






Outlier Robust Multivariate Polynomial Regression

Vipul Arora   


School of Computing, National University of Singapore, Singapore

Arnab Bhattacharyya   

School of Computing, National University of Singapore, Singapore

Mathews Boban   

School of Computing, National University of Singapore, Singapore

Venkatesan Guruswami   

Department of EECS, and Department of Mathematics, University of California, Berkeley, CA, USA

Esty Kelman   

CSAIL, Massachusetts Institute of Technology, Cambridge, MA, USA

Department of Computer Science, and Faculty of Computing & Data Sciences, Boston University, MA, USA

Abstract

We study the problem of *robust multivariate polynomial regression*: let $p: \mathbb{R}^n \rightarrow \mathbb{R}$ be an unknown n -variate polynomial of degree at most d in each variable. We are given as input a set of random samples $(\mathbf{x}_i, y_i) \in [-1, 1]^n \times \mathbb{R}$ that are noisy versions of $(\mathbf{x}_i, p(\mathbf{x}_i))$. More precisely, each \mathbf{x}_i is sampled independently from some distribution χ on $[-1, 1]^n$, and for each i independently, y_i is arbitrary (i.e., an outlier) with probability at most $\rho < 1/2$, and otherwise satisfies $|y_i - p(\mathbf{x}_i)| \leq \sigma$. The goal is to output a polynomial \hat{p} , of degree at most d in each variable, within an ℓ_∞ -distance of at most $O(\sigma)$ from p .

Kane, Karmalkar, and Price [FOCS'17] solved this problem for $n = 1$. We generalize their results to the n -variate setting, showing an algorithm that achieves a sample complexity of $O_n(d^n \log d)$, where the hidden constant depends on n , if χ is the n -dimensional Chebyshev distribution. The sample complexity is $O_n(d^{2n} \log d)$, if the samples are drawn from the uniform distribution instead. The approximation error is guaranteed to be at most $O(\sigma)$, and the run-time depends on $\log(1/\sigma)$. In the setting where each \mathbf{x}_i and y_i are known up to N bits of precision, the run-time's dependence on N is linear. We also show that our sample complexities are optimal in terms of d^n . Furthermore, we show that it is possible to have the run-time be independent of $1/\sigma$, at the cost of a higher sample complexity.

2012 ACM Subject Classification Theory of computation \rightarrow Continuous optimization

Keywords and phrases Robust Statistics, Polynomial Regression, Sample Efficient Learning

Digital Object Identifier 10.4230/LIPIcs.ESA.2024.12

Related Version *Full Version*: [arXiv:2403.09465](https://arxiv.org/abs/2403.09465) [2]

Funding *Vipul Arora*: Supported in part by NRF-AI Fellowship R-252-100-B13-281.

Arnab Bhattacharyya: Supported in part by NRF-AI Fellowship R-252-100-B13-281, Amazon Faculty Research Award, and Google South & Southeast Asia Research Award.

Mathews Boban: Supported in part by NRF-AI Fellowship R-252-100-B13-281.

Venkatesan Guruswami: Supported in part by NSF CCF-2211972 and a Simons Investigator Award.

Esty Kelman: Supported in part by an Amazon Faculty Research Award to AB, in part by ERC grant 834735, and in part by NSF TRIPODS program (award DMS-2022448).

Acknowledgements The authors would like to thank Yuval Filmus for fruitful discussions about some aspects of the robust regression problem.



© Vipul Arora, Arnab Bhattacharyya, Mathews Boban, Venkatesan Guruswami, and Esty Kelman; licensed under Creative Commons License CC-BY 4.0

32nd Annual European Symposium on Algorithms (ESA 2024).

Editors: Timothy Chan, Johannes Fischer, John Iacono, and Grzegorz Herman; Article No. 12; pp. 12:1–12:17

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

“Curve fitting” or *polynomial regression* is one of the oldest and most fundamental learning problems: find a polynomial that approximately satisfies the input-output relationship displayed by a collection of data points. Polynomial regression has a vast range of applications, from the physical sciences to statistics and machine learning; see, e.g., the books [13, 14] for discussions and references.

The focus of this work is on *multivariate* polynomial regression, which is the task of learning the class of bounded degree polynomials from random noisy samples. Multivariate polynomial regression is a natural requirement in many applications. For example, in computer vision, boundaries of objects are often modeled as low-degree bivariate polynomials, so it is well-motivated to fit curves to estimates of object boundaries. Our goal is to design *robust* regression algorithms, which can withstand having a constant fraction of the input data be arbitrary outliers in the same setting as in [1, 6, 8].

We next formally state the problem of robust multivariate regression. Let us denote by \mathcal{P}_d the class of all n -variate *individual* degree- d polynomials, which are the polynomials with degree at most d in each variable¹.

► **Robust Multivariate Polynomial Regression Problem.** Let $\sigma > 0$ be a noise bound, $C > 1$ be an approximation factor, $\rho \in [0, 1]$ be the outlier probability, χ be a probability distribution over $[-1, 1]^n$. Fix an unknown $p \in \mathcal{P}_d$ and let $S = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_M, y_M)\}$ be a set random samples where for each $i \in [M]$ independently, \mathbf{x}_i sampled from χ , and $y_i \in \mathbb{R}$ is an inlier satisfying $|y_i - p(\mathbf{x}_i)| \leq \sigma$ with probability $1 - \rho$ and otherwise, it may be an outlier, i.e., the noise may be arbitrarily large. The goal is to design an efficient algorithm that, given the set S of random samples as input, recovers a polynomial $\hat{p} \in \mathcal{P}_d$ satisfying

$$\max_{\mathbf{x} \in [-1, 1]^n} |p(\mathbf{x}) - \hat{p}(\mathbf{x})| \leq C\sigma,$$

with probability at least $1 - \delta$.

Note that though the locations of the outliers are random, i.e., each sample is an outlier with probability ρ independently, the noise for both the inliers and the outliers is still allowed to be chosen in an adversarial way (meaning an adversary can choose the values of all the y_i 's after seeing the entire sample set $\{\mathbf{x}_i\}$).

In the univariate setting, recovery for non-trivial values of ρ was first shown by Guruswami and Zuckerman [6] for $\rho < 1/\log d$. Previously, Arora and Khot [1] had shown $\rho < 1/2$ was information-theoretically necessary for unique recovery. Subsequently, Kane, Karmalkar, and Price [8] designed a simple and optimal (up to constants) algorithm that runs in polynomial time for any $\rho < 1/2$, uses $\Theta(d \log d)$ samples from the Chebyshev measure on $[-1, 1]$, or $\Theta(d^2)$ uniform samples and outputs a degree- d univariate polynomial \hat{p} satisfying $\max_{x \in [-1, 1]} |p(x) - \hat{p}(x)| \leq C\sigma$. They show how to achieve C as close to 2 as desired. In addition, they show that to solve the problem for $d = 2$ with probability at least $2/3$, $C > 1.09$ is needed, while for general d , to succeed with constant probability, one needs $C > 1 + \Omega(1/d^3)$.

¹ This is in contrast to the usual convention of the *total* degree being at most d . Note that the class of polynomials of *total* degree at most d is strictly included in \mathcal{P}_d . Our results (for \mathcal{P}_d) can be translated for the class of total degree- d polynomials; See discussion in Remark 1.10.

1.1 Main results

We wish to minimize the sample complexity M . Our algorithmic results are mainly when the measure χ is either the uniform distribution or the n -dimensional Chebyshev measure, i.e., the n -fold product of the Chebyshev measure on $[-1, 1]$, with the probability density function $\propto 1/\sqrt{1-x^2}$ for $x \in [-1, 1]$.

Note that when n is large, for some distributions, solving the multivariate polynomial regression problem requires $\exp(n)$ many samples, even for polynomials of *total* degree $d = 1$ (see Theorem 1.5). So, for sample-efficient algorithms, it is prudent to assume $n > 1$ being a constant. In this setup, then, the total degree of an individual degree- d polynomial is at most nd , i.e., $O(d)$, and hence the multivariate polynomial regression problem becomes oblivious to the degree being total or individual. Thus, we focus on learning the class \mathcal{P}_d of *individual* degree- d polynomials in a constant number of variables.

We now state our main results. Denote the cube $[-1, 1]^n$ by \mathcal{C}_n ; we will omit the subscript when the dimension is clear from the context. Let $\|\cdot\|_{\mathcal{C},\infty}$ denote the ℓ_∞ norm² over \mathcal{C}_n .

► **Theorem 1.1.** *Let $\sigma \geq 0, \eta > 0$, and ρ be any constant $< 1/2$. There is an algorithm that almost solves the Robust Multivariate Polynomial Regression Problem with a constant approximation factor, up to an additive error of η . The output of the algorithm is $\hat{p} \in \mathcal{P}_d$ that satisfies*

$$|p(\mathbf{x}) - \hat{p}(\mathbf{x})| \leq O(\sigma) + \eta, \quad \text{for all } \mathbf{x} \in \mathcal{C}_n,$$

with probability at least $2/3$. It uses $M = O_n(d^n \log d)$ samples drawn from the multidimensional Chebyshev distribution, or $M = \tilde{O}_n(d^{2n})$ if the samples are drawn from the uniform measure. Its run-time is at most $\text{poly}(\log\|p\|_{\mathcal{C},\infty}, M, \log(1/\eta))$.

The notations $\tilde{O}, \tilde{\Theta}$ hide factors proportional to $\log d$ above; the dependence on η , or σ is kept explicit. The n in the subscripts denotes that it is the non-asymptotic parameter.

► **Remark 1.2.** One may consider the case of non-constant values of $\rho < 1/2$. Here, the number of samples increases as $\rho \rightarrow 1/2$, since the dependence of M on ρ is $M \propto 1/(1-2\rho)^2$.

In case σ is known to be at least 2^{-N} , one may choose $\eta = 2^{-N}$ to guarantee $\|\hat{p} - p\|_{\mathcal{C},\infty} \leq O(\sigma)$ and run-time proportional to $\text{poly}(N)$. Generalizing this observation, we consider the N -bit precision setting, where both the sample locations \mathbf{x}_i and the labels y_i are truncated to N bits of precision; this is consistent with a computational model where real numbers can only be specified up to N bits of precision. We show that in the N -bit precision setting, a variant of our algorithm achieves a constant approximation factor without any additional additive error.

► **Theorem 1.3.** *Let N be the number of bits of precision, $\sigma \geq 2^{-N}$, and ρ be any constant $< 1/2$. There exists an algorithm for the Robust Multivariate Polynomial Regression Problem, wherein each \mathbf{x}_i is now drawn from a continuous distribution χ and then rounded to N bits of precision, and each y_i is similarly rounded. The output of the algorithm is $\hat{p} \in \mathcal{P}_d$, that satisfies*

$$|p(\mathbf{x}) - \hat{p}(\mathbf{x})| \leq O(\sigma), \quad \text{for all } \mathbf{x} \in \mathcal{C}_n,$$

with probability at least $2/3$. It uses $M = O_n(d^n \log d)$ samples drawn from the multidimensional Chebyshev distribution, or $M = \tilde{O}_n(d^{2n})$ if the samples are drawn from the uniform measure. Its run-time is at most $\text{poly}(\log\|p\|_{\mathcal{C},\infty}, M, N)$.

² See formal Definition 2.1.

12:4 Outlier Robust Multivariate Polynomial Regression

To avoid a run-time dependent on $\|p\|_{C,\infty}$ and $1/\eta$, in case they are unknown or too large, we also obtain a variant of the algorithm that achieves an explicit constant multiplicative approximation factor, as close to 2 as desired and independent of $\|p\|_{C,\infty}$ and $1/\eta$, at the cost of a higher sample complexity.

► **Theorem 1.4.** *Let $\varepsilon > 0$, $\sigma \geq 0$, and a constant $\rho < 1/2$. There exists an algorithm that solves the Robust Multivariate Polynomial Regression Problem. The output of the algorithm is $\hat{p} \in \mathcal{P}_d$, that satisfies*

$$|p(\mathbf{x}) - \hat{p}(\mathbf{x})| \leq (2 + \varepsilon)\sigma \quad \text{for all } \mathbf{x} \in \mathcal{C}_n,$$

with probability at least $2/3$. It uses $M = \text{poly}(d^{n^2}, 1/\varepsilon^n)$ samples drawn from either the multidimensional Chebyshev distribution or the uniform distribution. Its run-time is $\text{poly}(M)$.

We complement the above results by showing lower bounds on the sample complexity of robust multivariate polynomial regression.

► **Theorem 1.5.** *For any constant approximation factor $C > 1$, and any $\sigma < \frac{1}{2C}$, given $M = e^{o(n\sigma^2)}$ samples, drawn from any product distribution with mean 0, no algorithm can solve the Robust Multivariate Polynomial Regression Problem with failure probability $\delta < 1/4$, for any ρ (even for $\rho = 0$, i.e., even without outliers).*

In particular, for constant noise level σ , any algorithm requires, $e^{\Omega(n)}$ many samples to succeed with probability at least $3/4$.

This motivates our setting where $n > 1$ is constant, and d is the asymptotic parameter.

► **Theorem 1.6.** *For any approximation factor $C > 1$, there exists $c = c(C) > 0$ such that any algorithm, that solves the Robust Multivariate Polynomial Regression Problem, requires at least $(cd)^{2n}$ samples drawn from the uniform measure to succeed with probability more than $2/3$. This holds for any outlier probability ρ .*

This lower bound matches the upper bound of Theorem 1.3 up to lower order terms (in the case of uniform sampling) for constant n , and C , and holds even for $\rho = 0$, where there are no outliers.

The following result shows that our result in Theorem 1.3 for the multidimensional Chebyshev measure matches the optimal sample complexity over arbitrary distributions³.

► **Theorem 1.7.** *For any approximation factor $C > 1$, and any outlier probability $\rho > 0$, there exists $c = c(C, \rho) > 0$ such that any algorithm, that solves the Robust Multivariate Polynomial Regression Problem, requires at least $(cd)^n \log d$ samples drawn from any measure over $[-1, 1]^n$ to succeed with probability more than $2/3$.*

► **Remark 1.8.** The reason for choosing the Chebyshev measure is that it matches the distribution-free lower bound of Theorem 1.7. Even for the univariate case, as shown by KKP, the Chebyshev measure yields an optimal sample complexity of $\Theta(d \log d)$. While for uniform sampling, they show an $\Omega(d^2)$ lower bound, which inspires an $\Omega(d^{2n})$ lower bound of Theorem 1.6 in the n -variate case. Intuitively, this tightness is because of a classical result in approximation theory that the optimal points for polynomial interpolation are the Chebyshev nodes (roots of Chebyshev polynomials).

A comparison of the results across parameter regimes may be given via the following Table 1, wherein M is the sample complexity, and $C_p = \log \|p\|_{C,\infty}$:

³ Similarly, the lower bounds match the respective sample complexities of Theorem 1.1, when the additive error η approaches 0, since the algorithm's run-time grows as $\eta \rightarrow 0$, but the sample complexities remain unchanged.

■ **Table 1** The upper bounds in the first row follow from Theorem 1.1, the second row from Theorem 1.3, and the third row from Theorem 1.4. The lower bounds in the second row for the Uniform Measure follow from Theorem 1.6, and for the Chebyshev Measure from Theorem 1.7.

Setting	Approximation	Chebyshev Measure	Uniform Measure	Run-time
Exact	$O(\sigma) + \eta$	$O_n(d^n \log d)$	$\tilde{O}_n(d^{2n})$	$\text{poly}(C_p, M, \log(1/\eta))$
N -bit	$O(\sigma)$	$\Theta_n(d^n \log d)$	$\tilde{\Theta}_n(d^{2n})$	$\text{poly}(C_p, M, N)$
Small ε	$(2 + \varepsilon)\sigma$	$\text{poly}(d^{n^2}, 1/\varepsilon^n)$	$\text{poly}(d^{n^2}, 1/\varepsilon^n)$	$\text{poly}(M)$

► **Remark 1.9.** [8] were able to achieve both optimal sample complexity and efficient run-time independent of $\|p\|_\infty/\sigma$ with a single algorithm in the univariate setting. In contrast, as we elaborate in the proof overview, our Theorem 1.4 incurs an additional blowup in the sample complexity; we leave open the problem of realizing the error guarantees of Theorem 1.4 with the optimal number of samples.

► **Remark 1.10.** Both our upper and lower bounds hold for the class of total degree- d polynomials, when n is bounded, since total degree being at most d implies individual degree being at most d , and individual degree being at most d implies total degree being at most dn .

1.2 Main technical contributions

Our main technical contributions are twofold, and they may be of interest more broadly. First, consider some $m \geq d$, and let $\{\mathcal{C}_j\}_{j \in [m]^n}$ be a partition of the cube \mathcal{C}_n induced by the m -Chebyshev extremas on each axis. We call it the (m, n) -Chebyshev partition⁴ of \mathcal{C}_n . For $n = 1$, [8] showed how to approximate a univariate polynomial of degree at most d on $[-1, 1]$ by an appropriate piece-wise constant function with respect to the unidimensional Chebyshev partition. We extend their result to the multivariate case.

► **Theorem 1.11.** *[Multivariate Approximation by piece-wise constant functions] Let $p : \mathcal{C}_n \rightarrow \mathbb{R}$ be a polynomial of degree at most d in each variable, and $m \geq d$. Let $r : \mathcal{C}_n \rightarrow \mathbb{R}$ be a piece-wise constant function with respect to the (m, n) -Chebyshev partition, such that for every $\mathbf{j} \in [m]^n$, there exist a point $\mathbf{x}^{(\mathbf{j})} \in \mathcal{C}_{\mathbf{j}}$, such that $r(\mathbf{x}) = p(\mathbf{x}^{(\mathbf{j})})$, for all $\mathbf{x} \in \mathcal{C}_{\mathbf{j}}$. Then, there exists a universal constant C such that,*

$$\|p - r\|_{\mathcal{C}_{n,\infty}} \leq C \frac{dn}{m} \|p\|_{\mathcal{C}_{n,\infty}}.$$

Second, we show how to relate the maximum value of a bounded degree polynomial on \mathcal{C}_n with its ℓ_1 norm, on the same cube \mathcal{C}_n .

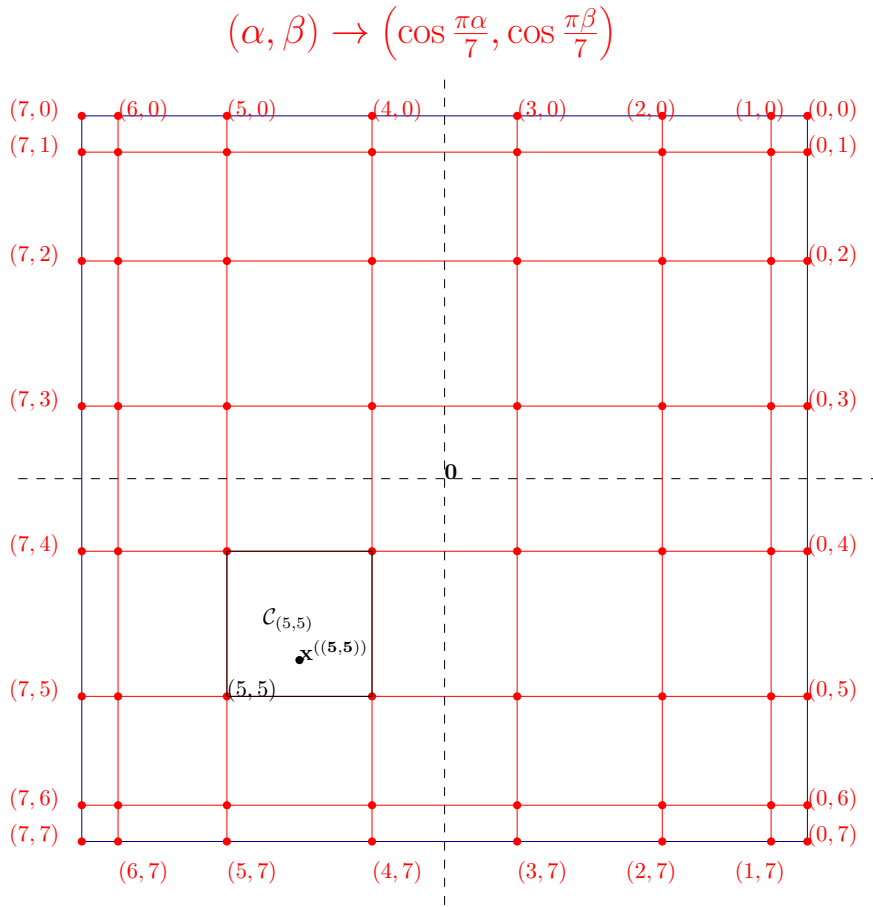
► **Theorem 1.12.** *There exists a global constant $C > 0$ such that, for every $p \in \mathcal{P}_d$:*

$$\|p\|_{\mathcal{C}_{n,\infty}} \leq C^n d^{2n} \|p\|_{\mathcal{C}_{n,1}}.$$

We also note the tightness of the relation between ℓ_∞ and ℓ_1 norms, using an observation from [7].

► **Proposition 1.13.** *There exists a global constant $c > 0$ such that for every odd d , there exists a family of individual degree- d polynomials $\{f_n\}_{n \in \mathbb{N}}$, where $f_n : \mathcal{C}_n \rightarrow \mathbb{R}$, such that $\|f_n\|_{\mathcal{C}_{n,\infty}} \geq c^n d^{2n} \|f_n\|_{\mathcal{C}_{n,1}}$.*

⁴ See formal Definitions 2.4–2.6, and Figure 1 for an illustration of a 2-dimensional Chebyshev partition.



■ **Figure 1** An illustration of a 2-dimensional $(7, 2)$ -Chebyshev partition (in red) super-imposed on the 2-dimensional solid cube $\mathcal{C}_2 = [-1, 1]^2$, with boundary in blue. The cells are indexed by their bottom-left Chebyshev extremas (the red points). Theorem 1.11 essentially proves that on any cell, for example, $\mathcal{C}_{(5,5)}$ (in black), any $p \in \mathcal{P}_d$ can be well approximated by its evaluation on one arbitrary point $\mathbf{x}^{((5,5))} \in \mathcal{C}_{(5,5)}$. As the partition grows finer, the approximation gets better.

1.3 Related Work

Given the fundamental nature of the polynomial regression problem, there is a long history of work on the problem, but mostly in the univariate setting. Arora and Khot [1] were the first to study this problem in our random outlier noise model, giving an algorithm that in $O(\frac{d^2}{\sigma} \log \frac{d}{\sigma})$ random noisy samples outputs an $O(\sigma)$ -approximation (in ℓ_∞) to the (actual) hidden polynomial, where the outlier rate $\rho = 0$. This was improved in a work by Guruswami and Zuckerman [6], who gave a computationally efficient algorithm for all $\rho < 1/\log d$. Finally, in a significant improvement, Kane, Karmalkar and Price [8] obtained computationally efficient algorithms for any $\rho < 1/2$, while having no additional requirements for σ or $\|p\|_\infty$. As far as we know, Daltrophe, Dolev and Lotker [3] were the first to consider the *multivariate* setting of this problem. For the two-dimensional case ($n = 2$), they gave an algorithm that with $O(\frac{d^4}{\sigma} \log \frac{d}{\sigma})$ random noisy samples outputs a $c(2) \cdot \sigma$ -approximation (in ℓ_∞), for any $\rho < \frac{1}{2}$, where $c(2) = 3$. A limitation of their result is that $c(n)$ grows exponentially in n . In contrast, we obtain a constant factor approximation for all n .

There has also been a surge of recent research in the related area of robust statistics. Here, instead of the outliers being randomly placed, their locations are chosen adversarially. For the setting when the *total* degree is fixed, and the dimension n is growing, Klivans, Kothari and Meka [9] gave an algorithm using the sum-of-squares method. However, their sample complexity is $\text{poly}(n^d)$, which is exponential in the degree; moreover, the output guarantee is with respect to the $\|\cdot\|_2$ norm, instead of the $\|\cdot\|_\infty$ norm in our setting. Other related works in this spirit are that of Diakonikolas, Kamath, Kane, Li, Steinhardt and Stewart [4] and Prasad, Suggala, Balakrishnan and Ravikumar [12]. The work of Diakonikolas, Kong, and Stewart [5] also studied the related problem of adversarially robust linear regression, but with the assumption that the \mathbf{x}_i 's are drawn from a Gaussian.

1.4 Technical Overview

We first sketch the algorithm designed by Kane, Karmalkar, and Price [8], henceforth KKP, and their analysis for the univariate case, $n = 1$. For univariate polynomial interpolation, the points at which the noisy samples are located play an important role in determining the interpolation error. Choosing the points to be the Chebyshev nodes, which are the roots of Chebyshev polynomials (see Definition 2.4), is a good starting point, as suggested by approximation theory literature. However, the algorithm receives random samples, which may not necessarily be located at the Chebyshev nodes. Instead, KKP argue that they have enough inliers around each Chebyshev node. For this, they define a partition of the interval $[-1, 1]$ on the extremal points of Chebyshev polynomials, which they call *the size- m Chebyshev partition*. In their algorithm and its analysis, they assume that the set of samples is *good*, in the sense that in every part of the partition, there is only a small fraction of outliers; this good event is guaranteed to happen with high probability.

1.4.1 KKP's Algorithm and Analysis

Formally, the *size- m Chebyshev partition* of $[-1, 1]$ is the set of intervals

$$\left\{ I_j = \left[\cos \frac{\pi j}{m}, \cos \frac{\pi(j-1)}{m} \right] \subsetneq [-1, 1], \forall j \in [m] \right\}.$$

Given a set of s samples (x_i, y_i) where x_i 's are drawn from some distribution over $[-1, 1]$, and y_i 's are the corresponding labels, the algorithm uses the idea of *median-based recovery*. For every interval I_j :

1. Let \tilde{y}_j be the median of y_i 's of samples for which $x_i \in I_j$. Since the set of samples is assumed to be *good*, i.e., the fraction of outliers in each interval is strictly less than $1/2$, \tilde{y}_j lies in between two inliers located in the interval I_j .
2. Let \tilde{x}_j be an arbitrary point in I_j .
3. Let \hat{p} be a minimizer, over all degree- d polynomials, of the empirical ℓ_∞ error $\max_j |\hat{p}(\tilde{x}_j) - \tilde{y}_j|$ over all $j \in [m]$.

As m grows, the partition gets finer, and the error gets better, though at a cost of higher sample complexity. Iteratively applying the median-based recovery on the residual left from previous iteration improves the approximation, and in $\log(\max_{x \in [-1, 1]} |p(x)|/\sigma)$ iterations, the error drops down to 3σ .

The backbone of their analysis is a technical result for approximating p on a size- m Chebyshev partition, by a piece-wise constant function (with respect to the same partition) that matches p on at least one point in every part of the partition.

► **Lemma 1.14.** [Lemma 2.1, [8]] Let $g : \mathbb{R} \rightarrow \mathbb{R}$ be a (univariate) degree- d polynomial. Let $\{I_j\}_{j \in [m]}$ denote the m -Chebyshev partition of $[-1, 1]$, for some $m \geq d$. Let $r : [-1, 1] \rightarrow \mathbb{R}$ be piece-wise constant, so that for each $k \in [m]$, there exists $x_k^* \in I_k$, such that $r(x) = g(x_k^*)$ for all $x \in I_k$. Then, there exists a universal constant C such that, for any $q \geq 1$,

$$\|g - r\|_q \leq \frac{Cd}{m} \|g\|_q.$$

To prove Lemma 1.14, they used Nevai’s inequality [11], an ℓ_q -version of Bernstein’s inequality, to bound the ℓ_q approximation error by a multiple of the ℓ_q norm of p . The multiple is linear in the degree d , and $1/m$. The bound from Nevai’s inequality works for all “inner” parts of the Chebyshev partition, as it relies on the fact that the length of any part I_j , where $j \notin \{1, m\}$, is at most $O(\sqrt{1 - x^2}/m)$, for every $x \in I_j$. To bound the approximation error on the peripheral parts I_1, I_m , they use Markov Brothers’ inequality (Lemma 2.3). Here they strongly rely on the fact that those parts are much narrower⁵ ($|I_1| = |I_m| = O(1/m^2)$) than the inner parts. This additional $1/m$ factor in the length compensates for the worse bound from Lemma 2.3.

Lemma 1.14, with q set to ∞ , is used to bound the error of the median-based recovery procedure, in terms of $\|p\|_{C_{1,\infty}}$. This allows the $\log \frac{\|p\|_{C_{1,\infty}}}{\sigma}$ iterations to be all that is further needed to bring the error down to 3σ .

To avoid the run-time dependence on $\max_{x \in [-1,1]} |p(x)|/\sigma$, which maybe unknown or too large, KKP first run an ℓ_1 regression, which gives an ℓ_∞ error of at most $O(d^2\sigma)$, and then run the median-based recovery algorithm on the residual polynomial, which in $\log d$ iterations drops the error further to at most 3σ . Lemma 1.14, with $q = 1$, is used to bound the ℓ_1 -error of the ℓ_1 -minimizer by $O(\sigma)$. A further application of Lemma 2.3 bounds the ℓ_∞ -error of the ℓ_1 -minimizer by $O(d^2\sigma)$. This then allows for a bound of $\log d$ on the number of iterations needed, and hence the algorithm’s run-time.

1.4.2 Our Results

Generalizing to the multivariate case ($n > 1$)

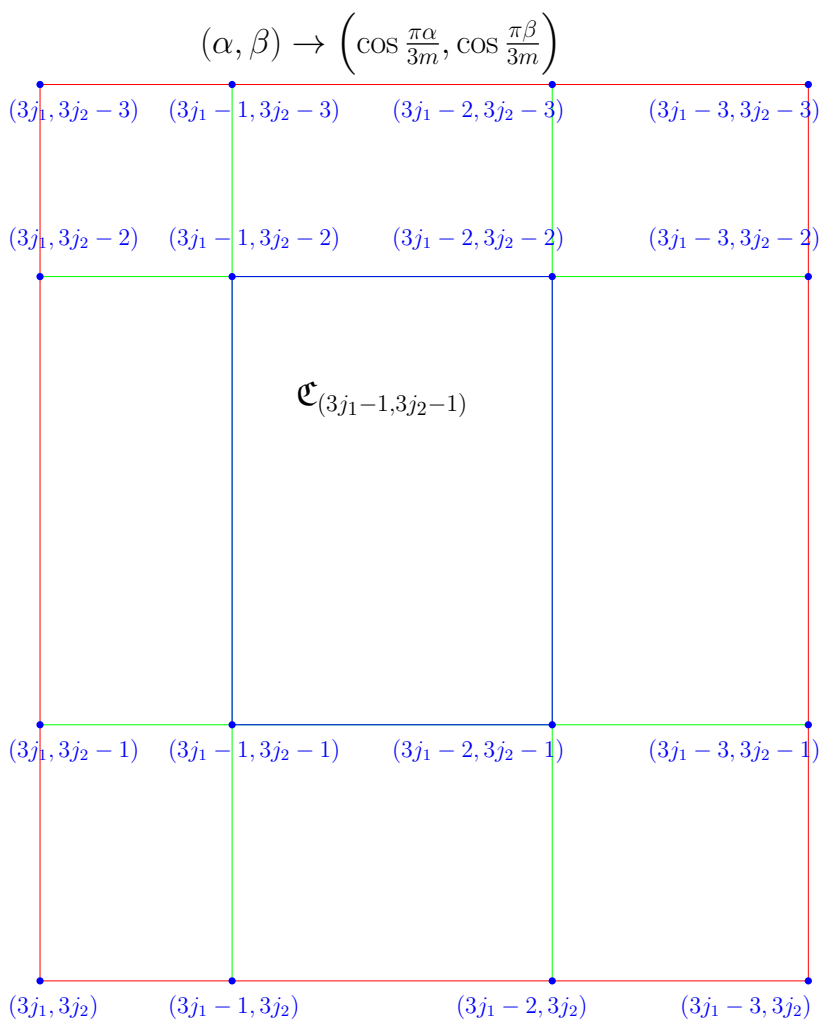
We show that the idea of KKP generalizes to the multivariate setting by considering a tensorization of the Chebyshev partition, i.e., we divide the cube $[-1, 1]^n$ into m^n cells according to a grid partition, where each axis is divided into m intervals, according to the size- m Chebyshev partition of $[-1, 1]$ defined by KKP. The analysis takes steps similar to the analysis done by KKP, and some of the proofs follow by “tensoring” KKP’s arguments in some sense.

There are some subtleties that we take care of along the way. We successfully show optimal sample complexity results, in terms of the dependence on d^n , for the median-based recovery algorithm, while for the ℓ_1 regression, we need more samples. For this reason, we first analyze the median based recovery algorithm, the running time (but not the sample complexity) of which, depends on $\max_{\mathbf{x} \in C_n} |p(\mathbf{x})|$. Later, we show that by running the ℓ_1 regression on weighted averages (with respect to cells) as the first step, we reach a constant approximation factor in bounded run-time at the cost of increasing the exponent in the sample complexity from n to $O(n^2)$.

⁵ A pictorial demonstration of this narrowness, for 2-dimensional partitions, can be observed in Figure 1.

Overview of the algorithms and analyses

For using the median-recovery algorithm, we devise a multivariate analog (Theorem 1.11) of Lemma 1.14 for the ℓ_∞ norm. Specifically, we show that, for large enough m , every n -variate, individual degree- d polynomial p is well approximated by any piece-wise constant function with respect to the (m, n) -Chebyshev partition that matches p on at least one point in each cell. This is proved by a repeated application of the *univariate* ℓ_∞ approximation statement from Lemma 1.14. Algorithmically, we then do median-based recovery on a fine enough Chebyshev partition of \mathcal{C}_n , and iteratively improve the output of the ℓ_∞ regression. After at most $\log(\|p\|_{\mathcal{C}_n, \infty}/\eta)$ iterations, we achieve an $O(\sigma) + \eta$ approximation. A poly($\log \|p\|_{\mathcal{C}_n, \infty}, M, \log(1/\eta)$) run-time is thus achieved. One may set $\eta = \sigma$ to achieve an $O(\sigma)$ approximation, in this case the run-time is dependent on $\log(1/\sigma)$ instead.



■ **Figure 2** An illustration of cell-refinement in 2-dimensional Chebyshev grids: a $(3m, 2)$ -grid (in green) super-imposed on a $(m, 2)$ -Chebyshev cell $\mathcal{C}_{(j_1, j_2)}$ (in red). The samples from middle-most cell $\mathcal{C}_{(3j_1-1, 3j_2-1)}$ (in blue) only are retained, and median-recovery is applied on them.

We also consider the *finite bit precision* setting where the samples are represented using at most N bits of precision. This forces $\sigma \geq 2^{-N}$, and the (location, evaluation) pairs of the random input samples are now rounded to N bits. In this case, the samples' locations

12:10 Outlier Robust Multivariate Polynomial Regression

are not exact, and hence we are uncertain as to which Chebyshev cell they belong to. To deal with it, we discard samples that lie in a small ℓ_1 neighborhood of the boundary of the cells. (See Figure 2 for an illustration.) We then apply the median-based recovery algorithm on only the remaining samples in the cells' interior. The interior *refined* sample points, by virtue of being far enough from their nearest cell boundary, would have remained in their respective cells, even after suffering from the rounding noise. Hence, we only have to ensure that all the interior regions have enough *good* samples, which we show increases the sample complexity by a factor dependent only on n . It still gives a tight upper bound on the sample complexity, in terms of d^n .

In order to avoid a run-time dependence on $(\|p\|_{\mathcal{C}_n, \infty}, \sigma)$, e.g., in case σ is unknown or $\|p\|_{\mathcal{C}_n, \infty}$ is too big, we compute an ℓ_1 minimizer \hat{p}_{ℓ_1} first as in KKP's approach. However, for this analysis, we need a multivariate analog of Lemma 1.14 for the ℓ_1 norm. The main difficulty here is the fact that now we have many more “peripheral” cells, i.e., cells on the boundary of \mathcal{C}_n (these cells correspond to the peripheral intervals I_1 and I_m from the 1-dimensional Chebyshev partition, that needed Markov Brothers' Inequality). Since these peripheral cells are narrower, and much more in number, as n grows, this issue becomes more crucial. For example, for $n = 1$, the fraction of “peripheral” intervals is $2/m$; but for $n = 2$, it is $\frac{4(m-1)}{m^2} = \frac{2}{m}(2 - 2/m) \gg \frac{1}{m^2}$. We circumvent this difficulty with our *second new technical contribution* (Theorem 1.12), that relates the ℓ_∞ and ℓ_1 norms of any individual degree- d , n -variate polynomial.

Relating ℓ_∞ and ℓ_1 norms of p

We inductively show the existence of a subset of points in \mathcal{C}_n , with a large measure (at least $1/(2d^2)^n$), on which the valuations of p can be guaranteed to be large, i.e., at least $\max_{\mathbf{x} \in \mathcal{C}_n} |p(\mathbf{x})|/2^n$. Thus we lower bound the ℓ_1 norm of p by a $1/\text{poly}(d^n)$ factor of its ℓ_∞ norm, in the form of Theorem 1.12. We also note the tightness of this bound, by showing a family of polynomials for which their (resp.) ℓ_1 norms are upper bounded by a matching $1/\text{poly}(d^n)$ factor of the (resp.) ℓ_∞ norms, in the form of Proposition 1.13.

To begin, for any point in $\mathbf{x} \in \mathcal{C}_n$, using Markov Brothers' inequality, we show the existence of a long enough line segment, on an axis-parallel line passing through it, such that p on that line segment has all valuations at least $p(\mathbf{x})/2$. For constructing (higher) $(k + 1)$ -dimensional cubes from k -dimensional cubes, in the induction step, we prove that all new unique line segments (one each corresponding to every point in the k -dimensional cube) can be translated to form a $(k + 1)$ -dimensional cube with a large enough Lebesgue measure. Thus, a sizable subset of points in n dimensions is constructed. On each of these points, the valuations of p are at least half of the valuations of p on the corresponding points in the k -dimensional cube. Using the inductive hypothesis, we conclude the argument.

Using Theorem 1.12 we bound the ℓ_∞ error of the ℓ_1 minimizer \hat{p}_{ℓ_1} by $\text{poly}(d^n)\sigma$. We then feed \hat{p}_{ℓ_1} to the median based recovery procedure, which in $O(n \log d)$ iterations⁶, brings down the error to $O(\sigma)$, thus proving Theorem 1.4.

Organization

We begin by setting up some preliminaries in Section 2. Discussion of the upper bounds follows, with Theorem 1.1 in Section 3, Theorem 1.3 in Section 4, and Theorem 1.4 in Section 5.

⁶ The number of iterations in this case additionally depends on $O(\log(1 - 2\rho))$, which diverges as $\rho \rightarrow 0.5$, thus agreeing with $\rho < 0.5$ being information-theoretically necessary, as shown by [1].

Due to space constraints, the discussion of lower bounds, and all proofs are omitted here. They are included in the full version [2] of our paper.

2 Preliminaries

Notations

As mentioned earlier, \mathcal{P}_d denotes the class individual degree- d polynomials, where a polynomial $p: \mathbb{R}^n \rightarrow \mathbb{R}$ is said to be of individual degree- d if it can be written as

$$p(x_1, \dots, x_n) = \sum_{\alpha \in \{0,1,\dots,d\}^n} c_{\alpha} x_1^{\alpha_1} \cdots x_n^{\alpha_n},$$

for some set of coefficients $c_{\alpha} \in \mathbb{R}$. We use $[m] = \{1, \dots, m\}$, and bold font for multi-indices. For example $\mathbf{j} = (j_1, \dots, j_n) \in [m]^n$ where each entry $j_i \in [m]$. We use the math bold font (for e.g., \mathbf{x}, \mathbf{y}) for vectors. To denote random uniform sampling from a set \mathcal{D} , we use $\sim \mathcal{D}$. We denote by $\mathcal{C}_n = [-1, 1]^n$ the n dimensional solid cube and omit the subscript n when it is clear from the context. Our main problem of interest is the Robust Multivariate Polynomial Regression Problem, formally described in Section 1.

► **Definition 2.1.** [Norms] For any bounded subset $S \subseteq \mathbb{R}^n$, for any $1 \leq q \in \mathbb{R}$, the ℓ_q norm of a function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ on S , provided it exists, is defined as:

$$\|f\|_{S,q} \triangleq \left(\int_S |f(\mathbf{x})|^q d\mathbf{x} \right)^{\frac{1}{q}} < \infty.$$

The supremum norm of f on S is defined as $\|f\|_{S,\infty} \triangleq \lim_{q \rightarrow \infty} \|f\|_{S,q} = \sup_{\mathbf{x} \in S} \{|f(\mathbf{x})|\}$.

► **Lemma 2.2.** [Hölder's Inequality] Let $\alpha, \beta, \gamma \in \mathbb{R}_{\geq 1}$ such that $\frac{1}{\alpha} + \frac{1}{\beta} = \frac{1}{\gamma}$. For all functions f and g with finite $\|f\|_{S,\alpha}$, and $\|g\|_{S,\beta}$, we have: $\|fg\|_{S,\gamma} \leq \|f\|_{S,\alpha} \|g\|_{S,\beta}$.

► **Lemma 2.3** (Markov Brothers' Inequality [10]). Let $p: \mathbb{R} \rightarrow \mathbb{R}$ be a degree- d polynomial. Then, for all $a < b \in \mathbb{R}$,

$$\|p'\|_{[a,b],\infty} \leq \frac{2d^2}{b-a} \|p\|_{[a,b],\infty}.$$

► **Definition 2.4** (Chebyshev Polynomials). Chebyshev polynomials of the first kind are degree- d polynomials $T_d: \mathbb{R} \rightarrow \mathbb{R}$, that follow the recurrence relation:

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_{d+1}(x) = 2xT_d(x) - T_{d-1}(x).$$

Their explicit trigonometric formulation is:

$$T_d(x) \triangleq \begin{cases} \cos(d \arccos(x)), & \text{if } |x| \leq 1, \\ \cosh(d \operatorname{arcosh}(x)), & \text{if } x \geq 1, \\ (-1)^d \cosh(d \operatorname{arcosh}(-x)), & \text{if } x \leq -1, \end{cases}.$$

► **Definition 2.5** (Chebyshev Extremas). For any $d \in \mathbb{Z}_{>0}$, the d Chebyshev extremas $\in [-1, 1]$ given by

$$x_k \triangleq \cos\left(\frac{k}{d}\pi\right), k \in [d]$$

are the extremas of T_d , the degree- d Chebyshev polynomial of the first kind, i.e., $T_d(x_k) \in \{\pm 1\}, \forall k \in [d]$.

Chebyshev Partition

We partition the cube \mathcal{C}_n into m^n cells by tensorizing the partition used by KKP for the line segment $[-1, 1]$.

► **Definition 2.6** (Chebyshev partition). *The (m, n) -Chebyshev partition of the cube \mathcal{C} is a set of m^n cells indexed by $\mathbf{j} \in [m]^n$ and denoted $\mathcal{C}_{\mathbf{j}}$, such that*

$$\mathcal{C}_{\mathbf{j}} = \left[\cos \frac{\pi \mathbf{j}_1}{m}, \cos \frac{\pi (\mathbf{j}_1 - 1)}{m} \right] \times \cdots \times \left[\cos \frac{\pi \mathbf{j}_n}{m}, \cos \frac{\pi (\mathbf{j}_n - 1)}{m} \right].$$

The grid is induced by partitioning $[-1, 1]$ between the extrema points of the degree m Chebyshev polynomial of the first kind, T_m , simultaneously along each axis.

We generalize KKP's notion of *goodness* that restricts the number of outliers in each cell:

► **Definition 2.7** (α -good sample set). *We say that a set of samples $S = \{(\mathbf{x}_i, y_i)\}$ is α -good for the (m, n) Chebyshev partition, if for every $\mathbf{j} \in [m]^n$, the set of samples in $\mathcal{C}_{\mathbf{j}}$ has size at least $1/\alpha$, and the fraction of outliers in the cell $\mathcal{C}_{\mathbf{j}}$ is less than α .*

3 Main algorithmic result

In this section, we present the algorithm that solves the Robust Multivariate Polynomial Regression Problem, proving the following theorem, handling an approximation factor as close to 2 as we want, and any success probability $1 - \delta$.

► **Theorem 3.1.** *[Generalized version of Theorem 1.1] Let $\varepsilon \in (0, 1/2]$, $\delta \in (0, \varepsilon]$, $\sigma \geq 0$, $\eta > 0$, and $\rho < 1/2$. There is an algorithm (Algorithm 2) that almost solves the Robust Multivariate Polynomial Regression Problem up to an additive error of η . The output of the algorithm is a polynomial \hat{p} of degree at most d in each variable, such that with probability at least $1 - \delta$ (over the random input samples), \hat{p} satisfies*

$$|p(\mathbf{x}) - \hat{p}(\mathbf{x})| \leq (2 + \varepsilon)\sigma + \eta \quad \text{for all } \mathbf{x} \in \mathcal{C}.$$

It uses $M = O_{n,\rho}((d/\varepsilon)^n \log(d/\delta))$ samples drawn from the multidimensional Chebyshev distribution, or $M = O_{n,\rho}((d/\varepsilon)^{2n} \log(d/\delta))$ if the samples are drawn from the uniform measure. Its run-time is that of solving $O(\log_{1/\varepsilon}(\|p\|_{\mathcal{C}_{n,\infty}}/\eta))$ linear programs with $(d+1)^n < M$ variables, and M constraints.

We remark that the idea is to show that we may achieve a multiplicative approximation factor C , as close to 2 as we want (as long as $C > 2$), at the cost of more samples. We may allow larger values of ε , and then run our algorithm⁷ with $\varepsilon' = \min\{\varepsilon, 1/2\}$. For $\varepsilon \geq 1/2$, the dependence on ε in the sample complexity becomes constant for constant values of n .

► **Remark 3.2.** In case $\sigma > 0$ is known, one may choose $\eta = \varepsilon\sigma/2$, and set the ε parameter to be half of the desired bound to guarantee $\|\hat{p} - p\|_{\mathcal{C}_{n,\infty}} \leq (2 + \varepsilon)\sigma$.

Our algorithm, given in Algorithm 2 with its subroutine Algorithm 1, is essentially the same algorithm proposed by KKP, which now uses the (m, n) -Chebyshev partition of the cube \mathcal{C} instead of the $(m, 1)$ -Chebyshev partition of the interval $[-1, 1]$ used in KKP. Compared to their algorithm, we don't use the ℓ_1 regression as the first step, but instead start with the 0 polynomial as the first approximation. We first describe the idea of the algorithm and the median-based recovery.

⁷ Having $\varepsilon' \leq 1/2$ is a limitation of the current analysis. An open question remains to make it work efficiently for any $\varepsilon' > 0$.

Median-based Recovery

As in KKP (a similar approach was taken by [3]), for every $\mathbf{j} \in [m]^n$, we take the median $\tilde{y}_{\mathbf{j}}$ of all the y_i 's corresponding to locations \mathbf{x}_i 's that land in the cell $\mathcal{C}_{\mathbf{j}}$. We assume that the sample set S is α -good, so the fraction of outliers in each cell is strictly less than one-half (α and ρ are related: $\alpha = \frac{2\rho+1}{4}$. So, $\alpha < 1/2$, since $\rho < 1/2$.) so that the median lies in between the values of the *inlier* samples. However, we may not be able to determine which domain point is associated with $\tilde{y}_{\mathbf{j}}$. Even if there is a sample (\mathbf{x}_i, y_i) in the cell $\mathcal{C}_{\mathbf{j}}$ which collide with the median value (for which $y_i = \tilde{y}_{\mathbf{j}}$), it might be the case (\mathbf{x}_i, y_i) that is an *outlier*. We generalize KKP's techniques to show that picking an arbitrary $\tilde{\mathbf{x}}_{\mathbf{j}} \in \mathcal{C}_{\mathbf{j}}$, and assigning $\tilde{y}_{\mathbf{j}}$ to it is enough. We then compute the polynomial r , that minimizes $\max_{\mathbf{j} \in [m]^n} |r(\tilde{\mathbf{x}}_{\mathbf{j}}) - \tilde{y}_{\mathbf{j}}|$, and show that r is $O(\sigma)$ -close to p in ℓ_∞ up to an additive error of $\varepsilon \|p\|_{C_{n,\infty}}$. To deal with this error, we iteratively refine the estimate r . After $\log_{1/\varepsilon}(\|p\|_{C_{n,\infty}}/\eta)$ iterations, the additive error becomes as small as η .

Algorithm 1 Refinement.

```

1 Procedure Refine( $S, \hat{p}$ )
   Input : A set of samples  $S = \{\mathbf{x}_i, y_i\}_{i=1}^M$ , and an estimate  $\hat{p}$ .
2   for  $\mathbf{j} \in [m]^n$  do
3      $\tilde{y}_{\mathbf{j}} \leftarrow \text{med}_{\mathbf{x}_i \in \mathcal{C}_{\mathbf{j}}}(y_i - \hat{p}(\mathbf{x}_i))$ ;
4     Choose an arbitrary  $\tilde{\mathbf{x}}_{\mathbf{j}} \in \mathcal{C}_{\mathbf{j}}$ ;
5     Fit a degree  $d$  polynomial  $r$  minimizing  $\|r(\tilde{\mathbf{x}}_{\mathbf{j}}) - \tilde{y}_{\mathbf{j}}\|_\infty$ ;
6      $\hat{p}' \leftarrow \hat{p} + r$ ;
7   Return  $\hat{p}'$ .
```

Algorithm 2 Median Based Recovery.

```

Input : A set of samples  $S = \{\mathbf{x}_i, y_i\}_{i=1}^M$ , approximation factor  $\varepsilon \leq 1/2$ , accuracy
parameter  $\eta > 0$ .
1  $\hat{p}^{(1)} \leftarrow \text{Refine}(S, 0)$ ; // Let  $\hat{p}^{(1)}(\mathbf{x}) = \sum_{\alpha \in \{0,1,\dots,d\}^n} c_\alpha \mathbf{x}^\alpha$ 
2 Let  $v_{\max} : |\hat{p}^{(1)}(\mathbf{x})| \leq v_{\max}$  for all  $\mathbf{x} \in \mathcal{C}$ ; // Set  $v_{\max} \triangleq \sum_{\alpha \in \{0,1,\dots,d\}^n} |c_\alpha|$ 
3  $N_2 \leftarrow O(\log_{1/\varepsilon}(v_{\max}/\eta))$ ;
4 for  $i \in \{1, \dots, N_2 - 1\}$  do
5    $\hat{p}^{(i+1)} \leftarrow \text{Refine}(S, \hat{p}^{(i)})$ ;
6 Return  $\hat{p}^{(N_2)}$ .
```

To prove Theorem 3.1, we show that for M as in the theorem, the set of samples is α -good with high probability, then we apply the following result.

► **Theorem 3.3** (Absolute ℓ_∞ error bound). *Let c be some absolute constant, and let $\varepsilon, \alpha < 1/2$, $0 < \eta \leq 1$, be parameters. For any $m \geq c d n / \varepsilon$, if the set $S = \{(\mathbf{x}_i, y_i)\}$ of M samples is α -good for the (m, n) -Chebyshev partition, then the median-based recovery Algorithm 2 returns an individual degree- d polynomial $\hat{p} = \hat{p}^{(N_2)}$, such that*

$$\|p - \hat{p}\|_{C_{n,\infty}} \leq (2 + \varepsilon)\sigma + \eta.$$

The first part of the proof of Theorem 3.3 follows the skeleton of the proof of [8, Theorem 1.4], whilst skipping the preliminary ℓ_1 regression.

12:14 Outlier Robust Multivariate Polynomial Regression

The main ingredient for proving Theorem 3.3 is the following technical result, bounding the ℓ_∞ error of the non-robust ℓ_∞ minimizer, i.e., a single run of Algorithm 1. This is later used to bound the error of the robust minimizer Algorithm 2.

► **Lemma 3.4** (Relative ℓ_∞ error bound, generalization of [8, Lemma 1.3]). *Let $c > 0$ be an absolute constant. Let $\varepsilon, \alpha < 1/2$, and $m \geq cdn/\varepsilon$. Let the set $S = \{(\mathbf{x}_i, y_i)\}$ of M samples is α -good for the (m, n) -Chebyshev partition. And for every $\mathbf{j} \in [m]^n$, let $\tilde{\mathbf{x}}_{\mathbf{j}}$ be an arbitrary point from the cell $\mathcal{C}_{\mathbf{j}}$, and $\tilde{y}_{\mathbf{j}} \triangleq \text{med}_S\{y_i : \mathbf{x}_i \in \mathcal{C}_{\mathbf{j}}\}$, i.e. the median of all those y_i 's in S , whose corresponding \mathbf{x}_i is in the cell $\mathcal{C}_{\mathbf{j}}$. Then, with*

$$\hat{p} \triangleq \arg \min_{q \in \mathcal{P}_d} \max_{\mathbf{j} \in [m]^n} |q(\tilde{\mathbf{x}}_{\mathbf{j}}) - \tilde{y}_{\mathbf{j}}|,$$

where the minimization is over the class \mathcal{P}_d of all individual degree- d polynomials over \mathbb{R}^n , we have

$$\|p - \hat{p}\|_{C_{n,\infty}} \leq (2 + \varepsilon)\sigma + \varepsilon\|p\|_{C_{n,\infty}}.$$

The proof of this statement mirrors the proof of its univariate counterpart [8, Lemma 1.3], with Theorem 1.11 replacing Lemma 1.14.

4 Dealing with finite precision representations

We prove a more precise statement of Theorem 1.3, giving an algorithm for handling an approximation factor close enough to 2, and for any success probability $1 - \delta$.

► **Theorem 4.1.** *[Generalized version of Theorem 1.3] Let N be the number of bits of precision, $\sigma \geq 2^{-N}$ and, constant $\rho < 1/2$. For any $\varepsilon \leq 1/2$ such that $\varepsilon = \Omega_n(d2^{-N/2})$, and $\delta \in (0, \varepsilon]$, there exists an algorithm (Algorithm 3) for the Robust Multivariate Polynomial Regression Problem. The output of the algorithm is $\hat{p}: \mathbb{R}^n \rightarrow \mathbb{R}$, a polynomial of degree at most d in each variable, that satisfies*

$$|p(\mathbf{x}) - \hat{p}(\mathbf{x})| \leq (2 + \varepsilon)\sigma \quad \text{for all } \mathbf{x} \in \mathcal{C}_n,$$

with probability at least $1 - \delta$. It uses $M = O_{n,\rho}((d/\varepsilon)^n \log(d/\delta))$ samples drawn from the multidimensional Chebyshev distribution, or $M = O_{n,\rho}((d/\varepsilon)^{2n} \log(d/\delta))$ if the samples are drawn from the uniform measure. Its run-time is that of solving $O(\log_{1/\varepsilon}\|p\|_{C_{n,\infty}} + N)$ linear programs with $(d + 1)^n < M$ variables, and M constraints.

■ **Algorithm 3** Median Based Recovery with Finite Precision.

Input : A set of samples $S = \{\mathbf{x}_i, y_i\}_{i=1}^M$ specified upto N bits of precision, approximation factor $\varepsilon \leq 1/2$, accuracy parameter $\eta = \varepsilon 2^{-N}$.

- 1 Sift S into $S' \triangleq \{(\mathbf{x} = (x_1, \dots, x_n), y) \in S : \forall i \in [n]. |k_{x_i} - x_i| > 2^{-N}\}$, where for all $i \in [n]$, k_{x_i} = the Chebyshev extrema (Definition 2.5) closest to x_i .
 - 2 Run Algorithm 2 on (S', ε, η) .
-

► **Remark 4.2.** We note that the condition on ε implies that in order to learn degree d polynomials using Algorithm 2, one needs at least $N = \Omega(\log d)$ bits of precision. The reason is that to get a good approximation we need to take a fine enough grid, but the grid's "fineness parameter" m is limited as well by the precision restriction, as we need the width of any cell to be at least 2^{-N} . Note that if $N = o(\log d)$, i.e. $d = 2^{\Omega(N)}$, then $\varepsilon = \Omega(1)$,

i.e. the approximation factor achieved in this setting is too large, compared to the factor of at most 3, achievable when $N = \Omega(\log d)$. If we would take another approach, and just consider the difference between $p(\mathbf{x})$ and $p(\mathbf{z})$, for some arbitrary $\mathbf{x}, \mathbf{z} \in \mathcal{C}_n$; the bound, using for example, Markov Brothers' inequality, would involve $\|p\|_{\mathcal{C}_n, \infty}$ and impose a restriction of $\|p\|_{\mathcal{C}_n, \infty} \leq 2^N$. This might make sense in some settings, if we consider N , not only as a restriction on the precision of the given sample set, but also as a restriction on the space complexity of the algorithm.

5 Robust multivariate regression algorithm

In this section, we show how to modify our algorithm to avoid having a run-time dependence on $(\|p\|_{\mathcal{C}_n, \infty}, \sigma, N)$ and thus prove the following theorem. In Algorithm 4, we use the same idea as in KKP: starting with \hat{p}_{ℓ_1} , the result of an ℓ_1 regression, we then iteratively refine the estimate, improving the ℓ_∞ error in each step.

► **Theorem 5.1.** *[Generalized form of Theorem 1.4] Let $\varepsilon \in (0, 1/2], \delta \in (0, \varepsilon], \sigma \geq 0$, and $\rho < 1/2$. There is an algorithm (Algorithm 4) that solves the Robust Multivariate Polynomial Regression Problem with approximation factor $C = 2 + \varepsilon$. The output of the algorithm is a polynomial $\hat{p}: \mathbb{R}^n \rightarrow \mathbb{R}$ of degree at most d in each variable, such that with probability (over the random input samples) at least $1 - \delta$, \hat{p} satisfies*

$$|p(\mathbf{x}) - \hat{p}(\mathbf{x})| \leq (2 + \varepsilon)\sigma, \quad \text{for all } \mathbf{x} \in \mathcal{C}.$$

It uses $M = O_{n, \rho} \left((d^{2n+1}/\varepsilon)^n \log(d/\delta) \right)$ samples drawn from the multidimensional Chebyshev distribution, or $M = O_{n, \rho} \left((d^{2n+1}/\varepsilon)^{2n} \log(d/\delta) \right)$ if the sampled are drawn from the uniform measure. Its run-time is $\text{poly}(M, \log_\varepsilon(1 - 2\rho))$.

■ **Algorithm 4** Median Based Recovery with ℓ_1 regression.

Input: A set of samples $S = \{\mathbf{x}_i, y_i\}_{i=1}^M$, of which a ρ fraction may be outliers.

- 1 $\hat{p}^{(0)} \leftarrow$ result of ℓ_1 regression: \hat{p}_{ℓ_1} ;
- 2 $N_4 \leftarrow O \left(n \log_{1/\varepsilon} d + \log_\varepsilon(1 - 2\rho) \right)$;
- 3 **for** $i \in \{0, \dots, N_4 - 1\}$ **do**
- 4 $\hat{p}^{(i+1)} \leftarrow \mathbf{Refine}(S, \hat{p}^{(i)})$;
- 5 **Return** $\hat{p}^{(N_4)}$.

As a result, the number of iterations here depends only on (d, n) , and improves as ε gets smaller. (In fact, it is linear in n , and logarithmic in d , and $1/\varepsilon$).

Here, as in Section 3, we prove that with enough samples, the set of samples is α -good (again, for $\alpha = \frac{2\rho+1}{4}$) with high probability, and separately we show that for any α -good (where $\alpha < 1/2$) set of samples, Algorithm 4 recovers p as required.

► **Theorem 5.2** (Absolute ℓ_∞ error bound). *Let $\varepsilon, \alpha < 1/2$. Let the set $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^M$ of samples be α -good for the (m, n) -Chebyshev partition where $m \geq (cd)^{2n+1}/\varepsilon$, for some large enough constant $c > 1$. Then the median recovery Algorithm 4, in $N_4 = O(n \log_{1/\varepsilon} d + \log_\varepsilon(1 - 2\alpha))$ iterations, returns an individual degree- d polynomial \hat{p} , such that $\|p - \hat{p}\|_{\mathcal{C}_n, \infty} \leq (2 + \varepsilon)\sigma$.*

For proving the above theorem, we bound the initial error $\|p - \hat{p}_{\ell_1}\|_{\mathcal{C}_n, \infty}$ by $\text{poly}(d^n)\sigma$. We briefly discuss this in Subsection 5.1, presented as Theorem 5.4. However, to use the ℓ_1 regression result, the underlying Chebyshev grid needs to be much finer, with $m = (cd)^{2n+1}/\varepsilon$,

for some absolute constant $c > 0$ (as compared to $c_0 dn/\varepsilon$ needed for just the ℓ_∞ regression). Hence the Chebyshev and Uniform sample complexities for Algorithm 4 are worse compared to Algorithm 2.

5.1 Bounding the ℓ_1 regression error

We next generalize KKP’s ℓ_1 regression. Similar to their regression on averages over Chebyshev intervals, we do regression on averages over Chebyshev cells.

► **Definition 5.3** (ℓ_1 Minimizer). *Let m be large enough integer. Given a set of M samples $S = \{(\mathbf{x}_i, y_i)\}$, for every $\mathbf{j} \in [m]^n$, let $S_{\mathbf{j}} \triangleq \{\beta \in [M] : \mathbf{x}_\beta \in \mathcal{C}_{\mathbf{j}}\}$. The ℓ_1 minimizer of S with respect to the (m, n) -Chebyshev partition, is the individual degree- d polynomial:*

$$\hat{p}_{\ell_1} \triangleq \arg \min_{f \in \mathcal{P}_d} \sum_{\mathbf{j} \in [m]^n} \frac{V_n(\mathcal{C}_{\mathbf{j}})}{|S_{\mathbf{j}}|} \sum_{\beta \in S_{\mathbf{j}}} |f(\mathbf{x}_\beta) - y_\beta|. \quad (1)$$

Using Theorem 1.12, we show that on a set of α -good samples, the ℓ_1 regression outputs an individual degree- d polynomial with $\text{poly}(d^n)$ error in ℓ_∞ .

► **Theorem 5.4** (ℓ_∞ error bound for the ℓ_1 minimizer). *Let $\alpha < 1/2$ be constant, $\varepsilon \leq (1-2\alpha)/2$, and for some constant $c > 1$, $m \geq (cd)^{2n+1}/\varepsilon$. Given a set S of samples that is α -good with respect to the (m, n) -Chebyshev partition, the ℓ_1 minimizer \hat{p}_{ℓ_1} from (1) satisfies*

$$\|p - \hat{p}_{\ell_1}\|_{c_{n,\infty}} = O_\alpha((8d^2)^n \sigma).$$

References

- 1 Sanjeev Arora and Subhash Khot. Fitting algebraic curves to noisy data. *Journal of Computer and System Sciences*, 67(2):325–340, 2003. Special Issue on STOC 2002. doi:10.1016/S0022-0000(03)00012-6.
- 2 Vipul Arora, Arnab Bhattacharyya, Mathews Boban, Venkatesan Guruswami, and Esty Kelman. Outlier Robust Multivariate Polynomial Regression, 2024. arXiv:2403.09465.
- 3 Hadassa Daltrophe, Shlomi Dolev, and Zvi Lotker. Big data interpolation using functional representation. *Acta Informatica*, 55:213–225, 2018. doi:10.1007/s00236-016-0288-8.
- 4 Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. Sever: A robust meta-algorithm for stochastic optimization. In *International Conference on Machine Learning*, pages 1596–1606. PMLR, 2019. URL: <https://arxiv.org/abs/1803.02815>.
- 5 Ilias Diakonikolas, Weihao Kong, and Alistair Stewart. Efficient algorithms and lower bounds for robust linear regression. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2745–2754. SIAM, 2019. URL: <https://arxiv.org/abs/1806.00040>.
- 6 V. Guruswami and D. Zuckerman. Robust Fourier and Polynomial Curve Fitting. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 751–759, Los Alamitos, CA, USA, October 2016. IEEE Computer Society. doi:10.1109/FOCS.2016.75.
- 7 Helmut. Norms on \mathcal{P}_N Vector Space of Polynomials up to Order N . Mathematics Stack Exchange. URL: <https://math.stackexchange.com/q/2693954>.
- 8 Daniel Kane, Sushrut Karmalkar, and Eric Price. Robust Polynomial Regression up to the Information Theoretic Limit. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 391–402, 2017. doi:10.1109/FOCS.2017.43.
- 9 Adam R. Klivans, Pravesh K. Kothari, and Raghu Meka. Efficient algorithms for outlier-robust regression. In Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet, editors, *Conference On Learning Theory, COLT 2018, Stockholm, Sweden, 6-9 July 2018*, volume 75 of *Proceedings of Machine Learning Research*, pages 1420–1430. PMLR, 2018. URL: <http://proceedings.mlr.press/v75/klivans18a.html>.

- 10 Andrey Andreyevich Markov. On a question by D. I. Mendeleev. *Zap. Imp. Akad. Nauk. St. Petersburg*, 62:1–24, 1890. URL: <https://history-of-approximation-theory.com/fpapers/markov4.pdf>.
- 11 Paul G Nevai. Bernstein's inequality in l_p for $0 < p < 1$. *Journal of Approximation Theory*, 27(3):239–243, 1979. doi:10.1016/0021-9045(79)90105-9.
- 12 Adarsh Prasad, Arun Sai Suggala, Sivaraman Balakrishnan, and Pradeep Ravikumar. Robust Estimation via Robust Gradient Estimation. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 82(3):601–627, 2020. URL: <https://arxiv.org/abs/1802.06485>.
- 13 John Wolberg. *Data analysis using the method of least squares: extracting the most information from experiments*. Springer Science & Business Media, 2006. doi:10.1007/3-540-31720-1.
- 14 Achim Zielesny. *From curve fitting to machine learning*, volume 18. Springer, 2011. doi:10.1007/978-3-319-32545-3.