP was proposed in [52] to analyze a reweighted ℓ_1-minimization algorithm for compressed sensing. With the notation of Proposition 1, we can see that checking (8) for all $\boldsymbol{d}$ in the range space of $\boldsymbol{A}$ is equivalent to checking it for all $\boldsymbol{d}$ in the nullspace of $\boldsymbol{D}$, the latter being the well-known *nullspace property* (NSP) [24, 28]. In other words, the (stable) RSP of $\boldsymbol{A}$ is equivalent to the (stable) NSP of $\boldsymbol{D}$.

In Sections 3.2 and 3.3, we will use the (k, η)-stable RSP as an assumption for analysis. Arguably, this is a weak assumption to make as the (stable) RSP is very close to a sufficient and necessary condition for exact recovery of the coefficient vector via ℓ_p-robust regression (1):

Proposition 2 (Exact Recovery $\Leftrightarrow$ RSP)**.** *Let $p \in (0, 1]$. For all $\boldsymbol{x}^*$ and $\boldsymbol{y}$ such that $\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$ is k-sparse $\boldsymbol{x}^*$ is the unique solution to (1) if and only if $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ satisfies the k-RSP.*

It follows from [81, Theorem 5] and Proposition 1 that checking whether a given matrix $\boldsymbol{A}$ satisfies the stable RSP is *co-NP-complete* [82]. On the other hand, we show that random Gaussian matrices of size $m \times n$ with sufficiently large m satisfy the stable RSP with high probability, which further justifies the usage of the stable RSP as an assumption in our analysis of $\texttt{IRLS}_p$:

Proposition 3 (Gaussian $\Rightarrow$ RSP)**.** *Suppose $m - n \geq 2k$. Let $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ be a matrix with i.i.d. $\mathcal{N}(0, 1)$ entries. Let $\delta \in (0, 1)$ and $\eta \in (0, 1]$ be fixed constants. If it holds that*

$$\frac{(m-n)^2}{m-n+1} \geq 2k \ln(em/k) \cdot \left(1.67 + \eta^{-1} + \frac{\sqrt{18 \ln(2.5\delta^{-1})}}{\sqrt{2k \ln(em/k)}}\right)^2, \tag{10}$$

then $\boldsymbol{A}$ satisfies the (k, η)-stable RSP with probability at least $1 - \delta$.

The assumption $m - n \geq 2k$ of Proposition 3 is necessary, as it is not hard to prove that, if $m - n < 2k$ and $\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$ is k-sparse, then the ℓ_0-minimization problem $\min_{x \in \mathbb{R}^n} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_0$ has multiple solutions. With this assumption, (10) roughly becomes $m - n \geq ck \log(em/k)$ for some constant c if $m - n$ is large enough; thus, ignoring logarithmic and constant factors, condition (10) becomes $m - n \geq \Theta(k)$, which is nearly optimal, as the condition $m - n \geq 2k$ is necessary.

Proposition 2 follows directly from [28, Theorem 4.9], so we omit its proof. The reader might also find Proposition 3 corresponds to [28, Corollary 9.34]. This corollary assumes $\boldsymbol{D}$ of (4) has i.i.d. $\mathcal{N}(0, 1)$ entries, from which is not immediate what the distribution of $\boldsymbol{A}$ is, thus it does not imply Proposition 3 directly; this is why we provide a complete proof for Proposition 3 in Appendix C.1.

3.2 Global Linear Convergence of $\texttt{IRLS}_1$

With the background provided in Section 2, we are now ready to state the first main result:

Theorem 1 (Global Linear Convergence)**.** *Suppose $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ obeys the (k, η)-stable RSP with $\eta \in (0, 3/4)$ (cf. Definition 1). Let $\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$ be k-sparse. If the smoothing parameter $\epsilon^{(t)}$ is updated as per (7) with $\alpha = k$ during the execution of the $\texttt{IRLS}_1$ Algorithm 1, then it holds for any $t \geq 1$ that*

$$\left\|\boldsymbol{A}\boldsymbol{x}^{(t+1)} - \boldsymbol{y}\right\|_1 - \left\|\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}\right\|_1 \leq \left(1 - \frac{(3-4\eta)^2}{294\eta m}\right)^t \cdot 3 \cdot \left\|\boldsymbol{A}\boldsymbol{x}^{(1)} - \boldsymbol{y}\right\|_1, \tag{11}$$

meaning that the $\texttt{IRLS}_1$ Algorithm 1 converges linearly and globally in objective value.

Discussion. The condition $\eta \in (0, 3/4)$ is better understood via Proposition 3, which asserts that this condition holds with high probability if $\boldsymbol{A}$ has i.i.d. Gaussian entries and if m is large enough

(compared to n, k). Moreover, if the (k, η)-stable RSP holds, Proposition 2 implies that $\boldsymbol{x}^*$ is a global minimizer of (1) with $p = 1$. The strength of Theorem 1 is in that it guarantees linear convergence for *any initialization, starting at the first iteration*. However, the price to pay is that the convergence rate seems conservative with constant $1/294$, and depends on the number m of samples: (11) predicts that $\texttt{IRLS}_1$ converges with accuracy δ in $O(m \log(1/\delta))$ iterations and m can be large. We argue that such dependency is not an artifact of our analysis, because similar dependencies have been established for IRLS variants for ℓ_p-regression (1) with $p > 2$ [2, Theorem 3.1], for interior point methods in convex optimization with m inequality constraints ([83, Chapter 11], [54, Chapter 5]), and for SGD in smooth & strongly convex optimization with a sum of m function components [84, Theorem 2.1]). Moreover, an adversarial initialization for which the rate empirically depends on m was pointed out for IRLS for basis pursuit in [27]. That being said, we conjecture that the linear dependency on m in (11) can be improved to $\sqrt{m} \log m$ for some different choice of $\epsilon^{(t)}$; see [2] and [83, Sections 11.5 and 11.8.2] for why this conjecture makes sense. Furthermore, Figure 1a empirically depicts that, with the least-squares initialization, i.e., with $w_i^{(0)} = 1$ for all i, $\texttt{IRLS}_1$ typically converges to $\boldsymbol{x}^*$ in 30 iterations. This hints to the possibility of alleviating the dependency on m by analyzing the least-squares initialization, a challenging task which we leave to future work.

Comparison to [17]. To our knowledge, [17] is the only paper that claimed the global linear convergence of an IRLS variant with $\epsilon^{(t)}$ updated as per (5) and with $p = 1$. In particular, [17] sets $\epsilon^{(t+1)} \leftarrow \beta\epsilon^{(t)}$ whenever $\|\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)}\|_2 \leq 2\beta\epsilon^{(t)}$; see Figure 1a for the convergence of IRLS under this rule. The update rule $\epsilon^{(t+1)} \leftarrow \beta\epsilon^{(t)}$ of [17] has the advantage of being faster to compute than (7). In spite of this, and of their beautiful proof idea, their result has several disadvantages compared to Theorem 1. First, their proof involves sophisticated probabilistic arguments, and is tailored towards a feature matrix that satisfies strong concentration properties. Our analysis, on the other hand, uses the (k, η)-stable RSP, which is close to a necessary and sufficient condition for the success of ℓ_1-regression (cf. Proposition 2) and expected to hold for a much larger range of matrices than just (sub-)Gaussian matrices (see [85, 86] for related results in the context of basis pursuit). Second, some statements in their proof are inaccurate (e.g., their Lemma 5 does not hold for $k < n$, the last paragraph in their proving Lemma 8 is not rigorous). Third, their update rule incurs two hyper-parameters, $\epsilon^{(0)}$ and β, while Theorem 1 is based on the decreasing rule (7) that is adaptive to data, and the only hyper-parameter α is easier to set (Figure 1c).

Connection to [27]. A related global linear convergence was established for the IRLS method for basis pursuit in [27, Theorem 3.2]. Our proof is inspired by [27], but also improves on its strategy. For example, their Theorem 3.2 involves two parameters for the *stable nullspace property*, while we only have a single RSP parameter η, simplifying matters. Also, we obtain a constant of $1/294$ in Theorem 1, which is better than the value of $1/768$ for the respective constant in [27, Theorem 3.2].

It is important to note that the proofs of [17, 27] and ours heavily rely on certain rules for decreasing the smoothing parameter $\epsilon^{(t)}$; all these proofs of global linear convergence for IRLS would break down if $\epsilon^{(t)}$ were fixed. While the main theorems of both [17] and [27] are limited to the $p = 1$ case, we present our Theorem 2 for $p \in (0, 1)$ next.

3.3 Local Superlinear Convergence of $\texttt{IRLS}_p$

The ℓ_p-regression problem (1) with $p \in (0, 1)$ is more challenging than the case $p = 1$ due to the lack of convexity. However, here we show that at least locally, $\texttt{IRLS}_p$ converges with a superlinear rate of order $2 - p$. Moreover, since it is valid to run $\texttt{IRLS}_p$ even with $p = 0$, we carefully design the proof such that the following result holds not only for $p \in (0, 1]$, but also for $p = 0$, in which case $\texttt{IRLS}_p$ can be interpreted as an algorithm minimizing a sum-of-logarithm objective (see Appendix A.2).

Theorem 2 (Local Superlinear Convergence)**.** *Run $\texttt{IRLS}_p$ with $p \in [0, 1]$ and update $\epsilon^{(t)}$ by (7) with $\alpha = k$. Assume $\boldsymbol{A}$ satisfies the (k, η)-stable RSP (Definition 1). Let $c \in (0, 1)$ be a sufficiently small constant such that $2c^{1-p}\eta(\eta + 1) < (1 - c)^{2-p}$. Let $\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$ be k-sparse with support S^*. Define*

$$\mu := 2\eta(\eta + 1)(1 - c)^{p-2} \cdot \min_{i \in S^*} |\boldsymbol{a}_i^\top \boldsymbol{x}^* - y_i|^{p-1}. \tag{12}$$

If the initialization $\boldsymbol{x}^{(1)}$ is in a neighborhood of $\boldsymbol{x}^*$, in the sense that

$$\left\|\boldsymbol{A}\boldsymbol{x}^{(1)} - \boldsymbol{A}\boldsymbol{x}^*\right\|_1 \leq c \cdot \min_{i \in S^*} |\boldsymbol{a}_i^\top \boldsymbol{x}^* - y_i|, \tag{13}$$

then $\texttt{IRLS}_p$ *achieves the following superlinear convergence rate of order* $2-p$ *for every* $t \geq 1$*:*

$$\left\|\boldsymbol{A}\boldsymbol{x}^{(t+1)} - \boldsymbol{A}\boldsymbol{x}^*\right\|_1 \leq \mu \cdot \left(\left\|\boldsymbol{A}\boldsymbol{x}^{(t)} - \boldsymbol{A}\boldsymbol{x}^*\right\|_1\right)^{2-p} < \left\|\boldsymbol{A}\boldsymbol{x}^{(t)} - \boldsymbol{A}\boldsymbol{x}^*\right\|_1 \tag{14}$$

In particular, for $p \in [0,1)$*, the superlinear rate* (14) *implies*

$$\left\|\boldsymbol{A}\boldsymbol{x}^{(t+1)} - \boldsymbol{A}\boldsymbol{x}^*\right\|_1 \leq \left(\mu^{1/(1-p)} \cdot \left\|\boldsymbol{A}\boldsymbol{x}^{(1)} - \boldsymbol{A}\boldsymbol{x}^*\right\|_1\right)^{(2-p)^t - 1} \cdot \left\|\boldsymbol{A}\boldsymbol{x}^{(1)} - \boldsymbol{A}\boldsymbol{x}^*\right\|_1. \tag{15}$$

Example 1 (*Quadratic versus linear rates*)**.** We illustrate the practical benefit of a superlinear (quadratic) convergence rate with the following simple calculation. Suppose that the inequality in (13) is barely fulfilled such that in the case of $p = 0$, $\boldsymbol{x}^{(1)}$ satisfies $\mu \cdot \|\boldsymbol{A}\boldsymbol{x}^{(1)} - \boldsymbol{A}\boldsymbol{x}^*\|_1 \leq 0.9999999$. Then (15) implies that after 30 iterations, the residual error is far below numerical precision already since $\|\boldsymbol{A}\boldsymbol{x}^{(31)} - \boldsymbol{A}\boldsymbol{x}^*\|_1 \leq 0.9999999^{2^{30}-1} \cdot \|\boldsymbol{A}\boldsymbol{x}^{(1)} - \boldsymbol{A}\boldsymbol{x}^*\|_1 \approx 10^{-47} \cdot \|\boldsymbol{A}\boldsymbol{x}^{(1)} - \boldsymbol{A}\boldsymbol{x}^*\|_1$. On the other hand, for $p = 1$, a linear convergence factor μ of $\mu \leq 0.9999999$ is only able to guarantee a decay of the order $\mu^{30} \leq 0.9999999^{30} \approx 0.999997$ after 30 iterations.

Discussion. Even though $\texttt{IRLS}_p$ with $p \in (0,1)$ is tailored to ℓ_p-regression, we measure the progress of $\texttt{IRLS}_p$ in ℓ_1-norm, e.g., we provide upper bounds on $\|\boldsymbol{A}\boldsymbol{x}^{(t+1)} - \boldsymbol{A}\boldsymbol{x}^*\|_1$; this is because doing so avoids the use of Hölder's inequality, which allows us to give tighter results that improve on [26] (see the next paragraph). The second point that deserves discussion is the local convergence neighborhood defined by the right-hand side of (13). Note that, we typically assume that every outlier sample $(\boldsymbol{a}_i, y_i)$ (with $i \in S^*$) would result in a relatively large residual at $\boldsymbol{x}^*$, say $|\boldsymbol{a}_i^\top \boldsymbol{x}^* - y_i| \gg 0$. Hence the minimization term of the right-hand side in (13) is well-behaved. Next, one might ask how to obtain such an initialization $\boldsymbol{x}^{(1)}$ that satisfies (13). One might run $\texttt{IRLS}_1$ to produce such an initialization; note though that the upper bound of (13) can not be computed, so one could not detect when to switch from $\texttt{IRLS}_1$ to $\texttt{IRLS}_p$ ($p \in [0,1)$). Or alternatively, one could run $\texttt{IRLS}_p$ with a least-squares initialization and count on empirical global (super)linear convergence of $\texttt{IRLS}_p$; the latter is what we did in the experiments and is what we recommend. A final and important remark is that ℓ_p-regression (1) with $p \in (0,1)$ is in general NP-hard [28, Exercise 2.10], but this does not contradict the theoretical local superlinear convergence of Theorem 2 and the empirical global convergence in Figures 1b and 2b, and this does not mean that $\texttt{IRLS}_p$ solves an NP-hard problem in polynomial time. The catch is that we operate under the assumption that $\boldsymbol{x}^*$ leads to a k-sparse residual and is a unique global minimizer of (1) (cf. Proposition 2). With this assumption and small enough k/m, ℓ_p-regression is tractable and can be solved via $\texttt{IRLS}_p$ to high accuracy (Figure 1b).

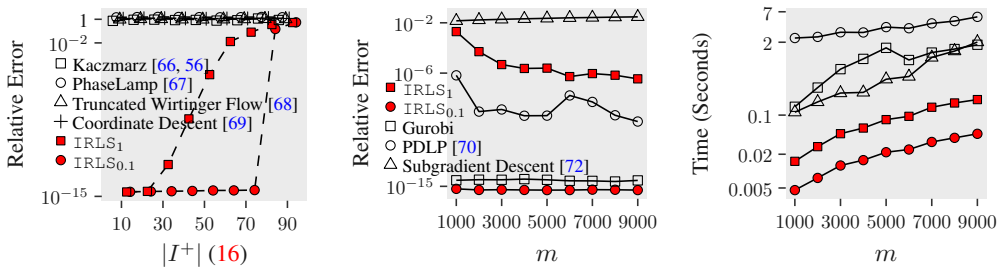
Connection to [26]. [26, Theorem 7.9] proves the local superlinear convergence of IRLS with update rule (6) for ℓ_p-basis pursuit (4), which motivates Theorem 2. However, [26, Theorem 7.9] requires c of (13) to satisfy $c = O(1/m^{2/p-1})$, which means that, as $p \to 0$ or $m \to \infty$, the eligible value of c quickly becomes vanishingly small, and thus the radius of local convergence (13) diminishes greatly. In contrast, in our condition $2c^{1-p}\eta(\eta+1) < (1-c)^{2-p}$, c has no direct dependency on m and is well-behaved even if $p \to 0$. The above difference is due partly to our improved proof strategy, and partly to the different update rules of the smoothing parameter $\epsilon^{(t)}$. Figure 1b showed that rule (6) of [26] does not work well for small p, in contrast to rule (7) that we use; a possible explanation for this phenomenon is that rule (6) decreases $\epsilon^{(t)}$ too fast, yielding a poor initialization for the next iteration of weighted least-squares (similarly to the interior point method, cf. [83]). Finally, [26, Theorem 7.9] does not hold for $p = 0$, for which Theorem 2 still holds and suggests a local quadratic rate.

4 Applications and Experiments

We now explore the performance of $\texttt{IRLS}_p$ in three different applications, *real phase retrieval* (Section 4.1), *linear regression without correspondences* (Section 4.2), and *face restoration* (Section 4.3). Note that the first two applications are special examples of the recent algebraic-geometric framework called *homomorphic sensing* [87], [88, Section 4], [89, Section 1.2]. In Section 4.4, we examine the behavior of $\texttt{IRLS}_p$ for different values of p and under noise.

4.1 Real Phase Retrieval

In real phase retrieval [55, 56], we are given a measurement matrix $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m]^\top$ and $\boldsymbol{y} = [y_1, \ldots, y_m]^\top$, where $y_i := |\boldsymbol{a}_i^\top \boldsymbol{x}^*|$, and we need to find either $\boldsymbol{x}^*$ or $-\boldsymbol{x}^*$. This problem



(a) $\frac{n}{m} = \frac{200}{399}$ (b) $n = 50$, $k/m = 0.5$ (c) $n = 50$, $k/m = 0.5$

Figure 2: Figure 2a: Relative error $\min\{\|\hat{\boldsymbol{x}} - \boldsymbol{x}^*\|_2, \|\hat{\boldsymbol{x}} + \boldsymbol{x}^*\|_2\}/\|\boldsymbol{x}^*\|_2$ of the methods that produce estimates $\hat{\boldsymbol{x}}$ for real phase retrieval. Figures 2b-2c: Relative errors $\|\hat{\boldsymbol{x}} - \boldsymbol{x}^*\|_2/\|\boldsymbol{x}^*\|_2$ and running times for linear regression without correspondences. $\texttt{IRLS}_p$ run for at most 50 iterations. 50 trials.

is a relative of the complex phase retrieval problem [64], where all data $\boldsymbol{y}$ and $\boldsymbol{A}$, as well as the ground-truth $\boldsymbol{x}^*$, are complex-valued, and which has applications in X-ray crystallography [90].

Here we show that real phase retrieval can be solved via ℓ_p-regression (1). Consider the index sets

$$I^+ := \{i : \boldsymbol{a}_i^\top \boldsymbol{x}^* > 0\}, \qquad I^- := \{i : \boldsymbol{a}_i^\top \boldsymbol{x}^* < 0\}. \tag{16}$$

We might assume $\boldsymbol{a}_i^\top \boldsymbol{x}^* \neq 0$ for every i without loss of generality. Then we know that either $\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$ has its ℓ_0 norm equal to $m - |I^-|$, or $\boldsymbol{A}(-\boldsymbol{x}^*) - \boldsymbol{y}$ has its ℓ_0 norm equal to $m - |I^+|$. With such sparsity patterns on these two residuals, we can minimize (1), which serves as a (non-convex) relaxation of ℓ_0-minimization, to recover $\boldsymbol{x}^*$ or $-\boldsymbol{x}^*$, whichever corresponds to a sparser residual. Here, we essentially treat one of the two clusters defined by (16) as inliers, and the other as outliers. This robust regression point of view on real phase retrieval seems to be known by experts [91, Section 1.2], but we have not found a paper that actually proposes to solve (1) for real phase retrieval.

Here we argue with Figure 2a that, solving (1) (by $\texttt{IRLS}_{0.1}$) for real phase retrieval can be beneficial. Indeed, theoretically, $\texttt{IRLS}_{0.1}$ converges to $\pm\boldsymbol{x}^*$ locally but superlinearly (as a corollary of Theorem 2). Empirically, Figure 2a shows that $\texttt{IRLS}_{0.1}$ succeeds in recovering $\pm\boldsymbol{x}^*$ by using only $m = 2n - 1$ samples, which is exactly the theoretical minimum for real phase retrieval [64, Proposition 2.5]. Figure 2a also shows that multiple other state-of-the-art methods[2] [56, 66, 67, 68, 69] fail to recover $\pm\boldsymbol{x}^*$ in this extreme situation, even though some of them have *nearly optimal* sample complexity, which requires roughly $O(n)$ samples up to a logarithmic factor to succeed (*caution*: nearly optimal $\neq$ optimal). As suggested by Proposition 3, for $|I^+|$ fixed, minimizing (1) with a Gaussian matrix $\boldsymbol{A}$ enjoys $m = O(n)$ sample complexity, up to also a logarithmic factor.

Despite these advantages, we also emphasize that solving (1) (via $\texttt{IRLS}_{0.1}$) for real phase retrieval has an inherent limit: The performance depends on the number $|I^+|$ of positive signs and the number m of samples; we might call $|I^+|/m$ the inlier rate or outlier rate. At $n/m = 200/399$, we see that $\texttt{IRLS}_{0.1}$ succeeds if $|I^+| \leq 70$, and if $|I^+| \geq 399 - 70$ by symmetry, but it fails if $|I^+|$ and $|I^-|$ get closer (Figure 2a). For $\texttt{IRLS}_{0.1}$ to successfully handle the case $|I^+| = |I^-|$, we need more samples, empirically, say $m \geq 5n$. It is an interesting future direction to design an algorithm that can handle the case of minimum samples ($m = 2n - 1$) and balanced data ($|I^+| = |I^-|$).

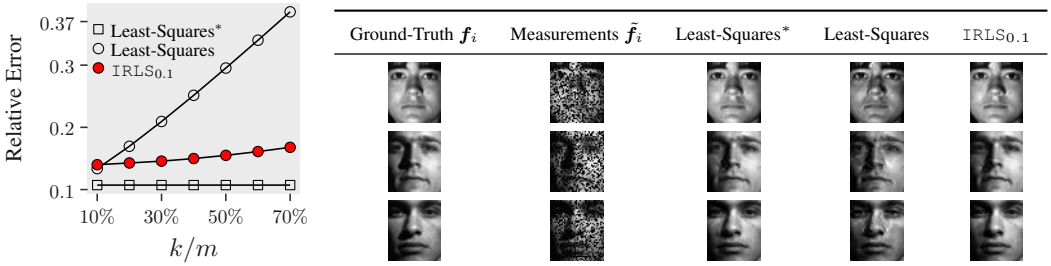
4.2 Linear Regression without Correspondences

Compared to (real) phase retrieval, the problem of *linear regression without correspondences* is an also important, but less developed subject; see [57, 93, 58, 59, 60, 94, 95, 96] for recent advances. In this problem we are given $\boldsymbol{y} \in \mathbb{R}^m$ and $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, and we need to solve the equations

$$\boldsymbol{y} = \boldsymbol{\Pi}\boldsymbol{A}\boldsymbol{x} \tag{17}$$

for some unknown permutation matrix $\boldsymbol{\Pi} \in \mathbb{R}^{m \times m}$ and vector $\boldsymbol{x} \in \mathbb{R}^n$, under the assumption that there exists a solution $(\boldsymbol{\Pi}^*, \boldsymbol{x}^*)$. Solving (17) is in general NP-hard for $n > 1$ [93, 97]. However, if

[2] Reviewing these algorithms is beyond the scope of this paper. Some of them are designed for complex phase retrieval but is applicable to the real case. We used the implementations from PhasePack [92].



(a) Quantitative results

(b) Qualitative results (randomly selected images)

Figure 3: Recover a face image of the Extended Yale B dataset [73] that is corrupted by *salt & pepper* noise [62]. Figure 3a: relative error as a function of the amount of random salt & pepper noise or sparsity level, averaged over 20 trials, all faces, and all individuals. Figure 3b: qualitative results.

we further assume that $\boldsymbol{\Pi}^*$ permutes at most k rows of $\boldsymbol{A}$, then we know that the residual $\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$ is k-sparse. This is an insight of [58], where it was proposed to estimate $\boldsymbol{x}^*$ by solving (1) with $p = 1$.

We show that $\texttt{IRLS}_p$ with $p \in (0, 1)$ is more suitable than several baselines for solving this robust regression problem reduced from linear regression without correspondences. As baselines, we use the PDLP [70] and Gurobi [71] solvers to solve (1) with $p = 1$ as a linear program, and we also use a subgradient descent method implemented in the FOM toolbox of Beck & Guttmann-Beck [72]. Figure 2b shows that $\texttt{IRLS}_{0.1}$ is the most accurate, and Figure 2c shows that $\texttt{IRLS}_{0.1}$ is 30 times faster than Gurobi and subgradient descent, and is more than 100 times faster than PDLP. Interestingly, $\texttt{IRLS}_1$ is not very competitive in terms accuracy (Figure 2b). Moreover, $\texttt{IRLS}_1$ is slower than $\texttt{IRLS}_{0.1}$, as $\texttt{IRLS}_{0.1}$ converges faster and thus terminates earlier than $\texttt{IRLS}_1$; we terminate $\texttt{IRLS}_p$ whenever the objective stops decreasing (up to a tolerance 10^{-15}). Finally, see also [44, 47, 48, 49, 50, 51] where their IRLS variants outperform a different set of baselines (for different problems).

4.3 Face Restoration from Sparsely Corrupted Measurements

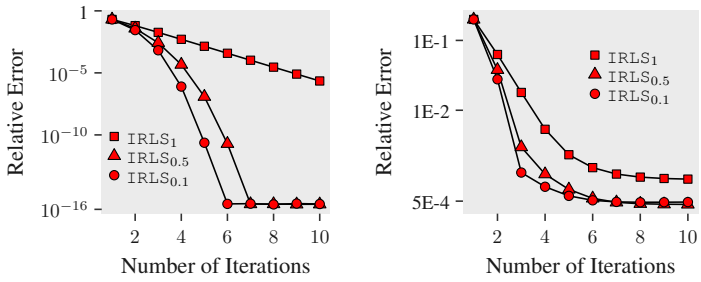
Here we consider a simple face restoration experiment on the Extended Yale B dataset [73], downsampled as per [98]. This dataset contains the face images of 38 individuals under about $n \approx 60$ different illuminations. Let $\boldsymbol{F} = [\boldsymbol{f}_1, \ldots, \boldsymbol{f}_{n+1}]$ be the matrix of faces of the same individual, where each $\boldsymbol{f}_i \in \mathbb{R}^m$ is a (vectorized) face image with $m = 2,016$ pixels. We assume that $\boldsymbol{F}$ is approximately low-rank (which is true under the *Lambertian reflectance* assumption [99]), thus each $\boldsymbol{f}_i$ can be approximately represented as a linear combination of other faces of the individual, i.e., $\boldsymbol{f}_i \approx \boldsymbol{F}_i \boldsymbol{x}_i^*$ for some $\boldsymbol{x}_i^* \in \mathbb{R}^n$, where $\boldsymbol{F}_i \in \mathbb{R}^{m \times n}$ is the same as $\boldsymbol{F}$, except with the i-th column excluded.

In our experiments of Figure 3, we corrupt $\boldsymbol{f}_i$ by random salt & pepper noise [62] and obtain the noisy face $\tilde{\boldsymbol{f}}_i$ as our measurement; so $\boldsymbol{F}_i \boldsymbol{x}_i^* - \tilde{\boldsymbol{f}}_i$ is approximately sparse. We consider three methods to estimate $\boldsymbol{x}_i^*$ and thus to find an estimate $\hat{\boldsymbol{f}}_i$ of $\boldsymbol{f}_i$: (1) Least-Squares that solves (1) with $\boldsymbol{A} = \boldsymbol{F}_i$, $\boldsymbol{y} = \tilde{\boldsymbol{f}}_i$, and $p = 2$, (2) Least-Squares* that solves (1) with $\boldsymbol{A} = \boldsymbol{F}_i$, $\boldsymbol{y} = \boldsymbol{f}_i$, and $p = 2$, used as a golden baseline, and (3) $\texttt{IRLS}_{0.1}$ which solves (1) with $\boldsymbol{A} = \boldsymbol{F}_i$ and $\boldsymbol{y} = \tilde{\boldsymbol{f}}_i$. Figure 3a shows that $\texttt{IRLS}_{0.1}$ is more robust to salt & pepper noise, whose amount is k/m, than Least-Squares, and has lower error $\|\hat{\boldsymbol{f}}_i - \boldsymbol{f}_i\|_2 / \|\boldsymbol{f}_i\|_2$ (recall though that Least-Squares is statistically optimal for Gaussian noise). Figure 3b shows some images randomly chosen from the dataset; observe that Least-Squares tends to "average" the illumination in all images $\boldsymbol{F}_i$, while $\texttt{IRLS}_{0.1}$ delivers faithful restoration.

4.4 The Choice of p and Performance of $\texttt{IRLS}_p$ Under Noise

Here we outline two experiments that illustrate two different aspects of $\texttt{IRLS}_p$ empirically.

In the first experiment, visualized in Figure 4a, we compare the decay of the relative errors of the iterates of $\texttt{IRLS}_p$ for different choices of p in the case that the residual is exactly k-sparse. We observe that $\texttt{IRLS}_1$ exhibits a linear error decay on the one hand and a superlinear decay for $\texttt{IRLS}_p$ with $p = 0.5$ and $p = 0.1$ that accelerates as p decreases towards 0, confirming the rates predicted by Theorem 1 and Theorem 2. ($p = 1, 0.5, 0.1$ are rather arbitrary choices in our experiments.)



(a) The exact k-sparse case (b) The approximate k-sparse case

Figure 4: Empirical convergence rates of $\texttt{IRLS}_p$ for different values of p ($m = 1000, n = 10, k = 200$, averaged over 20 trials). Figure 4a: $\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$ is exactly k-sparse. Figure 4b: $\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$ is approximately k-sparse with 1% noise (see Section D.2 for data generation).

In another experiment, visualized in Figure 4b, we explore the robustness of $\texttt{IRLS}_p$ if the model assumption of k-sparse residuals is not satisfied, but only approximately. In particular, the input vector $\boldsymbol{y} \in \mathbb{R}^m$ of $\texttt{IRLS}_p$ is now such that its restriction on S^* with $|S^*| = k$ has i.i.d. $\mathcal{N}(0, 1)$ entries (corresponding to the sparse corruptions), but its other coordinates are distributed as independent Gaussian variables with mean $\boldsymbol{a}_i^\top \boldsymbol{x}^*$ and variance 0.01, where $i \in (S^*)^c$ (corresponding to dense noise); see also the code provided in Appendix D.2. We observe that also for this case of only approximately k-sparse residuals, $\texttt{IRLS}_p$ performs well, with faster and more accurate convergence for $p = 0.1$ and $p = 0.5$ compared to $p = 1$. We note that, strictly speaking, Theorem 1 and Theorem 2 do not apply directly to this case of approximately k-sparse residuals, but it is not hard to generalize it to this case; see [27, Theorem A.1] for a similar result for IRLS applied to basis pursuit.

5 Discussion and Future Work

In this work, we established novel global linear and local superlinear rates of $\texttt{IRLS}_p$ for ℓ_p-regression (1) under the assumption of the stable range space property. Furthermore, we explored several applications for which $\texttt{IRLS}_p$ exhibits state-of-the-art results.

There are several directions that deserve further investigation. Theoretically, the Gaussian assumption of Proposition 3 might be weakened; similar results would hold for sub-Gaussian distributions [85, 86]. Note also that, in experiments (Figures 1 and 2), $\texttt{IRLS}_p$ works well even when $m - n \to 2k$. This leaves the question of whether the constant or logarithmic factors of condition (10) are too stringent for guaranteeing the stable RSP to hold; can this be improved? Finally, it is tempting to carry out a global rate analysis of $\texttt{IRLS}_p$ with $p \in (0, 1)$ for ℓ_p-regression; can one prove global (linear) rates under the stable RSP or other assumption?

Algorithmically, designing inexact solvers for the inner (weighted) least-squares problem (2) based on Krylov-subspace methods [100, 101] is expected to further accelerate our current IRLS implementation (our current implementation is attached in Appendix E). Also, our empirical experiments suggest that $\texttt{IRLS}_p$ (in particular $p \in (0, 1)$) is worth being adopted and applied to many other outlier-robust estimation tasks beyond robust regression [44, 47, 48, 49, 50, 51, 102, 103, 104, 105, 106, 107].

Acknowledgments and Disclosure of Funding

This work was supported by grants NSF 1704458, NSF 1934979, NSF-IIS-1837991, ONR MURI 503405-78051, and the NSF-Simons Collaboration on the Mathematical Foundations of Deep Learning (NSF grant 2031985).

References

[1] A. El Alaoui, X. Cheng, A. Ramdas, M. J. Wainwright, and M. I. Jordan, "Asymptotic behavior of ℓ_p-based Laplacian regularization in semi-supervised learning," in *Conference on Learning Theory*,

pp. 879–906, 2016.

[2] D. Adil, R. Peng, and S. Sachdeva, "Fast, provably convergent IRLS algorithm for p-norm linear regression," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[3] D. Slepcev and M. Thorpe, "Analysis of p-Laplacian regularization in semisupervised learning," *SIAM Journal on Mathematical Analysis*, vol. 51, no. 3, pp. 2085–2120, 2019.

[4] S. Bubeck, M. B. Cohen, Y. T. Lee, and Y. Li, "An homotopy method for ℓ_p regression provably beyond self-concordance and in input-sparsity time," in *ACM Symposium on Theory of Computing*, pp. 1130–1137, 2018.

[5] D. Adil, R. Kyng, R. Peng, and S. Sachdeva, "Iterative refinement for ℓ_p-norm regression," in *ACM-SIAM Symposium on Discrete Algorithms*, pp. 1405–1424, 2019.

[6] A. Jambulapati, Y. P. Liu, and A. Sidford, "Improved iteration complexities for overconstrained p-norm regression," tech. rep., arXiv:2111.01848v2 [cs.DS], 2021.

[7] R. J. Boscovich, "De litteraria expeditione per pontificiam ditionem, et synopsis amplioris operis, ac habentur plura ejus ex exemplaria etiam sensorum impessa," *Bononiensi Scientiarum et Artum Instuto Atque Academia Commentarii*, vol. 4, pp. 353–396, 1757.

[8] R. L. Plackett, "Studies in the History of Probability and Statistics. XXIX: The discovery of the method of least squares," *Biometrika*, vol. 59, pp. 239–251, 08 1972.

[9] J. Wright and Y. Ma, *High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications*. Cambridge University Press, 2020.

[10] P. J. Huber, "Robust estimation of a location parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964.

[11] E. Candes, M. Rudelson, T. Tao, and R. Vershynin, "Error correction via linear programming," in *IEEE Symposium on Foundations of Computer Science*, pp. 668–681, 2005.

[12] K. Bhatia, P. Jain, and P. Kar, "Robust regression via hard thresholding," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[13] K. Bhatia, P. Jain, P. Kamalaruban, and P. Kar, "Consistent robust regression," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[14] J. Yang, Y.-L. Chow, C. Ré, and M. W. Mahoney, "Weighted SGD for ℓ_p regression with randomized preconditioning," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 7811–7853, 2017.

[15] D. Durfee, K. A. Lai, and S. Sawlani, "ℓ_1 regression using lewis weights preconditioning and stochastic gradient descent," in *Conference On Learning Theory*, pp. 1626–1656, 2018.

[16] A. S. Suggala, K. Bhatia, P. Ravikumar, and P. Jain, "Adaptive hard thresholding for near-optimal consistent robust regression," in *Conference on Learning Theory*, pp. 2892–2897, 2019.

[17] B. Mukhoty, G. Gopakumar, P. Jain, and P. Kar, "Globally-convergent iteratively reweighted least squares for robust regression problems," in *International Conference on Artificial Intelligence and Statistics*, pp. 313–322, 2019.

[18] S. Pesme and N. Flammarion, "Online robust regression via sgd on the ℓ_1 loss," *Advances in Neural Information Processing Systems*, vol. 33, pp. 2540–2552, 2020.

[19] X. Chen and M. Derezinski, "Query complexity of least absolute deviation regression via robust uniform convergence," in *Conference on Learning Theory*, pp. 1144–1179, 2021.

[20] A. Parulekar, A. Parulekar, and E. Price, "ℓ_1 regression with Lewis weights subsampling," tech. rep., arXiv:2105.09433 [cs.LG], 2021.

[21] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[22] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[23] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution," *Communications on Pure and Applied Mathematics*, vol. 59, no. 6, pp. 797–829, 2006.

[24] A. Cohen, W. Dahmen, and R. DeVore, "Compressed sensing and best k-term approximation," *Journal of the American Mathematical Society*, vol. 22, no. 1, pp. 211–231, 2009.

[25] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.

[26] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, "Iteratively reweighted least squares minimization for sparse recovery," *Communications on Pure and Applied Mathematics*, vol. 63, no. 1, pp. 1–38, 2010.

[27] C. Kümmerle, C. Mayrink Verdun, and D. Stöger, "Iteratively reweighted least squares for basis pursuit with global linear convergence rate," *Advances in Neural Information Processing Systems*, 2021.

[28] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. Springer New York, 2013.

[29] R. Chartrand, "Nonconvex compressed sensing and error correction," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 3, pp. III–889, 2007.

[30] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Processing Letters*, vol. 14, no. 10, pp. 707–710, 2007.

[31] R. Chartrand and V. Staneva, "Restricted isometry properties and nonconvex compressive sensing," *Inverse Problems*, vol. 24, no. 3, p. 035020, 2008.

[32] S. Foucart and M.-J. Lai, "Sparsest solutions of underdetermined linear systems via ℓ_q-minimization for $0 < q \leq 1$," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 395–407, 2009.

[33] M. Wang, W. Xu, and A. Tang, "On the performance of sparse recovery via ℓ_p-minimization ($0 \leq p \leq 1$)," *IEEE Transactions on Information Theory*, vol. 57, no. 11, pp. 7255–7278, 2011.

[34] Q. Sun, "Recovery of sparsest signals via ℓ_q-minimization," *Applied and Computational Harmonic Analysis*, vol. 32, no. 3, pp. 329–341, 2012.

[35] D. Ba, B. Babadi, P. L. Purdon, and E. N. Brown, "Convergence and stability of iteratively re-weighted least squares algorithms," *IEEE Transactions on Signal Processing*, vol. 62, no. 1, pp. 183–195, 2013.

[36] L. Zheng, A. Maleki, H. Weng, X. Wang, and T. Long, "Does ℓ_p-minimization outperform ℓ_1-minimization?," *IEEE Transactions on Information Theory*, vol. 63, no. 11, pp. 6896–6935, 2017.

[37] G. Marjanovic and V. Solo, "On ℓ_q optimization and matrix completion," *IEEE Transactions on Signal Processing*, vol. 60, no. 11, pp. 5714–5724, 2012.

[38] K. Mohan and M. Fazel, "Iterative reweighted algorithms for matrix rank minimization," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 3441–3473, 2012.

[39] F. Nie, H. Huang, and C. Ding, "Low-rank matrix recovery via efficient Schatten p-norm minimization," in *AAAI Conference on Artificial Intelligence*, 2012.

[40] C. Kümmerle and J. Sigl, "Harmonic mean iteratively reweighted least squares for low-rank matrix recovery," *The Journal of Machine Learning Research*, vol. 19, no. 1, pp. 1815–1863, 2018.

[41] P. Giampouras, R. Vidal, A. Rontogiannis, and B. Haeffele, "A novel variational form of the Schatten-p quasi-norm," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21453–21463, 2020.

[42] C. Kümmerle and C. Mayrink Verdun, "A scalable second order method for ill-conditioned matrix completion from few samples," in *International Conference on Machine Learning*, pp. 5872–5883, 2021.

[43] G. Lerman, M. B. McCoy, J. A. Tropp, and T. Zhang, "Robust computation of linear models by convex relaxation," *Foundations of Computational Mathematics*, vol. 15, no. 2, pp. 363–410, 2015.

[44] M. C. Tsakiris and R. Vidal, "Dual principal component pursuit," *Journal of Machine Learning Research*, vol. 19, no. 18, pp. 1–50, 2018.

[45] Z. Zhu, Y. Wang, D. Robinson, D. Naiman, R. Vidal, and M. C. Tsakiris, "Dual principal component pursuit: Improved analysis and efficient algorithms," in *Advances in Neural Information Processing Systems*, 2018.

[46] T. Ding, Z. Zhu, T. Ding, Y. Yang, R. Vidal, M. C. Tsakiris, and D. Robinson, "Noisy dual principal component pursuit," in *International Conference on Machine Learning*, pp. 1617–1625, 2019.

[47] K. Aftab and R. Hartley, "Convergence of iteratively re-weighted least squares to robust M-estimators," in *IEEE Winter Conference on Applications of Computer Vision*, pp. 480–487, 2015.

[48] W. Dong, X.-j. Wu, and J. Kittler, "Sparse subspace clustering via smoothed ℓ_p minimization," *Pattern Recognition Letters*, vol. 125, pp. 206–211, 2019.

[49] D. Iwata, M. Waechter, W.-Y. Lin, and Y. Matsushita, "An analysis of sketched IRLS for accelerated sparse residual regression," in *European Conference on Computer Vision*, pp. 609–626, 2020.

[50] T. Ding, Y. Yang, Z. Zhu, D. P. Robinson, R. Vidal, L. Kneip, and M. C. Tsakiris, "Robust homography estimation via dual principal component pursuit," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6080–6089, 2020.

[51] H. Yang, P. Antonante, V. Tzoumas, and L. Carlone, "Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1127–1134, 2020.

[52] Y.-B. Zhao and D. Li, "Reweighted ℓ_1-minimization for sparse solutions to underdetermined linear systems," *SIAM Journal on Optimization*, vol. 22, no. 3, pp. 1065–1088, 2012.

[53] A. Beck, *First-Order Methods in Optimization*. Society for Industrial and Applied Mathematics, 2017.

[54] Y. Nesterov, *Lectures on Convex Optimization*. Springer, 2018.

[55] Y. Chen, Y. Chi, J. Fan, and C. Ma, "Gradient descent with random initialization: fast global convergence for nonconvex phase retrieval," *Mathematical Programming*, vol. 176, no. 1, pp. 5–37, 2019.

[56] Y. S. Tan and R. Vershynin, "Phase retrieval via randomized Kaczmarz: Theoretical guarantees," *Information and Inference: A Journal of the IMA*, vol. 8, no. 1, pp. 97–123, 2019.

[57] J. Unnikrishnan, S. Haghighatshoar, and M. Vetterli, "Unlabeled sensing with random linear measurements," *IEEE Transactions on Information Theory*, vol. 64, no. 5, pp. 3237–3253, 2018.

[58] M. Slawski and E. Ben-David, "Linear regression with sparsely permuted data," *Electronic Journal of Statistics*, vol. 13, no. 1, pp. 1–36, 2019.

[59] M. C. Tsakiris, L. Peng, A. Conca, L. Kneip, Y. Shi, and H. Choi, "An algebraic-geometric approach for linear regression without correspondences," *IEEE Transactions on Information Theory*, vol. 66, no. 8, pp. 5130–5144, 2020.

[60] L. Peng and M. C. Tsakiris, "Linear regression without correspondences via concave minimization," *IEEE Signal Processing Letters*, vol. 27, pp. 1580–1584, 2020.

[61] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2008.

[62] C. Solomon and T. Breckon, *Fundamentals of Digital Image Processing: A practical approach with examples in Matlab*. John Wiley & Sons, 2011.

[63] Y. Yao, L. Peng, and M. Tsakiris, "Unlabeled principal component analysis," *Advances in Neural Information Processing Systems*, 2021.

[64] R. Balan, P. Casazza, and D. Edidin, "On signal reconstruction without phase," *Applied and Computational Harmonic Analysis*, vol. 20, no. 3, pp. 345 – 356, 2006.

[65] T. Strohmer and R. Vershynin, "A randomized Kaczmarz algorithm with exponential convergence," *Journal of Fourier Analysis and Applications*, vol. 15, no. 2, pp. 262–278, 2009.

[66] K. Wei, "Solving systems of phaseless equations via Kaczmarz methods: A proof of concept study," *Inverse Problems*, vol. 31, no. 12, p. 125008, 2015.

[67] O. Dhifallah, C. Thrampoulidis, and Y. M. Lu, "Phase retrieval via linear programming: Fundamental limits and algorithmic improvements," in *Allerton Conference on Communication, Control, and Computing*, pp. 1071–1077, 2017.

[68] Y. Chen and E. Candes, "Solving random quadratic systems of equations is nearly as easy as solving linear systems," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[69] W.-J. Zeng and H.-C. So, "Coordinate descent algorithms for phase retrieval," tech. rep., arXiv:1706.03474 [cs.IT], 2017.

[70] D. Applegate, M. Díaz, O. Hinder, H. Lu, M. Lubin, B. O'Donoghue, and W. Schudy, "Practical large-scale linear programming using primal-dual hybrid gradient," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[71] "Gurobi 9.5.1." https://www.gurobi.com/, Date Accessed: May 10, 2022.

[72] A. Beck and N. Guttmann-Beck, "FOM – a MATLAB toolbox of first-order methods for solving convex optimization problems," *Optimization Methods and Software*, vol. 34, no. 1, pp. 172–193, 2019.

[73] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643–660, 2001.

[74] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[75] G. Lerman and T. Maunu, "Fast, robust and non-convex subspace recovery," *Information and Inference: A Journal of the IMA*, vol. 7, no. 2, pp. 277–336, 2018.

[76] Q. Qu, Z. Zhu, X. Li, M. C. Tsakiris, J. Wright, and R. Vidal, "Finding the sparsest vectors in a subspace: Theory, algorithms, and applications," tech. rep., arXiv:2001.06970 [cs.LG], 2020.

[77] T. F. Chan and P. Mulet, "On the convergence of the lagged diffusivity fixed point method in total variation image restoration," *SIAM Journal on Numerical Analysis*, vol. 36, no. 2, pp. 354–367, 1999.

[78] A. Beck, "On the convergence of alternating minimization for convex programming with applications to iteratively reweighted least squares and decomposition schemes," *SIAM Journal on Optimization*, vol. 25, no. 1, pp. 185–209, 2015.

[79] A. Y. Aravkin, J. V. Burke, and D. He, "IRLS for sparse recovery revisited: Examples of failure and a remedy," tech. rep., arXiv:1910.07095 [math.ST], 2019.

[80] X. Yang, J. Wang, and H. Wang, "Towards an efficient approach for the nonconvex ℓ_p-ball projection: Algorithm and analysis," tech. rep., arXiv:2101.01350v5 [math.OC], 2021.

[81] A. M. Tillmann and M. E. Pfetsch, "The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing," *IEEE Transactions on Information Theory*, vol. 60, no. 2, pp. 1248–1259, 2013.

[82] S. Arora and B. Barak, *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

[83] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[84] Z. Allen-Zhu, "Katyusha: The first direct acceleration of stochastic gradient methods," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 8194–8244, 2017.

[85] G. Lecué and S. Mendelson, "Sparse recovery under weak moment assumptions," *Journal of the European Mathematical Society*, vol. 19, no. 3, pp. 881–904, 2017.

[86] S. Dirksen, G. Lecué, and H. Rauhut, "On the gap between restricted isometry properties and sparse recovery conditions," *IEEE Transactions on Information Theory*, vol. 64, no. 8, pp. 5478–5487, 2016.

[87] M. C. Tsakiris and L. Peng, "Homomorphic sensing," in *International Conference on Machine Learning*, 2019.

[88] L. Peng, B. Wang, and M. Tsakiris, "Homomorphic sensing: Sparsity and noise," in *International Conference on Machine Learning*, pp. 8464–8475, 2021.

[89] L. Peng and M. C. Tsakiris, "Homomorphic sensing of subspace arrangements," *Applied and Computational Harmonic Analysis*, vol. 55, pp. 466–485, 2021.

[90] P. Grohs, S. Koppensteiner, and M. Rathmair, "Phase retrieval: Uniqueness and stability," *SIAM Review*, vol. 62, no. 2, pp. 301–350, 2020.

[91] M. Huang and Y. Wang, "Linear convergence of randomized Kaczmarz method for solving complex-valued phaseless equations," tech. rep., arXiv:2109.11811 [math.NA], 2021.

[92] R. Chandra, T. Goldstein, and C. Studer, "PhasePack: A phase retrieval library," in *International conference on Sampling Theory and Applications*, pp. 1–5, 2019.

[93] A. Pananjady, M. J. Wainwright, and T. A. Courtade, "Linear regression with shuffled data: Statistical and computational limits of permutation recovery," *IEEE Transactions on Information Theory*, vol. 64, no. 5, pp. 3286–3300, 2018.

[94] M. Slawski, G. Diao, and E. Ben-David, "A pseudo-likelihood approach to linear regression with partially shuffled data," *Journal of Computational and Graphical Statistics*, vol. 0, no. 0, pp. 1–31, 2021.

[95] Y. Xie, Y. Mao, S. Zuo, H. Xu, X. Ye, T. Zhao, and H. Zha, "A hypergradient approach to robust regression without correspondence," in *International Conference on Learning Representations*, 2021.

[96] F. Li, K. Fujiwara, F. Okura, and Y. Matsushita, "Generalized shuffled linear regression," in *IEEE/CVF International Conference on Computer Vision*, pp. 6474–6483, 2021.

[97] D. Hsu, K. Shi, and X. Sun, "Linear regression without correspondence," in *Advances in Neural Information Processing Systems*, 2017.

[98] C. You, D. Robinson, and R. Vidal, "Scalable sparse subspace clustering by orthogonal matching pursuit," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3918–3927, 2016.

[99] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 218–233, 2003.

[100] M. Fornasier, S. Peter, H. Rauhut, and S. Worm, "Conjugate gradient acceleration of iteratively re-weighted least squares methods," *Computational Optimization and Applications*, vol. 65, no. 1, pp. 205–259, 2016.

[101] S. Gazzola, C. Meng, and J. G. Nagy, "Krylov methods for low-rank regularization," *SIAM Journal on Matrix Analysis and Applications*, vol. 41, no. 4, pp. 1477–1504, 2020.

[102] A. Chatterjee and V. M. Govindu, "Robust relative rotation averaging," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 958–972, 2017.

[103] C. Sidhartha and V. M. Govindu, "It is all in the weights: Robust rotation averaging revisited," in *International Conference on 3D Vision*, pp. 1134–1143, 2021.

[104] L. Peng, M. C. Tsakiris, and R. Vidal, "ARCS: Accurate rotation and correspondences search," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11153–11163, 2022.

[105] S. H. Lee and J. Civera, "HARA: A hierarchical approach for robust rotation averaging," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15777–15786, 2022.

[106] L. Peng, M. Fazlyab, and R. Vidal, "Semidefinite relaxations of truncated least-squares in robust rotation search: Tight or not," in *European Conference on Computer Vision*, pp. 0–0, Springer, 2022.

[107] L. Carlone, "Estimation contracts for outlier-robust geometric perception," tech. rep., arXiv:2208.10521 [stat.ML], 2022.

[108] X. Chen, "Smoothing methods for nonsmooth, nonconvex minimization," *Mathematical Programming*, vol. 134, no. 1, pp. 71–99, 2012.

[109] Y. Nesterov and V. Spokoiny, "Random gradient-free minimization of convex functions," *Foundations of Computational Mathematics*, vol. 17, no. 2, pp. 527–566, 2017.

[110] A. Beck and S. Sabach, "Weiszfeld's method: Old and new results," *Journal of Optimization Theory and Applications*, vol. 164, no. 1, pp. 1–40, 2015.

[111] K. Lange, *MM Optimization Algorithms*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2016.

[112] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-Minimization algorithms in signal processing, communications, and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 3, pp. 794–816, 2017.

[113] M. Fazel, H. Hindi, and S. P. Boyd, "Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices," in *Proceedings of the American Control Conference*, vol. 3, pp. 2156–2162, 2003.

[114] E. Candès, M. B. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *The Journal of Fourier Analysis and Applications*, vol. 14, pp. 877–905, 2008.

[115] P. Ochs, A. Dosovitskiy, T. Brox, and T. Pock, "On iteratively reweighted algorithms for nonsmooth nonconvex optimization in computer vision," *SIAM Journal on Imaging Sciences*, vol. 8, no. 1, pp. 331–372, 2015.

[116] Y. Gordon, "On Milman's inequality and random subspaces which escape through a mesh in $\mathbb{R}^n$," in *Geometric Aspects of Functional Analysis*, pp. 84–106, Springer, 1988.

[117] M. Rudelson and R. Vershynin, "On sparse reconstruction from Fourier and Gaussian measurements," *Communications on Pure and Applied Mathematics*, vol. 61, no. 8, pp. 1025–1045, 2008.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] Furthermore, we made a lot of efforts in comparing the theoretical results of prior works and position our contribution in the literature.
   (b) Did you describe the limitations of your work? [Yes] See for example at Lines 268-274 on real phase retrieval. We also discussed several directions for improvements or future research at the end of the paper.
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] In particular, we used an entire subsection (Section 3.1) to justify the major assumption, the stable range space property.
   (b) Did you include complete proofs of all theoretical results? [Yes] We omitted the proofs of Proposition 1 and 2 as they follow directly from existing results; see Line 83 and Line 155. All other theoretical results are proved in the appendix.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We attached a copy of our code in the supplemental material.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] In the appendix, we use an entire section to describe our experimental setup. We did not do that in the main paper in interest of space.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] We did not report error bars for two reasons. The main reason is that some of our figures are quite dense, and adding error bars on them makes it harder to read. A slightly subjective reason is that we believe our current figures are informative enough, correctly deliver the message, and are fair for other methods.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes]

(b) Did you mention the license of the assets? [No] In the main paper we used the Extended Yale B face dataset, which is free for research purposes.

(c) Did you include any new assets either in the supplemental material or as a URL? [No]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A IRLS as Smoothing Method Minimizing Quadratic Models

In this section, we clarify the precise relationship between the steps of $\texttt{IRLS}_p$ as described in Algorithm 1 and the ℓ_p-objective on the residual, cf. (1).

A.1 $\texttt{IRLS}_p$ as Lp-Regression for $0 < p \leq 1$

In Section 2.2, we shed light on the different rules for choosing the *smoothing parameter* $\epsilon^{(t+1)}$ in Algorithm 1. In fact, this terminology is used as there is an intimate relationship between the least squares step (2) of $\texttt{IRLS}_p$ and a *smoothed* ℓ_p-objective $H_\epsilon(\cdot)$ which is, for given $\epsilon > 0$, defined as

$$H_\epsilon(\boldsymbol{r}) = \sum_{i=1}^{m} h_\epsilon(r_i) \tag{18}$$

where $h_\epsilon : \mathbb{R} \to \mathbb{R}$ is a symmetric smoothed one-dimensional function such that

$$h_\epsilon(r) = \begin{cases} \frac{1}{p}|r|^p & |r| > \epsilon \\ \frac{1}{2}\frac{r^2}{\epsilon^{2-p}} + \left(\frac{1}{p} - \frac{1}{2}\right)\epsilon^p & |r| \leq \epsilon \end{cases},$$

which makes H_ϵ a function akin to a scaled ℓ_p-quasinorm, but which is quadratic around 0. This definition corresponds to a scaled Huber loss in the case of $p = 1$ [10], but extends to the non-convex case if $0 < p < 1$.

By considering the derivative of $\epsilon \to h_\epsilon(r)$ for fixed r, it is easy to see that $h_\epsilon(r)$, and therefore also $H_\epsilon(\boldsymbol{r})$, is non-decreasing in ϵ, which implies that

$$\frac{1}{p}\left\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\right\|_p^p = \lim_{\epsilon' \to 0} H_{\epsilon'}(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}) \leq H_\epsilon(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}) \tag{19}$$

for any $\epsilon > 0$. We note that $H_\epsilon(\cdot)$ is continuously differentiable, so that the function $\mathbb{R}^m \times \mathbb{R}_{>0} \to \mathbb{R}$, $(\boldsymbol{r}, \epsilon) \mapsto H_\epsilon(\boldsymbol{r})$ can be considered as a *smoothing function* as defined in the non-smooth optimization literature [108, 109].

In this context, one interpretation of $\texttt{IRLS}_p$ (and of other IRLS methods such as [110, 26, 17, 27], with possibly different smoothing functions) is to consider it as a *smoothing method for ℓ_p-regression with quadratic majorizing models*. Indeed, defining the function $Q_\epsilon : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$,

$$Q_\epsilon(\boldsymbol{v}, \boldsymbol{r}) = \sum_{i=1}^{m} q_\epsilon(v_i, r_i), \text{ where } q_\epsilon(v, r) = h_\epsilon(r) + \frac{1}{2} \cdot \frac{v^2 - r^2}{\max\{|r|, \epsilon\}^{2-p}},$$

which is quadratic in $\boldsymbol{r}$, we observe that the least squares step (2) of $\texttt{IRLS}_p$ satisfies

$$\boldsymbol{x}^{(t+1)} = \operatorname*{argmin}_{\boldsymbol{x} \in \mathbb{R}^n} Q_\epsilon(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}, \boldsymbol{r}^{(t)}) \tag{20}$$

where $\boldsymbol{r}^{(t)} = \boldsymbol{A}\boldsymbol{x}^{(t)} - \boldsymbol{y}$. Since $Q_\epsilon(\cdot, \boldsymbol{r})$ locally coincides with the smoothed function $H_\epsilon(\cdot)$ so that

$$H_\epsilon(\boldsymbol{r}) = Q_\epsilon(\boldsymbol{r}, \boldsymbol{r})$$

for each $\boldsymbol{r} \in \mathbb{R}^m$, and furthermore, since it is not hard to show that $Q_\epsilon(\cdot, \boldsymbol{r})$ majorizes $H_\epsilon(\cdot)$ in the sense that

$$H_\epsilon(\boldsymbol{v}) \leq Q_\epsilon(\boldsymbol{v}, \boldsymbol{r}) \tag{21}$$

for any $\boldsymbol{v}, \boldsymbol{r} \in \mathbb{R}^m$ and $\epsilon > 0$, it is possible to establish the monotonous decrease of the smoothed ℓ_p-objective at each iteration of $\texttt{IRLS}_p$ without too much additional work: For each $t = 1, 2, \ldots$ we obtain that

$$H_{\epsilon^{(t+1)}}\left(\boldsymbol{r}^{(t+1)}\right) \leq H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t+1)}\right) \leq Q_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t+1)}, \boldsymbol{r}^{(t)}\right) \leq Q_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}, \boldsymbol{r}^{(t)}\right) = H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right), \tag{22}$$

where the first inequality holds as long as $\epsilon^{(t+1)} \leq \epsilon^{(t)}$, which is satisfied for the smoothing parameter update rules described in Section 2.2.

The monotonicity outlined in (22) enables the following interpretation of the iterates $\{\boldsymbol{x}^{(t)}\}_{t \geq 1}$ of $\texttt{IRLS}_p$: Recalling the definition $\boldsymbol{r}^{(t)} = \boldsymbol{A}\boldsymbol{x}^{(t)} - \boldsymbol{y}$, it follows that without any assumption—in

particular, without assuming any range space property on the feature matrix $\boldsymbol{A}$—each accumulation point of $\{\boldsymbol{r}^{(t)}\}_{t\geq 1}$ is a first-order stationary point of the $\bar{\epsilon}$-smoothed ℓ_p-objective $H_{\bar{\epsilon}}(\cdot)$ of (18) if $\bar{\epsilon}$ is defined as $\bar{\epsilon} = \lim_{t\to\infty} \epsilon^{(t)}$ (see, e.g., [26, Theorem 5.3] for a similar result).

We conclude this section by noting that $\texttt{IRLS}_p$ can be also interpreted within the framework of *majorization-minimization (MM)* [111, 112] algorithms—albeit, with the crucial difference that unlike for MM methods, the smoothing updates of $\texttt{IRLS}_p$ as described in Algorithm 1 change the underlying objective $H_{\epsilon^{(t)}}(\cdot)$ every time when $\epsilon^{(t)}$ changes, i.e., in many cases at each iteration.

A.2 $\texttt{IRLS}_p$ as Sum-of-Logarithm Minimization for $p = 0$

The considerations of Section A.1 are tailored for $\texttt{IRLS}_p$ with $0 < p \leq 1$. However, it is valid to use $\texttt{IRLS}_p$ with $p = 0$ by setting the weights $w_i^{(t+1)}$ accordingly in (3). In this case, it is still possible to interpret $\texttt{IRLS}_0$ as a smoothing method with respect to an underlying smoothed surrogate objective. In particular, we then define $H_\epsilon : \mathbb{R}^m \to \mathbb{R}$ as in (18) with

$$h_\epsilon(r) = \begin{cases} \log(|r|), & \text{if } |r| > \epsilon, \\ \frac{1}{2}\frac{r^2}{\epsilon^2} + \log(\epsilon) - \frac{1}{2}, & \text{if } |r| \leq \epsilon, \end{cases}$$

i.e., a smoothed *sum-of-logarithm* objective [113, 114, 115]. As the inequalities (21) and (22) still hold in this case, we conclude that $\texttt{IRLS}_0$ can be interpreted as a smoothing method for sum-of-logarithm minimization on the residuals $r = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}$. We note that while there is a close relationship between their minimizers, a pointwise majorization of an ℓ_0-objective $\left\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\right\|_0$ by the smoothed objective $H_\epsilon(\cdot)$ as in (19) is *not* possible for $p = 0$.

Another minor difference to the case $0 < p \leq 1$ is that for $p = 0$, the smoothed objective $H_{\epsilon^{(t)}}(\boldsymbol{r}^{(t)})$ of $\texttt{IRLS}_0$ residuals $\boldsymbol{r}^{(t)}$ might converge to $-\infty$ as $\epsilon \to 0$; however, this is rather a technicality than a deterrence for the numerical performance of the algorithm, which remains excellent.

On the other hand, the local convergence analysis put forward in Section 3.3 and Theorem 2 still applies: In particular, we obtain that $\texttt{IRLS}_0$ as described in Algorithm 1 exhibits *local quadratic convergence* under appropriate assumptions on the feature matrix, cf. Theorem 2.

B Proofs of Main Results

B.1 Global Linear Convergence for L1-Regression

Proof of Theorem 1. Define $\boldsymbol{r}^* := \boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$, $\boldsymbol{r}^{(t)} := \boldsymbol{A}\boldsymbol{x}^{(t)} - \boldsymbol{y}$, and $\boldsymbol{d}^{(t)} := \boldsymbol{r}^{(t)} - \boldsymbol{r}^* = \boldsymbol{A}\boldsymbol{x}^{(t)} - \boldsymbol{A}\boldsymbol{x}^*$. Recalling (22), for any $s > 0$ we have the following chain of inequalities:

$$H_{\epsilon^{(t+1)}}\left(\boldsymbol{r}^{(t+1)}\right) \leq H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t+1)}\right) \leq Q_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t+1)}, \boldsymbol{r}^{(t)}\right) \leq Q_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)} - s\boldsymbol{d}^{(t)}, \boldsymbol{r}^{(t)}\right) \tag{23}$$

In (23), the first inequality holds because $\epsilon^{(t+1)} \leq \epsilon^{(t)}$, and $\epsilon \mapsto H_\epsilon(\cdot)$ is a non-decreasing function in ϵ, the second inequality holds because $H_\epsilon(\cdot)$ is majorized by $Q_\epsilon(\cdot, \times)$, see also (22). Furthermore, the third inequality holds due to the optimality of $\boldsymbol{r}^{(t+1)}$ or $\boldsymbol{x}^{(t+1)}$ (cf. (2) and (20)). From (23) we obtain

$$\begin{aligned} H_{\epsilon^{(t+1)}}\left(\boldsymbol{r}^{(t+1)}\right) - H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) &\leq Q_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)} - s\boldsymbol{d}^{(t)}, \boldsymbol{r}^{(t)}\right) - H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) && (24) \\ &= \frac{1}{2}\sum_{i=1}^{m} \frac{\left(r_i^{(t)} - sd_i^{(t)}\right)^2 - \left(r_i^{(t)}\right)^2}{\max\{\left|r_i^{(t)}\right|, \epsilon^{(t)}\}} && (25) \\ &= s\sum_{i=1}^{m} \frac{-r_i^{(t)} \cdot d_i^{(t)}}{\max\left\{\left|r_i^{(t)}\right|, \epsilon^{(t)}\right\}} + \frac{s^2}{2}\sum_{i=1}^{m} \frac{\left(d_i^{(t)}\right)^2}{\max\left\{\left|r_i^{(t)}\right|, \epsilon^{(t)}\right\}}. && (26) \end{aligned}$$

We next derive upper bounds for the two sums of (26) respectively. Denote by S^* the support of $\boldsymbol{r}^*$, and define $R := \{i : |r_i^{(t)}| > \epsilon\} \subset \{1, \dots, m\}$. Then the first sum of (26) is

$$\sum_{i=1}^{m} \frac{-r_i^{(t)} \cdot d_i^{(t)}}{\max\{|r_i^{(t)}|, \epsilon^{(t)}\}} = \sum_{i \in S^*} \frac{-r_i^{(t)}}{\max\{|r_i^{(t)}|, \epsilon^{(t)}\}} \cdot d_i^{(t)} + \sum_{i \in (S^*)^c} \frac{-r_i^{(t)} \cdot r_i^{(t)}}{\max\{|r_i^{(t)}|, \epsilon^{(t)}\}} \tag{27}$$

$$\leq \sum_{i \in S^*} |d_i^{(t)}| - \sum_{i \in (S^*)^c \cap R} \frac{r_i^{(t)} \cdot r_i^{(t)}}{|r_i^{(t)}|} - \sum_{i \in (S^*)^c \cap R^c} \frac{r_i^{(t)} \cdot r_i^{(t)}}{\epsilon^{(t)}} \tag{28}$$

$$\leq (\eta - 1) \sum_{i \in (S^*)^c} |d_i^{(t)}| + \sum_{i \in (S^*)^c \cap R^c} |r_i^{(t)}| - \sum_{i \in (S^*)^c \cap R^c} \frac{r_i^{(t)} \cdot r_i^{(t)}}{\epsilon^{(t)}}. \tag{29}$$

In the last inequality, we applied (k, η)-RSP as our assumption, and used the fact $d_i^{(t)} = r_i^{(t)}$ for $i \in (S^*)^c$. Since $|r_i^{(t)}| \leq r_i^{(t)} \cdot r_i^{(t)}/\epsilon^{(t)} + \epsilon^{(t)}/4$, from the above we now arrive at

$$\sum_{i=1}^{m} \frac{-r_i^{(t)} \cdot d_i^{(t)}}{\max\{|r_i^{(t)}|, \epsilon^{(t)}\}} \leq (\eta - 1) \sum_{i \in (S^*)^c} |d_i^{(t)}| + |(S^*)^c \cap R^c| \cdot \frac{\epsilon^{(t)}}{4} \tag{30}$$

$$\leq (\eta - 1) \sum_{i \in (S^*)^c} |d_i^{(t)}| + \frac{m \cdot \epsilon^{(t)}}{4} \tag{31}$$

$$\leq (\eta - 1) \sum_{i \in (S^*)^c} |d_i^{(t)}| + \frac{1}{4} \sum_{i \in (S^*)^c} |d_i^{(t)}| = (\eta - \frac{3}{4}) \sum_{i \in (S^*)^c} |d_i^{(t)}|. \tag{32}$$

In the last inequality we used the definitions of $d_i^{(t)}$ and $\epsilon^{(t)}$. Next, the second summation of (26) is

$$\sum_{i=1}^{m} \frac{(d_i^{(t)})^2}{\max\{|r_i^{(t)}|, \epsilon^{(t)}\}} \leq \max_i |d_i^{(t)}| \cdot \frac{\sum_{i=1}^{m} |d_i^{(t)}|}{\epsilon^{(t)}} \tag{33}$$

$$\leq \eta(\eta + 1) \sum_{i \in (S^*)^c} |d_i^{(t)}| \cdot \frac{(\eta + 1) \sum_{i \in (S^*)^c} |d_i^{(t)}|}{\epsilon^{(t)}} \tag{34}$$

$$\leq \frac{49}{16} \eta \frac{\left(\sum_{i \in (S^*)^c} |d_i^{(t)}|\right)^2}{\epsilon^{(t)}}. \tag{35}$$

In the above we used (k, η)-stable RSP multiple times. With the two upper bounds on the sums of (26), and with $s' := s\big(\sum_{i \in (S^*)^c} |d_i^{(t)}|\big)$, we now obtain

$$H_{\epsilon^{(t+1)}}\left(\boldsymbol{r}^{(t+1)}\right) - H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) \leq s'(\eta - \frac{3}{4}) + \frac{(s')^2}{2} \cdot \frac{49}{16} \cdot \frac{\eta}{\epsilon^{(t)}}. \tag{36}$$

Since s is an arbitrary non-negative integer, let s be such that $s' = (12 - 16\eta)\epsilon^{(t)}/(49\eta)$. Then

$$H_{\epsilon^{(t+1)}}\left(\boldsymbol{r}^{(t+1)}\right) - H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) \leq -\frac{(3 - 4\eta)^2}{98\eta} \cdot \epsilon^{(t)}. \tag{37}$$

The definition of $\epsilon^{(t)}$ (7) implies there exists some $t' \leq t$ such that $\epsilon^{(t)} = \sigma^{(t')}/m$. Substituting this value of s' into the above inequality gives us the bound

$$H_{\epsilon^{(t+1)}}\left(\boldsymbol{r}^{(t+1)}\right) - \|\boldsymbol{r}^*\|_1 \leq H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) - \|\boldsymbol{r}^*\|_1 - \frac{(3 - 4\eta)^2}{98\eta} \cdot \frac{\sigma^{(t')}}{m} \tag{38}$$

$$\leq H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) - \|\boldsymbol{r}^*\|_1 - \frac{(3 - 4\eta)^2}{98\eta} \cdot \frac{H_{\epsilon^{(t')}}\left(\boldsymbol{r}^{(t')}\right) - \|\boldsymbol{r}^*\|_1}{3m} \tag{39}$$

$$\leq H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) - \|\boldsymbol{r}^*\|_1 - \frac{(3 - 4\eta)^2}{98\eta} \cdot \frac{H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) - \|\boldsymbol{r}^*\|_1}{3m} \tag{40}$$

$$= \left(1 - \frac{(3 - 4\eta)^2}{294\eta m}\right) \cdot \left(H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) - \|\boldsymbol{r}^*\|_1\right). \tag{41}$$

The last two inequalities above are due to Lemma 1 and the monotonicity of $H_{\epsilon^{(t)}}(\boldsymbol{r}^{(t)})$ in t. Finally, Note that $H_{\epsilon^{(t+1)}}(\boldsymbol{r}^{(t+1)})$ majorizes $\|\boldsymbol{r}^{(t+1)}\|_1$ and Lemma 1 implies

$$H_{\epsilon^{(1)}}(\boldsymbol{r}^{(1)}) - \|\boldsymbol{r}^*\|_1 \leq 3\sigma^{(1)} \leq 3\|\boldsymbol{r}^{(1)}\|_1. \tag{42}$$

We have thus obtained

$$\|\boldsymbol{r}^{(t+1)}\|_1 - \|\boldsymbol{r}^*\|_1 \leq H_{\epsilon^{(t+1)}}(\boldsymbol{r}^{(t+1)}) - \|\boldsymbol{r}^*\|_1 \leq \left(1 - \frac{(3-4\eta)^2}{294\eta m}\right)^t \cdot 3\|\boldsymbol{r}^{(1)}\|_1. \tag{43}$$

and the proof is now complete. □

B.2 Local Superlinear Convergence for Lp-Regression

Proof of Theorem 2. We first show that the right inequality of (14), $\mu \cdot \|\boldsymbol{A}\boldsymbol{x}^{(t)} - \boldsymbol{A}\boldsymbol{x}^*\|_1^{1-p} < 1$, holds true. With (13) and the definition of μ, this inequality holds whenever we have

$$2\eta(\eta+1)(1-c)^{p-2} \cdot \min_{i\in S^*} |\boldsymbol{a}_i^\top \boldsymbol{x}^* - y_i|^{p-1} \cdot \left(c \cdot \min_{i\in S^*} |\boldsymbol{a}_i^\top \boldsymbol{x}^* - y_i|\right)^{1-p} < 1, \tag{44}$$

$$\Leftrightarrow\ 2\eta(\eta+1)(1-c)^{p-2}c^{1-p} < 1, \tag{45}$$

which is true by the definition of c. This finishes proving the right inequality of (14). Next, one easily verifies (14) implies (15). Hence, it remains to prove the left inequality of (14).

Recall $\boldsymbol{r}^* := \boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$. Define $\boldsymbol{r}^{(t)} := \boldsymbol{A}\boldsymbol{x}^{(t)} - \boldsymbol{y}$, and $\boldsymbol{d}^{(t)} := \boldsymbol{r}^{(t)} - \boldsymbol{r}^* = \boldsymbol{A}\boldsymbol{x}^{(t)} - \boldsymbol{A}\boldsymbol{x}^*$. And recall that S^* denotes the support of the ground-truth residual $\boldsymbol{r}^*$. We have

$$\sum_{i=1}^m (r_i^* + d_i^{(t+1)}) d_i^{(t+1)} w_i^{(t)} = 0, \tag{46}$$

a known property for the global minimizer $\boldsymbol{x}^{(t+1)}$ of the weighted least-squares problem (2). So

$$\begin{aligned}
\sum_{i=1}^m (d_i^{(t+1)})^2 w_i^{(t)} &= -\sum_{i\in S^*} r_i^* d_i^{(t+1)} w_i^{(t)} && (47)\\
&\leq \sum_{i\in S^*} |r_i^*| \cdot |d_i^{(t+1)}| \max\{|r_i^{(t)}|, \epsilon^{(t)}\}^{p-2} && (48)\\
&\leq \sum_{i\in S^*} |r_i^*| \cdot |d_i^{(t+1)}| \cdot |r_i^{(t)}|^{p-2} && (49)\\
&= \sum_{i\in S^*} |r_i^*| \cdot |d_i^{(t+1)}| \cdot |d_i^{(t)} + r_i^*|^{p-2} && (50)\\
&\leq \sum_{i\in S^*} |r_i^*| \cdot |d_i^{(t+1)}| \cdot |r_i^*|^{p-2} \cdot (1-c)^{p-2}. && (51)
\end{aligned}$$

In the last step, we used the fact $|d_i^{(t)}| \leq c \cdot |r_i^*|$ for all $i \in S^*$ (13), which means $|d_i^{(t)} + r_i^*| \geq (1-c)|r_i^*|$. Thus, with the definition $C := (1-c)^{p-2} \cdot \min_{i\in S^*} |\boldsymbol{a}_i^\top \boldsymbol{x}^* - y_i|^{p-1}$, we get

$$\begin{aligned}
\sum_{i=1}^m (d_i^{(t+1)})^2 w_i^{(t)} &\leq \sum_{i\in S^*} |d_i^{(t+1)}| \cdot |r_i^*|^{p-1} \cdot (1-c)^{p-2}\\
&\leq C \sum_{i\in S^*} |d_i^{(t+1)}|.
\end{aligned} \tag{52}$$

From the above and the (k,η)-stable RSP, we can now arrive at

$$\sum_{i\in(S^*)^c} (d_i^{(t+1)})^2 w_i^{(t)} \leq \sum_{i=1}^m (d_i^{(t+1)})^2 w_i^{(t)} \leq C \sum_{i\in S^*} |d_i^{(t+1)}| \leq \eta \cdot C \sum_{i\in(S^*)^c} |d_i^{(t+1)}|. \tag{53}$$

Next, from the above and the Cauchy–Schwarz inequality, we get that

$$\begin{aligned}\sum_{i\in(S^*)^c}\left|d_i^{(t+1)}\right| &= \sum_{i\in(S^*)^c}\left|d_i^{(t+1)}\right|\cdot\left(w_i^{(t)}\right)^{\frac{1}{2}}\cdot\left(w_i^{(t)}\right)^{-\frac{1}{2}}\\ &\le \sqrt{\sum_{i\in(S^*)^c}\left(d_i^{(t+1)}\right)^2 w_i^{(t)}}\cdot\sqrt{\sum_{i\in(S^*)^c}\left(w_i^{(t)}\right)^{-1}}\\ &\le \sqrt{\eta\cdot C\sum_{i\in(S^*)^c}\left|d_i^{(t+1)}\right|}\cdot\sqrt{\sum_{i\in(S^*)^c}\left(w_i^{(t)}\right)^{-1}}.\end{aligned} \tag{54}$$

We next suppose $d_i^{(t+1)} = r_i^{(t+1)} - r_i^* \neq 0$ for some $i \in (S^*)^c$, for otherwise $\boldsymbol{r}^{(t+1)}$ is k-sparse and thus $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^*$ by the uniqueness of the k-sparse residual. Then the above inequality implies

$$\begin{aligned}\sum_{i\in(S^*)^c}\left|d_i^{(t+1)}\right| &\le \eta\cdot C\sum_{i\in(S^*)^c}\left(w_i^{(t)}\right)^{-1}\\ &= \eta\cdot C\cdot\sum_{i\in(S^*)^c}\max\left\{|r_i^{(t)}|,\epsilon^{(t)}\right\}^{2-p}\\ &\le \eta\cdot C\cdot\left(m\left(\epsilon^{(t)}\right)^{2-p}+\sum_{i\in(S^*)^c}\left|r_i^{(t)}\right|^{2-p}\right)\\ &\le \eta\cdot C\cdot\left(m\cdot\frac{\left(\sigma^{(t)}\right)^{2-p}}{m^{2-p}}+\sum_{i\in(S^*)^c}\left|d_i^{(t)}\right|^{2-p}\right)\\ &\le \eta\cdot C\cdot\left(\frac{1}{m^{1-p}}\cdot\Big(\sum_{i\in(S^*)^c}\left|d_i^{(t)}\right|\Big)^{2-p}+\Big(\sum_{i\in(S^*)^c}\left|d_i^{(t)}\right|\Big)^{2-p}\right)\\ &\le \eta\cdot C\cdot\left(\frac{1}{m^{1-p}}+1\right)\cdot\left(\sum_{i=1}^{m}\left|d_i^{(t)}\right|\right)^{2-p}\\ &\le 2\eta\cdot C\cdot\left(\sum_{i=1}^{m}\left|d_i^{(t)}\right|\right)^{2-p}.\end{aligned} \tag{55}$$

In the above, we used the definitions of $\epsilon^{(t)}, \sigma^{(t)}, r_i^{(t)}, d_i^{(t)}$, and S^*. We have proved (14) since $\sum_{i=1}^m \left|d_i^{(t+1)}\right| \le (1+\eta)\sum_{i\in(S^*)^c}\left|d_i^{(t+1)}\right|$ and by the definition of C. □

C Auxiliary Theoretical Results

Lemma 1 (Upper Bound of The Residual)**.** *Recall $\boldsymbol{r}^* = \boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{y}$. Run* `IRLS`$_1$ *Algorithm 1 with $p = 1$ and the update rule (7) for $\sigma^{(t)}$ and $\epsilon^{(t)}$, which yields the iterates $\{\boldsymbol{x}^{(t)}\}_{t>0}$. If A satisfies (k,η)-stable RSP, then for every $t \ge 1$ the residual $\boldsymbol{r}^{(t)} := \boldsymbol{A}\boldsymbol{x}^{(t)} - \boldsymbol{y}$ satisfies*

$$H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) - \left\|\boldsymbol{r}^*\right\|_1 \le 3\sigma^{(t)}, \tag{56}$$

where $H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right)$ is the smoothed ℓ_p-objective defined in (18) with $p = 1$.

Proof. Recall $R := \{i : \left|r_i^{(t)}\right| > \epsilon\} \subset \{1,\dots,m\}$ and the definition of $H_{\epsilon^{(t)}}$ (18), we have

$$H_{\epsilon^{(t)}}\left(\boldsymbol{r}^{(t)}\right) - \left\|\boldsymbol{r}^*\right\|_1 = \sum_{i\in R} r_i^{(t)} + \sum_{i\in R^c}\frac{1}{2}\left(\frac{r_i^{(t)}}{\epsilon^{(t)}}+\epsilon^{(t)}\right) - \left\|\boldsymbol{r}^*\right\|_1 \tag{57}$$

$$\le \left\|\boldsymbol{r}^{(t)}\right\|_1 + |R^c|\cdot\epsilon^{(t)} - \left\|\boldsymbol{r}^*\right\|_1 \tag{58}$$

$$\le \sigma^{(t)} + \left\|\boldsymbol{r}^{(t)}\right\|_1 - \left\|\boldsymbol{r}^*\right\|_1. \tag{59}$$

Then it remains to prove $\|\boldsymbol{r}^{(t)}\|_1 - \|\boldsymbol{r}^*\|_1 \le 2\sigma^{(t)}$. For this, define $S^{(t)}$ to be the support of the k largest entries of $\boldsymbol{r}^{(t)}$ in absolute values, then $\sum_{i\in(S^{(t)})^c} |r_i^{(t)}| = \sigma^{(t)}$, and we have that

$$\begin{aligned}
\sum_{i\in(S^{(t)})^c} |r_i^{(t)} - r_i^*| &\le \sum_{i\in(S^{(t)})^c} |r_i^{(t)}| + \sum_{i\in(S^{(t)})^c} |r_i^*| && (60)\\
&= \sigma^{(t)} + \|\boldsymbol{r}^*\|_1 - \sum_{i\in S^{(t)}} |r_i^*| && (61)\\
&= 2\sigma^{(t)} + \|\boldsymbol{r}^*\|_1 - \|\boldsymbol{r}^{(t)}\|_1 + \sum_{i\in S^{(t)}} |r^{(t)}| - \sum_{i\in S^{(t)}} |r_i^*| && (62)\\
&\le 2\sigma^{(t)} + \|\boldsymbol{r}^*\|_1 - \|\boldsymbol{r}^{(t)}\|_1 + \sum_{i\in S^{(t)}} |r_i^{(t)} - r_i^*|. && (63)
\end{aligned}$$

Since $\boldsymbol{A}$ satisfies (k,η)-stable RSP and $\boldsymbol{r}^{(t)} - \boldsymbol{r}^*$ is in the range space of $\boldsymbol{A}$, it holds that

$$\begin{aligned}
&\sum_{i\in S^{(t)}} |r_i^{(t)} - r_i^*| \le \eta \sum_{i\in(S^{(t)})^c} |r_i^{(t)} - r_i^*| && (64)\\
\Rightarrow &\sum_{i\in S^{(t)}} |r_i^{(t)} - r_i^*| \le \frac{\eta}{1-\eta}\Big(2\sigma^{(t)} + \|\boldsymbol{r}^*\|_1 - \|\boldsymbol{r}^{(t)}\|_1\Big). && (65)
\end{aligned}$$

In the last inequality we used (63) and rearranged terms. Using (63) and (65) we obtain

$$\begin{aligned}
\|\boldsymbol{r}^{(t)}\|_1 - \|\boldsymbol{r}^*\|_1 &\le \|\boldsymbol{r}^{(t)} - \boldsymbol{r}^*\|_1 && (66)\\
&= \sum_{i\in(S^{(t)})^c} |r_i^{(t)} - r_i^*| + \sum_{i\in S^{(t)}} |r_i^{(t)} - r_i^*| && (67)\\
&\le 2\sigma^{(t)} + \|\boldsymbol{r}^*\|_1 - \|\boldsymbol{r}^{(t)}\|_1 + 2\sum_{i\in S^{(t)}} |r_i^{(t)} - r_i^*| && (68)\\
&\le \frac{1+\eta}{1-\eta}\Big(2\sigma^{(t)} + \|\boldsymbol{r}^*\|_1 - \|\boldsymbol{r}^{(t)}\|_1\Big) && (69)
\end{aligned}$$

which implies

$$\|\boldsymbol{r}^{(t)}\|_1 - \|\boldsymbol{r}^*\|_1 \le \frac{1+\eta}{2}\cdot 2\sigma^{(t)} \le 2\sigma^{(t)}. \tag{70}$$

This finishes the proof. □

C.1 The Stable Range Space Property of Gaussian Matrices

Proof of Proposition 3. We need the notion of *Gaussian width* $w(\cdot)$ of a given set $\mathcal{K}$, defined as

$$w(\mathcal{K}) := \mathbb{E}_{\boldsymbol{g}\sim\mathcal{N}(0,\boldsymbol{I}_m)}\Big[\sup_{\boldsymbol{r}\in\mathcal{K}} \boldsymbol{g}^\top \boldsymbol{r}\Big]. \tag{71}$$

Write $\mathbb{S}^{m-1} := \{v \in \mathbb{R}^m : v^\top v = 1\}$. Denote by $\mathrm{Gr}(n,m)$ the set of n dimensional subspaces of $\mathbb{R}^m$, also known as the *Grassmannian* manifold or variety. With $\eta \in (0,1]$, consider the set

$$\mathcal{T}_k := \Big\{\boldsymbol{r} \in \mathbb{R}^m : \sum_{i\in S} |r_i| > \eta \sum_{i\in S^c} |r_i|, \text{ for some } S \subset \{1,\dots,m\} \text{ with of cardinality } k\Big\}. \tag{72}$$

The following lemma gives an upper bound on the Gaussian width of $\mathcal{T}_k \cap \mathbb{S}^{m-1}$:

Lemma 2 (Remark 9.30 and Proposition 9.33 of [28])**.** *If $m \ge 2k$ then we have:*

$$w\big(\mathcal{T}_k \cap \mathbb{S}^{m-1}\big) \le \sqrt{2k\ln(em/k)}\cdot\Big(1.67 + \eta^{-1}\Big). \tag{73}$$

Lemma 2 and (10) imply that $w\big(\mathcal{T}_k \cap \mathbb{S}^{m-1}\big) \leq \frac{m-n}{\sqrt{m-n+1}}$. Since $\boldsymbol{A}$ has i.i.d. $\mathcal{N}(0,1)$ entries, we can think of its range space $\mathrm{r}(\boldsymbol{A})$ as drawn uniformly at random from the Grassmannian $\mathrm{Gr}(n,m)$. Invoking Lemma 3 with $\mathcal{V} = \mathrm{r}(\boldsymbol{A})$ and $\mathcal{M} = \mathcal{T}_k \cap \mathbb{S}^{m-1}$, we see that

$$\Pr\big(\mathrm{r}(\boldsymbol{A}) \cap \mathcal{T}_k \cap \mathbb{S}^{m-1} = \varnothing\big) \geq 1 - 2.5\exp\bigg(-\frac{1}{18}\Big(\frac{m-n}{\sqrt{m-n+1}} - w(\mathcal{T}_k \cap \mathbb{S}^{m-1})\Big)^2\bigg). \tag{74}$$

Furthermore, Lemma 2 and condition (10) make sure that

$$\frac{m-n}{\sqrt{m-n+1}} - w(\mathcal{T}_k \cap \mathbb{S}^{m-1}) \geq \frac{m-n}{\sqrt{m-n+1}} - \sqrt{2k\ln(em/k)} \cdot \Big(1.69 + \eta^{-1}\Big) \tag{75}$$

$$\geq \sqrt{18\ln(2.5\delta^{-1})}. \tag{76}$$

Combining the above leads us to

$$\Pr\big(\mathrm{r}(\boldsymbol{A}) \cap \mathcal{T}_k \cap \mathbb{S}^{m-1} = \varnothing\big) \geq 1 - 2.5\exp\big(-\ln(2.5\delta^{-1})\big) = 1 - \delta. \tag{77}$$

The event $\mathrm{r}(\boldsymbol{A}) \cap \mathcal{T}_k \cap \mathbb{S}^{m-1} = \varnothing$ implies that the stable (k,η)-stable RSP holds true. □

Lemma 3 (Gordon's Escape Through a Mesh Theorem, Corollary 3.4 of [116], Theorem 4.3 of [117])**.** *Let $\mathcal{V}$ be a n-dimensional subspace of $\mathbb{R}^m$ drawn uniformly at random from the Grassmannian $Gr(n,m)$. Let $\mathcal{M}$ be a subset*[3] *of $\mathbb{S}^{m-1}$. If $w(\mathcal{M}) < \frac{m-n}{\sqrt{m-n+1}}$ then*

$$Pr\big(\mathcal{V} \cap \mathcal{M} = \varnothing\big) \geq 1 - 2.5\exp\bigg(-\frac{1}{18}\Big(\frac{m-n}{\sqrt{m-n+1}} - w(\mathcal{M})\Big)^2\bigg). \tag{78}$$

D Experimental Setup

D.1 How We Run Other Methods

In the real phase retrieval experiment (Figure 2a), all other baselines are implemented in PhasePack [92]. We use the following (quite standard) parameters:

```
opts.tol = 1e-11; opts.initMethod = 'Truncatedspectral';
```

In the experiments of linear regression without correspondences (Figures 2b and 2c), the implementation of PDLP [70] is here, and we run it with the command:

```
julia --project=scripts scripts/solve_qp.jl \
--instance_path test/trivial_lp_model.mps --iteration_limit 500 \
--method pdhg --output_dir [my directory]
```

Note that we run PDLP for a maximum of 500 iterations, while the recommended number of maximum iterations from their GitHub repo is 5000; this is mainly for the sake of efficiency. We use the newest version 9.5.1 of the Gurobi solver with default parameters. we employ the implementation of the FOM toolbox [72] for the (proximal) subgradient descent method. In particular, we invoke the function `prox_subgradient` with the function G being zero and F being $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_1$. We use 0 as initialization, and we observe similar performance when using the least-squares initialization. We set `par.alpha` to be $1/\|\boldsymbol{A}\|_2$, which corresponds to a stepsize of $1/\|\boldsymbol{A}\|_2/(t+1)$, where t is the number of iterations. We make this choice because the default `par.alpha` $= 1$ does not work well and sometimes diverges. Finally, we set the maximum number of iterations to be 10000, as the default choice 1000 leads to an estimate with large errors.

D.2 Synthetic Data Generation

In this section, we provide the Matlab codes generating synthetic data for experiments visualized in Figures 1, 2, and4, detailing the precise data generation process.

[3] The proof of Gordon [116] assumes $\mathcal{M}$ to be closed, but this assumption can be removed in view of the definition of the Gaussian width and the compactness of $\mathbb{S}^{m-1}$. The factor 3.5 of [116] can be improved to 2.5 [117]. To correct [117]; their condition "$w(S) < \sqrt{k}$" should be replaced by "$w(S) < k/\sqrt{k+1}$".

```
%% robust regression
function [y, A, x] = gen_RR(m, n, k, sigma)
    A = randn(m, n); x = randn(n,1);

    y = zeros(m,1);

    idx = datasample(1:m, k, 'Replace', false);

    y(idx) = randn(k,1);

    s = setdiff(1:m, idx);
    y(s) =  A(s, :) * x + sigma * randn(length(s), 1);
end

%% linear regression without correspondences
function [y, A, x] = gen_SLR(m, n, sigma, shuffle_ratio)
    A = randn(m, n); x = randn(n,1);

    w = sigma*randn(m, 1);

    y0 = A*x;
    k = int64(shuffle_ratio * m);
    partial_idx = datasample(1:m, k, 'Replace', false);
    y1 = y0(partial_idx, 1);
    y0(partial_idx, 1) = y1(randperm(k));

    y = y0 + w;
end

%% real phase retrieval
function [y, A, x] = gen_RPR(m, n, num_positive_sign)
    A = randn(m, n); x = randn(n,1); y = zeros(m,1);

    idx = datasample(1:m, num_positive_sign, 'Replace', false);

    y(idx) = A(idx, :) * x;

    s = setdiff(1:m, idx);
    y(s) = - A(s, :) * x;

    % make y positive
    idx = y < 0;
    y(idx) = -y(idx);

    A(idx,:) = -A(idx,:);
end
```

E Implementation

The below is our Matlab code that implements $\texttt{IRLS}_p$:

```
function [x_hat] = IRLSp(A, y, p, k, num_iter)
    [m, n] = size(A);

    q = 2 - p;  l = m - k; epsilon = inf;

    w = ones(m,1); x_old = zeros(n,1);
```

```
    for i = 1:num_iter
        x_hat = (A'* (w.*A)) \  (A' * (w.* y));

        abs_residual = abs(A*x_hat - y);

        sigma = sum(mink(abs_residual, l)) / m;

        epsilon = max(min(epsilon, sigma), 1e-16);

        w = 1./ (max(abs_residual, epsilon)).^(q);

        if norm(x_old - x_hat)/ norm(x_hat) < 1e-15
            break;
        end
        x_old = x_hat;
    end
end
```