Super Non-singular Decompositions of Polynomials and Their Application to Robustly Learning Low-Degree PTFs

Ilias Diakonikolas[*]
University of Wisconsin-Madison
Madison, USA
ilias.diakonikolas@gmail.com

Daniel M. Kane[†]
University of California
San Diego, USA
dakane@ucsd.edu

Vasilis Kontonis[‡]
University of Texas Austin
Austin, USA
kontonis@wisc.edu

Sihan Liu
University of California San Diego
San Diego, USA
sil046@ucsd.edu

Nikos Zarifis[§]
University of Wisconsin-Madison
Madison, USA
zarifis@wisc.edu

ABSTRACT

We study the efficient learnability of low-degree polynomial threshold functions (PTFs) in the presence of a constant fraction of adversarial corruptions. Our main algorithmic result is a polynomial-time PAC learning algorithm for this concept class in the strong contamination model under the Gaussian distribution with error guarantee $O_{d,c}(\mathrm{opt}^{1-c})$, for any desired constant $c > 0$, where opt is the fraction of corruptions. In the strong contamination model, an omniscient adversary can arbitrarily corrupt an opt-fraction of the data points and their labels. This model generalizes the malicious noise model and the adversarial label noise model. Prior to our work, known polynomial-time algorithms in this corruption model (or even in the weaker adversarial label noise model) achieved error $\tilde{O}_d(\mathrm{opt}^{1/(d+1)})$, which deteriorates significantly as a function of the degree d.

Our algorithm employs an iterative approach inspired by localization techniques previously used in the context of learning linear threshold functions. Specifically, we use a robust perceptron algorithm to compute a good partial classifier and then iterate on the unclassified points. In order to achieve this, we need to take a set defined by a number of polynomial inequalities and partition it into several well-behaved subsets. To this end, we develop new polynomial decomposition techniques that may be of independent interest.

CCS CONCEPTS

• **Theory of computation → Machine learning theory**.

[*] Supported by NSF Medium Award CCF-2107079, NSF Award CCF-1652862 (CAREER), a Sloan Research Fellowship, and a DARPA Learning with Less Labels (LwLL) grant.
[†] Supported by NSF Award CCF-1553288 (CAREER) and a Sloan Research Fellowship.
[‡] Supported in part by NSF Award CCF-2144298 (CAREER).
[§] Supported in part by NSF Award CCF-1652862 (CAREER) and a DARPA Learning with Less Labels (LwLL) grant.

STOC '24, June 24–28, 2024, Vancouver, BC, Canada
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0383-6/24/06
https://doi.org/10.1145/3618260.3649776

KEYWORDS

Machine Learning Theory, Polynomial Threshold Function, Agnostic Learning

ACM Reference Format:
Ilias Diakonikolas, Daniel M. Kane, Vasilis Kontonis, Sihan Liu, and Nikos Zarifis. 2024. Super Non-singular Decompositions of Polynomials and Their Application to Robustly Learning Low-Degree PTFs. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC '24), June 24–28, 2024, Vancouver, BC, Canada.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3618260.3649776

1 INTRODUCTION

A degree-d polynomial threshold function (PTF) is any Boolean function $f : \mathbb{R}^n \to \{\pm 1\}$ of the form $f(x) = \mathrm{sign}(p(\mathbf{x}))$[1], where $p : \mathbb{R}^n \to \mathbb{R}$ is a degree-d polynomial with real coefficients. For $d = 1$, we obtain Linear Threshold Functionss (LTFs) or halfspaces. PTFs are a fundamental class of Boolean functions that have been extensively studied in many contexts for at least the past five decades [11, 41, 42]. Over the past two decades, low-degree PTFs have been the focus of renewed research interest in various fields of theoretical computer science, including complexity theory [10, 13, 19, 31, 33, 40, 43–45] and learning theory [9, 14, 16, 22, 23, 29].

In this paper we study the problem of PAC learning degree-d PTFs in the presence of a constant fraction of adversarially corrupted data. More concretely, we define the following data contamination model considered in the current work.

Definition 1.1 (Strong Contamination Model). Let C be a class of Boolean functions on $\mathbb{R}^n$, $D_{\mathbf{x}}$ a distribution over $\mathbb{R}^n$, and f an unknown target function $f \in C$. For $0 < \mathrm{opt} < 1/2$, we say that a set T of m labeled examples is an opt-*corrupted* set of examples from C if it is obtained using the following procedure: First, we draw a set $S = \{(\mathbf{x}^{(i)}, y_i)\}$ of m labeled examples, $1 \le i \le m$, where for each i we have that $\mathbf{x}^{(i)} \sim D_{\mathbf{x}}$, $y_i = f(\mathbf{x}^{(i)})$, and the $\mathbf{x}^{(i)}$'s are independent. Then an omniscient adversary, upon inspecting the set S, is allowed to remove an opt-fraction of the examples and replace these examples by the same number of arbitrary examples of its choice. The modified set of labeled examples is the opt-corrupted set T.

[1] The function sign : $\mathbb{R} \to \{\pm 1\}$ is defined as $\mathrm{sign}(t) = 1$ if $t \ge 0$ and $\mathrm{sign}(t) = -1$ otherwise.

A learning algorithm in the strong contamination/nasty noise model is given as input an opt-corrupted set of examples from C and its goal is to output a hypothesis h such that with high probability the error $\mathbf{Pr}_{\mathbf{x}\sim D}[h(\mathbf{x}) \neq f(\mathbf{x})]$ is small, as compared to the information-theoretically optimal error of opt.

That is, in the nasty noise model [4], an omniscient adversary can arbitrarily corrupt a small constant fraction of both the data points and their labels. The nasty noise model is equivalent to the strong contamination model studied in the field of robust statistics [15, 17] and generalizes well-studied corruption models, including the agnostic (adversarial label noise) model [28, 35] and the malicious noise model [36, 48]. In the adversarial label noise (agnostic) model, the adversary can corrupt an opt-fraction of the labels, but cannot change the distribution of the unlabeled points. In the malicious model, the adversary can *add* an opt-fraction of corrupted labeled examples, but is not allowed to adversarially remove clean labeled examples.

The goal of this work is to understand the efficient learnability of degree-d PTFs under the Gaussian distribution in the presence of nasty noise. Our main algorithmic result is the following:

Theorem 1.2 (Main Learning Result). *There exists an algorithm that, given any $c, \epsilon \in (0,1)$, has sample and computational complexity $n^{O(d)}\mathrm{poly}_{d,c}(1/\epsilon)$, and learns the class of degree-d PTFs on $\mathbb{R}^n$ in the nasty noise model under the Gaussian distribution within 0-1 error $O_{c,d}(1)\,\mathrm{opt}^{1-c} + \epsilon$.*

Discussion. Some comments are in order. We start by noting that our learning algorithm is not proper. Specifically, the output hypothesis is a decision list whose leaves are degree-d PTFs.

It is instructive to quantitatively compare the complexity and error guarantee of Theorem 1.2 with prior work. The L_1-polynomial regression algorithm of [29] achieves the optimal error of opt + ϵ in the (weaker) adversarial label noise model with sample and computational complexity $n^{\mathrm{poly}(d/\epsilon)}$. Moreover, the exponential complexity dependence in $1/\epsilon$ is inherent [21, 46]. The latter computational lower bounds motivate the design of faster (ideally, fully-polynomial time) algorithms with relaxed error guarantees. When restricting to fully-polynomial time algorithms (i.e., with runtime $\mathrm{poly}_d(n/\epsilon)$), [22] gave a robust learner with error guarantee $\tilde{O}_d(\mathrm{opt}^{1/(d+1)}) + \epsilon$. For $d > 1$, this was the best previously known error guarantee (for $\mathrm{poly}_d(n/\epsilon)$ time algorithms) even in the weaker adversarial label noise model. (See the following subsection for a detailed summary of prior work.)

The latter error guarantee deteriorates dramatically as a function of the degree d. A natural question that motivated this work is whether it is possible to qualitatively nearly-match the $d = 1$ case — where polynomial-time algorithms with error $O(\mathrm{opt}) + \epsilon$ are known [1, 22] — for any constant degree d (or even for $d = 2$!). More concretely:

> *Is there a $\mathrm{poly}_d(n/\epsilon)$ time algorithm that, for any constant d, robustly learns*
> *degree-d PTFs with error $O_{c,d}(1)\,\mathrm{opt}^c$, where $c > 0$ is* independent *of d?*

Our main result answers this question in the affirmative. Moreover, we can take the parameter c above to be *any constant less than* 1. Achieving error $\tilde{O}_d(\mathrm{opt})$ or $O_d(\mathrm{opt})$ is left as an open question.

Finally, we reiterate that our algorithm is the first algorithm with this error guarantee *even in the weaker model of adversarial label noise.*

Interestingly, to obtain our algorithmic result, we generalize the *localization* technique [1, 3], developed in the context of learning *linear* threshold functions, for the problem of learning degree-d PTFs. To achieve this goal, we develop the algorithmic theory of *super non-singular polynomial decompositions*, which we believe is of broader interest beyond learning theory.

1.1 Prior Work

In the realizable PAC learning model (i.e., with clean/consistent labels), low-degree PTFs are known to be efficiently learnable in the distribution-free setting via a reduction to linear programming [39]. Specifically, the class of degree-d PTFs on $\mathbb{R}^n$ can be learned to 0-1 error ϵ with sample size $m = \tilde{O}(n^d/\epsilon)$ in $\mathrm{poly}(m)$ time. By standard VC-dimension arguments, this sample size is information-theoretically necessary for any learning algorithm.

In the presence of adversarial noise in the data (the focus of the current work), the learning problem becomes significantly more challenging computationally. Specifically, in the distribution-free setting, the agnostic learning problem (i.e., in the presence of adversarial label noise) is known to be computationally intractable, even for the special case of $d = 1$ and constant accuracy [7, 18, 46]. As a result, research in this area has focused on the *distribution-specific* setting, i.e., with respect to specific natural distributions on the domain, such as the Gaussian distribution.

In the distribution-specific agnostic model, the L_1-polynomial regression algorithm [29] learns degree-d PTFs within error opt + ϵ with sample and computational complexity $n^{O(d^2/\epsilon^4)}$ under the Gaussian distribution (and the uniform distribution on the hypercube) [14, 24, 25, 27, 30, 38]. Importantly, the exponential dependence in $1/\epsilon$ is inherent in the complexity of the problem, both in the Statistical Query model [20] and under standard cryptographic assumptions [21, 46] (even under the Gaussian distribution).

The aforementioned hardness results motivated the design of faster algorithms with relaxed error guarantees. Over the past fifteen years, substantial progress has been made in this direction, in particular for the special case of Linear Threshold Functions (corresponding to $d = 1$). Specifically, a sequence of works [1, 6, 22, 37] developed $\mathrm{poly}(n/\epsilon)$ time robust learners for LTFs in the malicious/nasty model (thus, also in the adversarial label noise model) under the Gaussian distribution and, in some cases, for isotropic (i.e., zero-mean, identity covariance) log-concave distributions. In more detail, [1] gave a malicious learning algorithm for *homogeneous* LTFs (i.e., halfspaces whose separating hyperplane goes through the origin) with near-optimal error guarantee of $O(\mathrm{opt}) + \epsilon$ under all isotropic log-concave distributions[2]. Subsequently, [22] gave an efficient algorithm that achieves error of $O(\mathrm{opt}) + \epsilon$ for arbitrary LTFs and succeeds under the Gaussian distribution. At the technical level, [1] developed a *localization method* (see also [2] for a precursor) which is crucial to obtain the near-optimal error guarantee of $O(\mathrm{opt})$. In fact, the algorithm of [22] for general halfspaces proceeds by a refinement of this idea.

[2] It turns out that the homogeneity assumption is important here. Specifically, the underlying algorithm does not extend to arbitrary LTFs with the same error guarantees.

For the case of degree-d PTFs, progress in this direction has been slow. The only prior algorithmic work on the topic is due to [22]. That work gave a $\mathrm{poly}(n^d/\epsilon)$ time algorithm that succeeds in the presence of nasty noise under the Gaussian distribution[3] and attains an error guarantee of $O_d(\mathrm{opt}^{1/(d+1)}) + \epsilon$.

2 TECHNICAL OVERVIEW

Prior Work: Learning via Degree-d Chow-Parameters. A polynomial threshold function (PTF) can be thought of as a linear threshold function (LTF) composed with the Veronese map $\mathbf{x} \mapsto \mathbf{x}^{\otimes d}$. Thus, we can think of the question of (robustly) learning a PTF of Gaussian inputs as the problem of learning an LTF with input $\widetilde{\mathbf{x}} := \mathbf{x}^{\otimes d}$.

A common approach to learning PTFs (and other related geometric concept classes) in the literature is via (low-degree) Chow-parameter fitting [4]; see, e.g. [8, 9, 12, 22, 47]. More precisely, in [22], the approach is to find an LTF (as a function of the tensor feature $\widetilde{\mathbf{x}}$) such that $\mathbf{E}_{\widetilde{\mathbf{x}}}[\mathrm{sign}(\mathbf{w} \cdot \widetilde{\mathbf{x}})\widetilde{\mathbf{x}}] = \mathbf{E}_{(\widetilde{\mathbf{x}},y)}[y\widetilde{\mathbf{x}}]$.

The Chow-parameter fitting approach requires two crucial assumptions about the distribution of $\widetilde{\mathbf{x}} = \mathbf{x}^{\otimes d}$. First, it requires concentration bounds in order to show that the adversarial noise cannot affect significantly the relevant Chow parameters $\mathbf{E}_{(\widetilde{\mathbf{x}},y)}[y\widetilde{\mathbf{x}}]$. Gaussian hypercontractivity indeed implies that polynomials of Gaussian random variables enjoy strong concentration. In addition to this, showing that a small error in the Chow parameters translates to a small error in the total variation distance of the corresponding threshold function, requires some anti-concentration properties of the underlying distribution. More precisely, it requires showing that any linear function is not-too-small with high probability. We can obtain this using results of [5]; but, unfortunately, the anti-concentration provided is weak, showing that $\mathbf{Pr}(|p(\mathbf{x})| < \epsilon \|p\|_{L_2}) < O_d(\epsilon^{1/d})$ for any degree-d polynomial $p(\cdot)$. This translates quantitatively to an algorithm that robustly learns degree-d PTFs to error $O(\mathrm{opt}^{1/O(d)})$ — a far cry from our goal of error $O(\mathrm{opt}^{1-c})$, especially when d is large.

Our Approach: Learning PTFs Using Perceptron and Localization. Our high-level plan to improve upon the error guarantee of $O(\mathrm{opt}^{1/O(d)})$ is via the method of localization, a powerful approach for learning with corrupted labels; see [1–3]. For technical reasons, our starting point is an early instantiation of this technique [3] developed in the context of learning LTFs with random label noise. At a high-level, localization consists of first learning some LTF that achieves good error for all large-margin points, and then conditioning on low-margin examples to learn a new (or improve the current) hypothesis. Importantly, all localization-based algorithms require that, *after conditioning on the low-margin region* $|\mathbf{w} \cdot \widetilde{\mathbf{x}}| < \epsilon$, the resulting distribution satisfies strong anti-concentration properties. While this property is true for learning LTFs under the Gaussian distribution, it completely fails to hold under the conditional distribution of low-margin points with respect to a PTF, i.e., $|p(\mathbf{x})| \leq \epsilon$. Our approach consists of two new ingredients: (i) a robust version of the localized margin-perceptron algorithm for learning PTFs under weaker (anti-)concentration assumptions, and (ii) a localization process for PTFs so that the corresponding conditional distributions satisfy (anti-)concentration. In the following presentation, we first focus on the localization process for PTFs, and then present our robust margin-perceptron learning algorithm.

2.1 PTF Localization via Partitioning

Naive localization fails. We first investigate why naively conditioning in the low-margin region $|p(\mathbf{x})| < \epsilon$ fails to satisfy the required anti-concentration property, when $p(\mathbf{x})$ has degree larger than 1. In particular, consider the polynomial $p(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^2\mathbf{x}_2^2$. To simplify the calculations, we first observe that the set $|\mathbf{x}_1^2\mathbf{x}_2^2| < \epsilon$ is similar to the union of two intervals $R = \{|\mathbf{x}_1^2| \leq \epsilon\} \cup \{|\mathbf{x}_2^2| \leq \epsilon\}$; see Figure 1. The probability of R under the standard Gaussian distribution is roughly $\sqrt{\epsilon}$, so we still need to learn a classifier inside it (note that we could simply ignore a region of mass $O(\epsilon)$). We examine the anti-concentration property of a different polynomial $q(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^2$ under the Gaussian distribution conditioned on the union of two rectangles R. It is not hard to see that the L_2-norm of q is $\Theta(1)$ under the conditional distribution (the points within the green rectangle in Figure 1 have large $\mathbf{x}_1$ coordinate with constant probability). To give some intuition of what would be "good" anti-concentration, we remark that for a polynomial q whose L_2-norm is constant, we would like the probability of $|q(\mathbf{x}_1, \mathbf{x}_2)| < \epsilon$ to be roughly $\mathrm{poly}(\epsilon)$ (as is indeed the case for the Gaussian, by [5]). However, q turns out to have much worse anti-concentration conditional on R, as we have $\mathbf{Pr}[\mathbf{x}_1^2 \leq \epsilon \mid \mathbf{x}_1^2\mathbf{x}_2^2 \leq \epsilon] \geq \mathbf{Pr}[\mathbf{x}_2^2 \leq 1]\,\mathbf{Pr}[\mathbf{x}_1^2 \leq \epsilon]/\mathbf{Pr}[R] = \Omega(1)$. Thus, a naive localization procedure — which tries to reapply a learner on the low-margin conditional distribution directly — is unlikely to work as long as the learner requires *any non-trivial* anti-concentration property.

Localization via Partitioning. A way to preserve the (anti)-concentration properties in the previous example is to (approximately) partition the region where $|\mathbf{x}_1^2\mathbf{x}_2^2| < \epsilon$ into small (axis-aligned) rectangles (see the right figure in Figure 1). The Gaussian distribution conditioned on each rectangle is a log-concave distribution, and thus has good concentration and anti-concentration. Hence, we could attempt to use the margin-perceptron learner on each of the conditional distributions. As one of our main contributions, we give an efficient algorithm that finds such a partition for *any* low-degree polynomial. In particular, for a degree-d polynomial p, we show that the low-margin area $|p(\mathbf{x})| \leq \epsilon$ can be partitioned into $O_d(1)$ many subsets such that the Gaussian distribution conditioned on each of them satisfies strong (anti-)concentration properties.

Theorem 2.1 (Informal – Partitioning the Low-Margin Region of Polynomials). *Fix $\epsilon \in (0, 1)$ and let $p : \mathbb{R}^n \mapsto \mathbb{R}$ be a polynomial of degree at most d. There exists an efficient algorithm that (approximately) decomposes the set $\{\mathbf{x} \in \mathbb{R}^n : |p(\mathbf{x})| < \epsilon\}$ into $m = \mathrm{poly}_d(1/\epsilon)$ sets $S^{(1)}, \cdots, S^{(m)} \subset \mathbb{R}^n$ such that $\mathcal{N}(\mathbf{0}, \mathbf{I})$ conditioned on $S^{(i)}$ satisfies good anti-concentration.*

Super Non-Singular Decomposition. To get some intuition of how the partition routine operates, we revisit the example $p(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^2\mathbf{x}_2^2$. What made this possible in this example is that the function

[3] We note that their algorithm works under a slightly more general class of distributions, whose moments up to order $2d$ are known a priori.

[4] The Chow parameters of a Boolean function $f(\mathbf{x})$ are defined as the vector $\mathbf{E}_{\mathbf{x}}[f(\mathbf{x})\mathbf{x}]$. Similarly, the degree-$d$ Chow-parameter tensor is defined as $\mathbf{E}_{\mathbf{x}}[f(\mathbf{x})\mathbf{x}^{\otimes d}]$.

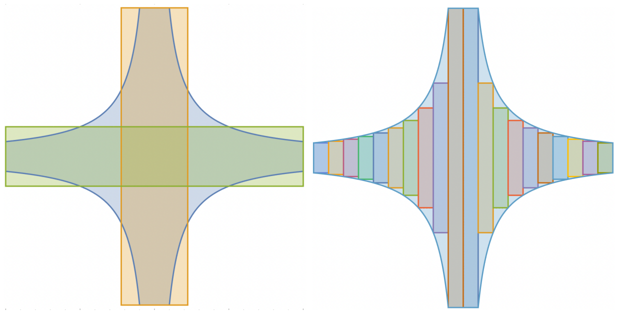

Figure 1: The localization region $|p(\mathbf{x}_1, \mathbf{x}_2)| = |\mathbf{x}_1^2\mathbf{x}_2^2| \le \epsilon$ is shown in blue. It is essentially a union of two rectangles (shown in the left figure) of width roughly $\sqrt{\epsilon}$. It is easy to see that (i) the total mass of the union is roughly $\sqrt{\epsilon}$); (ii) the expected value of $\mathbf{x}_1^2$ conditioned on the union is roughly $\Theta(1)$ (due to the contribution of the green rectangle). If the conditional distribution were a Gaussian, Carbery-Wright anti-concentration would imply that the conditional probability of $\left|\mathbf{x}_1^2\right| < \epsilon$ should be at most $\mathrm{poly}(\epsilon)$. In sharp contrast, the mass of the set $\left|\mathbf{x}_1^2\right| < \epsilon$ conditioned on the union is roughly $\Theta(1)$ (due to the contribution of the orange rectangle).
To mitigate the issue, we will partition the low-margin set $|p(\mathbf{x}_1, \mathbf{x}_2)| \le \epsilon$ into multiple rectangles as in the right figure. Since the Gaussian conditioned on each rectangle is a log-concave distribution, we have the desirable (anti-)concentration properties by [5].

can be decomposed into the *linear terms* $\mathbf{x}_1$ and $\mathbf{x}_2$. Conditioning the Gaussian density on a rectangle of the form $\mathbf{x}_1 \in I_1, \mathbf{x}_2 \in I_2$ yields a log-concave distribution with good anti-concentration. More generally, if we could always decompose a polynomial p into *a small number of* linear polynomials $q_1, \ldots, q_\ell$, then we would still have anti-concentration in the resulting conditional distributions after partitioning the region $|p(\mathbf{x})| < \epsilon$ into rectangles defined by the linear terms, i.e., every $q_i(\mathbf{x})$ lies in an interval I_i. Unfortunately, this is not possible for a general polynomial p. However, such a decomposition will exist if we allow the set of polynomials $q_1, \ldots, q_\ell$ (or more precisely the polynomial mapping $\mathbf{x} \mapsto \mathbf{q}(\mathbf{x}) := (q_1(\mathbf{x}), \ldots, q_\ell(\mathbf{x}))$) to only *resemble a linear transformation locally.* To achieve this, we will need to leverage and generalize the results of [32] on *non-singular decompositions*, which itself builds on the techniques of diffuse decompositions from [33]. In particular, we say that a collection of polynomials $q_1, q_2, \ldots, q_\ell$ is *non-singular* if there is only a negligible probability that the Jacobian of the (vector-valued) polynomial transformation $\mathbf{q}(\mathbf{x})$ (i.e., the matrix $[\nabla q_1(\mathbf{x})\nabla q_2(\mathbf{x}) \ldots \nabla q_\ell(\mathbf{x})]$) has small singular values. Intuitively, when this is the case, the polynomial transformation $\mathbf{q}$ will locally resemble a non-singular linear transformation. In [32], it is shown that for any polynomial of degree at most d, there exists a non-singular set of of $O_d(1)$ polynomials $q_1, \ldots, q_\ell$ so that p can approximately be written as a polynomial in the q_i's. It turns out that having a non-singular decomposition is not enough to establish the anti-concentration properties that we require. We introduce the notion of *super non-singularity*, which enforces "local linearity" by restricting the high-order derivatives of the polynomials. In particular, we establish two structural results on super non-singular sets of polynomials. First, we establish that the Gaussian distribution conditioned on a super non-singular set of polynomials, each lying in some interval, satisfies good anti-concentration and concentration properties.

Theorem 2.2 (Conditional (Anti-)Concentration for Super Non-Singular Transformation). *Let $d, \ell, K, C_{d,\ell,K}$ be positive integers and $\epsilon \in (0, 1)$. Let $S = \{q_1, \cdots, q_\ell\}$ be a set of harmonic polynomials, where each $q_i : \mathbb{R}^n \mapsto \mathbb{R}$ is of degree at most d. Define $\mathbf{q} : \mathbb{R}^n \mapsto \mathbb{R}^\ell$ to be the vector-valued polynomial such that $\mathbf{q}(\mathbf{x}) = (q_1(\mathbf{x}), \cdots, q_\ell(\mathbf{x}))$. Let $R \subset \mathbb{R}^\ell$ be an axis-aligned rectangle satisfying (i) each point in R is at most $\mathrm{poly}_{d,\ell}(\log(1/\epsilon))$-far from the origin, and (ii) $\mathbf{Pr}_{\mathbf{x}\sim\mathcal{N}(\mathbf{0},\mathbf{I})}[\mathbf{x} \in R] > \mathrm{poly}_{d,\ell}(\epsilon)$. Denote by D the distribution of $\mathcal{N}(\mathbf{0}, \mathbf{I})$ conditioned on $\{\mathbf{x} : \mathbf{q}(\mathbf{x}) \in R\}$. Suppose $C_{d,\ell,K}$ is sufficiently large given d, ℓ, K, and ϵ is sufficiently small given $d, \ell, K, C_{d,\ell,K}$. Assume that $\{q_1, \cdots, q_\ell\}$ is $(\epsilon^{1/(3d^2K)}, C_{d,\ell,K})$-super non-singular. Then for any polynomial $p : \mathbb{R}^n \mapsto \mathbb{R}$ of degree-d and $\epsilon < t < \epsilon^{2/K}$, it holds*

$$\Pr_{\mathbf{x}\sim D}\left[|p(\mathbf{x})| < t\ \ \|p\|_{D,L_2}\right] \le t^{1/(2d)}\,, \tag{1}$$

and, for all $t > 0$, it holds

$$\Pr_{\mathbf{x}\sim D}\left[|p(\mathbf{x})| > t\epsilon^{-1/K}\ \|p\|_{D,L_2}\right] \le O_{d,K}\left(t^{-K}\right). \tag{2}$$

Second, we give an efficient super non-singular decomposition algorithm. In particular, given a general degree-d polynomial $p : \mathbb{R}^n \mapsto \mathbb{R}$, there exists a computationally efficient algorithm that finds a super non-singular set of $m = O_d(1)$ polynomials $q_1, \ldots, q_m$ such that there exists a polynomial $h(\cdot)$ of degree at most d which satisfies $p(\mathbf{x}) \approx h(q_1(\mathbf{x}), \ldots, q_m(\mathbf{x}))$.

Theorem 2.3 (Extendible Super Non-singular Decomposition). *Let n, ℓ, d, M be positive integers, $f : \mathbb{Z}^+ \mapsto \mathbb{Z}^+$ be some function, and $\epsilon > 0$ be sufficiently small given ℓ, d, M, f. Let $S := \{q_1, \cdots, q_\ell\}$, where $q_i : \mathbb{R}^n \mapsto \mathbb{R}$, be a set of degree at most d harmonic polynomials and $p : \mathbb{R}^n \mapsto \mathbb{R}$ be some other degree-d polynomial. Suppose S is $(\epsilon^{1/3}, C_{d,\ell,f,M})$ super non-singular, where $C_{d,\ell,f,M}$ is sufficiently large given d, ℓ, f, M. Then there exists an algorithm which can extend S into $\bar{S} := \{q_1, \cdots, q_\ell, q_{\ell+1}, \cdots, q_m\}$ such that*

- *$\bar{S}$ is of size $m = O_{d,\ell,f,M}(1)$.*
- *$\bar{S}$ is $(\epsilon, f(m))$-super non-singular.*
- *There exists a polynomial $h : \mathbb{R}^m \mapsto \mathbb{R}$ with $\|h\|_{L_2} \le \epsilon^{-3d-1}$ such that*

$$\|p(\mathbf{x}) - h(q_1(\mathbf{x}), \cdots, q_m(\mathbf{x}))\|_{L_2} \le \epsilon^M.$$

Moreover, the algorithm runs in time $\mathrm{poly}(n)\ \mathrm{poly}_{d,\ell,M,f}(1/\epsilon)$.

Equipped with the above structural and algorithmic results, obtaining an efficient partition algorithm is relatively straightforward. After computing a super non-singular decomposition $q_1, \ldots, q_m$ of p using Theorem 2.3, we have that since p can be approximately expressed as a polynomial $h(q_1(\mathbf{x}), \ldots, q_m(\mathbf{x}))$, the value of $p(\mathbf{x})$ is (approximately) determined by the values of $q_i(\mathbf{x})$. Therefore, we can show that the set $\{\mathbf{x} \in \mathbb{R}^n : |p(\mathbf{x})| < \epsilon\}$ can be approximately covered by sets of the form $\{\mathbf{x} : (q_1(\mathbf{x}), \cdots, q_\ell(\mathbf{x}) \in R\}$, where R is an m-dimensional axis-aligned rectangle. Hence, anti-concentration properties of the conditional distributions on these sets follow.

This allows us to perform at least one round of localization by partitioning the set $L = \{\mathbf{x} : |p(\mathbf{x})| \le \epsilon\}$. Assuming that we have obtained a polynomial p' that achieves good error in the region R, we then need to "localize" on the region $L' = \{\mathbf{x} : |p(\mathbf{x})| \le \epsilon, |p'(\mathbf{x})| \le \epsilon\}$. We show that this is possible by a subtle "extendibility" property of our super non-singular decomposition algorithm. Specifically, assuming that we have a super non-singular decomposition $q_1, \cdots, q_\ell$ of p (which was used to compute p'), we can then extend it into a larger super non-singular set $\{q_1, \cdots, q_\ell, q_{\ell+1}, \cdots, q_m\}$ such that we can still approximately express $p'(\mathbf{x})$ as a polynomial in $q_1, \cdots, q_m$. For each rectangle R of the original partition of $p(\mathbf{x})$, we can now further cover the region $\{\mathbf{x} \in \mathbb{R}^n : |p'(\mathbf{x})| < \epsilon, (q_1(\mathbf{x}), \cdots, q_\ell(\mathbf{x})) \in R\}$ with sets of the form $\{\mathbf{x} : (q_1(\mathbf{x}), \cdots, q_\ell(\mathbf{x}) \in R, (q_{\ell+1}(\mathbf{x}), \cdots, q_m(\mathbf{x}) \in R')\}$, where R' is some other $(m-\ell)$-dimensional axis-aligned rectangle. As a slight digression, we note that this extendibility property is also what makes the proof of Theorem 2.2 possible.

2.2 Anti-concentration via Extendible Super Non-Singular Decomposition

We have already seen (see Figure 1) that the Gaussian distribution conditioned on sets of the form $|p_1(\mathbf{x})| < \epsilon, |p_2(\mathbf{x})| < \epsilon, \cdots$ for generic polynomials p_i does not satisfy good anti-concentration. To mitigate this issue, we need the polynomials appearing in the conditioning to collectively satisfy a strong non-singularity condition concerning their high-order derivatives. In the following definition, we denote by ∇_x the standard gradient operator and by $D_\mathbf{y}$ the derivative in the direction $\mathbf{y}$.

Definition 2.4 (Super Non-Singular Polynomial Transformation (SNPT)). *Let $\epsilon \in (0,1)$ and $N \in \mathbb{Z}_+$. Let $S := \{q_1, \cdots, q_m\}$, where $q_i : \mathbb{R}^n \mapsto \mathbb{R}$ is a set of harmonic (see Section 2.5 of [34] for the formal definition) real-valued polynomials of degree at most d. For $1 \le k \le d$, let $S_k \subseteq S$ be the set of degree-k harmonic polynomials (contained in S). The set of polynomials S is (ϵ, N)-super non-singular if for any integer $1 \le k \le d$ it holds that*

$$\left\| \nabla_\mathbf{x} D_{\mathbf{y}^{(k-1)}} \cdots D_{\mathbf{y}^{(1)}} \left(\sum_{i \in S_k} \mathbf{a}_i \, q_i(\mathbf{x}) \right) \right\|_2 < \epsilon$$

with probability at most ϵ^N, where the randomness is over $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \mathbf{y}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ for $1 \le i \le k-1$, for all $\mathbf{a} \in \mathbb{R}^m$ such that $\sum_{i \in S_k} \mathbf{a}_i^2 = 1$. We will also call (ϵ, N)-super non-singular a polynomial transformation $\mathbf{q}(\mathbf{x}) = (q_1(\mathbf{x}), \ldots, q_m(\mathbf{x}))$ defined by an (ϵ, N)-super non-singular set S.

We remark that Definition 2.4 resembles the definition of non-singular polynomials in [32], but imposes additional requirements on the high-order derivatives of the polynomials. This additional structure turns out to be crucial in proving Theorem 2.3, which is itself an important building block to establish Theorem 2.5. As one of our main contributions, we show that the distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$, conditioned on a set of super non-singular polynomials each lying in some interval (satisfying some mild conditions), satisfies good polynomial concentration and anti-concentration properties. For brevity, we henceforth refer to both concentration and anti-concentration as (anti-)concentration.

Theorem 2.5 (Informal – Conditional (anti-)concentration for SNPT, see Theorem 2.2). *Let $\mathbf{q}$ be a degree d, "sufficiently" super non-singular polynomial transformation (i.e., for large enough ϵ, N in Definition 2.4). Let $R \subseteq \mathbb{R}^m$ be an axis-aligned rectangle that is not too far from the origin and let D be $\mathcal{N}(\mathbf{0}, \mathbf{I})$ conditioned on the set $\{\mathbf{x} : \mathbf{q}(\mathbf{x}) \in R\}$. For any unit variance, mean-zero polynomial p of degree at most d we have:*

- *(Anti-Concentration)* $\mathbf{Pr}_{\mathbf{x} \sim D}[|p(\mathbf{x})| \le t] \le t^{1/(2d)}$.
- *(Concentration) For all $K \in \mathbb{Z}_+$ up to some constant* [5]*, it holds* $\mathbf{Pr}_{\mathbf{x} \sim D}[|p(\mathbf{x})| > t] \le t^{-1/K}$.

We now provide a sketch of the high-level ideas behind the proof of the above theorem. In what follows, we denote by $p(D)$ the distribution of the random variable $p(\mathbf{x})$ when $\mathbf{x} \sim D$. Let D be the distribution of $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ conditioned on $\mathbf{q}(\mathbf{x}) \in R$ for a rectangle R. Our goal is to show that for any low-degree polynomial p, the distribution $p(D)$ has good anti-concentration.

Constructing a Low-Dimensional Surrogate Distribution. As our first step, instead of directly analyzing the (anti-)concentration properties of p under the n-dimensional distribution D, which is challenging, we construct low-dimensional "surrogates" for D and p. Specifically, we consider a low-dimensional distribution Q together with a polynomial f, such that the outcome of $p(D)$ enjoys roughly the same concentration and anti-concentration properties as $f(Q)$. Given the construction, we can in turn focus on analyzing this low-dimensional surrogate pair.

The fact that super non-singular decompositions are "extendible" will play a critical role in this construction. In particular, the super non-singular polynomials $\{q_1, \cdots, q_\ell\}$ appearing in the conditioning of D will first be extended into a super non-singular decomposition for the target polynomial p such that there exists a set of super non-singular polynomials $\{q_1, \cdots, q_\ell, q_{\ell+1}, \cdots, q_m\}$, where $m = O_{d,\ell}(1)$, and a composition polynomial $f : \mathbb{R}^m \mapsto \mathbb{R}$ such that $p(\mathbf{x}) \approx f(q_1(\mathbf{x}), \cdots, q_m(\mathbf{x}))$. Define $\mathbf{q} : \mathbb{R}^n \mapsto \mathbb{R}^m$ to be the vector-valued polynomial whose i-th coordiniate is q_i, and Q to be the m-dimensional distribution of $\mathbf{q}(\mathcal{N}(\mathbf{0}, \mathbf{I}))$ conditioned on the set $\{\mathbf{y} \in \mathbb{R}^m : \mathbf{y}_i \in I_i \; \forall i = 1, \ldots, \ell\}$. Then, subject to the intervals appearing in the conditioning satisfying some mild conditions and the polynomials appearing in the conditioning being sufficiently super non-singular, one can verify that $p(D)$ enjoys roughly the same (anti-)concentration properties as $f(Q)$. For the details of this argument, we refer the readers to the full version of the paper.

Given such a construction, we can now shift our focus from the n-dimensional distribution D to the $m = O_{d,\ell}(1)$-dimensional conditional distribution Q defined by a set of super non-singular polynomials. In particular, if we use $\mathbf{q} : \mathbb{R}^n \mapsto \mathbb{R}^m$ to denote the vector-valued polynomial whose i-th coordinate is q_i, we are interested in the distribution of $\mathbf{q}\,(\mathcal{N}(\mathbf{0}, \mathbf{I}))$ conditioned on the event $\{\mathbf{q}_i(\mathbf{x}) \in I_i\}_{i=1}^{\ell}$, where I_i is some interval. To build some intuition as to why such conditional distributions may have desirable properties, we can start with the simple case where all q_i are linear functions. In that case, the distribution of $\mathbf{q}(\mathcal{N}(\mathbf{0}, \mathbf{I}))$ is simply some other Gaussian distribution $\mathcal{N}'$. Then, even if we condition on that the i-th

[5] The constant depends on "how" super non-singular the set is; see Theorem 2.2.

coordinate of $\mathbf{q}(\mathcal{N}(\mathbf{0},\mathbf{I}))$ is equal to $\mathbf{a}_i$, the resulting distribution will simply be some lower-dimensional Gaussian distribution. Under mild conditions on $\mathbf{a}_i$ and q_i, the resulting low-dimension Gaussian will be not too different from a standard Gaussian, in the sense that its mean is not very far from the origin and its covariance is bounded above and below by multiples of the identity.

Definition 2.6 ((δ,κ)-Reasonable Gaussian). *Let $\mathcal{N}(\mu,\Sigma)$ be a Gaussian distribution. Given $\delta \in (0,1)$ and $\kappa > 1$, we say $\mathcal{N}(\mu,\Sigma)$ is a (δ,κ)-reasonable Gaussian if $\|\mu\|_2 \le \kappa$ and $\delta\mathbf{I} \le \Sigma \le \kappa\mathbf{I}$.*

When the polynomials are of degree more than 1, it becomes hard to characterize the exact form of $\mathbf{q}(\mathcal{N}(\mathbf{0},\mathbf{I}))$. Nonetheless, the hope is that we can still compare its probability density function to that of some other more structured distribution family.

Definition 2.7 (Distribution Comparability). *Let Q, Q' be probability distributions with the same support. We say that Q and Q' are comparable if for all $\mathbf{x}$ in their common support, it holds $1/2\ Q'(\mathbf{x}) \le Q(\mathbf{x}) \le 2\ Q'(\mathbf{x})$.* [6]

We show that if two distributions are comparable to each other, then they will have similar (anti-)concentration properties — even under an arbitrary conditioning. The formal statement of this fact and its proof can be found in in the full version of the paper.

Super Non-Singular Polynomial Transformations of Gaussians are Reasonable. Given a polynomial transformation $\mathbf{x} \mapsto \mathbf{q}(\mathbf{x})$, we will say that $\mathbf{q}$ is super non-singular if the set of its polynomial coordinates $\mathbf{q}_i(\mathbf{x})$ is super non-singular. We show that a super non-singular transformation $\mathbf{q}$ behaves similarly to a linear transformation, in the sense that the distribution $q(\mathcal{N}(\mathbf{0},\mathbf{I}))$ is comparable to a mixture of reasonable Gaussians.

Proposition 2.8 (Informal – Super Non-Singular Polynomial Transformations are Reasonable). *Let $\mathbf{q}$ be a $(\delta^{1/(3d)}, C)$-super non-singular polynomial transformation for some sufficiently small δ and large C. Then, $\mathbf{q}(\mathcal{N}(\mathbf{0},\mathbf{I}))$ is $O(\delta^N)$-close in total variation distance to some distribution that is comparable to the mixture distribution $\int N_\theta d\theta$, where each N_θ is a $(\delta, \log^{O(d)}(1/\delta))$-reasonable Gaussian.*

We now provide a proof sketch for Proposition 2.8. We first note that $\mathbf{q}(\mathcal{N}(\mathbf{0},\mathbf{I}))$ has a distribution identical to $\mathbf{q}(\sqrt{1-\delta^2}\mathbf{x} + \delta\mathbf{z})$ for $\delta \in (0,1)$ an appropriately chosen real number and $\mathbf{x},\mathbf{z}$ distributed as two i.i.d. Gaussians. Fixing the value of $\mathbf{x}$ and Taylor expanding around $\mathbf{z}$, we find that $\mathbf{q}(\sqrt{1-\delta^2}\mathbf{x}+\delta\mathbf{z})$ is approximately $\mathbf{g}_\mathbf{x} + \delta\,\mathrm{Jac}_\mathbf{q}(\mathbf{x})\ \mathbf{z} + O(\delta^2)\mathbf{e}_\mathbf{x}(\mathbf{z})$, where $\mathrm{Jac}_\mathbf{q}$ represents the Jacobian of the transformation $\mathbf{q}$, $\mathbf{g}_\mathbf{x}$ is some vector that depends only on $\mathbf{x}$, and $\mathbf{e}_\mathbf{x}$ is some degree-d polynomial. It turns out if $\mathbf{q}$ consists of super non-singular polynomials, $\mathrm{Jac}_\mathbf{q}(\mathbf{x})$ will have no small singular values with high probability. Conditioned on some fixed value of $\mathbf{x}$ that makes $\mathrm{Jac}_\mathbf{q}(\mathbf{x})$ non-singular, the distribution of $\mathbf{g}_\mathbf{x} + \delta\,\mathrm{Jac}_\mathbf{q}(\mathbf{x})\mathbf{z}$ where $\mathbf{z} \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ will be a reasonable Gaussian. We remark that the transformation still has the high-order error term $O(\delta^2)\mathbf{e}_\mathbf{x}(\mathbf{z})$ that we have to bound. We notice that the coefficient in front of the high-order term is significantly smaller than the minimum singular value of the linear component. As a result, the distribution produced by the transformation will still be close in total variation distance to some distribution *comparable* to a reasonable Gaussian distribution. We refer to the full version of the paper for more details.

We remark that all of the above analysis is done for a fixed value of $\mathbf{x}$ that ensures non-singularity of $\mathrm{Jac}_\mathbf{q}(\mathbf{x})$. Hence, to conclude the proof, we simply need to take a mixture over the values of $\mathbf{x}$ following the standard Gaussian distribution. By super non-singularity, $\mathrm{Jac}_\mathbf{q}(\mathbf{x})$ has no small singular values with high probability. Consequently, most of the distributions within the mixture will be comparable to a reasonable Gaussian distribution. Proposition 2.8 thereby follows.

Given Proposition 2.8, and the definition of comparability, we conclude that the transformation $\mathbf{q}$ conditioned on an axis-aligned rectangle enjoys good (anti)-concentration properties. By our construction, the target polynomial p under the target distribution D enjoys roughly the same (anti)-concentration properties as some polynomial h under $\mathbf{q}(\mathcal{N}(\mathbf{0},\mathbf{I}))$ conditioned on an axis-aligned rectangle. The proof of Theorem 2.2 follows.

2.3 Efficiently Extending a Super Non-Singular Decomposition

In this section, we discuss our efficient algorithm for obtaining and extending a super non-singular decomposition. [32] shows that any polynomial of degree at most d can be approximately decomposed into a non-singular polynomial set of size at most $O_d(1)$. In Theorem 2.9, we show that this is also true for the notion of super non-singularity. Theorem 2.9 extends and strengthens the result of [32] in two ways: (i) we are able to decompose *multiple* (as opposed to just one, as in [32]) generic polynomials into a *common* set of super non-singular polynomials, and (ii) we are able to do so when the generic polynomials arrive in an *online* fashion. In particular, given a super non-singular set of polynomials Q obtained while decomposing some polynomials $p_1, \cdots, p_t$ in the past rounds, after receiving the new polynomial p_{t+1}, we are able to extend Q into a larger set of super non-singular polynomials Q' and decompose p_{t+1} in terms of Q'. We remark that the fact that we can keep extending a super non-singular set of polynomials to ensure it can be used to represent increasingly more polynomials is a unique characteristic of super non-singular decomposition (compared to its "non-super" counterpart). Crucially, this additional "extendibility" property of the decomposition is what makes the (anti-)concentration result (Theorem 2.5) and the polynomial set partitioning routine (Theorem 2.1) possible. In the following result, we present our efficient algorithm for extending a super non-singular decomposition.

Theorem 2.9 (Informal – Extendible Super Non-singular Decomposition). *Let $\ell, d, N' \in \mathbb{Z}_+$ and $\epsilon > 0$ be sufficiently small given ℓ, d. Let $S := \{q_1, \cdots, q_\ell\}$, where $q_i : \mathbb{R}^n \mapsto \mathbb{R}$, be a set of polynomials of degree at most d and $p : \mathbb{R}^n \mapsto \mathbb{R}$ be another polynomial of degree at most d. Suppose that S is $(\epsilon^{1/3}, N)$ super non-singular, where N is sufficiently large given d, ℓ, N'. Then there exists an algorithm which can extend S into a set of $m = O_{d,\ell}(1)$ polynomials $\bar{S} := \{q_1, \cdots, q_\ell, q_{\ell+1}, \cdots, q_m\}$ such that*

- *$\bar{S}$ is (ϵ, N')-super non-singular.*
- *There exists a polynomial $h : \mathbb{R}^m \mapsto \mathbb{R}$ of degree at most d such that $\|p(\mathbf{x}) - h(\mathbf{q}(\mathbf{x}))\|_{L_2} \le \epsilon^2$, where $\mathbf{q} = (q_1(\mathbf{x}), \ldots, q_m(\mathbf{x}))$.*

[6] For readers who are familiar with the notion of Rényi divergence, this is equivalent to stating that the symmetrized Rényi divergence of infinite order between the two distributions is bounded by some constant.

Moreover, the algorithm runs in time $\mathrm{poly}(n)\ \mathrm{poly}_{\ell,d,N'}(1/\epsilon)$.

Suppose we only want to compute a non-singular decomposition for a polynomial p. The process given in [32] maintains a data-structure to which we refer as a partial decomposition. Informally, the data-structure keeps track of a list of polynomials $q_1, \cdots, q_\ell$ (which is not necessarily non-singular), a coefficient vector $\mathbf{b} \in \mathbb{R}^\ell$, and a composition polynomial h such that $p(\mathbf{x}) = h(\mathbf{b}_1 q_1(\mathbf{x}), \cdots, \mathbf{b}_\ell q_\ell(\mathbf{x}))$. If the list of polynomials is already non-singular, we are done. Otherwise, following the definition of non-singularity, there exists a linear combination of the polynomials $q^*(\mathbf{x}) = \sum_i \mathbf{b}_i q_i(\mathbf{x})$ such that the gradient of the combined polynomial q^* is small with non-trivial probability under the Gaussian distribution. In the second case, we show that we can approximately decompose the combined polynomial q^* into a set of lower-degree polynomials $\alpha_1, \cdots, \alpha_m$. Hence, we can rewrite one of the polynomials q_i which has non-trivial weight in the linear combination with the set of newly obtained lower-degree polynomials α_i and the remaining polynomials in the linear combination. We then end up with a new partial decomposition consisting of the polynomials $q_1, \cdots, q_{i-1}, q_{i+1}, \cdots, q_\ell, \alpha_1, \cdots, \alpha_m$, a new coefficient vector $\mathbf{b}'$ and a new composition polynomial h'. It turns out such a rewriting strategy will always decrease the total weights of the polynomials that have the same degree as q_i in $\mathbf{b}'$, but may end up increasing the weights of the other polynomials in the linear combination. However, if we always choose to rewrite the highest (or one of the highest) degree among the polynomials in the linear combination, it is then guaranteed that we will have fewer and fewer high-degree polynomials in the decomposition. This process must then eventually terminate and give us a super non-singular set of polynomials.

In order to adapt the above strategy for extendible super non-singular decomposition, a caveat here is that for the additional extendibility property to hold, we are now allowed to rewrite any of the initial polynomials $q_1, \cdots, q_\ell$. Fortunately, if a set of polynomials does not satisfy super non-singularity, we are always capable of finding a linear combination of the polynomials *of the same degree* such that the combined polynomial has small gradients with non-trivial probability. Therefore, no matter which polynomial q_i we choose to rewrite, it will always have the highest degree among the polynomials in the linear combination (since all of them have the same degree!). The induction argument for showing the termination of the process will then go through, giving us a super non-singular decomposition algorithm.

2.4 Learning via Localization and Margin-Perceptron

At a high-level, our learning algorithm can be viewed as a robust version of the margin-perceptron algorithm of [26]. In particular, we establish the following: given sample access to a distribution corrupted by opt-nasty noise, the margin-perceptron is a semi-agnostic LTF learner when the underlying (uncorrupted) $\mathbf{x}$-marginal is "sufficiently" (anti)-concentrated.

Proposition 2.10. *Let* $\epsilon \in (0,1)$, $K \in \mathbb{Z}_+$, $C, C_1 > 0$, *and* D *be a distribution on* $\mathbb{R}^n \times \{\pm 1\}$. *Assume the following statements are true.*

(1) *The samples* $\mathbf{x}$ *are labeled with respect to an unknown function* $f(\mathbf{x}) = \mathrm{sign}(\mathbf{w}^* \cdot \mathbf{x})$ *for some* $\mathbf{w}^* \in \mathbb{R}^n$.
(2) *The covariance matrix of the marginal distribution* $D_\mathbf{x}$ *satisfies* $\mathbf{E}_{\mathbf{x}\sim D_\mathbf{x}}[\mathbf{x}^\top \mathbf{x}] \preccurlyeq 2\mathbf{I}$.
(3) *The* $\mathbf{x}$*-marginal distribution* $D_\mathbf{x}$ *satisfies anti-concentration, i.e.,*

$$\Pr_{\mathbf{x}\sim D_\mathbf{x}}[|\mathbf{v}\cdot\mathbf{x}| \le t\|\mathbf{v}\|_{D_\mathbf{x},L_2}] \le t^C ,$$

for some number $C > 0$, *and for all* $t \in (\epsilon, \epsilon^{2/\sqrt{K}})$.
(4) $D_\mathbf{x}$ *has concentration, i.e.,*

$$\Pr_{\mathbf{x}\sim D_\mathbf{x}}[|\mathbf{v}\cdot\mathbf{x}| > t\|\mathbf{v}\|_{D_\mathbf{x},L_2}] \le C_1 t^{-K} ,$$

for some $C_1 > 0$ *and all* $t > 0$

Then, given access to an ϵ*-corrupted version of* $\mathrm{poly}(n/\epsilon)$ *i.i.d. samples from* D, *there exists an algorithm that runs in sample-polynomial time, and with probability at least* $2/3$ *computes a weight vector* $\widehat{\mathbf{w}}$ *such that*

$$\Pr_{(\mathbf{x},y)\sim D}[\mathrm{sign}(\widehat{\mathbf{w}}\cdot\mathbf{x}) \neq y \mid B] \le C_1^{1/K}\ O_{C,K,d}(\epsilon^{1-O(1/\sqrt{K})}) ,$$

where $B = \{\mathbf{x} : |\mathbf{w}\cdot\mathbf{x}| \ge \gamma\|\mathbf{w}\|_{D_\mathbf{x},L_2}\}$ *for* $\gamma = \epsilon^{4/\sqrt{K}}$.

Our modified perceptron algorithm of Proposition 2.10 uses a robust sub-routine of [22] for estimating the Chow-parameters of the LTF under nasty noise. For example, in the first round, we learn a PTF that achieves error opt^{1-c} for all high-margin points in the set $\{\mathbf{x} : |p(\mathbf{x})| \ge \epsilon^c \|p\|_2\}$. We then localize (condition) on the set of low-margin points $\{\mathbf{x} : |p(\mathbf{x})| \le \epsilon^c \|p\|_2\}$, and use the partitioning algorithm of Theorem 2.1 to partition the above region into sets $S^{(1)}, \ldots, S^{(m)}$ such that the standard normal conditional on those sets has good (anti)-concentration. We then again condition on each set of this partition and use the robust-perceptron algorithm of Proposition 2.10 to learn a PTF inside each set; and continue recursively until the probability mass of the "unclassified" low-region is at most $O(\epsilon)$. Our final hypothesis is therefore a decision list of degree-d PTFs: one PTF for each set of the partition. For more details, we refer the reader to the full version of the paper.

REFERENCES

[1] P. Awasthi, M. F. Balcan, and P. M. Long. 2017. The Power of Localization for Efficiently Learning Linear Separators with Noise. *J. ACM* 63, 6 (2017), 50:1–50:27.

[2] M. Balcan and P. Long. 2013. Active and passive learning of linear separators under log-concave distributions. In *Conference on Learning Theory*. PMLR, 288–316.

[3] A. Blum, A. M. Frieze, R. Kannan, and S. Vempala. 1996. A Polynomial-Time Algorithm for Learning Noisy Linear Threshold Functions. In *37th Annual Symposium on Foundations of Computer Science, FOCS '96*. 330–338.

[4] N. Bshouty, N. Eiron, and E. Kushilevitz. 2002. PAC Learning with Nasty Noise. *Theoretical Computer Science* 288, 2 (2002), 255–275.

[5] A. Carbery and J. Wright. 2001. Distributional and L^q norm inequalities for polynomials over convex bodies in R^n. *Mathematical Research Letters* 8, 3 (2001), 233–248.

[6] A. Daniely. 2015. A PTAS for Agnostically Learning Halfspaces. In *Proceedings of The 28th Conference on Learning Theory, COLT 2015*. 484–502.

[7] A. Daniely. 2016. Complexity theoretic limitations on learning halfspaces. In *Proceedings of the 48th Annual Symposium on Theory of Computing, STOC 2016*. 105–117.

[8] C. Daskalakis, I. Diakonikolas, and R. A. Servedio. 2012. Learning Poisson Binomial Distributions. In *Proceedings of the 44th Symposium on Theory of Computing*. 709–728.

[9] A. De, I. Diakonikolas, V. Feldman, and R. A. Servedio. 2014. Nearly Optimal Solutions for the Chow Parameters Problem and Low-Weight Approximation of Halfspaces. *J. ACM* 61, 2 (2014), 11:1–11:36.

[10] A. De and R. A. Servedio. 2014. Efficient deterministic approximate counting for low-degree polynomial threshold functions. In *Symposium on Theory of Computing, STOC 2014, 2014*. ACM, 832–841.

[11] M. Dertouzos. 1965. *Threshold Logic: A Synthesis Approach.* MIT Press, Cambridge, MA.

[12] I. Diakonikolas, S. Goel, S. Karmalkar, A. R. Klivans, and M. Soltanolkotabi. 2020. Approximation Schemes for ReLU Regression. In *Conference on Learning Theory, COLT (Proceedings of Machine Learning Research, Vol. 125)*. PMLR, 1452–1485.

[13] I. Diakonikolas, P. Gopalan, R. Jaiswal, R. Servedio, and E. Viola. 2010. Bounded independence fools halfspaces. *SIAM J. Comput.* 39, 8 (2010), 3441–3462.

[14] I. Diakonikolas, P. Harsha, A. Klivans, R. Meka, P. Raghavendra, R. A. Servedio, and L. Y. Tan. 2010. Bounding the average sensitivity and noise sensitivity of polynomial threshold functions. In *STOC*. 533–542.

[15] I. Diakonikolas, G. Kamath, D. M. Kane, J. Li, A. Moitra, and A. Stewart. 2016. Robust Estimators in High Dimensions without the Computational Intractability. In *Proceedings of FOCS'16*. 655–664.

[16] I. Diakonikolas and D. M. Kane. 2019. Degree-d Chow parameters robustly determine degree-d PTFs (and algorithmic applications). In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, 2019*. ACM, 804–815.

[17] I. Diakonikolas and D. M. Kane. 2023. *Algorithmic high-dimensional robust statistics*. Cambridge University Press.

[18] I. Diakonikolas, D. M. Kane, P. Manurangsi, and L. Ren. 2022. Cryptographic Hardness of Learning Halfspaces with Massart Noise. *CoRR* abs/2207.14266 (2022). https://doi.org/10.48550/arXiv.2207.14266 arXiv:2207.14266 Conference version in NeurIPS'22..

[19] I. Diakonikolas, D. M. Kane, and J. Nelson. 2010. Bounded Independence Fools Degree-2 Threshold Functions. In *FOCS*. 11–20.

[20] I. Diakonikolas, D. M. Kane, T. Pittas, and N. Zarifis. 2021. The Optimality of Polynomial Regression for Agnostic Learning under Gaussian Marginals in the SQ Model. In *Proceedings of The 34th Conference on Learning Theory, COLT*.

[21] I. Diakonikolas, D. M. Kane, and L. Ren. 2023. Near-Optimal Cryptographic Hardness of Agnostically Learning Halfspaces and ReLU Regression under Gaussian Marginals. In *ICML*. https://doi.org/10.48550/arXiv.2302.06512 arXiv:2302.06512

[22] I. Diakonikolas, D. M. Kane, and A. Stewart. 2018. Learning geometric concepts with nasty noise. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018*. 1061–1073.

[23] I. Diakonikolas, R. O'Donnell, R. A. Servedio, and Y. Wu. 2011. Hardness Results for Agnostically Learning Low-Degree Polynomial Threshold Functions. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, 2011*. SIAM, 1590–1606.

[24] I. Diakonikolas, P. Raghavendra, R. A. Servedio, and L. Y. Tan. 2014. Average Sensitivity and Noise Sensitivity of Polynomial Threshold Functions. *SIAM J. Comput.* 43, 1 (2014), 231–253.

[25] I. Diakonikolas, R. Servedio, L.-Y. Tan, and A. Wan. 2010. A regularity lemma, and low-weight approximators, for low-degree polynomial threshold functions. In *CCC*. 211–222.

[26] J. Dunagan and S. Vempala. 2004. A simple polynomial-time rescaling algorithm for solving linear programs. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*. 315–320.

[27] P. Harsha, A. R. Klivans, and R. Meka. 2014. Bounding the Sensitivity of Polynomial Threshold Functions. *Theory of Computing* 10 (2014), 1–26.

[28] D. Haussler. 1992. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and Computation* 100 (1992), 78–150.

[29] A. Kalai, A. Klivans, Y. Mansour, and R. Servedio. 2008. Agnostically Learning Halfspaces. *SIAM J. Comput.* 37, 6 (2008), 1777–1805. Special issue for FOCS 2005..

[30] D. M. Kane. 2010. The Gaussian surface area and noise sensitivity of degree-d polynomial threshold functions. In *CCC*. 205–210.

[31] D. M. Kane. 2011. k-Independent Gaussians Fool Polynomial Threshold Functions. In *IEEE Conference on Computational Complexity*. 252–261.

[32] D. M. Kane. 2012. A Pseudorandom Generator for Polynomial Threshold Functions of Gaussian with Subpolynomial Seed Length. *CoRR* abs/1210.1280 (2012).

[33] D. M. Kane. 2012. A Structure Theorem for Poorly Anticoncentrated Gaussian Chaoses and Applications to the Study of Polynomial Threshold Functions. In *FOCS*. 91–100.

[34] D. M. Kane. 2017. A structure theorem for poorly anticoncentrated polynomials of Gaussians and applications to the study of polynomial threshold functions. *Ann. Probab.* 45, 3 (2017), 1612–1679. https://doi.org/10.1214/16-AOP1097

[35] M. Kearns, R. Schapire, and L. Sellie. 1994. Toward Efficient Agnostic Learning. *Machine Learning* 17, 2/3 (1994), 115–141.

[36] M. J. Kearns and M. Li. 1993. Learning in the presence of malicious errors. *SIAM J. Comput.* 22, 4 (1993), 807–837.

[37] A. Klivans, P. Long, and R. Servedio. 2009. Learning Halfspaces with Malicious Noise. *Journal of Machine Learning Research* 10 (2009), 2715–2740.

[38] A. Klivans, R. O'Donnell, and R. Servedio. 2008. Learning Geometric Concepts via Gaussian Surface Area. In *Proc. 49th IEEE Symposium on Foundations of Computer Science (FOCS)*. Philadelphia, Pennsylvania, 541–550.

[39] W. Maass and G. Turan. 1994. How fast can a threshold gate learn?. In *Computational Learning Theory and Natural Learning Systems*, S. Hanson, G. Drastal, and R. Rivest (Eds.). MIT Press, 381–414.

[40] R. Meka and D. Zuckerman. 2010. Pseudorandom generators for polynomial threshold functions. In *STOC*. 427–436.

[41] M. Minsky and S. Papert. 1968. *Perceptrons: an introduction to computational geometry*. MIT Press, Cambridge, MA.

[42] S. Muroga. 1971. *Threshold logic and its applications*. Wiley-Interscience, New York.

[43] R. O'Donnell, R. A. Servedio, and L.-Y. Tan. 2020. Fooling Gaussian PTFs via local hyperconcentration. In *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, 2020*. ACM, 1170–1183.

[44] A. Sherstov. 2009. The intersection of two halfspaces has high threshold degree. In *Proc. 50th IEEE Symposium on Foundations of Computer Science (FOCS)*.

[45] A. A. Sherstov and P. Wu. 2019. Near-optimal lower bounds on the threshold degree and sign-rank of AC^0. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, 2019*. ACM, 401–412.

[46] S. Tiegel. 2023. Hardness of Agnostically Learning Halfspaces from Worst-Case Lattice Problems. In *COLT*. arXiv:2207.14030

[47] L. Trevisan, M. Tulsiani, and S. P. Vadhan. 2009. Regularity, Boosting, and Efficiently Simulating Every High-Entropy Distribution. In *IEEE Conference on Computational Complexity*. 126–136.

[48] L. Valiant. 1985. Learning disjunctions of conjunctions. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*. 560–566.

Received 13-NOV-2023; accepted 2024-02-11