A Fully Polynomial-Time Algorithm for Robustly Learning Halfspaces over the Hypercube

Gautam Chandrasekaran*
gautamc@cs.utexas.edu
UT Austin

Adam R. Klivans†
klivans@cs.utexas.edu
UT Austin

Konstantinos Stavropoulos‡
kstavrop@utexas.edu
UT Austin

Arsen Vasilyan§
arsenvasilyan@gmail.com
UT Austin

Abstract

We give the first fully polynomial-time algorithm for learning halfspaces with respect to the uniform distribution on the hypercube in the presence of contamination, where an adversary may corrupt some fraction of examples and labels arbitrarily. We achieve an error guarantee of $\eta^{O(1)}+\varepsilon$ where η is the noise rate. Such a result was not known even in the agnostic setting, where only labels can be adversarially corrupted. All prior work over the last two decades has a superpolynomial dependence in $1/\varepsilon$ or succeeds only with respect to continuous marginals (such as log-concave densities).

Previous analyses rely heavily on various structural properties of continuous distributions such as anti-concentration. Our approach avoids these requirements and makes use of a new algorithm for learning Generalized Linear Models (GLMs) with only a polylogarithmic dependence on the activation function's Lipschitz constant. More generally, our framework shows that supervised learning with respect to discrete distributions is not as difficult as previously thought.

*Supported by the NSF AI Institute for Foundations of Machine Learning (IFML).

†Supported by NSF award AF-1909204 and the NSF AI Institute for Foundations of Machine Learning (IFML).

‡Supported by the NSF AI Institute for Foundations of Machine Learning (IFML) and by the 2025 Apple Scholars in AI/ML PhD fellowship.

§Supported by the NSF AI Institute for Foundations of Machine Learning (IFML).

1 Introduction

Distribution-specific supervised learning is a central and long-standing problem in computational learning theory. In the case of *continuous* high-dimensional distributions, a wide range of polynomial-time learning algorithms are known, including [Vem10, ABL17, Dan15, DKS18a, DKK+21, DZ24]. For example, the celebrated work of [ABL17] gives a polynomial-time algorithm for agnostically learning linear classifiers under log-concave distributions. These results highlight a mature algorithmic understanding of learning in well-studied continuous settings.

For learning under *discrete* distributions, however, our algorithmic understanding is much less complete. None of the aforementioned results extend to the most basic and well-studied discrete high-dimensional distribution: the uniform distribution over the hypercube $\{\pm 1\}^d$. In fact, with very few exceptions (such as monotone $O(\log n)$ juntas), the only known function classes that admit polynomial-time learnability with respect to discrete distributions seem to be function classes that can be learned with respect to *all* distributions.

The main reason for this striking gap between the continuous and discrete settings is that the existing algorithms rely crucially on *geometric* tools of high-dimensional distributions—such as the *anti-concentration property* and the *localization technique* [ABL17]—which fail for high-dimensional discrete distributions. These geometric tools underpin nearly all known polynomial-time learning algorithms for continuous distributions, but they do not hold, even approximately, in discrete domains such as the hypercube.

This motivates the following fundamental question:

Is supervised learning under discrete distributions inherently harder than learning under continuous distributions?

In this work, we take a major step toward answering this question by giving the *first polynomial-time algorithm* for arguably the most fundamental problem in supervised learning under discrete high-dimensional distributions — the problem of agnostically learning halfspaces under the uniform distribution on the Boolean hypercube. Halfspaces constitute one of the most extensively studied concept classes in learning theory, while the uniform distribution on the Boolean cube is among the most canonical and well-analyzed distributions in this context.

Specifically, our algorithm achieves classification error $\mathrm{opt}^{O(1)} + \epsilon$, where opt is the optimal halfspace error. All previous polynomial-time algorithms for related problems in continuous settings relied heavily on anti-concentration properties of the underlying distribution. Our result bypasses this limitation entirely: the uniform distribution on $\{\pm 1\}^d$ is not anti-concentrated, yet our algorithm still achieves a fully polynomial-time guarantee.

The fastest previously known algorithm for this problem was given by [OS08], which runs in time $\mathrm{poly}(d) \cdot 2^{\mathrm{poly}(1/\epsilon)}$. Our algorithm has an exponentially better runtime while maintaining the $\mathrm{opt}^{O(1)} + \epsilon$ accuracy guarantee. (See Table 1 for a detailed comparison.)

Attempts to improve on [OS08] have led to significant follow-up research. In particular, [DDFS14] developed a faster (though not polynomial-time) algorithm based on *Chow parameter matching*. Their algorithm achieves quasi-polynomial accuracy blow-up $2^{-\Omega(\sqrt[3]{\log(1/\mathrm{opt})})} + \epsilon$ and runs in quasi-polynomial time $\mathrm{poly}(d) \cdot 2^{O(\log^3(1/\epsilon))}$. Compared to this result, our algorithm achieves a quasi-polynomial improvement in both runtime and accuracy. Moreover, [DDFS14] prove that even with improved Chow-reconstruction techniques, an $\mathrm{opt}^{O(1)} + \epsilon$ error bound *cannot* be achieved by any approach based solely on Chow parameter matching. Thus, our results fundamentally go beyond the limitations of prior methods.

To overcome these obstacles, we develop a new technique for learning under high-dimensional discrete distributions. At a high level, our approach combines ideas from *Generalized Linear Models (GLMs)* with the analysis of *high-influence variables*. As a byproduct of our analysis, we also obtain the first polynomial-time algorithm for the fundamental task of distribution-free learning of *sigmoidal Generalized Linear Models* whose runtime scales *polylogarithmically* with the Lipschitz constant of the activation function. This is an exponential improvement over the runtime of previous algorithms, all of which scaled *linearly* with the Lipschitz constant.

Finally, our algorithm extends naturally to the more challenging setting of learning under contamination (a.k.a. *nasty noise*). When an η-fraction of the samples are arbitrarily corrupted, our algorithm outputs a halfspace with classification error $\eta^{O(1)} + \epsilon$.

Error Guarantee	Runtime	Reference
$\mathrm{opt} + \epsilon$	$d^{O(1/\epsilon^2)}$	[KKMS08]
$\mathrm{opt}^{\Omega(1)} + \epsilon$	$\mathrm{poly}(d) \cdot 2^{\mathrm{poly}(1/\epsilon)}$	[OS08]
$2^{-\Omega(\sqrt[3]{\log(1/\mathrm{opt})})} + \epsilon$	$\mathrm{poly}(d) \cdot 2^{O(\log^3(1/\epsilon))}$	[DDFS14]
$\mathrm{opt}^{\Omega(1)} + \epsilon$	$\mathrm{poly}(d, 1/\epsilon)$	**This work**

Table 1: Upper bounds for agnostic learning of halfspaces over the uniform distribution on $\{\pm 1\}^d$.

Related Work

Polynomial-Time Robust Learning Algorithms under Continuous Distributions. There has been a large body of work on $\mathrm{poly}(d/\epsilon)$-time algorithms for robust learning over continuous high-dimensional distributions, such as the Gaussian distribution and log-concave distributions.

The work of [Vem10] gives polynomial-time algorithms for (non-robust) learning intersections of halfspaces with respect to the Gaussian distribution. The work of [ABL17] gives a polynomial-time algorithm for learning halfspaces under log-concave distributions. The work of [DKTZ20b] extends the result of [ABL17] to a broader family of high-dimensional continuous distributions satisfying certain well-behavedness assumptions, such as concentration, anti-concentration, and so-called anti-anti-concentration.

The work of [Dan15] gives a polynomial-time approximation scheme (PTAS) for learning halfspaces under the Gaussian distribution, achieving error $(1+\mu)\mathrm{opt} + \epsilon$ for any small constant μ. The work of [DKK+21] further improves on [Dan15] by giving a proper PTAS under the Gaussian distribution that outputs a halfspace as the learned hypothesis.

The works of [KLS09, ABL17, DKS18a] give polynomial-time learning algorithms for learning halfspaces over log-concave distributions under contamination (also known as nasty noise). The work of [DKS18a] also applies to learning constant-degree polynomial threshold functions (PTFs) under log-concave distributions with contamination. The more recent work of [DKK+24] gives a polynomial-time algorithm for the same task with improved robustness under the Gaussian distribution.

There has also been a long line of work on polynomial-time algorithms for learning halfspaces in the Massart noise model under log-concave distributions [ABHU15, ABHZ16, YZ17, ZLC17, MV19, DKTZ20a]. The algorithm of [DKTZ20a] runs in time polynomial in d/ϵ and achieves an error of $\mathrm{opt} + \epsilon$.

None of the polynomial-time algorithms mentioned above extend to discrete high-dimensional distributions, such as the uniform distribution over the Boolean cube. This demonstrates a large gap in our understanding between robust PAC learning under continuous and discrete high-dimensional distributions.

Robust Learning of Halfspaces over the Hypercube. The algorithm of [KKMS08, DGJ+10] runs in time $d^{\tilde{O}(1/\epsilon^2)}$ and achieves agnostic learning of halfspaces over the Boolean cube up to error $\mathrm{opt} + \epsilon$. There is a strong evidence that this run-time is optimal: as noted in [KKMS08] even an improvement in the run-time of $d^{\tilde{O}(1/\epsilon^{(2-\beta)})}$ for positive constant β will yield faster algorithms for the parity with noise problem. See also [DSFT+14] for a statistical query lower bound.

As mentioned earlier, the work of [OS08] gives an algorithm that runs in time $\mathrm{poly}(d) \cdot 2^{\mathrm{poly}(1/\epsilon)}$ and achieves an error of $\mathrm{opt}^{O(1)} + \epsilon$ in the agnostic learning setting.

The subsequent work of [DDFS14] introduced a faster — though still not polynomial-time — algorithm based on *Chow parameter matching*. This algorithm achieves a quasi-polynomial accuracy guarantee of $2^{-\Omega(\sqrt[3]{\log(1/\mathrm{opt})})} + \epsilon$ and operates in quasi-polynomial time $\mathrm{poly}(d) \cdot 2^{O(\log^3(1/\epsilon))}$.

The limitations of the Chow parameter matching technique were also explored in [DDFS14], and it was shown that no method based solely on Chow parameter matching can achieve an error of $\mathrm{poly}(\mathrm{opt}) + \epsilon$. Even more, it was shown that Chow parameter matching cannot yield an accuracy better than $\mathrm{opt}^{\omega(1/(\log\log \mathrm{opt}))}$. Our techniques break through this barrier.

The approach of [DK19] can be used to extend the result of [DDFS14] into the setting of learning with contaminated data (nasty noise).

Learning Generalized Linear Models. Generalized Linear Models have a long history in the literature of machine learning [Ros58, NW72, DB18]. They correspond to the scenario where the input samples $(\boldsymbol{x}, y) \in \mathcal{X} \times \{\pm 1\}$ are generated by some distribution such that $\mathbb{E}[y|\boldsymbol{x}] = \sigma(\boldsymbol{w}^* \cdot \boldsymbol{x})$ for some known, monotone activation function $\sigma : \mathbb{R} \to [-1, 1]$ and some unknown vector $\boldsymbol{w}^*$. Learning halfspaces is a special case of GLM learning with the sign activation, which is, however, discontinuous. Several works study efficient learnability under continuous GLMs [KS09, KKSK11, KM17, CKMY20]. Many of these and other related results, including ours, make use of the notion of the matching loss, initially proposed by [AHW95]. Most related to our work, the classical result of [KKSK11] showed that GLMtron — an algorithmic variant of the perceptron algorithm — learns any GLM with known, Lipschitz activation with runtime that scales polynomially with $\|\boldsymbol{w}^*\|_2$. Our results exponentially improve the run-time dependence on $\|\boldsymbol{w}^*\|_2$ compared to the GLMtron [KKSK11], by establishing, for the first time, that a poly-logarithmic dependence on $\|\boldsymbol{w}^*\|_2$ is achievable for the sigmoidal activation.

A wide range of related results on learning GLMs with continuous activations have been provided in the literature. Several works consider the case where the labels are real-valued and noisy, and the goal of the learner is to output a hypothesis whose error competes with the optimum error achieved by a function of the form $\sigma(\boldsymbol{w} \cdot \boldsymbol{x})$ (often referred to as *robustly learning a single neuron*). Such results fall into two main categories, depending on whether the activation σ is known [SSSS10, DGK+20, DKTZ22, ATV23, WZDD23, WZDD24, GV24] or unknown [DH18, GGKS23, ZWDD24, RBH+25, WZDD25]. These results typically make strong assumptions on the marginal distribution on $\boldsymbol{x}$. There is strong evidence that strong assumptions are necessary when the labels are noisy [DKMR22]. In fact, most of the aforementioned results assume that $\boldsymbol{x}$ follows some continuous distribution, such as the Gaussian. It remains an open question whether our results on robust halfspace learning over the hypercube — which corresponds to agnostic GLM learning with respect to the sign activation — can be extended to other activations as well.

2 Brief Technical Overview

Algorithm Outline. Our algorithm receives m i.i.d. examples drawn from some distribution over pairs $(\boldsymbol{x}, y) \in \{\pm 1\}^d \times \{\pm 1\}$ such that $\boldsymbol{x}$ follows the uniform distribution over the d-dimensional hypercube and there is an unknown halfspace $f^*(\boldsymbol{x}) = \mathrm{sign}(\boldsymbol{v}^* \cdot \boldsymbol{x} + \tau^*)$ such that $\Pr[y \neq f^*(\boldsymbol{x})] = \mathrm{opt}$. The output of the algorithm is guaranteed to have error $O(\mathrm{opt}^{1/c}) + \varepsilon$ for some constant c.[1,2] We proceed in the following phases (see also Algorithm 1 for more details).

- **Phase 1: Finding indices of influential variables.** We first find a set of indices $j_1, j_2, \ldots, j_k \in [d]$ that correspond to the k largest (in absolute value) coordinates of some vector $\tilde{\boldsymbol{v}}^*$ such that:

$$\Pr[f^*(\boldsymbol{x}) \neq \mathrm{sign}(\tilde{\boldsymbol{v}}^* \cdot \boldsymbol{x} + \tau^*)] \leq 2k \cdot \mathrm{opt} \tag{1}$$

- **Phase 2: Learn a structured halfspace.** We find a halfspace $h_{\mathrm{r}}(\boldsymbol{x}) = \mathrm{sign}(\widehat{\boldsymbol{v}}_{\mathrm{head}} \cdot \boldsymbol{x} + \widehat{\boldsymbol{v}}_{\mathrm{tail}} \cdot \boldsymbol{x} + \widehat{\tau})$, where $\widehat{\boldsymbol{v}}_{\mathrm{head}}$ is supported on $j_1, \ldots, j_k$ and $\widehat{\boldsymbol{v}}_{\mathrm{tail}}$ is supported on the remaining coordinates.
  (a) First, we compute $(\widehat{\boldsymbol{v}}_{\mathrm{head}}, \widehat{\tau})$ through Algorithm 3.
  (b) Then, we compute $\widehat{\boldsymbol{v}}_{\mathrm{tail}}$ through Algorithm 4 using $(\widehat{\boldsymbol{v}}_{\mathrm{head}}, \widehat{\tau})$ as part of the input.
- **Phase 3: Learn a sparse halfspace.** We attempt to find $h_{\mathrm{sp}}(\boldsymbol{x}) = \mathrm{sign}(\widehat{\boldsymbol{v}}_{\mathrm{sp}} \cdot \boldsymbol{x} + \widehat{\tau})$ defined by a coefficient vector $\widehat{\boldsymbol{v}}_{\mathrm{sp}}$ which is only supported on the coordinates $j_1, j_2, \ldots, j_k$ that perfectly classifies a set of $\mathrm{poly}(k, 1/\varepsilon)$ input examples. This can be done through linear programming.

Proof Outline. Our proof is quite technical and splits into several cases as depicted by the tree of reductions in Figure 1. It starts with a robust correspondence between coefficients of large magnitude and the coordinate influences of the unknown halfspace (see Section 2.3 for a more thorough overview). This enables us to reduce the initial halfspace learning problem to one where the coordinates corresponding to the k largest coefficients are known, by incurring an $O(k)$ error amplification (Phase 1). We proceed with the *critical index* machinery introduced by Servedio [Ser06] (we use a statement from [GOWZ10]), which asserts that any halfspace over the Boolean hypercube is either close to a sparse halfspace or is *structured*, meaning that beyond a small number of *heavy (or head)* coefficients, independent of the input dimension, all remaining weights are of roughly comparable magnitude (the *regular or tail* coefficients). This allows us to decompose the learning task into two subproblems: learning either a structured halfspace (Phase 2) or a sparse halfspace (Phase 3) — we try both cases in parallel and check afterwards to see which case was successful.

Structured Case. The structured case comprises the bulk of the technical work of the paper and requires several new ideas. First, we establish a connection between learning structured halfspaces and learning generalized linear models (GLMs), which allows us to learn the heavy coefficients (see Section 2.1 for more background on GLMs and an overview of our GLM learning results). Concretely, we treat the aggregate contribution of the tail variables as approximately Gaussian noise and represent each example as $(\tilde{\boldsymbol{x}}, y)$, where $\tilde{\boldsymbol{x}}$ corresponds to the heavy variables and $\mathbb{E}[y|\tilde{\boldsymbol{x}}]$ is given by a GLM with (approximately) known activation. This viewpoint is justified by the regularity property of the tail weights, which enables the use of

[1] In our proofs, we assume that $\mathrm{opt} \leq \varepsilon^c$ for some absolute constant c, which is without loss of generality since one can always redefine ε as a new value $\varepsilon' \geq \mathrm{opt}^{1/c}$.

[2] The constant we provide is $c = 25$, but it is possible that it can be improved.

the Berry-Esseen theorem to predict the collective behavior of the inner product between the tail variables and the corresponding (regular) coefficients.

A key observation is that the number of heavy coefficients k can be chosen small enough relative to the input noise level so that the effect of noise becomes negligible when learning the corresponding coefficients. Specifically, if the error of some halfspace f is η, then for $m \ll 1/\eta$ labeled i.i.d. samples, with high probability all labels are consistent with f.

In our analysis, we choose the halfspace f to be the one whose k largest coefficients are known and has error $\eta \leq (2k+1)\mathrm{opt}$ (Phase 1, Eq. (1)). The choice $m \ll 1/\eta$ is feasible because the effective dimension of the problem is $\dim(\tilde{\boldsymbol{x}}) = k \ll 1/\mathrm{opt}^c$ for some sufficiently large constant c. Moreover, according to the critical index lemma, if f is structured, then the heavy coefficients coincide with the k largest coefficients of f, whose positions are known. This enables us to identify the heavy variables $\tilde{\boldsymbol{x}}$.

Finally, once the heavy coefficients are fixed, we estimate the tail coefficients robustly and efficiently by minimizing a convex surrogate loss (see Section 2.2 for more details). The crucial observation here is that the tail distribution is anti-concentrated — again, by Berry-Esseen — ensuring that the value of the hinge loss is representative of the misclassification error of the optimal halfspace.

Sparse Case. Once more, we use the fact that the number of relevant variables k in the sparse case can be chosen small enough relative to the input noise level. Since the positions of the relevant variables are known (due to Phase 1), we can use a simple linear program to recover a sparse halfspace consistent with the data and guaranteed to be close to the optimum.

2.1 Learning the Heavy Coefficients: A Connection with GLM Learning

Recall that in the structured case (Phase 2 of our algorithm), we first learn the heavy coefficients. We will now sketch the main idea behind the algorithm used for this purpose. First, we focus on a seemingly unrelated problem called GLM (Generalized Linear Model) learning, which turns out to be closely related to the problem of learning the heavy coefficients in the structured case. In GLM learning, there is a distribution over pairs $(\boldsymbol{x}, y) \in \mathbb{R}^d \times \{\pm 1\}$, where the conditional expectation $\mathbb{E}[y|\boldsymbol{x}]$ is specified by some function of the form $\sigma(\boldsymbol{w}^* \cdot \boldsymbol{x} + \tau^*)$, where $\sigma : \mathbb{R} \to [-1, 1]$ is known, and $\boldsymbol{w}, \tau$ are unknown. The goal is to learn some $\widehat{w}, \widehat{\tau}$ such that:

$$\mathbb{E}\Big[\big(\sigma(\boldsymbol{w}^* \cdot \boldsymbol{x} + \tau^*) - \sigma(\widehat{\boldsymbol{w}} \cdot \boldsymbol{x} + \widehat{\tau})\big)^2\Big] \leq \varepsilon\,.$$

One of our main technical contributions is a GLM learning algorithm whose sample complexity is independent of the magnitude $w_{\max}$ of associated coefficients, and whose runtime scales logarithmically with $w_{\max}$. In particular, we obtain the first efficient algorithm for learning sigmoidal GLMs (activation $\sigma^{\mathrm{sig}}(t) = (1 - e^{-t})/(1 + e^{-t})$) with sample complexity independent of the weights:

Theorem 1. *There is an algorithm that, given sample access to a distribution over pairs $(\boldsymbol{x}, y) \in \mathbb{R}^d \times \{\pm 1\}$ where $\|\boldsymbol{x}\|_2 \leq \sqrt{d}$ and $\mathbb{E}[y|\boldsymbol{x}] = \sigma^{\mathrm{sig}}(\boldsymbol{w}^* \cdot \boldsymbol{x} + \tau^*)$ for some unknown $\boldsymbol{w}^*, \tau^*$ with $\|\boldsymbol{w}^*\|_2 + |\tau^*| \leq w_{\max}$, outputs $\widehat{\boldsymbol{w}}, \widehat{\tau}$ such that with probability at least 0.99 we have:*

$$\mathbb{E}\Big[\big(\sigma^{\mathrm{sig}}(\boldsymbol{w}^* \cdot \boldsymbol{x} + \tau^*) - \sigma^{\mathrm{sig}}(\widehat{\boldsymbol{w}} \cdot \boldsymbol{x} + \widehat{\tau})\big)^2\Big] \leq \varepsilon\,.$$

The sample complexity of the algorithm is $\tilde{O}(d/\varepsilon^2)$ and its time complexity is $\mathrm{poly}(d, 1/\varepsilon, \log w_{\max})$.

The proof of the above theorem combines two key ingredients: (1) a known connection between GLM learning and a class of surrogate losses often referred to as *matching losses* (see, e.g., [Kan18]), and (2) the guarantees provided by the ellipsoid algorithm. To ensure that the sample complexity does not depend on $w_{\max}$, we first establish the result for GLMs with *clipped activations* — that is, activations $\sigma(t)$ whose absolute value is fixed at 1 whenever $|t|$ exceeds a threshold α. We then argue that for sufficiently large α, since the sigmoid approaches 1 exponentially fast, the distribution induced by the clipped sigmoid GLM is within vanishing total variation distance of the original GLM distribution, relative to the number of samples needed for generalization. Hence, the result for the clipped model directly extends to the original GLM.

A modified version of the above result (see Theorem 15) is crucial in our approach here, but we believe that Theorem 1 is of independent interest, since, to our knowledge, it is the first GLM learning result to achieve a logarithmic runtime dependence on the magnitude of the coefficients, which essentially quantifies how steep the transition of $\mathbb{E}[y|\boldsymbol{x}]$ from -1 to 1 is, as $\boldsymbol{x}$ moves along the direction of $\boldsymbol{w}^*$. See section Section 5 for the proof of Theorem 15 and Section B.2 for the proof of Theorem 1.

In order to apply the result in our case, we identify GLM learning as a particular subproblem of learning a halfspace $f(\boldsymbol{x}) = \mathrm{sign}(\tilde{\boldsymbol{v}}^* \cdot \boldsymbol{x} + \tau^*)$ over the d-dimensional hypercube whose k-largest coefficients are in known positions. Specifically, due the critical index lemma, one important scenario is that $\tilde{\boldsymbol{v}}^* = \boldsymbol{v}^*_{\mathrm{head}} + \boldsymbol{v}^*_{\mathrm{tail}}$, where $\boldsymbol{v}^*_{\mathrm{head}}$ is supported on the k known coordinates (where k does not depend on d), and $\boldsymbol{v}^*_{\mathrm{tail}}$ is supported on the remaining coordinates and is regular, i.e., $\|\boldsymbol{v}^*_{\mathrm{tail}}\|_2 = 1$ and $\|\boldsymbol{v}^*_{\mathrm{tail}}\|_4 \ll 1$. By focusing on the coordinates corresponding to $\boldsymbol{v}^*_{\mathrm{head}}$ — which are known due to the definition of $\tilde{\boldsymbol{v}}^*$ in Phase 1, Eq. (1) — we obtain a new GLM learning problem over a distribution of pairs $(\tilde{\boldsymbol{x}}, y)$ where $\tilde{\boldsymbol{x}} \in \mathbb{R}^k$ and:

$$\mathbb{E}[y|\tilde{\boldsymbol{x}}] = \mathop{\mathbb{E}}_{\boldsymbol{x}' \sim \{\pm 1\}^{d-k}} \left[\mathrm{sign}\left(\boldsymbol{u}^*_{\mathrm{head}} \cdot \tilde{\boldsymbol{x}} + \boldsymbol{u}^*_{\mathrm{tail}} \cdot \boldsymbol{x}' + \tau^* \right) \right], \tag{2}$$

where $\boldsymbol{u}^*_{\mathrm{head}}$ is the restriction of $\boldsymbol{v}^*_{\mathrm{head}}$ on the k known coordinates and $\boldsymbol{u}^*_{\mathrm{tail}}$ the restriction of $\boldsymbol{v}^*_{\mathrm{tail}}$ to the remaining coordinates. We show that a solution $(\widehat{\boldsymbol{v}}_{\mathrm{head}}, \widehat{\tau})$ to the corresponding GLM problem can be used in place of $(\boldsymbol{v}^*_{\mathrm{head}}, \tau^*)$, increasing the classification distance from the initial halfspace by only a small amount. There are several additional technical complications.

First, the GLM defined by Equation (2) has unknown activation, since $\boldsymbol{v}^*_{\mathrm{tail}}$ is unknown, but the surrogate loss technique requires knowledge of the activation. To address this, we apply Berry-Esseen theorem (Theorem 33), which implies that the distribution of $\boldsymbol{v}^*_{\mathrm{tail}} \cdot \boldsymbol{x}'$ is very close to the standard Gaussian. This leads to a sigmoid-like GLM, which has all the required properties to obtain a result similar to Theorem 1, but is slightly misspecified.[3] This is because we do not know the activation values exactly, but only approximately. Nevertheless, we show that our technique is robust to such misspecification.

Second, we still require some bound on $\|\boldsymbol{v}^*_{\mathrm{head}}\|_2$, since the runtime of our GLM algorithm scales poly-logarithmically to the magnitude of the coefficients. Such a bound can be obtained through a classical result in Boolean analysis showing that every halfspace over the d-dimensional hypercube can be exactly represented by integer coefficients of magnitude at most $w_{\max} = 2^{O(d \log d)}$ [Ser06]. Since we obtain a logarithmic dependence on $w_{\max}$, our runtime remains polynomial.

Finally, Equation (2) holds only when the input distribution is realized by the halfspace f^*, yet our input distribution is actually noisy. Nevertheless, since the effective dimension of the corresponding GLM problem is $k \ll 1/\mathrm{opt}^c$ for some constant c (due to the critical index lemma), the number of samples required to solve the problem is small compared to the label noise rate (opt) and the labels are effectively realized by f^*. Therefore, our method is robust to noise.

[3] The actual activation is $\psi(z) \coloneqq \mathbb{E}_{x \sim \mathcal{N}(0,1)}[\mathrm{sign}(z + x)]$, which has similar properties to the sigmoid.

2.2 Learning the Regular Coefficients: Hinge Loss Minimization

Another important ingredient of our approach is an algorithm that learns the tail coefficients, once we have estimates for the heavy coefficients. A simpler version of the result we use in our proof is the following result, which solves a particular case of the noisy halfspace learning problem over the hypercube when the coefficients are regular.

Theorem 2. *There is an algorithm that, given sample access to a distribution over pairs $(\boldsymbol{x}, y) \in \{\pm 1\}^d \times \{\pm 1\}$ where $\boldsymbol{x}$ is uniform over $\{\pm 1\}^d$, outputs $\widehat{\boldsymbol{v}} \in \mathbb{R}^d$ such that with probability at least 0.99 we have:*

$$\Pr\Big[\mathrm{sign}(\widehat{\boldsymbol{v}} \cdot \boldsymbol{x}) \neq y\Big] \leq \frac{C}{\varepsilon} \cdot \log \frac{1}{\varepsilon} \cdot \mathrm{opt}_\varepsilon + C \cdot \varepsilon\,, \;\; \textit{for} \;\; \mathrm{opt}_\varepsilon := \min_{f \in \mathcal{C}_\varepsilon} \Pr\Big[f(\boldsymbol{x}) \neq y\Big]$$

where $\mathcal{C}_\varepsilon$ is the class of ε-regular halfspaces, i.e., $f(\boldsymbol{x}) = \mathrm{sign}(\boldsymbol{v} \cdot \boldsymbol{x})$ where $\|\boldsymbol{v}\|_2 = 1$ and $\|\boldsymbol{v}\|_4^2 \leq \varepsilon$ and C is some sufficiently large constant. The sample and time complexity of the algorithm is $\mathrm{poly}(d, 1/\varepsilon)$.

The algorithm minimizes an ε-rescaled version of the hinge loss, i.e., it minimizes the empirical expected value of the function $\ell_\varepsilon(y, \boldsymbol{v} \cdot \boldsymbol{x}) = \mathsf{ReLU}(1 - y(\boldsymbol{v} \cdot \boldsymbol{x})/\varepsilon)$ over the input samples $(\boldsymbol{x}, y)$. It is a well-known fact that the hinge loss is a surrogate for the missclassification error, i.e., $\mathbb{1}\{y \neq \mathrm{sign}(\boldsymbol{v} \cdot \boldsymbol{x})\} \leq \ell_\varepsilon(y, \boldsymbol{v} \cdot \boldsymbol{x})$. Therefore, achieving a small value for the hinge loss leads to low missclassification error.

The non-trivial part of our proof is showing that the minimum value of the hinge loss is indeed shrinking with opt_ε. To this end, we focus on the value of the hinge loss on $\boldsymbol{v}_\varepsilon^*$, where $\boldsymbol{v}_\varepsilon^*$ defines the regular halfspace $f_\varepsilon^* \in \mathcal{C}_\varepsilon$ that achieves opt_ε. Due to regularity of $f_\varepsilon^*(\boldsymbol{x}) = \mathrm{sign}(\boldsymbol{v}_\varepsilon^* \cdot \boldsymbol{x})$ and uniformity of $\boldsymbol{x}$, we may apply the Berry-Esseen theorem (Theorem 33) to show that the distribution of $\boldsymbol{v}_\varepsilon^* \cdot \boldsymbol{x}$ is anticoncentrated. Moreover, the distribution of $\boldsymbol{v}_\varepsilon^* \cdot \boldsymbol{x}$ is concentrated, since $\boldsymbol{x}$ is subgaussian. These two properties suffice to relate opt_ε and $\mathbb{E}[\ell_\varepsilon(y, \boldsymbol{v}_\varepsilon^* \cdot \boldsymbol{x})]$.

In our proof, we use a slightly more general version of Theorem 2 (Theorem 22), which enables the learner to fix the heavy coefficients $\boldsymbol{v}_{\mathrm{heavy}}$ and the bias τ to their estimated values $(\widehat{\boldsymbol{v}}_{\mathrm{heavy}}, \widehat{\tau})$ (from the GLM component) and only optimize over the regular tail $\boldsymbol{v}_{\mathrm{tail}}$. The additional complication here is that there could be some points $\boldsymbol{x}$ where $|\widehat{\boldsymbol{v}}_{\mathrm{head}} \cdot \boldsymbol{x} + \widehat{\tau}| \gg |\boldsymbol{v}_{\mathrm{tail}} \cdot \boldsymbol{x}|$ leading to large values of the surrogate loss, even for the optimal choice of $\boldsymbol{v}_{\mathrm{tail}}$. To overcome this obstacle, we may simply filter out the points $\boldsymbol{x}$ for which $|\widehat{\boldsymbol{v}}_{\mathrm{head}} \cdot \boldsymbol{x} + \widehat{\tau}| \gg \log(1/\varepsilon)$, as the missclassification error on them is not influenced by the choice of $\boldsymbol{v}_{\mathrm{tail}}$, because the sign is determined by $\widehat{\boldsymbol{v}}_{\mathrm{head}} \cdot \boldsymbol{x} + \widehat{\tau}$. See Section 6 for more details.

2.3 Finding the Positions of Influential Variables

One crucial part of our proof is finding the coordinates $i_1, \ldots, i_k \in [d]$ that correspond to coefficients of the optimum halfspace with the k largest magnitudes. This is important for robustness, since it enables us to learn a sparse halfspace in the sparse case, as well as the heavy coefficients in the structured case, by solving appropriate low-dimensional problems (see Phase 1, Eq. (1)). If the dimensionality of the problem is low compared to the noise rate, then robustness can be guaranteed in a black-box manner, due to a total variation distance-based argument.

The coordinates $i_1, \ldots, i_k$, however, cannot be recovered exactly, due to the label noise, as well as sampling errors. Instead, we provide the following robust coordinate identification result, which is sufficient for our purposes.

Theorem 3 (Simplified version of Theorem 5)**.** *There is an algorithm that, given sample access to a distribution over pairs $(\boldsymbol{x}, y) \in \{\pm 1\}^d \times \{\pm 1\}$ where $\boldsymbol{x}$ is uniform over $\{\pm 1\}^d$ and $\Pr[y \neq \mathrm{sign}(\boldsymbol{v}^* \cdot \boldsymbol{x} + \tau^*)] \leq \eta$, outputs a set of k indices $\{j_1, \ldots, j_k\}$ such that with probability at least 0.99:*

$$\Pr\Big[\mathrm{sign}(\boldsymbol{v}^* \cdot \boldsymbol{x} + \tau^*) \neq \mathrm{sign}(\tilde{\boldsymbol{v}}^* \cdot \boldsymbol{x} + \tau^*)\Big] \leq 4\eta k\,,$$

where $\tilde{\boldsymbol{v}}^*$ is such that the k largest (in absolute value) coefficients correspond to the indices $j_1, \ldots, j_k$. The algorithm has time and sample complexity $\mathrm{poly}(d, 1/\eta)$.

The above result is essentially the strongest possible statement one could hope for, at least from a qualitative perspective. It implies that for all purposes, we may assume that the returned indices $j_1, \ldots, j_k$ correspond to the coefficients of largest magnitudes, by only incurring an error amplification factor of $4k$. Moreover, the critical index lemma (Theorem 25) precisely identifies the large-magnitude coefficients with the relevant coordinates in the sparse case and the heavy coefficients in the structured case.

The proof of the above theorem is based on carefully combining the following three main ingredients. First, we use the fact that the ordering of the coordinate influences of a halfspace over the hypercube coincides with the ordering of the absolute values of its coefficients. Second, we show that the absolute difference between the influences of two coordinates i, j gives an upper bound on the disagreement between two halfspaces that correspond to coefficient vectors $\boldsymbol{v}, \boldsymbol{v}'$, where $\boldsymbol{v}$ and $\boldsymbol{v}'$ are the same in all coordinates except i, j, where the magnitudes are swapped, i.e., $v'_i = |v_j| \cdot \mathrm{sign}(v_i)$ and $v'_j = |v_i| \cdot \mathrm{sign}(v_j)$. Finally, we observe that the coordinate influences can be efficiently estimated through the Chow parameters (see [OS08]). See Section 4 for more details.

2.4 Tolerating Contamination

Our method works even in the more challenging setting where both the features and the labels are adversarially contaminated for a bounded fraction of the input examples. In particular, our results apply to the following model called bounded contamination (or nasty noise), initially defined by [BEK02].

Definition 4 (Learning with Bounded Contamination [BEK02])**.** We say that a set S_{inp} of N labeled samples is an η-contaminated (uniform) sample with respect to some class $\mathcal{C} \subseteq \{\{\pm 1\}^d \to \{\pm 1\}\}$, where $N \in \mathbb{N}$ and $\eta \in (0, 1)$, if it is formed by an adversary as follows.

1. The adversary receives a set of N clean i.i.d. labeled examples S_{cln}, drawn from the uniform distribution over $\{\pm 1\}^d$ and labeled by some unknown concept f^* in $\mathcal{C}$.
2. The adversary removes an arbitrary set S_{rem} of $\lfloor \eta N \rfloor$ labeled examples from S_{cln} and substitutes it with an adversarial set of $\lfloor \eta N \rfloor$ labeled examples S_{adv}.

In this model, given $\epsilon \in (0, 1)$ and S_{inp} for large enough N, the goal of the learner is to output, with probability 0.95, a hypothesis $h : \{\pm 1\}^d \to \{\pm 1\}$ such that $\Pr_{\boldsymbol{x} \sim \{\pm 1\}^d}[h(\boldsymbol{x}) \neq f^*(\boldsymbol{x})] \leq \mathrm{poly}(\eta) + \epsilon$.

Once more, we obtain a polynomial-time algorithm for the problem based on a similar approach as the one for the label noise case, using the following additional ingredients. See Section 8 for more details.

Equivalence Between Adaptive and Oblivious Adversaries. It follows from results in [BV25] that learning some class $\mathcal{C}$ with bounded contamination over the hypercube is computationally equivalent to learning using samples from some distribution D over $\{\pm 1\}^d \times \{\pm 1\}$ whose total variation distance from the clean distribution D_{clean} is η. The marginal of D_{clean} on $\{\pm 1\}^d$ is uniform and the clean labels are generated by some $f \in \mathcal{C}$. This is useful for our analysis, since it allows us to consider the input samples independent.

Finding Positions of Influential Variables under Contamination. We provide an analogue of Theorem 3 for the setting of (oblivious) contamination. The crucial observation here is that the way we estimate the positions of influential variables is by estimating the expected value of the quantities yx_i for $i \in [d]$, whose absolute value is bounded by 1 for any $y \in \{\pm 1\}, \boldsymbol{x} = (x_1, \ldots, x_d) \in \{\pm 1\}^d$. Therefore, the estimation error is bounded by the total variation distance between the input distribution and the clean distribution.

Learning the Heavy Coefficients or a Sparse Halfspace under Contamination. Recall that in both the sparse case and in learning the heavy coefficients of the structured case, the required sample size m can be chosen to be vanishingly small compared to the noise rate. This enables safely ignoring the label noise through a total variance argument between the joint distribution of m samples with or without noise. The same argument works even in the presence of contamination and, therefore, the noise can be ignored in this case as well.

Learning the Regular Coefficients under Contamination. To learn the regular coefficients, we once more use an appropriate hinge loss minimization routine. However, we must ensure that the minimum value of the hinge loss remains bounded by a function of the contamination rate η. This requires two properties of the input distribution: (i) concentration in every direction, and (ii) anti-concentration along the direction of the optimal tail coefficient vector. Regarding the anti-concentration property, we use the fact that the input distribution is close to the uniform distribution over the hypercube, and that the optimum tail coefficient vector is regular (since we are in the structured case). To ensure concentration, we perform an outlier removal procedure on the input samples, ensuring that the moments of constant degree are bounded along any direction. Such procedures have been widely used in the literature of learning with contamination [KLS09, ABL17, DKS18b, GSSV24, KSV24]. Here, we use the formulation of [KSV24] as a black box, and obtain an algorithm for learning the regular coefficients that is robust to contamination.

3 Notation and Roadmap

Notation. We use bold letters to denote vectors. For a vector $\boldsymbol{x} \in \mathbb{R}^d$, we denote with x_i the value of the i-th coordinate, and for $H \subseteq [d]$, we denote with $\boldsymbol{x}_H$ the vector in $\mathbb{R}^{|H|}$ corresponding to the restriction of $\boldsymbol{x}$ on the coordinates in H, i.e., $\boldsymbol{x}_H = (x_i)_{i \in H}$. For vectors that are denoted with subscript notation, e.g., $\boldsymbol{x}_{\text{example}}$, we use the notation $(\boldsymbol{x}_{\text{example}})_H$ to denote their restriction.

We use D to denote a labeled distribution over pairs $(\boldsymbol{x}, y) \in \{\pm 1\}^d \times \{\pm 1\}$, and D_X to denote the marginal of D on $\{\pm 1\}^d$. Moreover, we use the notation $D_{y|\boldsymbol{x}}$ to denote the conditional distribution of y given $\boldsymbol{x}$ when $(\boldsymbol{x}, y)$ is drawn according to D. We use the notation $(\boldsymbol{x}, y) \sim D$ to assert that the pair $(\boldsymbol{x}, y)$ is drawn according to the distribution D, and for a set S of points in $\{\pm 1\}^d \times \{\pm 1\}$, we use the notation $(\boldsymbol{x}, y) \sim S$ to assert that $(\boldsymbol{x}, y)$ is drawn from the uniform distribution over the elements in S.

We denote with $\text{sign} : \mathbb{R} \to \{\pm 1\}$ the sign function and with $\mathbb{1}_E \in \{0, 1\}$ the indicator of the event E.

Roadmap. Our main result is achieved by Algorithm 1. In the following sections, we provide a proof for our main result Theorem 24, following the structure of Algorithm 1, as well as the proof outline presented in Figure 1. In particular:

- In Section 4 we provide an important tool (Theorem 5) that reduces the problem of learning an arbitrary halfspace over the hypercube to learning a halfspace which is slightly more noisy, but whose large-magnitude coefficients are in known positions.
- In Section 5 we show how to learn the heavy coefficients of a structured halfspace, i.e., a halfspace such that, except from a small number of heavy coefficients whose locations are known, has coefficients of roughly comparable magnitudes.
- In Section 6 we prove that the regular coefficients of a structured halfspace can be learned robustly and efficiently, as long as we are given accurate estimates for the heavy coefficients.
- Finally, in Section 7 we show how all these ingredients can be combined with the critical index lemma (Theorem 25) to obtain our main result of Theorem 24.

Moreover, in Section 8 we show how to extend our results to the setting of learning under bounded contamination, where a bounded fraction of both datapoints $\boldsymbol{x}$ and their labels y can be adversarially corrupted.

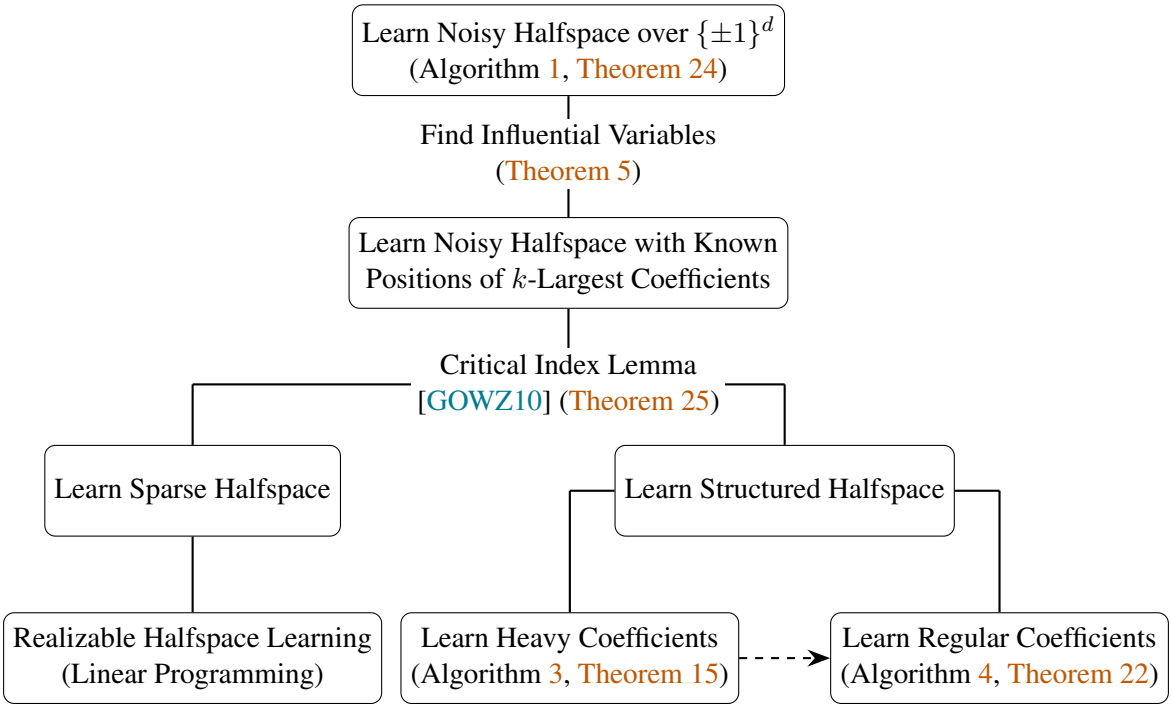

Figure 1: Reduction tree for the proof of our main result (Theorem 24). Each internal node corresponds to a learning problem that reduces to solving all of its children. The dashed arrow indicates a dependency between two leaf nodes, where the output of one serves as input to the other.

Algorithm 1: LEARNHALFSPACE(ε, D)

Input: Parameter $\varepsilon \in (0,1)$, an oracle to independent samples $(\boldsymbol{x}, y) \sim D$
Output: A halfspace $\widehat{h}(\boldsymbol{x}) = \mathrm{sign}(\widehat{\boldsymbol{v}} \cdot \boldsymbol{x} + \widehat{\tau})$ described by $\widehat{\boldsymbol{v}} \in \mathbb{R}^d, \widehat{\tau} \in \mathbb{R}$

/* Initialization */

Let $C \geq 1$ be a sufficiently large universal constant;
Set $k = C^{0.1} \log^{16}(1/\varepsilon)/\varepsilon^8$ and $\eta = \varepsilon^{25}/C$; ▷ η is an upper bound for opt

/* Finding Indices of Influential Variables (see Section 4) */

Let S_1 be a set of $C\sqrt{\log d}/\eta^2$ i.i.d. samples from D;
Compute the quantities $\widehat{\mathrm{I}}_i = \mathbb{E}_{(\boldsymbol{x},y)\sim S_1}[yx_i]$ for $i \in [d]$;
Let $H_k = \{j_1, j_2, \ldots, j_k\}$ be such that $|\widehat{\mathrm{I}}_i| \leq |\widehat{\mathrm{I}}_j|$ for all $i \in [d]$, $j \in H_k$; ▷ k largest values $|\widehat{\mathrm{I}}_i|$

/* Finding Structured Halfspace */

for $\Delta = 0, 1, 2, \ldots, k$ **do**

  /* Part I: Heavy Coefficients (see Section 5) */

  Let $H_\Delta = \{j_1, j_2, \ldots, j_\Delta\}$ and let D_Δ the distribution of $(\boldsymbol{x}_{H_\Delta}, y)$ where $(\boldsymbol{x}, y) \sim D$;
  Set $\varepsilon_{\mathrm{hv}} = C^{0.01} \log^2(k)/\sqrt{k}$ and $u_{\max} = d2^{20d \log d}$;
  Let $(\widehat{\boldsymbol{u}}, \widehat{\tau}_\Delta)$ be the output of FINDHEAVYCOEFFICIENTS$(\varepsilon_{\mathrm{hv}}, \Delta, u_{\max}, D_\Delta)$;
  Let $\widehat{\boldsymbol{v}}_{\mathrm{hv}} \in \mathbb{R}^d$ such that $(\widehat{\boldsymbol{v}}_{\mathrm{hv}})_{H_\Delta} = \widehat{\boldsymbol{u}}$ and $(\widehat{\boldsymbol{v}}_{\mathrm{hv}})_{[d]\setminus H_\Delta} = 0$;

  /* Part II: Regular Coefficients (see Section 6) */

  Set $\varepsilon_{\mathrm{reg}} = \varepsilon/C^{0.01}$;
  Let $\widehat{\boldsymbol{v}}_\Delta$ be the output of FINDREGULARCOEFFICIENTS$(\varepsilon_{\mathrm{reg}}, d, H_\Delta, \widehat{\boldsymbol{v}}_{\mathrm{hv}}, \widehat{\tau}_\Delta, D)$;
  Let $h_\Delta(\boldsymbol{x}) = \mathrm{sign}(\widehat{\boldsymbol{v}}_\Delta \cdot \boldsymbol{x} + \widehat{\tau}_\Delta)$;

/* Finding Sparse Halfspace (see Section 7). */

Let S_2 be a set of $\sqrt{C}\frac{k}{\varepsilon^2}\log^2\frac{k}{\varepsilon}$ i.i.d. samples from D;
Use linear programming to attempt to find a halfspace $h_{\mathrm{sp}}(\boldsymbol{x}) = \mathrm{sign}(\widehat{\boldsymbol{v}}_{\mathrm{sp}} \cdot \boldsymbol{x} + \widehat{\tau})$, where $\widehat{\boldsymbol{v}}_{\mathrm{sp}} \in \mathbb{R}^d$ is supported on the indices in H_k and $\widehat{\tau} \in \mathbb{R}$, such that $(\boldsymbol{x}, h_{\mathrm{sp}}(\boldsymbol{x})) = (\boldsymbol{x}, y)$ for all $(\boldsymbol{x}, y) \in S_2$;
If the linear program is infeasible then set $h_{\mathrm{sp}} \equiv +1$.

/* Selecting the Best Hypothesis. */

Let S_3 be a set of $C\sqrt{\log k}/\varepsilon^2$ i.i.d. samples from D;
Choose $\widehat{h}$ to be the hypothesis in $\{h_0, \ldots, h_k\} \cup \{h_{\mathrm{sp}}\}$ that minimizes the empirical error on S_3;

4 Finding Indices of Influential Variables

In this section, we prove the following, slightly more technical version of Theorem 3, which reduces the problem of learning an arbitrary noisy halfspace over the hypercube to learning a halfspace that is slightly more noisy, but whose large magnitude coefficients are in known positions.

Theorem 5. *Let D be a distribution over pairs $(\boldsymbol{x}, y) \in \{\pm 1\}^d \times \{\pm 1\}$ where $\boldsymbol{x}$ is uniform over $\{\pm 1\}^d$ and $\Pr[y \neq \operatorname{sign}(\boldsymbol{v}^* \cdot \boldsymbol{x} + \tau^*)] \leq \eta$. Let $H_k = \{j_1, \ldots, j_k\}$ be such that:*

$$\left| \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}[yx_i] \right| \leq \left| \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}[yx_j] \right| \text{, for all } i \in [d] \setminus H_k \text{ and } j \in H_k \,,$$

where S is a set of $C\sqrt{\log d}/\eta^2$ i.i.d. samples from D for some sufficiently large universal constant $C \geq 1$. Then, with probability at least 0.99, we have:

$$\Pr_{\boldsymbol{x}\sim\{\pm 1\}^d}\left[\operatorname{sign}(\boldsymbol{v}^* \cdot \boldsymbol{x} + \tau^*) \neq \operatorname{sign}(\tilde{\boldsymbol{v}}^* \cdot \boldsymbol{x} + \tau^*)\right] \leq 4\eta k \,,$$

where $\tilde{\boldsymbol{v}}^*$ is such that the k largest (in absolute value) coefficients correspond to the indices $j_1, \ldots, j_k$. The algorithm has time and sample complexity $\operatorname{poly}(d, 1/\eta)$.

We give a more explicit construction for the vector $\tilde{\boldsymbol{v}}^*$ of Theorem 5. To this end, we will make use of the following definitions.

Definition 6. For any $\boldsymbol{v} \in \mathbb{R}^d$ and a pair of distinct $i, j \in [d]$, let $\boldsymbol{v}_{i\leftrightarrow j}$ be defined as the vector one obtains by swapping the absolute values of the weights v_i and v_j but keeping their signs. Formally:

$$(\boldsymbol{v}_{i\leftrightarrow j})_\ell = \begin{cases} v_\ell & \text{if } \ell \notin \{i, j\} \,, \\ \operatorname{sign}(v_i)\,|v_j| & \text{if } \ell = i, \\ \operatorname{sign}(v_j)\,|v_i| & \text{if } \ell = j. \end{cases}$$

If $i = j$, then $\boldsymbol{v}_{i\leftrightarrow j}$ is defined to be equal to $\boldsymbol{v}$.

The following definition extends the notion of $\boldsymbol{v}_{i\leftrightarrow j}$ to larger permutations on $[d]$:

Definition 7. Let $\pi : [d] \to [d]$ be a permutation, then for a vector $\boldsymbol{v}$ we define a vector $\boldsymbol{v}^\pi$ as follows

$$(\boldsymbol{v}^\pi)_i = \operatorname{sign}(v_i)\left|v_{\pi(i)}\right| .$$

For a linear threshold function $g(\boldsymbol{x}) = \operatorname{sign}(\boldsymbol{v} \cdot \boldsymbol{x} + \tau)$ we define

$$g^\pi(\boldsymbol{x}) = \operatorname{sign}(\boldsymbol{v}^\pi \cdot \boldsymbol{x} + \tau)$$

To state our result, we will need some observations regarding weights of a halfspace and influences of the coordinates:

Definition 8. For a vector $\boldsymbol{v} \in \mathbb{R}^d$, we denote $\operatorname{Inf}_i[\boldsymbol{v}]$ the influence of the function $\operatorname{sign}(\boldsymbol{v} \cdot \boldsymbol{x})$ defined as

$$\operatorname{Inf}_i[\boldsymbol{v}] = \Pr_{\boldsymbol{x}\sim\{\pm 1\}^d}\left[\operatorname{sign}(\boldsymbol{v} \cdot \boldsymbol{x}) \neq \operatorname{sign}\left(\boldsymbol{v} \cdot \boldsymbol{x}^{\otimes i}\right)\right],$$

where $\boldsymbol{x}^{\otimes i}$ denotes the vector $\boldsymbol{x}$ with the i-th coordinate negated. More generally, for a function $g : \{\pm 1\}^d \to \{\pm 1\}$ we define

$$\operatorname{Inf}_i[g] = \Pr_{\boldsymbol{x}\sim\{\pm 1\}^d}\left[g(\boldsymbol{x}) \neq g\left(\boldsymbol{x}^{\otimes i}\right)\right].$$

We provide the following result, as we will show shortly, implies Theorem 5.

Lemma 9. *Let $\eta \in (0,1)$, let $\boldsymbol{v}$ be a vector on $\mathbb{R}^d$, let $i_1, i_2, \cdots i_k \in [d]$ be the indices with the k largest weights for the linear classifier $g(\boldsymbol{x}) = \operatorname{sign}(\boldsymbol{v} \cdot \boldsymbol{x} + \tau)$, i.e.*

$$i_j \leftarrow \operatorname*{arg\,max}_{i \in [d] \setminus \{i_1, i_2, \cdots i_{j-1}\}} [|v_i|],$$

and let $\widehat{i}_1, \widehat{i}_2, \cdots \widehat{i}_k \in [d]$ satisfy

$$\operatorname{Inf}_{\widehat{i}_j}[g] \geq \max_{i \in [d] \setminus \left\{\widehat{i}_1, \widehat{i}_2, \cdots \widehat{i}_{j-1}\right\}} \operatorname{Inf}_i[g] - \eta,$$

then there exists a permutation $\pi : [d] \to [d]$ such that:

1. *π permutes only indices inside $\{i_1, i_2, \cdots i_k\} \cup \{\widehat{i}_1, \widehat{i}_2, \cdots \widehat{i}_k\}$.*
2. *For all $j \in [k]$, it holds that $\pi\left(\widehat{i}_j\right) = i_j$ and therefore*

$$(\boldsymbol{v}^\pi)_{\widehat{i}_j} = \operatorname{sign}\left(v_{\widehat{i}_j}\right) \left|v_{i_j}\right|. \tag{3}$$

3. *We have $\Pr_{\boldsymbol{x} \sim \{\pm 1\}^d}[g(\boldsymbol{x}) \neq g^\pi(\boldsymbol{x})] \leq \eta k$.*

Before proving Theorem 9, we state the following observation regarding influences of a halfspace:

Claim 10. For any $\boldsymbol{v} \in \mathbb{R}^d$, $\tau \in \mathbb{R}$ and $i, j \in [d]$, if it is the case that $|v_i| \geq |v_j|$ then the halfspace $f = \operatorname{sign}(\boldsymbol{v} \cdot \boldsymbol{x} + \tau)$ satisfies

$$\operatorname{Inf}_i[f] \geq \operatorname{Inf}_j[f]$$

Furthermore, it holds that

$$\Pr_{\boldsymbol{x} \sim \{\pm 1\}^d}[\operatorname{sign}(\boldsymbol{v} \cdot \boldsymbol{x} + \tau) \neq \operatorname{sign}(\boldsymbol{v}_{i \leftrightarrow j} \cdot \boldsymbol{x} + \tau)] \leq |\operatorname{Inf}_i[f] - \operatorname{Inf}_j[f]|.$$

Proof. Without loss of generality, assume that $i = 1$, $j = 2$ and $v_1 \geq v_2 \geq 0$.

First, we consider the case when $d = 2$. By inspecting all functions $\{\pm 1\}^2 \to \{\pm 1\}$, we see that $f(x_1, x_2) = \operatorname{sign}(v_1 x_1 + v_2 x_2 + \tau)$ is one of the following functions:

- f_1 that equals to -1 on all of $\{\pm 1\}^2$.
- f_2 that equals to $+1$ on $\{+1, +1\}$ and equals to -1 on all other points of $\{\pm 1\}^2$.
- f_3 that equals to $+1$ on $\{+1, +1\}$ and $\{+1, -1\}$, and equals to -1 on all other points of $\{\pm 1\}^2$.
- f_4 that equals to $+1$ on $\{+1, +1\}$, $\{+1, -1\}$ and $\{-1, +1\}$, and equals to -1 on $\{-1, -1\}$.
- f_5 that equals to -1 on $\{+1, +1\}$, $\{+1, -1\}$ and $\{-1, +1\}$, and equals to -1 on all of $\{\pm 1\}^2$.

Note that $\text{sign}(v_1x_1 + v_2x_2 + \tau)$ with $v_1 \geq v_2 \geq 0$ cannot represent the function g that equals to $+1$ on $\{+1,+1\}$ and $\{-1,+1\}$, and equals to -1 on all other points of $\{\pm 1\}^2$. This is the case because $g(-1,+1) > g(+1,-1)$ whereas $\text{sign}(-v_1 + v_2 + \tau) \leq \text{sign}(v_1 - v_2 + \tau)$ because $v_1 \geq v_2 \geq 0$. Thus, $f(x_1, x_2)$ has to be one of the five functions above.

We see that for each function f among the functions in $\{f_1, f_2, f_3, f_4, f_5\}$ we have

$$\text{Inf}_1[f] = \Pr_{\boldsymbol{x}\sim\{\pm1\}^2}\left[f(\boldsymbol{x}) \neq f(\boldsymbol{x}^{\otimes 1})\right] \geq \Pr_{\boldsymbol{x}\sim\{\pm1\}^2}\left[f(\boldsymbol{x}) \neq f(\boldsymbol{x}^{\otimes 2})\right] = \text{Inf}_2[f], \tag{4}$$

and, again by considering all functions in $\{f_1, f_2, f_3, f_4, f_5\}$ that $f(x) = \text{sign}(v_1x_2 + v_2x_2 + \tau)$ can represent, we check that for each of them it holds that

$$\begin{aligned}\Pr_{\boldsymbol{x}\sim\{\pm1\}^2}\left[\text{sign}\left(v_1x_1 + v_2x_2 + \tau\right) \neq \text{sign}\left(v_2x_1 + v_1x_2 + \tau\right)\right] &= \Pr_{\boldsymbol{x}\sim\{\pm1\}^2}\left[f(x_1, x_2) \neq f(x_2, x_1)\right] \\ &\leq \text{Inf}_1\left[f\right] - \text{Inf}_2\left[f\right]. \end{aligned} \tag{5}$$

finishing the proof for $d = 2$.

Now we consider the case of general d (but still, without loss of generality, assume $i = 1$, $j = 2$ and $v_1 \geq v_2 \geq 0$). Let $f^{x_3,\cdots x_d}$ be the restriction of f with specific settings of $x_3, \cdots x_d$ (while keeping x_1 and x_2 free). We have

$$\begin{aligned}\text{Inf}_1[f] &= \mathop{\mathbb{E}}_{x_3,\cdots x_d\sim\{\pm1\}^{d-2}}[\text{Inf}_1\left[f^{x_3,\cdots x_d}\right]] \\ \text{Inf}_2[f] &= \mathop{\mathbb{E}}_{x_3,\cdots x_d\sim\{\pm1\}^{d-2}}[\text{Inf}_2\left[f^{x_3,\cdots x_d}\right]]\end{aligned}$$

Applying Equation 4 to each two-dimensional linear threshold function $f^{x_3,\cdots x_d}$ we get $\text{Inf}_1[f] \geq \text{Inf}_2[f]$ as required.

Combining the two equalities above, and using Equation 5 we further have

$$\begin{aligned}\text{Inf}_1[f] - \text{Inf}_2[f] &= \mathop{\mathbb{E}}_{x_3,\cdots x_d\sim\{\pm1\}^{d-2}}[\text{Inf}_1\left[f^{x_3,\cdots x_d}\right] - \text{Inf}_2\left[f^{x_3,\cdots x_d}\right]] \\ &\geq \mathop{\mathbb{E}}_{x_3,\cdots x_d\sim\{\pm1\}^{d-2}}\left[\Pr_{\boldsymbol{x}\sim\{\pm1\}^2}\left[f^{x_3,\cdots x_d}(x_1, x_2) \neq f^{x_3,\cdots x_d}(x_2, x_1)\right]\right] \\ &= \Pr_{\boldsymbol{x}\sim\{\pm1\}^d}\left[\text{sign}\left(\boldsymbol{v}\cdot\boldsymbol{x} + \tau\right) \neq \text{sign}\left(\boldsymbol{v}_{i\leftrightarrow j}\cdot\boldsymbol{x} + \tau\right)\right],\end{aligned}$$

finishing the proof of the claim. □

The claim above allows us to use an alternative characterization of $i_1, i_2, \cdots i_k$ in Lemma 9

$$i_j = \mathop{\arg\max}_{i\in[d]\setminus\{i_1,i_2,\cdots i_{j-1}\}}\left[\text{Inf}_i\left[g\right]\right].$$

To prove Lemma 9, we also will need the following basic observation:

Claim 11. Suppose we are given a sequence of positive numbers $\{b_1, b_2, \cdots, b_d\}$ such that for every $\ell \in [k]$ we have

$$b_\ell \geq \max_{i\in[d]\setminus\{1,2,\cdots,\ell-1\}}\left[b_i\right] - \eta. \tag{6}$$

suppose we obtain a new sequence $\{b'_1, b'_2, \cdots, b'_d\}$ by swapping a pair b_{j_1} and b_{j_2} for which $j_2 > j_1$ and $b_{j_2} \geq b_{j_1}$ (and keeping all other values the same). Then for every ℓ in $\ell \in [k]$ we again have

$$b'_\ell \geq \max_{i\in[d]\setminus\{1,2,\cdots,\ell-1\}}\left[b'_i\right] - \eta. \tag{7}$$

Proof. The claim follows by considering the following cases for $\ell \in [k]$:

- If $\ell > j_2$ or $\ell < j_1$, both sides of Equation 9 are the same as for Equation 6.
- If $\ell = j_1$, we have

$$b'_\ell = b'_{j_1} = b_{j_2} \geq b_{j_1} \geq \max_{i \in [d] \setminus \{1,2,\cdots,\ell-1\}} [b_i] - \eta \geq \max_{i \in [d] \setminus \{1,2,\cdots,\ell\}} \left[b'_i\right] - \eta,$$

  which implies

$$b'_\ell \geq \max_{i \in [d] \setminus \{1,2,\cdots,\ell-1\}} \left[b'_i\right] - \eta.$$

- If $\ell \in (j_1, j_2)$ we have

$$b'_\ell = b_\ell \geq \max_{i \in [d] \setminus \{1,2,\cdots,\ell-1\}} [b_i] - \eta \geq \max_{i \in [d] \setminus \{1,2,\cdots,\ell-1\}} \left[b'_i\right] - \eta.$$

- If $\ell = j_2$, we have

$$b'_{j_2} = b_{j_1} \geq \max_{i \in [d] \setminus \{1,2,\cdots,j_1-1\}} [b_i] - \eta \geq \max_{i \in [d] \setminus \{1,2,\cdots,j_2-1\}} [b_i] - \eta \geq \max_{i \in [d] \setminus \{1,2,\cdots,j_2-1\}} \left[b'_i\right] - \eta.$$

We have concluded the proof of Theorem 11. □

Now, we are ready to to prove Theorem 9.

Proof of Lemma 9. Without loss of generality, assume that $\widehat{i}_j = j$. We will use the procedure 2 to form π.

Algorithm 2: Permutation Procedure

Input: Vector $\boldsymbol{v} \in \mathbb{R}^d$, set of indices $\{i_1, \ldots, i_k\} \subseteq [d]$
Output: Vectors $\boldsymbol{v}^1, \ldots, \boldsymbol{v}^k \in \mathbb{R}^d$

Set $\boldsymbol{v}^0 \leftarrow \boldsymbol{v}$;
Set $i^0_\ell \leftarrow i_\ell$ for all $\ell \in [k]$; ▷ These values keep track of where we mapped indices in $\{i_1, \cdots, i_k\}$
for $j = 1, 2, \ldots, k$ **do**
  Set $a_j \leftarrow i^{j-1}_j$; ▷ This is where the permutation which we built so far maps i_j
  Set $\boldsymbol{v}^j \leftarrow \left(\boldsymbol{v}^{j-1}\right)_{a_j \leftrightarrow j}$;
  Set $i^j_j \leftarrow j$; **If** $j = i^{j-1}_\ell$ for some ℓ **then** set $i^j_\ell \leftarrow a_j$; ▷ Keeping track of the swap
  For all $\ell \in [k] \setminus \{a_j, j\}$, set $i^j_\ell \leftarrow i^{j-1}_\ell$;

Procedure 2 implicitly defines a sequence of permutations $\pi^0, \pi^1, \cdots, \pi^k$ for which $\boldsymbol{v}^j = \boldsymbol{v}^{\pi^j}$. We also have a sequence of linear classifiers $g^0, g^1, \cdots g^k$ for which

$$g^j(\boldsymbol{x}) = \operatorname{sign}(\boldsymbol{v}^j \cdot \boldsymbol{x} + \tau) = g^{\pi^j}(\boldsymbol{x}).$$

From the construction it is immediate that the procedure only permutes indices inside $\{i_1, i_2, \cdots i_k\} \cup \{1, 2, \cdots, k\}$, and does not permute any other indices.

We claim that the following invariants hold throughout the process above:

- Invariant A: for every step j and index $\ell \in [k]$ we have $i_\ell^j = \pi^j(\ell)$.
- Invariant B: At every step j, for $\ell \in \{1, \cdots, j\}$ we have $i_\ell^j = \ell$, it is the case that
$$\left(\boldsymbol{v}^j\right)_\ell = \operatorname{sign}\left(v_\ell\right)\left|v_{i_\ell}\right|,$$
and additionally for $\ell \in \{j+1, \cdots, k\}$ we have $i_\ell^j > \ell$.
- Invariant C: At every step j, for every $\ell \in [k]$ we have
$$\operatorname{Inf}_\ell\left[g^j\right] \geq \max_{i \in [d] \setminus \{1,2,\cdots,\ell-1\}} \left[\operatorname{Inf}_i\left[g^j\right]\right] - \eta. \tag{8}$$

Invariants A and B follow directly by inspecting the procedure and induction on j. Invariant B in particular implies that for all $j \in [k]$, it holds that $\pi^j(j) = i_j$ as required in part (2) of the claim (then Equation 3 follows by Definition 7).

Invariant C also can be shown by induction as follows. First of all, the base case $j = 0$ holds by the premise of the claim. At every step j we obtain $\boldsymbol{v}^j$ as $\left(\boldsymbol{v}^{j-1}\right)_{a_j \leftrightarrow j}$. Invariant A allows us to conclude that $\operatorname{Inf}_a\left[g^{j-1}\right] = \operatorname{Inf}_{i_j}[\boldsymbol{g}]$ and the inductive assumption implies that for every $\ell \in [k]$ we have
$$\operatorname{Inf}_\ell\left[g^{j-1}\right] \geq \max_{i \in [d] \setminus \{1,2,\cdots,\ell-1\}} \left[\operatorname{Inf}_i\left[g^{j-1}\right]\right] - \eta. \tag{9}$$

Since at step j we have $a_j = i_j^{j-1}$, invariants A and B (together with the definition of i_j in the claim statement) imply that
$$\operatorname{Inf}_{a_j}\left[g^{j-1}\right] = \operatorname{Inf}_{i_j}[g] = \max_{i \in [d] \setminus \{1,2,\cdots,j-1\}} \left[\operatorname{Inf}_i\left[g^{j-1}\right]\right], \tag{10}$$

so in particular since Invariant B implies that $a_j \geq j$ we have
$$\operatorname{Inf}_{a_j}\left[g^{j-1}\right] \geq \operatorname{Inf}_j\left[g^{j-1}\right].$$

The three observations that (a) $a_j \geq j$ (b) $\operatorname{Inf}_{a_j}\left[g^{j-1}\right] \geq \operatorname{Inf}_j\left[g^{j-1}\right]$ and (c) $\boldsymbol{v}^j \leftarrow \left(\boldsymbol{v}^{j-1}\right)_{a_j \leftrightarrow j}$ allow us to use Claim 11 to conclude that Equation 8 holds (note that for the case $a_j = j$ this follows trivially because then $\boldsymbol{v}^j = \boldsymbol{v}^{j-1}$). This establishes Invariant C.

Claim 10 implies that for every j in $[k]$ we have
$$\Pr_{\boldsymbol{x} \sim \{\pm 1\}^d}\left[g^{j-1}(\boldsymbol{x}) \neq g^j(\boldsymbol{x})\right] \leq \left|\operatorname{Inf}_j\left[g^{j-1}\right] - \operatorname{Inf}_{a_j}\left[g^{j-1}\right]\right|.$$

By Equation 10: $\operatorname{Inf}_{a_j}\left[\boldsymbol{v}^{j-1}\right] \geq \operatorname{Inf}_j\left[\boldsymbol{v}^{j-1}\right]$. By Invariant C we have $\operatorname{Inf}_j\left[g^{j-1}\right] \geq \operatorname{Inf}_{a_j}\left[g^{j-1}\right] - \eta$, which implies that
$$\Pr_{\boldsymbol{x} \sim \{\pm 1\}^d}\left[g^{j-1}(\boldsymbol{x}) \neq g^j(\boldsymbol{x})\right] \leq \eta.$$

Summing over j, we obtain:
$$\begin{aligned}
\Pr_{\boldsymbol{x} \sim \{\pm 1\}^d}\left[g(\boldsymbol{x}) \neq g^\pi(\boldsymbol{x})\right] &= \Pr_{\boldsymbol{x} \sim \{\pm 1\}^d}\left[g^{\pi^0}(\boldsymbol{x}) \neq g^{\pi^k}(\boldsymbol{x})\right] \\
&\leq \sum_{j=1}^{k} \Pr_{\boldsymbol{x} \sim \{\pm 1\}^d}\left[g^{j-1}(\boldsymbol{x}) \neq g^j(\boldsymbol{x})\right] \\
&\leq k\eta,
\end{aligned}$$

which concludes the proof of Theorem 9. □

We are now ready to prove Theorem 5, by using Theorem 9.

Proof of Theorem 5. By the premise of the theorem, we have that for all i

$$\widehat{\mathrm{I}}_i := \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}[yx_i] = \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}[f^*(\boldsymbol{x})x_i] \pm 2\eta\,.$$

The following claim is a standard result that can be found, for example, in [O'D14]:

Claim 12. For any linear threshold function $g(\boldsymbol{x}) = \mathrm{sign}(\boldsymbol{v}\cdot\boldsymbol{x}+\tau)$ and $i \in [d]$, it is the case that

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\{\pm1\}^d}[x_i \cdot g(\boldsymbol{x})] = \mathrm{sign}\,(v_i)\,\mathrm{Inf}_i\,[g]\,.$$

Therefore, we have the following:

$$\left|\widehat{\mathrm{I}}_i\right| = \mathrm{Inf}_i\,[f^*] \pm 2\eta,$$

which by the definition of $j_1,\ldots,j_k$ tells us that for every $t \in \{0,\cdots,k\}$

$$\mathrm{Inf}_{j_t}\,[f^*] \geq \max_{i\in[d]\setminus\{j_1,j_2,\cdots j_{t-1}\}}[\mathrm{Inf}_i\,[f^*]] - 4\eta.$$

The proof follows by applying Theorem 9 on the choice $(\widehat{i}_1,\ldots,\widehat{i}_k) = (j_1,\ldots,j_k)$. □

5 Learning the Heavy Coefficients: The GLM component

In this section, we will present the algorithm and proof for the task of learning Generalized Linear Models (GLMs) with run-time scaling polylogarithmically in the magnitude of weights. This is an important sub-routine in our final algorithm for learning halfspaces: it is the main tool we use to find the heavy coefficients (see Figure 1). Using the same techniques, we also prove a theorem of independent interest on learning sigmoidal activations with run-time scaling polylogarithmically in the magnitudes of the weights (see Theorem 1).

5.1 Algorithm and Theorem Statement

Recall the definition of a generalized linear model.

Definition 13 (Generalized linear model (GLM))**.** A distribution D_{GLM} on $\mathbb{R}^\Delta \times \{\pm1\}$ is called a generalized linear model if there exists a monotone function σ and a vector $\boldsymbol{w}^* \in \mathbb{R}^\Delta$ such that

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}[y \mid \boldsymbol{x}] = \sigma(\boldsymbol{w}^*\cdot\boldsymbol{x}).$$

We consider the class of ϵ-regular GLM's which are important for our application of learning halfspaces.

Definition 14. We say that a function $\sigma^{\mathrm{reg}} : \mathbb{R} \to [-1,1]$ is ε-regular if for some integer d and unit vector $\boldsymbol{u} \in \mathbb{R}^d$ satisfying $\|\boldsymbol{u}\|_2 = 1$ and $\|\boldsymbol{u}\|_4^2 \leq \varepsilon$ we have

$$\sigma^{\mathrm{reg}}(z) = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\{\pm1\}^d}[\mathrm{sign}\,(z+\boldsymbol{x}\cdot\boldsymbol{u})]$$

Note that d in the definition above can be arbitrarily large. The goal of this section is to show the following theorem.

Theorem 15. *There exists an algorithm* FINDHEAVYCOEFFICIENTS$(\varepsilon, \Delta, w_{\max}, D)$ *that takes as inputs* $\varepsilon \in (0,1)$*, positive integer* Δ*, a value* $w_{\max} \in \mathbb{R}_{\geq 1}$*, and a sample access to a distribution* D *supported on* $\left\{\boldsymbol{x} \in \mathbb{R}^{\Delta} : \|\boldsymbol{x}\|_2 \leq \sqrt{\Delta}\right\} \times \{\pm 1\}$*, satisfying the following:*

- *The algorithm* FINDHEAVYCOEFFICIENTS *takes* $O\left(\frac{\Delta}{\varepsilon^2} \cdot \left(\log \frac{\Delta}{\varepsilon}\right)^3\right)$ *samples from* D *and runs in time* $\operatorname{poly}\left(\frac{\Delta}{\varepsilon} \log w_{\max}\right)$.
- *When* D *is a GLM with an arbitrary* $\mathbb{R}^{\Delta}$*-marginal supported on* $\left\{\boldsymbol{x} \in \mathbb{R}^{\Delta} : \|\boldsymbol{x}\|_2 \leq \sqrt{\Delta}\right\}$*, and an activation* $\sigma^{reg}(\boldsymbol{v^*} \cdot \boldsymbol{x} + \tau^*)$*, where* σ^{reg} *is a* ε*-regular function (Definition 14), and* $\|\boldsymbol{v}^*\|_2 + |\tau^*| \leq w_{\max}$*. Then with probability at least* 0.95 FINDHEAVYCOEFFICIENTS *will outputs* $\boldsymbol{v}_0 \in \mathbb{R}^{\Delta}$ *and* $t_0 \in \mathbb{R}$ *for which*

$$\mathop{\mathbb{E}}_{(x,y)\sim D}\left[\left(\sigma^{reg}(\boldsymbol{v^*} \cdot \boldsymbol{x} + \tau^*) - \sigma^{reg}(\boldsymbol{v}_0 \cdot \boldsymbol{x} + t_0)\right)^2\right] \leq O\left(\varepsilon \cdot \log(\Delta/\varepsilon)\right).$$

Algorithm 3: FINDHEAVYCOEFFICIENTS$(\varepsilon, \Delta, w_{\max}, D)$

Input: Parameters $\varepsilon \in (0,1), w_{\max} > 0$, and an oracle to independent examples $(\boldsymbol{x}, y) \sim D$
Output: A pair $(\widehat{\boldsymbol{v}}, \widehat{\tau})$ where $\widehat{\boldsymbol{v}} \in \mathbb{R}^{\Delta}, \widehat{\tau} \in \mathbb{R}$

`/* Initialization */`

Let $C \geq 1$ be a sufficiently large universal constant and set $\rho = C + C \log \Delta/\varepsilon$;
Let S be a set of $C\frac{\Delta\rho^2}{\varepsilon^2}\log(\Delta\rho/\varepsilon) + C$ i.i.d. examples from D;
Let $\psi(z) = \mathbb{E}_{x\sim\mathcal{N}(0,1)}[\operatorname{sign}(x+z)]$ and let $\psi_{\mathrm{clp}} : \mathbb{R} \to [-1,1]$ be the function that can be computed as follows for any given $z \in \mathbb{R}$:

$$\psi_{\mathrm{clp}}(z) = \begin{cases} \psi(z)\,, \text{ if } |z| \leq \rho - 1 \\ \psi(\rho-1) + (1-\psi(\rho-1))(z-\rho+1)\,, \text{ if } z \in (\rho-1, \rho] \\ \psi(-\rho+1) - (1+\psi(-\rho+1))(z-\rho+1)\,, \text{ if } z \in [-\rho, -\rho+1) \\ \operatorname{sign}(z)\,, \text{ if } |z| > \rho \end{cases}$$

`/* Optimization via Ellipsoid Algorithm */`

Let $\tilde{S} = \{((\boldsymbol{x}, 1), y) : (\boldsymbol{x}, y) \in S\}$;
Compute the solution $\widehat{\boldsymbol{w}} \in \mathbb{R}^{\Delta+1}$ of the following convex program using the ellipsoid algorithm:

$$\mathop{\arg\min}_{\boldsymbol{w}:\|\boldsymbol{w}\|_2 \leq w_{\max}} \sum_{(\tilde{\boldsymbol{x}}, y)\in\tilde{S}} \int_0^{\boldsymbol{w}\cdot\tilde{\boldsymbol{x}}} (\psi_{\mathrm{clp}}(z) - y)\, dz$$

Set $(\widehat{\boldsymbol{v}}, \widehat{\tau}) = \widehat{\boldsymbol{w}}$;

We highlight a few distinctions between our result and the standard realizable GLM learning results from literature. The most important feature of our algorithm is that run-time is polynomial in $\log w_{\max}$ as opposed to polynomial in $w_{\max}$. We note that sample complexity does not depend on $w_{\max}$. Another strength of our algorithm is that it does not need to know the function σ in advance: it works universally over the whole family of ε-regular σ. We will exploit this strength in our halfspace learning algorithm. A final distinction is that the activations we consider are not Lipschitz (although we will see that in a strong sense they are well-approximated by a Lipschitz activation). The overall GLM learning algorithm can be found in Algorithm 3.

5.2 Proof

We first state a homogeneous (no bias term) version of the result. Theorem 15 will immediately follow due to distribution free nature of the result.

Theorem 16. *There exists an algorithm $\mathcal{A}_{\mathrm{regGLM}}(\varepsilon, \Delta, w_{\max}, D)$ that takes as inputs $\varepsilon \in (0,1)$, positive integer Δ, a value $w_{\max} \in \mathbb{R}_{\geq 1}$, and sample access to a distribution D over $\{\boldsymbol{x} \in \mathbb{R}^{\Delta} : \|\boldsymbol{x}\|_2 \leq \sqrt{\Delta}\} \times \{\pm 1\}$, satisfying the following:*

- *The algorithm $\mathcal{A}_{\mathrm{regGLM}}$ takes $O\left(\frac{\Delta}{\varepsilon^2} \cdot \left(\log \frac{\Delta}{\varepsilon}\right)^3\right)$ samples from D and runs in time $\mathrm{poly}\left(\frac{\Delta}{\varepsilon} \log w_{\max}\right)$.*
- *When D is a GLM with an arbitrary $\mathbb{R}^{\Delta}$-marginal supported on $\left\{\boldsymbol{x} \in \mathbb{R}^{\Delta} : \|\boldsymbol{x}\|_2 \leq \sqrt{\Delta}\right\}$, and an activation $\sigma^{reg}(\boldsymbol{v^*} \cdot \boldsymbol{x})$, where σ^{reg} is ε-regular and $\|\boldsymbol{v}^*\|_2 \leq w_{\max}$. Then with probability at least 0.95 $\mathcal{A}_{\mathrm{regGLM}}$ will output a vector $\boldsymbol{v}_0$ for which*

$$\mathop{\mathbb{E}}_{(x,y)\sim D}\left[\left(\sigma^{reg}(\boldsymbol{v^*} \cdot \boldsymbol{x}) - \sigma^{reg}(\boldsymbol{v}_0 \cdot \boldsymbol{x})\right)^2\right] \leq O\left(\varepsilon \cdot \log(\Delta/\varepsilon)\right).$$

We will now complete the proof of Theorem 15 given the above theorem.

Proof of Theorem 15. Define a new distribution D' supported on $\mathbb{R}^{\Delta+1} \times \{\pm 1\}$ and formed from D as follows: sample $(\boldsymbol{x}, y) \sim D$ and output $((\boldsymbol{x}, 1), y)$. Clearly, D' is a GLM with an ε-regular activation $\sigma(\boldsymbol{z}) \coloneqq \sigma^{\mathrm{reg}}(\boldsymbol{w}^* \cdot \boldsymbol{z})$ where $\boldsymbol{w}^* = (\boldsymbol{v}^*, \tau^*)$. The algorithm FINDHEAVYCOEFFICIENTS$(\varepsilon, \Delta, w_{\max}, D)$ (see Algorithm 3) implements the $\mathcal{A}_{\mathrm{regGLM}}(\varepsilon, \Delta, w_{\max}, D')$ from Theorem 16. Now, applying Theorem 16 to D', we obtain a vector $\widehat{\boldsymbol{w}} = (\widehat{\boldsymbol{v}}, \widehat{t})$ such that:

$$\begin{aligned}\mathop{\mathbb{E}}_{(z,y)\sim D'}[(\sigma(\widehat{\boldsymbol{w}} \cdot \boldsymbol{z}) - \sigma(\boldsymbol{w}^* \cdot \boldsymbol{z}))^2] &= \mathop{\mathbb{E}}_{(x,y)\sim D}\left[\left(\sigma^{\mathrm{reg}}(\boldsymbol{v^*} \cdot \boldsymbol{x} + \tau^*) - \sigma^{\mathrm{reg}}(\widehat{\boldsymbol{v}} \cdot \boldsymbol{x} + \widehat{t})\right)^2\right] \\ &\leq O(\varepsilon \cdot \log(\Delta/\varepsilon))\end{aligned}$$

The bound on runtime and sample complexity follows from the analogous bounds in Theorem 16. □

In order to prove Theorem 16, we prove the following theorem on misspecified GLM learning corresponding to Lipschitz, truncated activations.

Theorem 17. *Let $\alpha \in \mathbb{R}_{\geq 1}$ and suppose σ is a an activation function $\mathbb{R} \to [-1, 1]$ that satisfies the following three conditions:*

- *σ is monotone non-decreasing.*

- *σ is 1-Lipschitz.*
- *$\sigma(z) = \operatorname{sign}(z)$ whenever $|z| \geq \alpha$.*

Then, there exists an algorithm $\mathcal{A}^{\sigma}_{\mathrm{misspGLM}}(\varepsilon, \alpha, w_{\max}, D)$ that

- *Draws $O\left(\frac{\Delta\alpha^2}{\varepsilon^2} \ln \frac{\Delta\alpha}{\varepsilon}\right)$ samples $\{x_i\}$ from a distribution D over $\left\{\boldsymbol{x} \in \mathbb{R}^{\Delta} : \|\boldsymbol{x}\|_2 \leq \sqrt{\Delta}\right\} \times \{\pm 1\}$ and outputs a vector $\boldsymbol{v} \in \mathbb{R}^{\Delta}$.*
- *Runs in time $\operatorname{poly}\left(\frac{\alpha\Delta}{\varepsilon} \log w_{\max}\right)$.*
- *When D is a GLM with an arbitrary $\mathbb{R}^{\Delta}$-marginal supported on $\left\{\boldsymbol{x} \in \mathbb{R}^{\Delta} : \|\boldsymbol{x}\|_2 \leq \sqrt{\Delta}\right\}$, and a monotone activation $\sigma'(\boldsymbol{v}^* \cdot \boldsymbol{x})$, where*
  - *$\|\boldsymbol{v}^*\|_2 \leq w_{\max}$,*
  - *$\sigma'(z) = \operatorname{sign}(z)$ whenever $|z| \geq \alpha$,*
  - *for all $z \in \mathbb{R}$ we have $|\sigma'(z) - \sigma(z)| \leq \varepsilon$*

  then with probability at least 0.99 the algorithm $\mathcal{A}^{\sigma}_{\mathrm{misspGLM}}(D, \varepsilon, \alpha, w_{\max})$ will output a vector $\boldsymbol{v}_0$ satisfying

$$\mathop{\mathbb{E}}_{\boldsymbol{x} \sim D_X}\left[\left(\sigma'(\boldsymbol{x} \cdot \boldsymbol{v}^*) - \sigma'(\boldsymbol{x} \cdot \boldsymbol{v}_0)\right)^2\right] \leq 24\alpha\varepsilon \tag{11}$$

  where D_X is the $\mathbb{R}^{\Delta}$-marginal of D. (Note that no assumption is made on D_X.)

The proof of Theorem 17 follows the lines of Varun Kanade's notes [Kan18] on GLM learning through minimizing a special kind of surrogate loss, typically called the matching loss. Our approach deviates from this paradigm in that (1) we use the ellipsoid method for the corresponding convex minimization problem instead of iterative descent to obtain a logarithmic runtime dependence on the parameter $w_{\max}$, and (2) our analysis crucially uses a truncated version of the surrogate loss in order to provide a sample complexity bound that does not depend on $w_{\max}$.

Proof. We define the surrogate losses ℓ^{σ} and $\ell^{\sigma'}$ as follows:

$$\ell^{\sigma}(\boldsymbol{v}; \boldsymbol{x}, y) = \int_0^{\boldsymbol{v}\cdot\boldsymbol{x}} (\sigma(z) - y)\, dz.$$

$$\ell^{\sigma'}(\boldsymbol{v}; \boldsymbol{x}, y) = \int_0^{\boldsymbol{v}\cdot\boldsymbol{x}} (\sigma(z) - y)\, dz.$$

The surrogate loss is convex because σ is monotone. The algorithm $\mathcal{A}^{\sigma}_{\mathrm{misspGLM}}(\varepsilon, \alpha, w_{\max}, D)$ does the following:

- Draw a collection S of $C\frac{\Delta\alpha^2}{\varepsilon^2} \ln \frac{\Delta\alpha}{\varepsilon} + C$ samples from D, where C is a sufficiently large absolute constant.
- Using the Ellipsoid algorithm, find

$$\boldsymbol{v}_0 \leftarrow \mathop{\arg\min}_{\boldsymbol{v}:\ \|\boldsymbol{v}\|_2 \leq w_{\max}} \mathop{\mathbb{E}}_{(\boldsymbol{x}, y) \in S} \left[\ell^{\sigma}(\boldsymbol{v}; \boldsymbol{x}, y)\right]$$

- Output $\boldsymbol{v}_0$.

The run-time is dominated by the second step, which runs in time $\mathrm{poly}\left(\frac{\alpha\Delta}{\varepsilon}\log w_{\max}\right)$. To analyze the algorithm, we define the following auxilliary functions:

$$\mathrm{trunc}(z) := \min(\alpha, \max(-\alpha, z))$$

$$\ell^{\sigma}_{\mathrm{trunc}}(\boldsymbol{v}; \boldsymbol{x}, y) = \int_0^{\mathrm{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - y)\, dz.$$

Since $\sigma(z) = \mathrm{sign}(z)$ and $\sigma'(z) = \mathrm{sign}(z)$ whenever $|z| \geq \alpha$, for every $z \in \mathbb{R}$ we have

$$\sigma(z) = \sigma(\mathrm{trunc}(z)) = \sigma'(z). \tag{12}$$

We now make the following observation:

Claim 18. For every $(\boldsymbol{x}, y)$ in the support of D we have

$$\ell^{\sigma}(\boldsymbol{v}^*; \boldsymbol{x}, y) = \ell^{\sigma}_{\mathrm{trunc}}(\boldsymbol{v}^*; \boldsymbol{x}, y) \in [-2\alpha, 2\alpha]$$

Proof. The proof is deferred to the appendix where it is identical to the proof of Claim 36. □

Now, since $|\ell^{\sigma}(\boldsymbol{v}^*; \boldsymbol{x}, y)| \leq 2\alpha$, we can use the Hoeffding bound to conclude that with probability 0.99 over the choice of our sample set S we have

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}[\ell^{\sigma}(\boldsymbol{v}^*; \boldsymbol{x}, y)] = \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}[\ell^{\sigma}(\boldsymbol{v}^*; \boldsymbol{x}, y)] \pm O\left(\frac{\alpha}{\sqrt{|S|}}\right),$$

Since v_0 minimizes the empirical surrogate loss ℓ^{σ} among all vectors whose norm is at most $w_{\max}$, and $\|\boldsymbol{v}^*\|_2 \leq w_{\max}$ we also have

$$\begin{aligned}
\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}[\ell^{\sigma}(\boldsymbol{v}_0; \boldsymbol{x}, y)] &\leq \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}[\ell^{\sigma}(\boldsymbol{v}^*; \boldsymbol{x}, y)] \\
&= \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}[\ell^{\sigma}(\boldsymbol{v}^*; \boldsymbol{x}, y)] \pm O\left(\frac{\alpha}{\sqrt{|S|}}\right) \\
&= \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}[\ell^{\sigma}(\boldsymbol{v}^*; \boldsymbol{x}, y)] \pm \frac{\varepsilon}{4},
\end{aligned} \tag{13}$$

where in the last step we substituted $|S| = C\frac{\Delta\alpha^2}{\varepsilon^2}\ln\frac{\Delta\alpha}{\varepsilon} + C$ and took C to be a sufficiently large absolute constant. We now show some useful properties of the truncated surrogate loss.

Claim 19. For every $\boldsymbol{v}, \boldsymbol{x}, y$, the following hold:

1. $|\ell^{\sigma}_{\mathrm{trunc}}(\boldsymbol{v}; \boldsymbol{x}, y)| \leq \int_0^{\mathrm{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} |\sigma(z) - y|\, dz \leq 2\alpha$
2. $\ell^{\sigma}_{\mathrm{trunc}}(\boldsymbol{v}; \boldsymbol{x}, y) \leq \ell^{\sigma}(\boldsymbol{v}; \boldsymbol{x}, y)$.

Proof. The first property follows immediately from definition of trunc and the fact that σ and y are bounded by 1. We show the second property in cases:

- If $\boldsymbol{v} \cdot \boldsymbol{x} \in [-\alpha, \alpha]$, we have $\text{trunc}(\boldsymbol{v} \cdot \boldsymbol{x}) = \boldsymbol{v} \cdot \boldsymbol{x}$ and we have $\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v}; \boldsymbol{x}, y) = \ell^{\sigma}(\boldsymbol{v}; \boldsymbol{x}, y)$.
- If $\boldsymbol{v} \cdot \boldsymbol{x} > \alpha$ we have

$$\ell^{\sigma}(\boldsymbol{v}; \boldsymbol{x}, y) = \int_{0}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - y)\, dz + \int_{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}^{\boldsymbol{v}\cdot\boldsymbol{x}} (1 - y)\, dz \geq \int_{0}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - y)\, dz$$

- If $\boldsymbol{v} \cdot \boldsymbol{x} < -\alpha$ we have

$$\ell^{\sigma}(\boldsymbol{v}; \boldsymbol{x}, y) = \int_{0}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - y)\, dz + \int_{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}^{\boldsymbol{v}\cdot\boldsymbol{x}} (-1 - y)\, dz \geq \int_{0}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - y)\, dz$$

□

We also need the following claim.

Claim 20. For a sufficiently large value of the absolute constant C, with probability at least 0.999 over the choice of the set S, we have the following uniform convergence bound for $\ell^{\sigma}_{\text{trunc}}$:

$$\max_{\boldsymbol{v}' \in \mathbb{R}^d} \left| \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D} \left[\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v}'; \boldsymbol{x}, y)\right] - \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S} \left[\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v}'; \boldsymbol{x}, y)\right] \right| \leq \frac{\varepsilon}{4} \tag{14}$$

Proof. The proof is exactly the same as in the case of Equation 54 and is deferred to the appendix. □

For the GLM D, define $D_{y|\boldsymbol{x}}$ to be the distribution of the label y conditioned on a specific point $\boldsymbol{x} \in \mathbb{R}^d$. We now compare the value of the truncated surrogate loss for a candidate vector $\boldsymbol{v}$ and the optimal vector $\boldsymbol{v}^*$ as follows (this argument is similar to the argument in [Kan18]):

$$\begin{aligned}
&\mathop{\mathbb{E}}_{y\sim D_{y|\boldsymbol{x}}} [\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v}; \boldsymbol{x}, y)] - \mathop{\mathbb{E}}_{y\sim D_{y|\boldsymbol{x}}} [\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v}^*; \boldsymbol{x}, y)] = \mathop{\mathbb{E}}_{y\sim D_{y|\boldsymbol{x}}} \left[\int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - y)\, dz \right] \\
&\quad = \int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - \sigma'(\boldsymbol{v}^* \cdot \boldsymbol{x}))\, dz = \int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - \sigma'(\text{trunc}(\boldsymbol{v}^* \cdot \boldsymbol{x})))\, dz \\
&\quad \geq \int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - \sigma(\text{trunc}(\boldsymbol{v}^* \cdot \boldsymbol{x})))\, dz - \int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} \left|\sigma(\text{trunc}(\boldsymbol{v}^* \cdot \boldsymbol{x})) - \sigma'(\text{trunc}(\boldsymbol{v}^* \cdot \boldsymbol{x}))\right|\, dz \\
&\quad \geq \int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - \sigma(\text{trunc}(\boldsymbol{v}^* \cdot \boldsymbol{x})))\, dz - 2\alpha\varepsilon \\
&\quad \geq \frac{1}{2} \left(\sigma(\text{trunc}(\boldsymbol{v} \cdot \boldsymbol{x})) - \sigma(\text{trunc}(\boldsymbol{v}^* \cdot \boldsymbol{x}))\right)^2 - 2\alpha\varepsilon = \frac{1}{2} \left(\sigma(\boldsymbol{v} \cdot \boldsymbol{x}) - \sigma(\boldsymbol{v}^* \cdot \boldsymbol{x})\right)^2 - 2\alpha\varepsilon.
\end{aligned} \tag{15}$$

The first equality is due to linearlity of expectation and the definition of $\ell^{\sigma}_{\text{trunc}}$. The second equality is because $D_{y|\boldsymbol{x}}$ is the conditional probability distribution of a GLM with vector $\boldsymbol{v}^*$ and activation σ'. The third equality is due to Equation 12. The fourth inequality is due to the triangle inequality. The fifth inequality uses the premise that $|\sigma(z') - \sigma'(z')| \leq \varepsilon$ for all $z' \in \mathbb{R}$. The inequality $\int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - \sigma(\text{trunc}(\boldsymbol{v}^* \cdot \boldsymbol{x})))\, dz \geq \frac{1}{2} \left(\sigma(\text{trunc}(\boldsymbol{v} \cdot \boldsymbol{x})) - \sigma(\text{trunc}(\boldsymbol{v}^* \cdot \boldsymbol{x}))\right)^2$ is due to σ being monotonically increasing and 1-Lipschitz. The last equality is due to Equation 12.

We therefore have for any vector $\boldsymbol{v} \in \mathbb{R}^d$

$$\begin{aligned}\frac{1}{2}\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\left(\sigma(\boldsymbol{v}\cdot\boldsymbol{x})-\sigma(\boldsymbol{v}^*\cdot\boldsymbol{x})\right)^2\right]-2\alpha\varepsilon &\le \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\ell^\sigma_{\text{trunc}}(\boldsymbol{v};\boldsymbol{x},y)-\ell^\sigma_{\text{trunc}}(\boldsymbol{v}^*;\boldsymbol{x},y)\right]\\ &= \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\ell^\sigma_{\text{trunc}}(\boldsymbol{v};\boldsymbol{x},y)-\ell^\sigma(\boldsymbol{v}^*;\boldsymbol{x},y)\right]\\ &\le \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}\left[\ell^\sigma_{\text{trunc}}(\boldsymbol{v};\boldsymbol{x},y)\right]-\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\ell^\sigma(\boldsymbol{v}^*;\boldsymbol{x},y)\right]+\varepsilon/4\\ &\le \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}\left[\ell^\sigma(\boldsymbol{v};\boldsymbol{x},y)\right]-\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\ell^\sigma(\boldsymbol{v}^*;\boldsymbol{x},y)\right]+\varepsilon/4.\end{aligned}$$

The first step uses Equation 15. The second uses Claim 18. The third uses Equation 14 and the fourth step uses Claim 19 (2). Finally, combining with Equation 13, we have

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim D_X}\left[\left(\sigma(\boldsymbol{x}\cdot\boldsymbol{v}^*)-\sigma(\boldsymbol{x}\cdot\boldsymbol{v}_0)\right)^2\right]\le 8\alpha\varepsilon \tag{16}$$

Recalling that for all $z\in\mathbb{R}$ we have $|\sigma'(z)-\sigma(z)|\le\varepsilon$, which implies that for every z_1 and z_2 in $\mathbb{R}$ we have

$$\begin{aligned}&\left|(\sigma'(z_1)-\sigma'(z_2))^2-(\sigma(z_1)-\sigma(z_2))^2\right|\\ &\qquad\le 2\left|\sigma'(z_1)-\sigma(z_1)-(\sigma'(z_2)-\sigma(z_2))\right|\cdot\left|\sigma'(z_1)-\sigma'(z_2)+(\sigma(z_1)-\sigma(z_2))\right|\le 16\varepsilon.\end{aligned}$$

Setting $z_1=\boldsymbol{x}\cdot\boldsymbol{v}^*$, $z_2=\boldsymbol{x}\cdot\boldsymbol{v}_0$, and combining with Equation 16 yields the desired Equation 11. □

Putting it all together: the case of ε-regular activation σ^{reg}. We are now ready to prove Theorem 16. In Theorem 17, we gave a polynomial time learning algorithm for GLMs which were misspecified, truncated and had weights with magnitued potentially exponential in dimension. Recall that the goal of this section was to learn ϵ-regular GLMs, which naturally arose in the context of our application to halfspace learning. We now argue that learning ϵ-regular GLMs can be reduced to learning mis-specified GLMs with activation $\psi(z)\coloneqq\mathbb{E}_{x\sim\mathcal{N}(0,1)}[\text{sign}(z+x)]$.

Proof. Recall that in the setting of Theorem 16, the algorithm is given inputs $\varepsilon\in(0,1)$, positive integer Δ, a value $w_{\max}\in\mathbb{R}_{\ge 1}$, and a sample access to a distribution D supported on $\{\boldsymbol{x}\in\mathbb{R}^\Delta:\|\boldsymbol{x}\|_2\le\sqrt{\Delta}\}\times\{\pm1\}$. The distribution D is a GLM with an arbitrary $\mathbb{R}^\Delta$-marginal supported on $\{\boldsymbol{x}\in\mathbb{R}^\Delta:\|\boldsymbol{x}\|_2\le\sqrt{\Delta}\}$, and an activation $\sigma^{\text{reg}}(\boldsymbol{v}^*\cdot\boldsymbol{x})$, where σ^{reg} is ε-regular and $\|\boldsymbol{v}^*\|_2\le w_{\max}$. With probability at least 0.95 $\mathcal{A}_{\text{regGLM}}$ the algorithm should output a vector $\boldsymbol{v}_0$ for which

$$\mathop{\mathbb{E}}_{(x,y)\sim D}\left[\left(\sigma^{\text{reg}}(\boldsymbol{v}^*\cdot\boldsymbol{x})-\sigma^{\text{reg}}(\boldsymbol{v}_0\cdot\boldsymbol{x})\right)^2\right]\le O\left(\varepsilon\cdot\log(\Delta/\varepsilon)\right). \tag{17}$$

We now state the algorithm $\mathcal{A}_{\text{regGLM}}$ we use for Theorem 16. The algorithm $\mathcal{A}_{\text{regGLM}}(\varepsilon,\Delta,w_{\max},D)$ will use the algorithm $\mathcal{A}_{\text{misspGLM}}$ from Theorem 17, specifically running $\mathcal{A}^{\psi_{\text{clp}}}_{\text{misspGLM}}(4\varepsilon,\rho,w_{\max},D)$ with the choice of activation $\sigma=\psi_{\text{clp}}$ and parameter ρ to be defined below. Afterwards, the algorithm $\mathcal{A}_{\text{regGLM}}$ outputs the same vector $\widehat{v}$ that it received from $\mathcal{A}_{\text{misspGLM}}$.

Below, we describe the function $\psi_{\text{clp}}:\mathbb{R}\to[-1,1]$ used in our algorithm and prove Theorem 16.

Recall that, since σ^{reg} is ε-regular (Definition 14), for some integer d and unit vector $\boldsymbol{u}\in\mathbb{R}^d$ satisfying $\|\boldsymbol{u}\|_2=1$ and $\|\boldsymbol{u}\|_4^2\le\varepsilon$ we have

$$\sigma^{\text{reg}}(z)=\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\{\pm1\}^d}\left[\text{sign}\left(z+\boldsymbol{x}\cdot\boldsymbol{u}\right)\right] \tag{18}$$

In addition to activation σ^{reg}, we define the following function

$$\psi(z) = \mathop{\mathbb{E}}_{x\sim\mathcal{N}(0,1)}\left[\text{sign}\left(z+x\right)\right].$$

Since σ^{reg} is an ε-regular activation, the Berry-Esseen theorem (Fact 34) implies that

$$\max_{z\in\mathbb{R}} |\psi(z) - \sigma^{\text{reg}}(z)| \le 2\varepsilon \tag{19}$$

Now, we are ready to define the function ψ_{clp} that is used in our algorithm. Let $\rho = C(\log\frac{\Delta}{\varepsilon}+1)$ be a real-valued parameter, where C is a sufficiently large absolute constant. Define the following "clipped" versions of σ^{reg} and ψ:

$$\sigma^{\text{reg}}_{\text{clipped}}(z) = \begin{cases} \sigma^{\text{reg}}(z) & \text{if } |z| \le \rho \\ \text{sign}(z) & \text{if } |z| > \rho \end{cases} \tag{20}$$

$$\psi_{\text{clp}}(z) = \begin{cases} \psi(z) & \text{if } |z| \le \rho-1 \\ \psi(\rho-1) + (1-\psi(\rho-1))\cdot(z-(\rho-1)) & \text{if } z\in(\rho-1,\rho] \\ \psi(-(\rho-1)) - (1+\psi(-(\rho-1)))\cdot(-z-(\rho-1)) & \text{if } z\in[-\rho,-\rho+1) \\ \text{sign}(z) & \text{if } |z| > \rho \end{cases} \tag{21}$$

Additionally, recall that

The function $\sigma^{\text{reg}}_{\text{clipped}}(z)$ equals to $\sigma^{\text{reg}}(z)$ in $[-(\rho-1),(\rho-1)]$ and equals to $\text{sign}(z)$ outside of this interval. The function $\psi_{\text{clp}}(z)$ likewise equals to $\psi(z)$ in $[-(\rho-1),(\rho-1)]$ and equals $\text{sign}(z)$ in $\mathbb{R}\setminus[-\rho,\rho]$. In the intervals $[-\rho,-\rho+1]$ and $[\rho-1,\rho]$, the function $\psi_{\text{clp}}(z)$ interpolates linearly between the values of $\psi(z)$ and $\text{sign}(z)$ on the respective edges of these two intervals.

Also, note that the function $\psi_{\text{clp}}(z)$ is defined without any reference to σ^{reg}, and thus one can run the algorithm $\mathcal{A}^{\psi_{\text{clp}}}_{\text{misspGLM}}(4\varepsilon,\rho,w_{\max},D)$ without knowing σ^{reg} in advance. Below, we will show that for any ε-regular σ^{reg}, with probability at least 0.95 the resulting vector $\boldsymbol{v}_0$ will satisfy Equation 17.

Recall, that the distribution D is a GLM with an arbitrary $\mathbb{R}^\Delta$-marginal supported on $\{\boldsymbol{x}\in\mathbb{R}^\Delta : \|\boldsymbol{x}\|_2 \le \sqrt{\Delta}\}$, and an activation $\sigma^{\text{reg}}(\boldsymbol{v}^*\cdot\boldsymbol{x})$.

From Hoeffding inequality and the standard Gaussian tail bound respectively, we have the following two inequalities

$$\Pr_{\boldsymbol{x}\sim\{\pm1\}^d}\left[|\boldsymbol{x}\cdot\boldsymbol{u}| \ge \rho-1\right] \le 2e^{-(\rho-1)^2/2}, \tag{22}$$

$$\Pr_{x\sim\mathcal{N}(0,1)}\left[|x| \ge \rho-1\right] \le 2e^{-(\rho-1)^2/2}, \tag{23}$$

which allow us to conclude that

$$\max_{z\in\mathbb{R}}\left|\sigma^{\text{reg}}_{\text{clipped}}(z) - \sigma^{\text{reg}}(z)\right| \le 2e^{-(\rho-1)^2/2} \tag{24}$$

$$\max_{z\in\mathbb{R}}|\psi_{\text{clp}}(z)-\psi(z)| \le 2e^{-(\rho-1)^2/2}. \tag{25}$$

We define D_{clipped} to be a GLM with the same $\mathbb{R}^\Delta$-marginal as D but activation $\sigma^{\text{reg}}_{\text{clipped}}$. We are now ready to use Theorem 17 to prove the following claim about D_{clipped}:

Claim 21. Let $\boldsymbol{v}_0$ be obtained by running $\mathcal{A}^{\psi_{\mathrm{clp}}}_{\mathrm{misspGLM}}(4\varepsilon, \rho, w_{\max}, D_{\mathrm{clipped}})$. Then, with probability at least 0.99, the vector $\boldsymbol{v}_0$ satisfies Equation 17.

Proof. We see that the choice of $\sigma = \psi_{\mathrm{clp}}$ satisfies the requirements on σ in Theorem 17. Indeed,

- The function ψ_{clp} is monotone non-decreasing, because so is ψ. (Monotonicity of ψ follows directly from the definition ψ.)
- The function ψ_{clp} is 1-Lipschitz because,
  - For z in $[-(\rho-1), \rho-1]$ we have $\psi_{\mathrm{clp}}(z) = \psi(z)$ and we have
  $$\frac{d}{dz}\psi(z) = \frac{1}{\sqrt{2\pi}} e^{-z^2/2} \leq 1,$$
  and hence ψ_{clp} is 1-Lipschitz in this interval.
  - In the interval $(\rho-1, \rho]$, the function ψ_{clp} is linear, with slope $1 - \psi(\rho-1)$. Since $\rho \geq 1$, it follows from the definition of ψ that $\psi(\rho-1) \geq 0$, and thus ψ_{clp} is 1-Lipschitz in the interval $(\rho-1, \rho]$. The case of interval $[-\rho, \rho-1)$ is fully analogous.
  - Outside the interval $[-\rho, \rho]$ we have $\psi_{\mathrm{clp}}(z) = \mathrm{sign}(z)$, which is trivially 1-Lipschitz.
- By construction, we have $\psi_{\mathrm{clp}}(z) = \mathrm{sign}(z)$ whenever z is not in $[-\rho, \rho]$.

By combining Equation 24, Equation 25 and Equation 19, using the triangle inequality and substituting $\rho = C(\log \frac{\Delta}{\varepsilon} + 1)$ we see that for sufficiently large absolute constant C we have

$$\max_{z\in\mathbb{R}} \left|\psi_{\mathrm{clp}}(z) - \sigma^{\mathrm{reg}}_{\mathrm{clipped}}(z)\right| \leq 2\varepsilon + 4e^{-(\rho-1)^2/2} \leq 4\varepsilon.$$

Overall, we see that all the premises of Theorem 17 are satisfied for $\sigma = \psi_{\mathrm{clp}}$ and $\sigma' = \sigma^{\mathrm{reg}}_{\mathrm{clipped}}$. Therefore, with probability at least 0.99, the vector $\boldsymbol{v}_0$, obtained by running $\mathcal{A}^{\psi_{\mathrm{clp}}}_{\mathrm{misspGLM}}(4\varepsilon, \rho, w_{\max}, D_{\mathrm{clipped}})$, satisfies

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim D_X}\left[\left(\sigma^{\mathrm{reg}}_{\mathrm{clipped}}(\boldsymbol{x}\cdot\boldsymbol{v}^*) - \sigma^{\mathrm{reg}}_{\mathrm{clipped}}(\boldsymbol{x}\cdot\boldsymbol{v}_0)\right)^2\right] \leq 96\rho\varepsilon.$$

Combining the inequality above with Equation 24, using the basic inequality $\left|(a+b)^2 - a^2\right| \leq 2\left|ab\right| + b^2$, and recalling that $|\sigma^{\mathrm{reg}}(z)| \leq 1$ we see that

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim D_X}\left[\left(\sigma^{\mathrm{reg}}(\boldsymbol{x}\cdot\boldsymbol{v}^*) - \sigma^{\mathrm{reg}}(\boldsymbol{x}\cdot\boldsymbol{v}_0)\right)^2\right] \leq 96\rho\varepsilon + 8e^{-(\rho-1)^2/2} + 4e^{-(\rho-1)^2}.$$

Substituting $\rho = C(\log \frac{\Delta}{\varepsilon} + 1)$ we see that for sufficiently large absolute constant C the inequality above implies Equation 17, which finishes the proof. □

Finally, we compare the distributions D and D_{clipped}. First, for every specific z, the TV distance between a pair of $\{\pm 1\}$-valued random variables with expectations $\sigma^{\mathrm{reg}}_{\mathrm{clipped}}(z)$ and $\sigma^{\mathrm{reg}}(z)$ respectively is bounded by $\left|\sigma^{\mathrm{reg}}_{\mathrm{clipped}}(z) - \sigma^{\mathrm{reg}}(z)\right|/2$, which is at most $e^{-(\rho-1)^2/2}$ by Equation 24. Comparing this with the definitions of D and D_{clipped} we see that

$$\mathrm{dist}_{\mathrm{TV}}(D, D_{\mathrm{clipped}}) \leq \frac{1}{2}\max_{z\in\mathbb{R}}\left|\sigma^{\mathrm{reg}}_{\mathrm{clipped}}(z) - \sigma^{\mathrm{reg}}(z)\right| \leq e^{-(\rho-1)^2/2}.$$

Theorem 17 implies that the algorithm $\mathcal{A}^{\psi_{\mathrm{clp}}}_{\mathrm{misspGLM}}$ uses at most $O\left(\frac{\Delta\rho^2}{\varepsilon^2}\ln\frac{\Delta\rho}{\varepsilon}\right)$ samples from the distribution it accesses. This, together with the data-processing inequality and the bound above, lets us conclude that

$$\begin{aligned}&\mathrm{dist}_{\mathrm{TV}}\left(\mathcal{A}^{\psi_{\mathrm{clp}}}_{\mathrm{misspGLM}}(4\varepsilon,\rho,w_{\max},D),\mathcal{A}^{\psi_{\mathrm{clp}}}_{\mathrm{misspGLM}}(4\varepsilon,\rho,w_{\max},D_{\mathrm{clipped}})\right)\\&\quad\le O\left(\frac{\Delta\rho^2}{\varepsilon^2}\ln\frac{\Delta\rho}{\varepsilon}\right)\mathrm{dist}_{\mathrm{TV}}(D,D_{\mathrm{clipped}})\le O\left(\frac{\Delta\rho^2}{\varepsilon^2}\ln\frac{\Delta\rho}{\varepsilon}\right)\cdot e^{-(\rho-1)^2/2}.\end{aligned}$$

Substituting $\rho = C(\log\frac{\Delta}{\varepsilon}+1)$, we see that for sufficiently large absolute constant C the expression above is at most 0.05. Combining this with Claim 21, we see that the vector $\boldsymbol{v}_0$ given by $\mathcal{A}^{\psi_{\mathrm{clp}}}_{\mathrm{misspGLM}}(4\varepsilon,\rho,w_{\max},D)$ with probability at least $0.99-0.05=0.95$ satisfies Equation 17, finishing the proof of correctness for the algorithm $\mathcal{A}_{\mathrm{regGLM}}(\varepsilon,\Delta,w_{\max},D)$ we use for Theorem 16 (recall $\mathcal{A}_{\mathrm{regGLM}}(\varepsilon,\Delta,w_{\max},D) = \mathcal{A}^{\psi_{\mathrm{clp}}}_{\mathrm{misspGLM}}(4\varepsilon,\rho,w_{\max},D)$).

Finally, we check the required sample complexity and run-time bounds. Referring to Theorem 17, and substituting $\rho = C(\log\frac{\Delta}{\varepsilon}+1)$, we see that the sample complexity of the resulting algorithm is $O\left(\frac{\Delta}{\varepsilon^2}\cdot\left(\log\frac{\Delta}{\varepsilon}\right)^3\right)$, as required by Theorem 16. The resulting run-time is $\mathrm{poly}\left(\frac{\Delta}{\varepsilon}\log w_{\max}\right)$, as required by Theorem 16. □

6 Learning the Regular Coefficients

In this section, we give an algorithm for learning the regular tail coefficients once we have a good enough estimate for the head coefficients in the structured case (see Figure 1). We provide the following result.

Theorem 22. *There exists an algorithm* FindRegularCoefficients$(\varepsilon,d,H,\boldsymbol{v}_{\mathrm{head}},\tau,D)$ *(Algorithm 4) that takes as inputs $\varepsilon\in(0,1)$, positive integer d, a set $H\subset[d]$, a vector $\boldsymbol{v}_{\mathrm{head}}\in\mathbb{R}^d$ that is supported on H, and a sample access to a distribution D supported on $\{\pm1\}^d\times\{\pm1\}$ whose $\{\pm1\}^d$-marginal is the uniform distribution. The algorithm runs in time $\mathrm{poly}(d/\varepsilon)$ and outputs, with probability at least 0.995, a vector $\widehat{\boldsymbol{v}}\in\mathbb{R}^d$ for which*

$$\Pr_{(\boldsymbol{x},y)\sim D}\left[\mathrm{sign}(\widehat{\boldsymbol{v}}\cdot\boldsymbol{x}+\tau)\neq y\right]\le O\left(\frac{\mathrm{opt}_{\mathrm{reg}}}{\varepsilon}\log\frac{1}{\varepsilon}+\varepsilon\right), \tag{26}$$

where $\mathrm{opt}_{\mathrm{reg}}=\mathrm{opt}_{\mathrm{reg}}(H,\varepsilon)$ is defined as follows for $\mathcal{V}(H,\varepsilon)=\{\boldsymbol{v}\in\mathbb{R}^d:\|\boldsymbol{v}\|_2=1,\|\boldsymbol{v}\|_4^2\le\varepsilon,\boldsymbol{v}_H=0\}$:

$$\mathrm{opt}_{\mathrm{reg}}:=\min_{\boldsymbol{v}\in\mathcal{V}(H,\varepsilon)}\Pr_{(\boldsymbol{x},y)\sim D}\left[\mathrm{sign}(\boldsymbol{v}_{\mathrm{head}}\cdot\boldsymbol{x}+\boldsymbol{v}\cdot\boldsymbol{x}+\tau)\neq y\right] \tag{27}$$

Proof of Theorem 22. We define the following loss (which can be interpreted as a modified hinge loss):

$$\lambda(z)=\begin{cases}0 & \text{if } z\ge\varepsilon,\\ 1-\frac{z}{\varepsilon} & \text{otherwise.}\end{cases}$$

Observe that $\lambda(z)=\frac{1}{\varepsilon}\mathsf{ReLU}(\varepsilon-z)$. Moreover, we let $\phi:\mathbb{R}^d\to\mathbb{R}$ be the function defined as follows.

$$\phi(\boldsymbol{x})=\boldsymbol{v}_{\mathrm{head}}\cdot\boldsymbol{x}+\tau \tag{28}$$

Algorithm 4: FINDREGULARCOEFFICIENTS$(\varepsilon, d, H, \boldsymbol{v}_{\text{head}}, \tau, D)$

Input: Parameters $\varepsilon \in (0,1), d \in \mathbb{N}, H \subseteq [d], \boldsymbol{v}_{\text{head}} \in \mathbb{R}^d, \tau \in \mathbb{R}$ where $(\boldsymbol{v}_{\text{head}})_{[d]\setminus H} = 0$, and an oracle to independent samples $(\boldsymbol{x}, y) \sim D$.

Output: A vector $\widehat{\boldsymbol{v}} \in \mathbb{R}^d$

/* Initialization */

Let $C \geq 1$ be a sufficiently large universal constant;

Let S be a set of $C(d/\varepsilon)^C$ i.i.d. samples from D;

Let $S' = \{(\boldsymbol{x}, y) \in S : |\boldsymbol{v}_{\text{head}} \cdot \boldsymbol{x} + \tau| \leq \log 1/\varepsilon\}$;

/* Hinge Loss minimization */

Let $\mathcal{V} = \{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 \leq 1, \boldsymbol{v}_H = 0\}$;

Compute the solution $\widehat{\boldsymbol{v}}_{\text{tail}} \in \mathbb{R}^d$ of the following convex program:

$$\arg\min_{\boldsymbol{v} \in \mathcal{V}} \sum_{(\boldsymbol{x},y) \in S'} \mathsf{ReLU}(\varepsilon - y(\boldsymbol{v}_{\text{head}} \cdot \boldsymbol{x} + \boldsymbol{v} \cdot \boldsymbol{x} + \tau))$$

Set $\widehat{\boldsymbol{v}} = \widehat{\boldsymbol{v}}_{\text{tail}} + \boldsymbol{v}_{\text{head}}$;

Claim 23. Let $\boldsymbol{v}^*_{\text{tail}}$ be a minimizer in Equation 27, then with probability at least 0.999, it is the case that

$$\frac{1}{|S|} \sum_{(\boldsymbol{x},y) \in S_{\text{filt}}} [\lambda\left((\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y\right)] \leq O\left(\frac{\text{opt}_{\text{reg}}}{\varepsilon} \log \frac{1}{\varepsilon} + \varepsilon\right). \tag{29}$$

Proof. We see that $\lambda(z) \leq \frac{|z|}{\varepsilon} + 1$, and therefore

$$\begin{aligned}\left(\mathbb{1}_{|\phi(\boldsymbol{x})| \leq \log \frac{1}{\varepsilon}} \cdot \lambda\left((\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y\right)\right)^2 &\leq \mathbb{1}_{|\phi(\boldsymbol{x})| \leq \log \frac{1}{\varepsilon}} O\left(\frac{(\phi(\boldsymbol{x}))^2 + (\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})^2}{\varepsilon^2}\right) \\ &\leq O\left(\frac{\left(\log \frac{1}{\varepsilon}\right)^2 + (\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})^2}{\varepsilon^2}\right).\end{aligned}$$

Taking the expectations of both sides and recalling that $\|\boldsymbol{v}^*_{\text{tail}}\|_2 = 1$, we see that

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},y) \sim D}\left[\left(\mathbb{1}_{|\phi(\boldsymbol{x})| \leq \log \frac{1}{\varepsilon}} \cdot \lambda\left((\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y\right)\right)^2\right] = O\left(\frac{\left(\log \frac{1}{\varepsilon}\right)^2 + 1}{\varepsilon^2}\right),$$

which allows us to use the Chebyshev's inequality to conclude that if C is a sufficiently large absolute constant the $C\left(\frac{d}{\varepsilon}\right)^C$-sized sample set S with probability at least 0.999 satisfies

$$\frac{1}{|S|} \sum_{(\boldsymbol{x},y) \in S_{\text{filt}}} [\lambda\left((\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y\right)] = \mathop{\mathbb{E}}_{(\boldsymbol{x},y) \sim D}\left[\mathbb{1}_{|\phi(\boldsymbol{x})| \leq \log \frac{1}{\varepsilon}} \cdot \lambda\left((\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y\right)\right] \pm \varepsilon.$$

Picking $B \geq 1$ to be a positive parameter to be set later, we can write

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{|\phi(\boldsymbol{x})|\leq \log\frac{1}{\varepsilon}} \cdot \lambda\left((\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y\right)\right] \leq$$
$$\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\leq B}\mathbb{1}_{|\phi(\boldsymbol{x})|\leq \log\frac{1}{\varepsilon}} \cdot \lambda\left((\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y\right)\right]$$
$$+ \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq B}\mathbb{1}_{|\phi(\boldsymbol{x})|\leq \log\frac{1}{\varepsilon}} \cdot \lambda\left((\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y\right)\right] \quad (30)$$

We now bound the two terms above as follows. To bound the first term, we make the following three observations:

1. From the definition of λ, we see that $\lambda(z) \leq \frac{|z|}{\varepsilon} + 1$, and that $\lambda(z) = 0$ if $z \geq \varepsilon$.

2. By Berry-Esseen (Fact 34) and the fact that $\|\boldsymbol{v}^*_{\text{tail}}\|_4^2 \leq \varepsilon$ and $\|\boldsymbol{v}^*_{\text{tail}}\|_2 = 1$, for any fixed $a \in \mathbb{R}$ we have
$$\Pr_{\boldsymbol{x}\sim\{\pm1\}^d}\left[|\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x} + a| \leq \varepsilon\right] \leq O(\varepsilon)$$
Further recalling that $\boldsymbol{v}^*_{\text{tail}}$ and $\boldsymbol{v}_{\text{head}}$ are supported on disjoint indices in $[d]$, we see that $\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x}$ and $\phi(\boldsymbol{x})$ are indipendent random variables when $\boldsymbol{x}$ is drawn uniformly from $\{\pm1\}^d$. This allows us to average over $a = \phi(\boldsymbol{x})$ to conclude that
$$\Pr_{\boldsymbol{x}\sim\{\pm1\}^d}\left[|\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x} + \phi(\boldsymbol{x})| \leq \varepsilon\right] \leq O(\varepsilon)$$

3. We see that $(\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y < 0$ if and only if $\text{sign}(\boldsymbol{v}_{\text{head}} \cdot \boldsymbol{x} + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x} + \tau) \neq y$, since we have defined $\phi(\boldsymbol{x}) = \boldsymbol{v}_{\text{head}} \cdot \boldsymbol{x} + \tau$.

The three observations above, together with the observation that $\lambda(z) = 0$ whenever $z > \varepsilon$, allow us to conclude that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\leq B}\mathbb{1}_{|\phi(\boldsymbol{x})|\leq \log\frac{1}{\varepsilon}} \cdot \lambda\left((\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y\right)\right] \leq \\
&\quad\leq \Pr_{\boldsymbol{x}\sim\{\pm1\}^d}\left[y\left(\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x} + \phi(\boldsymbol{x})\right) \in [0,\varepsilon]\right] + \left(\frac{B}{\varepsilon} + 1\right) \cdot \Pr_{(\boldsymbol{x},y)\sim D}\left[(\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})y < 0\right] \\
&\quad= O(\varepsilon) + \left(\frac{B}{\varepsilon} + 1\right) \cdot \Pr_{(\boldsymbol{x},y)\sim D}\left[\text{sign}(\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x}) \neq y\right] \\
&\quad\leq O(\varepsilon) + \left(\frac{B}{\varepsilon} + 1\right)\text{opt}_{\text{reg}}, \qquad (31)
\end{aligned}$$

where the first inequality follows from the first inequality above, the second inequality from the other two observations, and the last equality is due to $\boldsymbol{v}^*_{\text{tail}}$ being a minimizer in Equation 27.

Using again the inequality $\lambda(z) \leq \frac{|z|}{\varepsilon} + 1$, we bound the second term in Equation 30 as follows:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq B}\mathbb{1}_{|\phi(\boldsymbol{x})|\leq\log\frac{1}{\varepsilon}}\cdot\lambda\left((\phi(\boldsymbol{x})+\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x})y\right)\right] \leq \\
&\quad\leq \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq B}\left(\frac{1}{\varepsilon}\left(\log\frac{1}{\varepsilon}+|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\right)+1\right)\right] \\
&\quad= \left(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon}+1\right)\Pr_{(\boldsymbol{x},y)\sim D}\left[|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq B\right]+\frac{1}{\varepsilon}\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq B}\left|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}\right|\right].
\end{aligned}$$

Combining this with the Hoeffding inequality, we see

$$\Pr_{(\boldsymbol{x},y)\sim D}\left[|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq B\right]\leq 2\exp\left(-2B^2\right),$$

$$\begin{aligned}
\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq B}\left|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}\right|\right] &\leq B\cdot\Pr\left[|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq B\right]+\int_{z=B}^{\infty}\Pr\left[|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq z\right]\,dz \\
&\leq 2B\exp\left(-2B^2\right)+2\int_{z=B}^{\infty}\exp\left(-2z^2\right)\,dz \\
&\leq 2B\exp\left(-2B^2\right)+2\int_{z=B}^{\infty}z\exp\left(-2z^2\right)\,dz \\
&\leq O\left(B\exp\left(-2B^2\right)\right).
\end{aligned}$$

therefore

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq B}\mathbb{1}_{|\phi(\boldsymbol{x})|\leq\log\frac{1}{\varepsilon}}\cdot\lambda\left((\phi(\boldsymbol{x})+\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x})y\right)\right]\leq O\left(\frac{B}{\varepsilon}\log\frac{1}{\varepsilon}\exp\left(-2B^2\right)\right),$$

which combined with Equations 30 and 31 yields

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{|\phi(\boldsymbol{x})|\leq\log\frac{1}{\varepsilon}}\cdot\lambda\left((\phi(\boldsymbol{x})+\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x})y\right)\right]\leq\left(\frac{B}{\varepsilon}+1\right)\text{opt}_{\text{reg}}+O\left(\varepsilon+\frac{B}{\varepsilon}\log\left(\frac{1}{\varepsilon}\right)e^{-2B^2}\right).$$

Substituting $B=\log\frac{1}{\varepsilon}$ yields the desired Equation 29. □

Using Claim 23, and the observation that $\lambda\left((\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})y\right)\geq\mathbb{1}_{\text{sign}(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})\neq y}$, we see that:

$$\begin{aligned}
O\left(\frac{\text{opt}_{\text{reg}}}{\varepsilon}\log\frac{1}{\varepsilon}+\varepsilon\right) &\geq \frac{1}{|S|}\sum_{(\boldsymbol{x},y)\in S_{\text{filt}}}\left[\lambda\left((\phi(\boldsymbol{x})+\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x})y\right)\right] \\
&\geq \frac{1}{|S|}\sum_{(\boldsymbol{x},y)\in S_{\text{filt}}}\left[\lambda\left((\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})y\right)\right] \\
&\geq \frac{1}{|S|}\sum_{(\boldsymbol{x},y)\in S_{\text{filt}}}\mathbb{1}_{\text{sign}(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})\neq y} \\
&= \Pr_{(\boldsymbol{x},y)\in S}\left[\left(|\phi(\boldsymbol{x})|\leq\log\frac{1}{\varepsilon}\right)\wedge\left(\text{sign}\left(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x}\right)\neq y\right)\right] \qquad (32)
\end{aligned}$$

Since $\|\boldsymbol{v}^*_{\text{tail}}\|_2$ and $\|\widehat{\boldsymbol{v}}_{\text{tail}}\|_2$ are bounded by 1, we can conclude (using the Hoeffding inequality for $\boldsymbol{x}\cdot\boldsymbol{v}^*_{\text{tail}}$ and $\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x}$ respectively) that:

$$\begin{aligned}\text{opt}_{\text{reg}} \geq \Pr_{(\boldsymbol{x},y)\sim D}\left[\left(|\phi(\boldsymbol{x})| > \log\frac{1}{\varepsilon}\right) \wedge (\text{sign}(\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}) \neq y)\right] = \\ \Pr_{(\boldsymbol{x},y)\sim D}\left[\left(|\phi(\boldsymbol{x})| > \log\frac{1}{\varepsilon}\right) \wedge (\text{sign}(\phi(\boldsymbol{x})) \neq y)\right] \pm \underbrace{\Pr_{(\boldsymbol{x},y)\sim D}\left[|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}| \geq \log\frac{1}{\varepsilon}\right]}_{\leq O(\varepsilon)},\end{aligned}$$

and by the same token

$$\begin{aligned}\Pr_{(\boldsymbol{x},y)\sim D}\left[\left(|\phi(\boldsymbol{x})| > \log\frac{1}{\varepsilon}\right) \wedge (\text{sign}(\phi(\boldsymbol{x}) + \widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x}) \neq y)\right] = \\ \Pr_{(\boldsymbol{x},y)\sim D}\left[\left(|\phi(\boldsymbol{x})| > \log\frac{1}{\varepsilon}\right) \wedge (\text{sign}(\phi(\boldsymbol{x})) \neq y)\right] \pm \underbrace{\Pr_{(\boldsymbol{x},y)\sim D}\left[|\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x}| \geq \log\frac{1}{\varepsilon}\right]}_{\leq O(\varepsilon)},\end{aligned}$$

which together tell us that

$$\Pr_{(\boldsymbol{x},y)\sim D}\left[\left(|\phi(\boldsymbol{x})| > \log\frac{1}{\varepsilon}\right) \wedge (\text{sign}(\phi(\boldsymbol{x}) + \widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x}) \neq y)\right] \leq \text{opt}_{\text{reg}} + O(\varepsilon).$$

Using the above together with Chebyshev's inequality we see that if C is a sufficiently large absolute constant, then the $C\left(\frac{d}{\varepsilon}\right)^C$-sized sample set S with probability at least 0.999 satisfies

$$\Pr_{(\boldsymbol{x},y)\sim S}\left[\left(|\phi(\boldsymbol{x})| > \log\frac{1}{\varepsilon}\right) \wedge (\text{sign}(\phi(\boldsymbol{x}) + \widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x}) \neq y)\right] \leq \text{opt}_{\text{reg}} + O(\varepsilon),$$

which together with 32 and the definition of ϕ (28) implies that

$$\Pr_{(\boldsymbol{x},y)\sim S}\left[(\text{sign}(\boldsymbol{v}_{\text{head}}\cdot\boldsymbol{x} + \widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x} + \tau) \neq y)\right] \leq O\left(\frac{\text{opt}_{\text{reg}}}{\varepsilon}\log\frac{1}{\varepsilon} + \varepsilon\right). \tag{33}$$

Finally, a standard uniform-convergence VC-dimension argument for halfspaces implies that if C is a sufficiently large absolute constant, then the $C\left(\frac{d}{\varepsilon}\right)^C$-sized sample set S with probability at least 0.999 satisfies the following inequality, uniformly over $\boldsymbol{v}_{\text{tail}} \in \mathbb{R}^d$:

$$\left|\Pr_{(\boldsymbol{x},y)\sim S}\left[(\text{sign}(\boldsymbol{v}_{\text{head}}\cdot\boldsymbol{x} + \boldsymbol{v}_{\text{tail}}\cdot\boldsymbol{x} + \tau) \neq y)\right] - \Pr_{(\boldsymbol{x},y)\sim D}\left[(\text{sign}(\boldsymbol{v}_{\text{head}}\cdot\boldsymbol{x} + \boldsymbol{v}_{\text{tail}}\cdot\boldsymbol{x} + \tau) \neq y)\right]\right| \leq \varepsilon,$$

which together with Equation 33 implies Equation 26 finishing the proof of Theorem 22. □

7 Putting everything together

In this section, we complete the proof of our main result by combining all the ingredients we obtained in the previous sections. In particular, we will prove the following theorem.

Theorem 24. *There is an algorithm that, given access to a distribution D over $\{\pm 1\}^d \times \{\pm 1\}$ whose marginal on $\{\pm 1\}^d$ is the uniform distribution, outputs, with probability at least 0.8, a hypothesis $\widehat{h} : \{\pm 1\}^d \to \{\pm 1\}$ such that*

$$\Pr_{(\boldsymbol{x},y)\sim D}\left[\widehat{h}(\boldsymbol{x}) \neq y\right] \leq O(\mathrm{opt}^{1/25}) + \varepsilon\,, \text{ where } \mathrm{opt} = \min_{\substack{\boldsymbol{v}\in\mathbb{R}^d \\ \tau\in\mathbb{R}}} \Pr_{(\boldsymbol{x},y)\sim D}\left[\mathrm{sign}(\boldsymbol{v}\cdot\boldsymbol{x}+\tau) \neq y\right].$$

The algorithm has time and sample complexity $\mathrm{poly}(d, 1/\varepsilon)$.

If $\mathrm{opt} \leq \varepsilon^{25}/C$ for some sufficiently large constant C, then Algorithm 1, run on input ε, achieves the guarantee of Theorem 24. Otherwise, we may choose the input parameter ε' to be such that $(\varepsilon')^{25}/(2C) \leq \mathrm{opt} \leq (\varepsilon')^{25}/C$ by trying all possible estimates for opt within a grid of appropriate size and choose the one that leads to the hypothesis with the minimum error.

Remark 1. The probability of success can be amplified to $1-\delta$ through $O(\log 1/\delta)$ repetitions and selection of the hypothesis with the best accuracy.

One key ingredient of our proof is the, by now well-known, critical index lemma which states that any halfspace over the boolean hypercube is either close to a sparse halfspace, or most of its coefficients are roughly of the same magnitude. Here, we use the following version of the critical index lemma appearing in [GOWZ10].

Lemma 25 ([GOWZ10])**.** *Let* $\mathrm{sign}\,(\boldsymbol{v}\cdot\boldsymbol{x}+\tau)$ *be a halfspace over* $\{\pm 1\}^d$*. Define for every* $j \in [d]$

$$i_j = \operatorname*{arg\,max}_{i\in[d]\setminus\{i_1,i_2,\cdots i_{j-1}\}} [|v_i|]\,.$$

Let k *be a positive integer. Then, one of the following two cases must hold:*

1. *(Sparse). There is a halfspace* $\mathrm{sign}\,(\boldsymbol{v}_{\mathrm{sp}}\cdot\boldsymbol{x}+\tau)$ *with* $\boldsymbol{v}_{\mathrm{sp}}$ *supported on at most* k *indices* $\{i_1, i_2, \cdots i_k\}$ *for which*
$$\Pr_{\boldsymbol{x}\sim\{\pm 1\}^d}\left[\mathrm{sign}\,(\boldsymbol{v}\cdot\boldsymbol{x}+\tau) \neq \mathrm{sign}\,(\boldsymbol{v}_{\mathrm{sp}}\cdot\boldsymbol{x}+\tau)\right] \leq \frac{1}{k^{100}}.$$
2. *(Structured). There exists some* $\Delta \in \{0, 1, \cdots, k-1\}$ *for which* $\frac{\|\boldsymbol{u}_{\mathrm{tail}}\|_4^2}{\|\boldsymbol{u}_{\mathrm{tail}}\|_2^2} \leq O\left(\frac{\log^2 k}{\sqrt{k}}\right)$*, where* $\boldsymbol{u}_{\mathrm{tail}}$ *denotes the restriction of* $\boldsymbol{v}$ *onto* $[d] \setminus \{i_1, i_2, \cdots i_\Delta\}$.

We are now ready to prove Theorem 24.

Proof of Theorem 24. As we mentioned after the statement of Theorem 24, without loss of generality, we may assume that $\mathrm{opt} \leq \eta$ for $\eta = \varepsilon^{25}/C$.

Recall that D is a distribution over $\{\pm 1\}^d \times \{\pm 1\}$ that has a uniform $\{\pm 1\}^d$-marginal and for some halfspace $f^*(\boldsymbol{x}) = \mathrm{sign}\,(\boldsymbol{v}^*\cdot\boldsymbol{x}+\tau^*)$ satisfies

$$\Pr_{(\boldsymbol{x},y)\sim D}[f^*(\boldsymbol{x}) \neq y] \leq \eta = \varepsilon^{25}/C. \tag{34}$$

Without loss of generality $\boldsymbol{v}^*$ has integer weights whose values are bounded by $2^{10d\log d}$ [Ser06]. The algorithm computes

$$\widehat{\mathrm{I}}_i = \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}[yx_i] \pm \eta$$

and obtains $H = \{j_1, j_2, \cdots j_k\}$ as

$$j_t \leftarrow \operatorname*{arg\,max}_{i \in [d] \setminus \{j_1, j_2, \cdots j_{t-1}\}} \left[\left| \widehat{\mathrm{I}}_i \right| \right]. \tag{35}$$

This allows us to apply Theorem 5 to obtain a halfspace $g^*(\boldsymbol{x}) = \operatorname{sign}(\tilde{\boldsymbol{v}}^* \cdot \boldsymbol{x} + \tau)$ so that the vector $\tilde{\boldsymbol{v}}^*$ satisfies

1. The top k weights of $\tilde{\boldsymbol{v}}^*$ (in absolute value) are located respectively at the indices $j_1, j_2, \cdots j_k$.
2. All weights of $\tilde{\boldsymbol{v}}^*$are integers with weights bounded by $2^{10d \log d}$ (as are the weights of $\boldsymbol{v}^*$).
3. It is the case that
$$\Pr_{\boldsymbol{x} \sim \{\pm 1\}^d} [g^*(\boldsymbol{x}) \neq f^*(\boldsymbol{x})] \leq 4\eta k.$$

The last equation together with Equation 34 implies that

$$\Pr_{(\boldsymbol{x}, y) \sim D} [g^*(\boldsymbol{x}) \neq y] \leq 4(k+1)\eta \tag{36}$$

Now, we can apply Lemma 25 with the halfspace defined by $\tilde{\boldsymbol{v}}^*$. Suppose we are in case (1), then for some vector $\boldsymbol{v}_{\mathrm{sp}}$ supported of $\{\widehat{i}_1, \cdots, \widehat{i}_k\}$ we have

$$\Pr_{\boldsymbol{x} \sim \{\pm 1\}^d} [g^*(\boldsymbol{x}) \neq \operatorname{sign}(\boldsymbol{v}_{\mathrm{sp}} \cdot \boldsymbol{x} + \tau)] \leq \frac{1}{k^{100}},$$

which together with the previous inequality implies that

$$\Pr_{(\boldsymbol{x}, y) \sim D} [\operatorname{sign}(\boldsymbol{v}_{\mathrm{sp}} \cdot \boldsymbol{x} + \tau) \neq y] \leq \frac{1}{k^{100}} + 4(k+1)\eta.$$

A union bound implies that with all examples in S_2 will be consistent with $\operatorname{sign}(\boldsymbol{v}_{\mathrm{sp}} \cdot \boldsymbol{x} + \tau)$ with probability

$$\left(\frac{1}{k^{100}} + 4(k+1)\eta \right) |S_2|.$$

Substituting $k = \frac{C^{0.1} \log^{16} \frac{1}{\varepsilon}}{\varepsilon^8}$ and $\eta = \varepsilon^{25}/C$ and $S_2 = \sqrt{C} \frac{k}{\varepsilon^2} \log^2 \frac{k}{\varepsilon}$, we see that the above is at most 0.001 for a sufficiently large absolute constant C. Thus, the set S_2 will be linearly separable, and the algorithm will succeed in finding a classifier $h_{\mathrm{sp}}(\boldsymbol{x}) = \operatorname{sign}(\widehat{\boldsymbol{v}}_{\mathrm{sp}} \cdot x + \widehat{\tau})$ that perfectly classifies the set S_2. Given the size of S_2, the standard VC bound for linear classifiers implies that (for a sufficiently large absolute constant C) with probability at least 0.999 we have

$$\Pr_{(\boldsymbol{x}, y) \sim D} [\operatorname{sign}(\widehat{\boldsymbol{v}}_{\mathrm{sp}} \cdot \boldsymbol{x} + \widehat{\tau}) \neq y] \leq \varepsilon,$$

concluding our consideration of the sparse case of Lemma 25.

For the rest of the section we will consider the second case of Lemma 25, where $\tilde{\boldsymbol{v}}^* = \boldsymbol{v}^*_{\mathrm{head}} + \boldsymbol{v}^*_{\mathrm{tail}}$ for some vector $\boldsymbol{v}^*_{\mathrm{tail}}$ whose projection outside H_Δ is regular. We will focus on the iteration Δ of our algorithm,

fixing Δ to be the value given to us by Lemma 25. We tacitly will equate $\{\pm 1\}^d$ with $\{\pm 1\}^\Delta \times \{\pm 1\}^{d-\Delta}$ using the bijective map

$$\boldsymbol{x} \leftrightarrow (\boldsymbol{x}_{\text{head}}, \boldsymbol{x}_{\text{tail}}), \text{ where } \boldsymbol{x}_{\text{head}} = \left[x_{\widehat{i}_1}, x_{\widehat{i}_2}, \cdots, x_{\widehat{i}_\Delta}\right]$$

and $\boldsymbol{x}_{\text{tail}}$ is the coordinates of $\boldsymbol{x}$ on the rest of $[d]$ (ordered in some arbitrary but fixed way). In the notation of our main algorithm, $\boldsymbol{x}_{\text{head}} = \boldsymbol{x}_{H_\Delta}$. Moreover, we let $\boldsymbol{u}^*_{\text{head}} = (\tilde{\boldsymbol{v}}^*)_{H_\Delta}$ and $\boldsymbol{u}^*_{\text{tail}} = (\tilde{\boldsymbol{v}}^*)_{[d]\setminus H_\Delta}$. We have:

$$\begin{aligned} g^*(\boldsymbol{x}) &= \operatorname{sign}\left(\tilde{\boldsymbol{v}}^* \cdot \boldsymbol{x} + \tau^*\right) = \operatorname{sign}\left(\boldsymbol{u}^*_{\text{head}} \cdot \boldsymbol{x}_{\text{head}} + \boldsymbol{u}^*_{\text{tail}} \cdot \boldsymbol{x}_{\text{tail}} + \tau^*\right) \\ &= \operatorname{sign}\left(\frac{\boldsymbol{u}^*_{\text{head}} \cdot \boldsymbol{x}_{\text{head}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \frac{\boldsymbol{u}^*_{\text{tail}} \cdot \boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \frac{\tau^*}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}\right), \end{aligned} \tag{37}$$

where $\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2 \neq 0$ because otherwise we would fall under the first case of Lemma 25. Since $\boldsymbol{u}^*_{\text{tail}}$ has integer weights bounded by $2^{10d\log d}$, the vector $\frac{\boldsymbol{u}^*_{\text{head}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}$ will likewise have weights bounded by $2^{10d\log d}$. Since the second case of Lemma 25 holds, we see that

$$\frac{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_4^2}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2^2} \leq O\left(\frac{\log^2 k}{\sqrt{k}}\right) \tag{38}$$

and therefore Fact 34 implies that $\boldsymbol{u}^*_{\text{tail}}$ is such that for all $\tau \in \mathbb{R}$

$$\Pr_{\boldsymbol{x}\sim\{\pm 1\}^d}\left[\frac{\boldsymbol{u}^*_{\text{tail}} \cdot \boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} \geq \tau\right] = \Pr_{z\sim\mathcal{N}(0,1)}[z \geq \tau] \pm O\left(\frac{\log^2 k}{\sqrt{k}}\right), \tag{39}$$

which implies that for any fixed value of $\boldsymbol{x}_{\text{head}}$ we have:

$$\begin{aligned} &\Pr_{\boldsymbol{x}_{\text{tail}}\sim\{\pm 1\}^{d-\Delta}}\left[\operatorname{sign}\left(\frac{\boldsymbol{u}^*_{\text{head}} \cdot \boldsymbol{x}_{\text{head}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \frac{\boldsymbol{u}^*_{\text{tail}} \cdot \boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \frac{\tau^*}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}\right) = 1\right] \\ &\qquad = \Pr_{z\sim\mathcal{N}(0,1)}\left[\frac{\boldsymbol{u}^*_{\text{head}} \cdot \boldsymbol{x}_{\text{head}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \frac{\tau^*}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + z \geq 0\right] \pm O\left(\frac{\log^2 k}{\sqrt{k}}\right) \end{aligned} \tag{40}$$

Define $\mathcal{D}_{\text{GLM}}$ to be the probability distribution over $\mathbb{R}^\Delta \times \{\pm 1\}$ distributed as the tuple of random variables $(\boldsymbol{x}_{\text{head}}, g^*(\boldsymbol{x}))$, where $\boldsymbol{x}$ is uniform on $\{\pm 1\}^d$, and recall that D_Δ is the distribution of pairs of the form $(\boldsymbol{x}_{\text{head}}, y)$, where $\boldsymbol{x}_{\text{head}} = \boldsymbol{x}_{H_\Delta}$ and $(\boldsymbol{x}, y) \sim D$. We will first consider what happens when we run FINDHEAVYCOEFFICIENTS$(\varepsilon_{\text{hv}}, \Delta, u_{\max}, \mathcal{D}_{\text{GLM}})$ and then use it to make conclusions about FINDHEAVYCOEFFICIENTS$(\varepsilon_{\text{hv}}, \Delta, u_{\max}, D_\Delta)$. Recall that, in Algorithm 1, we make the following choices for the input parameters of FINDHEAVYCOEFFICIENTS: $\varepsilon_{\text{hv}} = C^{0.01}\log^2(k)/\sqrt{k}$ and $u_{\max} = d2^{20d\log d}$.

The distribution $\mathcal{D}_{\text{GLM}}$ is a generalized linear model such that:

$$\mathop{\mathbb{E}}_{(\boldsymbol{x}_{\text{head}}, y)\sim\mathcal{D}_{\text{GLM}}}\left[y \;\middle|\; \boldsymbol{x}_{\text{head}}\right] = \sigma\left(\frac{\boldsymbol{u}^*_{\text{head}} \cdot \boldsymbol{x}_{\text{head}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \frac{\tau^*}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}\right),$$

where the activation function $\sigma(\cdot)$ is defined as follows:

$$\begin{aligned}\sigma(t) &= \mathop{\mathbb{E}}_{x_{\text{tail}}\sim\{\pm1\}^{d-\Delta}}\left[\operatorname{sign}\left(\frac{\boldsymbol{u}^*_{\text{tail}}\cdot\boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}+t\right)\right]\\ &= 2\left(\Pr_{x_{\text{tail}}\sim\{\pm1\}^{d-\Delta}}\left[\frac{\boldsymbol{u}^*_{\text{tail}}\cdot\boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}+t\geq 0\right]-1\right).\end{aligned} \tag{41}$$

Recalling the definition of a regular activation function (Definition 14), we conclude that Equation 38 implies that the function σ is $O(\log^2(k)/\sqrt{k})$-regular. Recalling that we use a regularity parameter $\varepsilon_{\text{hv}} = C^{0.01}\log^2(k)/\sqrt{k}$ when running FINDHEAVYCOEFFICIENTS, we see that σ is indeed ε_{hv}-regular when C is a sufficiently large absolute constant. Additionally, as we noted earlier, each coefficient of $\boldsymbol{u}^*_{\text{head}}/\|\boldsymbol{u}^*_{\text{tail}}\|_2$ is bounded by $2^{10d\log d}$ in absolute value. Thus, in this case we can apply Corollary 15[4] to conclude that with probability at least 0.95 the algorithm FINDHEAVYCOEFFICIENTS$(\varepsilon_{\text{hv}},\Delta,u_{\max},\mathcal{D}_{\text{GLM}})$ will output a vector $\widehat{\boldsymbol{u}}$ and a scalar $\widehat{\tau}_\Delta$ for which

$$\mathop{\mathbb{E}}_{(\boldsymbol{x}_{\text{head}},y)\sim\mathcal{D}_{\text{GLM}}}\left[\left(\sigma\left(\widehat{\boldsymbol{u}}\cdot\boldsymbol{x}_{\text{head}}+\widehat{\tau}_\Delta\right)-\sigma\left(\frac{\boldsymbol{u}^*_{\text{head}}\cdot\boldsymbol{x}_{\text{head}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}+\frac{\tau^*}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}\right)\right)^2\right]\leq O\left(\varepsilon_{\text{hv}}\log(\Delta/\varepsilon_{\text{hv}})\right),$$

which in particular implies (via Jensen's inequality)

$$\mathop{\mathbb{E}}_{\tilde{\boldsymbol{x}}\sim\{\pm1\}^{\Delta}}\left[\left|\sigma\left(\widehat{\boldsymbol{u}}\cdot\tilde{\boldsymbol{x}}+\widehat{\tau}_\Delta\right)-\sigma\left(\frac{\boldsymbol{u}^*_{\text{head}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}\cdot\tilde{\boldsymbol{x}}+\frac{\tau^*}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}\right)\right|\right]\leq O\left(\frac{C^{0.005}\log^{1.5}(k/\varepsilon)}{k^{1/4}}\right) \tag{42}$$

Furthermore, Equations 37 and 36 give us a bound of $4(k+1)\eta$ on the statistical distance between $\mathcal{D}_{\text{GLM}}$ and a sample $(\boldsymbol{x}_{\text{head}},y)$ with $(\boldsymbol{x}_{\text{head}},\boldsymbol{x}_{\text{tail}},y)\sim D$ (in our main algorithm we call this distribution D_Δ). Using the bound on the sample complexity m_{hv} of FINDHEAVYCOEFFICIENTS allows us to bound the total variation distance between the output of FINDHEAVYCOEFFICIENTS$(\varepsilon_{\text{hv}},\Delta,u_{\max},D_\Delta)$ and the output of FINDHEAVYCOEFFICIENTS$(\varepsilon_{\text{hv}},\Delta,u_{\max},\mathcal{D}_{\text{GLM}})$ by the following quantity:

$$m_{\text{hv}}\operatorname{dist}_{\text{TV}}(D_\Delta,\mathcal{D}_{\text{GLM}})\leq\tilde{O}(\Delta/\varepsilon^2_{\text{hv}})(k+1)\eta\leq\eta\,\tilde{O}(k^3)$$

Thus, with probability at least $0.95-\eta\,\tilde{O}(k^3)\geq 0.94$, the output $(\widehat{\boldsymbol{u}},\widehat{\tau}_\Delta)$ of FINDHEAVYCOEFFICIENTS, obtained in iteration Δ of our main algorithm, satisfies Equation 42.

Now, we observe that the definition of the activation function σ (i.e. Equation 41) implies that for any pair of fixed values a and b in $\mathbb{R}$ with $a\leq b$ we have

$$\begin{aligned}&\Pr_{\boldsymbol{x}_{\text{tail}}\sim\{\pm1\}^{d-\Delta}}\left[\operatorname{sign}\left(a+\frac{\boldsymbol{u}^*_{\text{tail}}\cdot\boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}\right)\neq\operatorname{sign}\left(b+\frac{\boldsymbol{u}^*_{\text{tail}}\cdot\boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}\right)\right]=\\ &\quad=\left|\Pr_{\boldsymbol{x}_{\text{tail}}\sim\{\pm1\}^{d-\Delta}}\left[-b\leq\frac{\boldsymbol{u}^*_{\text{tail}}\cdot\boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}<-a\right]\right|\\ &\quad=\left|\Pr_{x_{\text{tail}}\sim\{\pm1\}^{d-\Delta}}\left[\frac{\boldsymbol{u}^*_{\text{tail}}\cdot\boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}+a\geq 0\right]-\Pr_{x_{\text{tail}}\sim\{\pm1\}^{d-\Delta}}\left[\frac{\boldsymbol{u}^*_{\text{tail}}\cdot\boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2}+b\geq 0\right]\right|\\ &\quad=\frac{1}{2}\left|\sigma(a)-\sigma(b)\right|.\end{aligned}$$

[4]Without loss of generality, the bias τ^* of the ground truth half-space is bounded by $d2^{10d\log d}$ as the inputs are from $\{\pm1\}^d$.

Analogously, we also see that the same expression holds when $a \geq b$. We can thus apply the identity above together with Equation 42 to obtain

$$\Pr_{\substack{\boldsymbol{x}_{\text{head}} \sim \{\pm 1\}^{\Delta}, \\ \boldsymbol{x}_{\text{tail}} \sim \{\pm 1\}^{d-\Delta}}} \left[\operatorname{sign}\left(\frac{\boldsymbol{u}^*_{\text{head}} \cdot \boldsymbol{x}_{\text{head}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \frac{\boldsymbol{u}^*_{\text{tail}} \cdot \boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \frac{\tau^*}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} \right) \neq \operatorname{sign}\left(\widehat{\boldsymbol{u}} \cdot \boldsymbol{x}_{\text{head}} + \frac{\boldsymbol{u}^*_{\text{tail}} \cdot \boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \widehat{\tau}_{\Delta} \right) \right] =$$

$$\mathop{\mathbb{E}}_{\boldsymbol{x}_{\text{head}} \sim \{\pm 1\}^{\Delta}} \left[\left| \sigma\left(\frac{\boldsymbol{u}^*_{\text{head}} \cdot \boldsymbol{x}_{\text{head}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \frac{\tau^*}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} \right) - \sigma\left(\widehat{\boldsymbol{u}} \cdot \boldsymbol{x}_{\text{head}} + \widehat{\tau}_{\Delta} \right) \right| \right] \leq O\left(\frac{C^{0.005} \log k}{k^{1/4}} \right),$$

which together with Equations 37 and 36 gives us

$$\begin{aligned} \Pr_{\boldsymbol{x}, y \sim D} \left[\operatorname{sign}\left(\widehat{\boldsymbol{u}} \cdot \boldsymbol{x}_{\text{head}} + \frac{\boldsymbol{u}^*_{\text{tail}} \cdot \boldsymbol{x}_{\text{tail}}}{\left\|\boldsymbol{u}^*_{\text{tail}}\right\|_2} + \widehat{\tau}_{\Delta} \right) \neq y \right] &\leq O\left(\frac{C^{0.005} \log^{1.5}(k/\varepsilon)}{k^{1/4}} \right) + (k+1)\eta \\ &\leq \frac{\varepsilon^2}{C^{0.05} \log^{1.5} \frac{1}{\varepsilon}}, \end{aligned} \tag{43}$$

where the last step holds for a sufficiently large value of C when we substitute the values of k, η.

Now, we will analyze the vector $\widehat{\boldsymbol{v}}_{\Delta}$ obtained as the output of the algorithm for finding regular coefficients FINDREGULARCOEFFICIENTS$(\varepsilon_{\text{reg}}, d, H_{\Delta}, \widehat{\boldsymbol{v}}_{\text{hv}}, \widehat{\tau}_{\Delta}, D)$. Recalling Theorem 22 that analyzes FINDREGULARCOEFFICIENTS, we see that it is stated in terms of the following quantity, where we define $\mathcal{V}(H_{\Delta}, \varepsilon_{\text{reg}}) = \{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 = 1, \|\boldsymbol{v}\|_4^2 \leq \varepsilon_{\text{reg}}, \boldsymbol{v}_{H_{\Delta}} = 0\}$:

$$\text{opt}_{\text{reg}} := \min_{\boldsymbol{v} \in \mathcal{V}(H_{\Delta}, \varepsilon_{\text{reg}})} \Pr_{(\boldsymbol{x}, y) \sim D} \left[\operatorname{sign}(\widehat{\boldsymbol{v}}_{\Delta} \cdot \boldsymbol{x} + \boldsymbol{v} \cdot \boldsymbol{x}_{\text{tail}} + \widehat{\tau}_{\Delta}) \neq y \right] \tag{44}$$

Comparing Equation 44 and Equation 43, we observe that, taking $\boldsymbol{v} = \boldsymbol{v}^*_{\text{tail}} / \|\boldsymbol{v}^*_{\text{tail}}\|_2$, using Equation 38 and substituting the value of k, we have

$$\frac{\|\boldsymbol{v}^*_{\text{tail}}\|_4^2}{\|\boldsymbol{v}^*_{\text{tail}}\|_2^2} \leq O\left(\frac{\log^2 k}{\sqrt{k}} \right) = O\left(C^{-0.05} \varepsilon^4 \right) \leq \varepsilon / C^{0.01},$$

where the last inequality holds for a sufficiently large absolute constant C. Therefore, comparing Equation 44 and Equation 43, we see that $\text{opt}_{\text{reg}} \leq \varepsilon^2 / (C^{0.05} \log^{1.5}(1/\varepsilon))$. This allows us to use Theorem 22 to conclude that when we invoke

$$\widehat{\boldsymbol{v}}_{\Delta} \leftarrow \text{FINDREGULARCOEFFICIENTS}\left(\varepsilon_{\text{reg}}, d, H_{\Delta}, \widehat{\boldsymbol{v}}_{\text{hv}}, \widehat{\tau}_{\Delta}, D\right)$$

with probability at least 0.95 we obtain a vector $\widehat{\boldsymbol{v}}_{\Delta}$ satisfying

$$\begin{aligned} \Pr_{(\boldsymbol{x}, y) \sim D} \left[\operatorname{sign}(\widehat{\boldsymbol{v}}_{\Delta} \cdot \boldsymbol{x} + \widehat{\tau}_{\Delta}) \neq y \right] &\leq O\left(C^{0.01} \frac{\text{opt}_{\text{reg}}}{\varepsilon} \log \frac{C^{0.01}}{\varepsilon} + \frac{\varepsilon}{C^{0.01}} \right) \\ &\leq O\left(\frac{\varepsilon}{C^{0.04} \log^{1.5} \frac{1}{\varepsilon}} \log \frac{C^{0.01}}{\varepsilon} + \frac{\varepsilon}{C^{0.01}} \right) \leq \frac{\varepsilon}{10}, \end{aligned}$$

where the last inequality follows by taking C to be a sufficiently large absolute constant.

Finally, we see that with overall probability of at least 0.81, in the iteration Δ of our main algorithm (with Δ given by the second case of Lemma 25) we add a hypothesis $h_{\Delta}(\boldsymbol{x}) = \operatorname{sign}(\widehat{\boldsymbol{v}}_{\Delta} \cdot \boldsymbol{x} + \widehat{\tau}_{\Delta})$ into the

set V whose prediction error is at most $\varepsilon/10$. The standard Hoeffding bound implies that for a sufficiently large value of C, each estimate $\widehat{\mathrm{err}}(h)$ produced by our algorithm is indeed $\varepsilon/4$-accurate (with probability at least 0.995). Since there exists a hypothesis h_Δ in V with error at most $\varepsilon/10$, we see that

$$\min_{h \in V} \widehat{\mathrm{err}}(h) \leq \frac{3\varepsilon}{4}$$

and therefore, the minimizer $\widehat{h}$ that our algorithm obtains satisfies

$$\Pr_{(\boldsymbol{x},y)\sim D}\left[\widehat{h}(\boldsymbol{x}) \neq y\right] \leq \widehat{\mathrm{err}}(\widehat{h}) + \frac{\varepsilon}{4} \leq \varepsilon,$$

finishing the proof. □

8 Learning with Contamination

In this section, we will show that our main result extends to a much more challenging noise model, called bounded contamination (or nasty noise) — see Theorem 4. While we aim to capture the bounded contamination model, it will be more convenient to phrase our proofs in terms of learning under a seemingly weaker noise model, which was recently shown to be equivalent to bounded contamination when the size of the domain is finite [BV25].

Definition 26 (Learning with Oblivious Contamination)**.** We say that an algorithm learns a class $\mathcal{C}$ under oblivious contamination of rate $\eta \in (0,1)$ with respect to the uniform distribution over $\{\pm 1\}^d$ if it satisfies the following specifications. The algorithm is given $\varepsilon \in (0,1)$ and access to a large enough set of i.i.d. examples from some distribution D over $\{\pm 1\}^d \times \{\pm 1\}$ such that $\mathrm{dist}_{\mathrm{TV}}(D, D_{\mathrm{clean}}) \leq \eta$ where D_{clean} is a distribution over samples $(\boldsymbol{x}, y)$ such that $\boldsymbol{x}$ is the uniform over $\{\pm 1\}^d$ and $y = f^*(\boldsymbol{x})$ for some unknown $f^* \in \mathcal{C}$. The output of the algorithm is some hypothesis $h : \{\pm 1\}^d \times \{\pm 1\}$ such that, with probability at least 0.95, we have:

$$\Pr_{(\boldsymbol{x},f^*(\boldsymbol{x}))\sim D_{\mathrm{clean}}}[h(\boldsymbol{x}) \neq f^*(\boldsymbol{x})] \leq \mathrm{poly}(\eta) + \varepsilon$$

In particular, the following result from [BV25] shows that learning under bounded contamination is essentially equivalent to learning under oblivious contamination of the same rate.

Theorem 27 (Special Case of Theorem 2 from [BV25])**.** *If a class $\mathcal{C} \subseteq \{\{\pm 1\}^d \times \{\pm 1\}\}$ can be learned under oblivious contamination of rate η up to error $\gamma = \gamma(\eta)$ with sample complexity m and in time T, then it can be learned under bounded contamination of rate η up to error $\gamma(\eta) + \varepsilon$ with sample complexity $\mathrm{poly}(m, d, 1/\varepsilon)$ in time $\mathrm{poly}(m, d, 1/\varepsilon) + T$.*[5]

We will show the following theorem, which, together with Theorem 27, yields a $\mathrm{poly}(d, 1/\varepsilon)$-time algorithm in the bounded contamination case with error $\mathrm{poly}(\eta) + \varepsilon$.

Theorem 28. *There is an algorithm that learns the class $\mathcal{C} = \{f(\boldsymbol{x}) = \mathrm{sign}(\boldsymbol{v} \cdot \boldsymbol{x} + \tau) : \boldsymbol{v} \in \mathbb{R}^d, \tau \in \mathbb{R}\}$ under oblivious contamination of rate η with respect to the uniform distribution over $\{\pm 1\}^d$ up to error $O(\eta^{1/c}) + \varepsilon$ for some sufficiently large constant $c \geq 1$. The algorithm has time and sample complexity $\mathrm{poly}(d, 1/\varepsilon)$.*

[5]The equivalence preserves the failure probability for any choice $\delta \in (0,1)$. This is important, since amplifying the success probability under oblivious contamination is straightforward, but the same is not true for bounded contamination, where the samples are not independent.

The algorithm is essentially the same as the one used for the label noise case (Algorithm 1), with the only difference being that in place of FINDREGULARCOEFFICIENTS, we use the algorithm with the specifications of the following theorem.

Theorem 29. *There exists an algorithm* FINDREGULARCONTAMINATED$(\varepsilon, d, H, \boldsymbol{v}_{\mathrm{head}}, \tau, D)$ *that satisfies the following specifications. The algorithm takes as inputs* $\varepsilon \in (0,1)$*, positive integer* d*, a set* $H \subset [d]$*, a vector* $\boldsymbol{v}_{\mathrm{head}} \in \mathbb{R}^d$ *that is supported on* H*, a real-valued* τ*, and a sample access to a distribution* D *supported on* $\{\pm 1\}^d \times \{\pm 1\}$ *such that* $\mathrm{dist}_{\mathrm{TV}}(D, D_{\mathrm{clean}}) \le \eta$*, where for* $(\boldsymbol{x}, y) \sim D_{\mathrm{clean}}$ *we have (i)* $\boldsymbol{x}$ *is uniformly random on* $\{\pm 1\}^d$ *(ii)* $y = \mathrm{sign}(\boldsymbol{v}_{\mathrm{head}} \cdot \boldsymbol{x} + \boldsymbol{v}^*_{\mathrm{tail}} \cdot \boldsymbol{x} + \tau)$ *where* $\|\boldsymbol{v}^*_{\mathrm{tail}}\|_2 = 1$*, for all* $i \in H$ *we have* $(\boldsymbol{v}^*_{\mathrm{tail}})_i = 0$ *and* $\|\boldsymbol{v}^*_{\mathrm{tail}}\|_4^2 \le \varepsilon$*. The algorithm runs in time* $\mathrm{poly}(d/\varepsilon)$ *and outputs, with probability at least* 0.995*, a vector* $\widehat{\boldsymbol{v}} \in \mathbb{R}^d$ *for which*

$$\Pr_{(\boldsymbol{x},y)\sim D}[\mathrm{sign}(\widehat{\boldsymbol{v}} \cdot \boldsymbol{x} + \tau) \neq y] \le O\left(\frac{\eta}{\varepsilon^3} + \varepsilon\right), \tag{45}$$

Compared to FINDREGULARCOEFFICIENTS, the algorithm FINDREGULARCONTAMINATED performs an additional outlier removal step to ensure that the uniform distribution over the input samples is concentrated in every direction. In particular, we use the following outlier removal guarantee which originates to classical results from the literature of learning with contamination. We use a direct corollary of a version of the outlier removal guarantee appearing in the recent work of [KSV24].

Lemma 30 (Outlier removal [KLS09, DKS18b, KSV24])**.** *There is an algorithm that satisfies the following specifications. The algorithm is given* $\delta \in (0,1)$*,* $r \in \mathbb{N}$*, and a set* S *of* m *i.i.d. examples from some distribution over* $\{\pm 1\}^d$ *whose total variation distance from the uniform is at most* η*. If* $m \ge C(3d)^{4r}\log(1/\delta)$ *for some sufficiently large universal constant* $C \ge 1$*, then the algorithm outputs a set* $S' \subseteq S$ *such that:*

1. *With probability at least* $1-\delta$*, we have* $|S \setminus S'| \le 4\eta m$*.*
2. *For any* $A > 0$ *and any* $\boldsymbol{v} \in \mathbb{R}^d$ *such that* $\mathbb{E}_{\boldsymbol{x}\sim\{\pm 1\}^d}[(\boldsymbol{v}\cdot\boldsymbol{x})^{2r}] \le A$*, we have:*

$$\sum_{\boldsymbol{x}\in S'} (\boldsymbol{v}\cdot\boldsymbol{x})^{2r} \le 8Am \quad \textit{with probability at least } 1-\delta.$$

While the proof of Theorem 29 closely follows the proof of Theorem 22, we provide its proof here for completeness.

Proof of Theorem 29. We define $\lambda(z) = \mathsf{ReLU}(1 - z/\varepsilon)$. For some absolute constant C, the algorithm FINDREGULARCONTAMINATED does the following

- Draw a $C\left(\frac{d}{\varepsilon}\right)^C$-sized sample set S from D.
- Run the algorithm of Theorem 30 on inputs $(\delta = 1/C, r = 1, S)$ and let S' be its output.
- Form a subset S_{filt} of S' as follows for $\phi_{\max} = C/\sqrt{\varepsilon}$:

$$S_{\mathrm{filt}} \leftarrow \left\{(\boldsymbol{x}, y) \in S' : |\boldsymbol{v}_{\mathrm{head}} \cdot \boldsymbol{x} + \tau| \le \phi_{\max}\right\}$$

- Let $\mathcal{V} = \{\boldsymbol{v} \in \mathbb{R}^d : \|\boldsymbol{v}\|_2 \le 1 \text{ and } \boldsymbol{v}_H = 0\}$, where recall that $\boldsymbol{v}_H$ is a vector in $\mathbb{R}^{|H|}$ corresponding to the restriction of $\boldsymbol{v}$ to the coordinates in the set H.

- Compute the following quantity:

$$\widehat{\boldsymbol{v}}_{\text{tail}} \leftarrow \arg\min_{\boldsymbol{v}\in\mathcal{V}} \sum_{(\boldsymbol{x},y)\in S_{\text{filt}}} \lambda\Big((\boldsymbol{v}_{\text{head}}\cdot\boldsymbol{x} + \boldsymbol{v}\cdot\boldsymbol{x} + \tau)\cdot y\Big),$$

  Note that the minimization task above can be performed in time $\mathrm{poly}(d/\varepsilon)$ because the function λ is convex, and therefore the minimization above is a convex minimization task.

- Output $\widehat{\boldsymbol{v}} := \widehat{\boldsymbol{v}}_{\text{tail}} + \boldsymbol{v}_{\text{head}}$.

For the following, we let $\phi : \mathbb{R}^d \to \mathbb{R}$ be the function defined as follows.

$$\phi(\boldsymbol{x}) = \boldsymbol{v}_{\text{head}}\cdot\boldsymbol{x} + \tau \tag{46}$$

Claim 31. Let $\boldsymbol{v}^*_{\text{tail}}$ be a as specified in the statement of Theorem 29, then with probability at least 0.999, it is the case that

$$Q := \frac{1}{|S|}\sum_{(\boldsymbol{x},y)\in S_{\text{filt}}} \left[\lambda\left((\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x})y\right)\right] \le O\left(\frac{\eta}{\varepsilon^3} + \varepsilon\right). \tag{47}$$

Proof. We have that $\lambda(z) \le \frac{|z|}{\varepsilon} + 1$ for all z and $\lambda(z) = 0$, for all $z \ge \varepsilon$. Moreover, $(\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x})y < 0$ if and only if $\mathrm{sign}(\boldsymbol{v}_{\text{head}}\cdot\boldsymbol{x} + \boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x} + \tau) \neq y$, since we have defined $\phi(\boldsymbol{x}) = \boldsymbol{v}_{\text{head}}\cdot\boldsymbol{x} + \tau$. Therefore:

$$Q \le \frac{1}{|S|}\sum_{(\boldsymbol{x},y)\in S_{\text{filt}}} \left(2\cdot\mathbb{1}_{\phi(\boldsymbol{x})+\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}\in[0,\varepsilon]} + \left(1 + \frac{|\phi(\boldsymbol{x})| + |\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|}{\varepsilon}\right)\mathbb{1}_{y\neq\mathrm{sign}(\phi(\boldsymbol{x})+\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x})}\right)$$

We split the right side of the above inequality in two terms, Q_1 and Q_2. We have:

$$\begin{aligned}
Q_1 &:= \frac{2}{|S|}\sum_{(\boldsymbol{x},y)\in S_{\text{filt}}} \mathbb{1}_{\phi(\boldsymbol{x})+\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}\in[0,\varepsilon]} \\
&\le \frac{2}{|S|}\sum_{(\boldsymbol{x},y)\in S} \mathbb{1}_{\phi(\boldsymbol{x})+\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}\in[0,\varepsilon]} \\
&\le 2\Pr_{(\boldsymbol{x},y)\sim D}\left[\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}\in[0,\varepsilon]\right] + O(\varepsilon) \\
&\le 2\Pr_{\boldsymbol{x}\sim\{\pm1\}^d}\left[\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}\in[0,\varepsilon]\right] + O(\varepsilon+\eta) \\
&\le O(\varepsilon+\eta),
\end{aligned}$$

where the second inequality follows from a union bound, combined with the fact that $|S| \ge C(d/\varepsilon)^C$, the third inequality follows from the total variation distance bound between D and D_{clean}, and the final inequality follows from the application of the Berry-Esseen theorem (Fact 34). Specifically, since $\|\boldsymbol{v}^*_{\text{tail}}\|_4^2 \le \varepsilon$ and $\|\boldsymbol{v}^*_{\text{tail}}\|_2 = 1$, for any fixed $a \in \mathbb{R}$ we have

$$\Pr_{\boldsymbol{x}\sim\{\pm1\}^d}\left[|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x} + a| \le \varepsilon\right] \le O(\varepsilon)$$

Further recalling that $\boldsymbol{v}^*_{\text{tail}}$ and $\boldsymbol{v}_{\text{head}}$ are supported on disjoint indices in $[d]$, we see that $\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x}$ and $\phi(\boldsymbol{x})$ are indipendent random variables when $\boldsymbol{x}$ is drawn uniformly from $\{\pm 1\}^d$. This allows us to average over $a = \phi(\boldsymbol{x})$ to conclude that

$$\Pr_{\boldsymbol{x} \sim \{\pm 1\}^d} [|\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x} + \phi(\boldsymbol{x})| \le \varepsilon] \le O(\varepsilon)$$

For the second term Q_2 of Q we have the following:

$$\begin{aligned} Q_2 &:= \frac{1}{|S|} \sum_{(\boldsymbol{x},y) \in S_{\text{filt}}} \left(1 + \frac{|\phi(\boldsymbol{x})| + |\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x}|}{\varepsilon}\right) \mathbb{1}_{y \neq \text{sign}(\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})} \\ &\le (1 + \phi_{\max}/\varepsilon) \Pr_{(\boldsymbol{x},y) \sim S} [y \neq \text{sign}(\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})] + \frac{1}{|S|} \sum_{(\boldsymbol{x},y) \in S_{\text{filt}}} \left(\frac{|\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x}|}{\varepsilon}\right) \mathbb{1}_{y \neq \text{sign}(\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})} , \end{aligned}$$

where the inequality follows from the definition of S_{filt}, where we have removed the points such that $|\phi(\boldsymbol{x})| > \phi_{\max}$. Let $B = C/\varepsilon^2$. We split the summation of the second term of the above expression in two parts regarding the value of $|\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x}|$: let S_1 contain the points such that $|\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x}| \le B$ and $S_2 = S_{\text{filt}} \setminus S_1$.

$$Q_2 \le \left(1 + \frac{\phi_{\max} + B}{\varepsilon}\right) \Pr_{(\boldsymbol{x},y) \sim S} \left[y \neq \text{sign}(\phi(\boldsymbol{x}) + \boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})\right] + \frac{1}{|S|} \sum_{(\boldsymbol{x},y) \in S_{\text{filt}}} \left(\frac{|\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x}|}{\varepsilon}\right) \mathbb{1}_{(\boldsymbol{x},y) \in S_2}$$

For the second term of the above expression, we apply the Cauchy-Schwarz inequality to obtain:

$$\begin{aligned} \frac{1}{|S|} \sum_{(\boldsymbol{x},y) \in S_{\text{filt}}} \left(\frac{|\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x}|}{\varepsilon}\right) \mathbb{1}_{(\boldsymbol{x},y) \in S_2} &\le \frac{1}{\varepsilon |S|} \sqrt{\sum_{(\boldsymbol{x},y) \in S_{\text{filt}}} (\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})^2 \cdot \sum_{(\boldsymbol{x},y) \in S_{\text{filt}}} \mathbb{1}_{|\boldsymbol{v}_{\text{tail}} \cdot \boldsymbol{x}| > B}} \\ &\le \frac{1}{\varepsilon} \sqrt{\frac{8}{|S|} \sum_{(\boldsymbol{x},y) \in S_{\text{filt}}} \mathbb{1}_{|\boldsymbol{v}_{\text{tail}} \cdot \boldsymbol{x}| > B}} \\ &\le \frac{1}{\varepsilon} \sqrt{\frac{8}{|S| B^2} \sum_{(\boldsymbol{x},y) \sim S_{\text{filt}}} (\boldsymbol{v}^*_{\text{tail}} \cdot \boldsymbol{x})^2} \\ &\le \frac{8}{\varepsilon B} \le O(\varepsilon) , \end{aligned}$$

where the first inequality follows from Cauchy-Schwarz, the second follows from the guarantee of the outlier removal procedure (see Theorem 30) and the fact that $S_{\text{filt}} \subseteq S'$, the third inequality follows from Markov's inequality, and the fourth inequality follows again by the guarantee of the outlier removal procedure.

For the first term of the upper bound for Q_2, we use Hoeffding's inequality, as well as the bound on the total variation distance between D and D_{clean} to overall obtain:

$$Q_2 \le \left(1 + \frac{\phi_{\max} + B}{\varepsilon}\right) \cdot \eta + O(\varepsilon) \le O(\eta/\varepsilon^3) + O(\varepsilon) ,$$

as desired. □

Using the conclusion of Claim 31, and the fact that $\lambda\left((\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})y\right)\geq\mathbb{1}_{\text{sign}(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})\neq y}$, we see that

$$\begin{aligned}
O\left(\frac{\eta}{\varepsilon^3}+\varepsilon\right) &\geq \frac{1}{|S|}\sum_{(\boldsymbol{x},y)\in S_{\text{filt}}}\left[\lambda\left((\phi(\boldsymbol{x})+\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x})y\right)\right] \geq \frac{1}{|S|}\sum_{(\boldsymbol{x},y)\in S_{\text{filt}}}\left[\lambda\left((\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})y\right)\right] \\
&\geq \frac{1}{|S|}\sum_{(\boldsymbol{x},y)\in S_{\text{filt}}}\mathbb{1}_{\text{sign}(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})\neq y} \geq \frac{1}{|S|}\sum_{(\boldsymbol{x},y)\in S'}\mathbb{1}_{\text{sign}(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})\neq y}\mathbb{1}_{|\phi(\boldsymbol{x})|\leq\phi_{\max}} \\
&\geq \frac{1}{|S|}\sum_{(\boldsymbol{x},y)\in S}\mathbb{1}_{\text{sign}(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})\neq y}\mathbb{1}_{|\phi(\boldsymbol{x})|\leq\phi_{\max}} - 4\eta \\
&= \Pr_{(\boldsymbol{x},y)\in S}\Big[\left(|\phi(\boldsymbol{x})|\leq\phi_{\max}\right)\wedge\left(\text{sign}\left(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x}\right)\neq y\right)\Big] - 4\eta\,, \qquad (48)
\end{aligned}$$

where the last inequality follows from the guarantee of the outlier removal procedure that $|S\setminus S'|\leq 4\eta|S|$.

Since $\|\boldsymbol{v}^*_{\text{tail}}\|_2$ and $\|\widehat{\boldsymbol{v}}_{\text{tail}}\|_2$ are bounded by 1, we conclude that:

$$\begin{aligned}
\eta \geq \Pr_{(\boldsymbol{x},y)\sim D}\left[\left(|\phi(\boldsymbol{x})|>\phi_{\max}\right)\wedge\left(\text{sign}(\phi(\boldsymbol{x})+\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x})\neq y\right)\right] = \\
\Pr_{(\boldsymbol{x},y)\sim D}\left[\left(|\phi(\boldsymbol{x})|>\phi_{\max}\right)\wedge\left(\text{sign}(\phi(\boldsymbol{x}))\neq y\right)\right] \pm \underbrace{\Pr_{(\boldsymbol{x},y)\sim D}\left[|\boldsymbol{v}^*_{\text{tail}}\cdot\boldsymbol{x}|\geq\phi_{\max}\right]}_{\leq\eta+O(\varepsilon)},
\end{aligned}$$

and similarly

$$\begin{aligned}
\Pr_{(\boldsymbol{x},y)\sim D}\left[\left(|\phi(\boldsymbol{x})|>\phi_{\max}\right)\wedge\left(\text{sign}(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})\neq y\right)\right] = \\
\Pr_{(\boldsymbol{x},y)\sim D}\left[\left(|\phi(\boldsymbol{x})|>\phi_{\max}\right)\wedge\left(\text{sign}(\phi(\boldsymbol{x}))\neq y\right)\right] \pm \underbrace{\Pr_{(\boldsymbol{x},y)\sim D}\left[|\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x}|\geq\phi_{\max}\right]}_{\leq\eta+O(\varepsilon)},
\end{aligned}$$

which together tell us that

$$\Pr_{(\boldsymbol{x},y)\sim D}\left[\left(|\phi(\boldsymbol{x})|>\phi_{\max}\right)\wedge\left(\text{sign}(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})\neq y\right)\right]\leq O(\varepsilon+\eta).$$

Using the above together with Chebyshev's inequality we see that if C is a sufficiently large absolute constant, then the $C\left(\frac{d}{\varepsilon}\right)^C$-sized sample set S with probability at least 0.999 satisfies

$$\Pr_{(\boldsymbol{x},y)\sim S}\left[\left(|\phi(\boldsymbol{x})|>\phi_{\max}\right)\wedge\left(\text{sign}(\phi(\boldsymbol{x})+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x})\neq y\right)\right]\leq O(\varepsilon+\eta),$$

which together with 48 and the definition of ϕ (46) implies that

$$\Pr_{(\boldsymbol{x},y)\sim S}\left[\left(\text{sign}(\boldsymbol{v}_{\text{head}}\cdot\boldsymbol{x}+\widehat{\boldsymbol{v}}_{\text{tail}}\cdot\boldsymbol{x}+\tau)\neq y\right)\right]\leq O\left(\frac{\eta}{\varepsilon^3}+\varepsilon\right). \qquad (49)$$

Finally, a standard uniform-convergence VC-dimension argument for halfspaces tells us that if C is a sufficiently large absolute constant, then the $C\left(\frac{d}{\varepsilon}\right)^C$-sized sample set S with probability at least 0.999 satisfies the uniform convergence property for the class of halfspaces, which together with Equation 49 implies Equation 45 finishing the proof of Theorem 29. □

Another important ingredient for the proof of Theorem 28 is the following analogue of Theorem 3, which ensures that we may consider the positions of the k-largest magnitude coefficients to be known, by only incurring an $O(k)$ error amplification, even in the presence of contamination.

Theorem 32. *There is an algorithm that, given sample access to a distribution over pairs $(\boldsymbol{x}, y) \sim \{\pm 1\}^d \times \{\pm 1\}$ that has total variation distance at most η from another distribution over $(\boldsymbol{x}, y)$ such that $\boldsymbol{x}$ is uniform over $\{\pm 1\}^d$ and $y = \mathrm{sign}(\boldsymbol{v}^* \cdot \boldsymbol{x} + \tau^*)$ for some unknown $\boldsymbol{v}^*, \tau^*$, outputs, with probability at least 0.995, a set of k indices $\{j_1, \ldots, j_k\}$ such that:*

$$\Pr_{\boldsymbol{x} \sim \{\pm 1\}^d} \left[\mathrm{sign}(\boldsymbol{v}^* \cdot \boldsymbol{x} + \tau^*) \neq \mathrm{sign}(\tilde{\boldsymbol{v}}^* \cdot \boldsymbol{x} + \tau^*) \right] \leq 4\eta k \,,$$

where $\tilde{\boldsymbol{v}}^*$ is such that the k largest (in absolute value) coefficients correspond to the indices $j_1, \ldots, j_k$. The algorithm has time and sample complexity $\mathrm{poly}(d, 1/\eta)$.

Proof. The algorithm draws a set S of sufficiently many samples from the input distribution and computes the quantities $\widehat{I_i} = \mathbb{E}_{(\boldsymbol{x},y)\sim S}[yx_i]$ and let $j_1, j_2, \ldots, j_k$ be the k indices i corresponding to the k largest values of $|\widehat{I_i}|$. Since $|yx_i| = 1$ and the total variation distance between the input distribution and the clean distribution is η, we have that the choice $(\widehat{i_1}, \ldots, \widehat{i_k}) = (j_1, \ldots, j_k)$ satisfies, with high probability, the premise of Theorem 9 for the choice $\eta = 4\eta$, which concludes the proof. □

We are now ready to prove Theorem 28.

Proof of Theorem 28. As in the proof of Theorem 24, we may, without loss of generality assume that $\eta \leq \varepsilon^{49}/C$ for some sufficiently large constant $C \geq 1$. The algorithm that achieves the guarantee of Theorem 28 is very similar to Algorithm 1, with the main differences being that (1) the choice of the parameter k for the critical index lemma is $\tilde{O}(1/\varepsilon^{c_1})$ for $c_1 = 49/3$, and (2) in order to learn the regular coefficients (part II of the structured case), the algorithm uses FINDREGULARCONTAMINATED in place of FINDREGULARCOEFFICIENTS. The proof of correctness closely follows the proof of Theorem 24. Since, in this case, the noise is not only on the labels, but also on the features, we use Theorem 32 to find the influential coordinates. For the sparse case, as well as learning the heavy coefficients in the structured case, recall that we used appropriate total variation distance arguments to show that the sample complexity is small enough with respect to the noise rate, so that we can safely assume that the noise is negligible, and the performance of the algorithms will be, with high probability, identical to the case where the labels are realized by the corresponding sparse or structured halfspace. In fact, these total variation distance arguments are even stronger and they hold identically in the setting of learning with contamination. Finally, to learn the regular coefficients, we use Theorem 29, which tolerates contamination but incurs a worse error dependence compared to its analogue for the label noise case. In particular, we overall achieve an error guarantee of $O(\eta^{1/49}) + \varepsilon$. □

Remark 2. In order to obtain an error guarantee that nearly matches the guarantee of $O(\mathrm{opt}^{1/25}) + \varepsilon$ that we achieved in the label noise case, we may apply Theorem 30 for some large enough constant r and obtain a tighter version of Theorem 29. However, this would also lead to a worse runtime of $O(d)^{4r}$. Here, we chose $r = 1$ for simplicity of presentation.

References

[ABHU15] Pranjal Awasthi, Maria-Florina Balcan, Nika Haghtalab, and Ruth Urner. Efficient learning of linear separators under bounded noise. In *Conference on Learning Theory*, pages 167–190. PMLR, 2015. 2

[ABHZ16] Pranjal Awasthi, Maria-Florina Balcan, Nika Haghtalab, and Hongyang Zhang. Learning and 1-bit compressed sensing under asymmetric noise. In *Conference on Learning Theory*, pages 152–192. PMLR, 2016. 2

[ABL17] Pranjal Awasthi, Maria Florina Balcan, and Philip M Long. The power of localization for efficiently learning linear separators with noise. *Journal of the ACM (JACM)*, 63(6):1–27, 2017. 1, 2, 9

[AHW95] Peter Auer, Mark Herbster, and Manfred KK Warmuth. Exponentially many local minima for single neurons. *Advances in neural information processing systems*, 8, 1995. 3

[ATV23] Pranjal Awasthi, Alex Tang, and Aravindan Vijayaraghavan. Agnostic learning of general reLU activation using gradient descent. In *The Eleventh International Conference on Learning Representations*, 2023. 3

[BEK02] Nader H. Bshouty, Nadav Eiron, and Eyal Kushilevitz. Pac learning with nasty noise. *Theoretical Computer Science*, 288(2):255–275, 2002. Algorithmic Learning Theory. 8

[BV25] Guy Blanc and Gregory Valiant. Adaptive and oblivious statistical adversaries are equivalent. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 2031–2042, 2025. 9, 36

[CKMY20] Sitan Chen, Frederic Koehler, Ankur Moitra, and Morris Yau. Classification under misspecification: Halfspaces, generalized linear models, and evolvability. *Advances in Neural Information Processing Systems*, 33:8391–8403, 2020. 3

[Dan15] Amit Daniely. A ptas for agnostically learning halfspaces. In *Conference on Learning Theory*, pages 484–502. PMLR, 2015. 1, 2

[DB18] Annette J Dobson and Adrian G Barnett. *An introduction to generalized linear models*. Chapman and Hall/CRC, 2018. 3

[DDFS14] Anindya De, Ilias Diakonikolas, Vitaly Feldman, and Rocco A Servedio. Nearly optimal solutions for the chow parameters problem and low-weight approximation of halfspaces. *Journal of the ACM (JACM)*, 61(2):1–36, 2014. 1, 2, 3

[DGJ+10] Ilias Diakonikolas, Parikshit Gopalan, Ragesh Jaiswal, Rocco A Servedio, and Emanuele Viola. Bounded independence fools halfspaces. *SIAM Journal on Computing*, 39(8):3441–3462, 2010. 3

[DGK+20] Ilias Diakonikolas, Surbhi Goel, Sushrut Karmalkar, Adam R Klivans, and Mahdi Soltanolkotabi. Approximation schemes for relu regression. In *Conference on learning theory*, pages 1452–1485. PMLR, 2020. 3

[DH18] Rishabh Dudeja and Daniel Hsu. Learning single-index models in gaussian space. In Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet, editors, *Proceedings of the 31st Conference On Learning Theory*, volume 75 of *Proceedings of Machine Learning Research*, pages 1887–1930. PMLR, 06–09 Jul 2018. 3

[DK19] Ilias Diakonikolas and Daniel M Kane. 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*, pages 804–815, 2019. 3

[DKK+21] Ilias Diakonikolas, Daniel M. Kane, Vasilis Kontonis, Christos Tzamos, and Nikos Zarifis. Agnostic proper learning of halfspaces under gaussian marginals. In Mikhail Belkin and Samory Kpotufe, editors, *Conference on Learning Theory, COLT 2021, 15-19 August 2021, Boulder, Colorado, USA*, volume 134 of *Proceedings of Machine Learning Research*, pages 1522–1551. PMLR, 2021. 1, 2

[DKK+24] Ilias Diakonikolas, Daniel M Kane, Vasilis Kontonis, Sihan Liu, and Nikos Zarifis. 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*, pages 152–159, 2024. 2

[DKMR22] Ilias Diakonikolas, Daniel Kane, Pasin Manurangsi, and Lisheng Ren. Hardness of learning a single neuron with adversarial label noise. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 8199–8213. PMLR, 28–30 Mar 2022. 3

[DKS18a] Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. Learning geometric concepts with nasty noise. In Ilias Diakonikolas, David Kempe, and Monika Henzinger, editors, *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018*, pages 1061–1073. ACM, 2018. 1, 2

[DKS18b] Ilias Diakonikolas, Daniel M Kane, and Alistair Stewart. Learning geometric concepts with nasty noise. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1061–1073, 2018. 9, 37

[DKTZ20a] Ilias Diakonikolas, Vasilis Kontonis, Christos Tzamos, and Nikos Zarifis. Learning halfspaces with massart noise under structured distributions. In *Conference on Learning Theory*, pages 1486–1513. PMLR, 2020. 2

[DKTZ20b] Ilias Diakonikolas, Vasilis Kontonis, Christos Tzamos, and Nikos Zarifis. Non-convex sgd learns halfspaces with adversarial label noise. *Advances in Neural Information Processing Systems*, 33:18540–18549, 2020. 2

[DKTZ22] Ilias Diakonikolas, Vasilis Kontonis, Christos Tzamos, and Nikos Zarifis. Learning a single neuron with adversarial label noise via gradient descent. In *Conference on learning theory*, pages 4313–4361. PMLR, 2022. 3

[DSFT+14] Dana Dachman-Soled, Vitaly Feldman, Li-Yang Tan, Andrew Wan, and Karl Wimmer. Approximate resilience, monotonicity, and the complexity of agnostic learning. In *Proceedings of*

the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms, pages 498–511. SIAM, 2014. 3

[DZ24] Ilias Diakonikolas and Nikos Zarifis. A near-optimal algorithm for learning margin halfspaces with massart noise. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems, NeurIPS*, 2024. 1

[GGKS23] Aravind Gollakota, Parikshit Gopalan, Adam Klivans, and Konstantinos Stavropoulos. Agnostically learning single-index models using omnipredictors. *Advances in Neural Information Processing Systems*, 36:14685–14704, 2023. 3

[GOWZ10] Parikshit Gopalan, Ryan O'Donnell, Yi Wu, and David Zuckerman. Fooling functions of halfspaces under product distributions. In *2010 IEEE 25th Annual Conference on Computational Complexity*, pages 223–234. IEEE, 2010. 4, 10, 31

[GSSV24] Surbhi Goel, Abhishek Shetty, Konstantinos Stavropoulos, and Arsen Vasilyan. Tolerant algorithms for learning with arbitrary covariate shift. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 124979–125018. Curran Associates, Inc., 2024. 9

[GV24] Anxin Guo and Aravindan Vijayaraghavan. Agnostic learning of arbitrary relu activation under gaussian marginals. In *Annual Conference Computational Learning Theory*, 2024. 3

[Kan18] Varun Kanade. Lecture notes: Learning real-valued functions, 2018. 6, 20, 22, 47, 50

[KKMS08] Adam Tauman Kalai, Adam R Klivans, Yishay Mansour, and Rocco A Servedio. Agnostically learning halfspaces. *SIAM Journal on Computing*, 37(6):1777–1805, 2008. 2, 3

[KKSK11] Sham M Kakade, Varun Kanade, Ohad Shamir, and Adam Kalai. Efficient learning of generalized linear and single index models with isotonic regression. *Advances in Neural Information Processing Systems*, 24, 2011. 3

[KLS09] Adam R Klivans, Philip M Long, and Rocco A Servedio. Learning halfspaces with malicious noise. *Journal of Machine Learning Research*, 10(12), 2009. 2, 9, 37

[KM17] Adam Klivans and Raghu Meka. Learning graphical models using multiplicative weights. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 343–354. IEEE, 2017. 3

[KS09] Adam Tauman Kalai and Ravi Sastry. The isotron algorithm: High-dimensional isotonic regression. In *COLT*, volume 1, page 9, 2009. 3

[KSV24] Adam R Klivans, Konstantinos Stavropoulos, and Arsen Vasilyan. Learning constant-depth circuits in malicious noise models. *arXiv preprint arXiv:2411.03570*, 2024. 9, 37

[MV19] Oren Mangoubi and Nisheeth K Vishnoi. Nonconvex sampling with the metropolis-adjusted langevin algorithm. In *Conference on learning theory*, pages 2259–2293. PMLR, 2019. 2

[NW72] John Ashworth Nelder and Robert WM Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 135(3):370–384, 1972. 3

[O'D14] Ryan O'Donnell. *Analysis of boolean functions*. Cambridge University Press, 2014. 17

[OS08] Ryan O'Donnell and Rocco A Servedio. The chow parameters problem. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 517–526, 2008. 1, 2, 3, 8

[RBH+25] Dhruv Rohatgi, Adam Block, Audrey Huang, Akshay Krishnamurthy, and Dylan J. Foster. Computational-statistical tradeoffs at the next-token prediction barrier: Autoregressive and imitation learning under misspecification (extended abstract). In Nika Haghtalab and Ankur Moitra, editors, *Proceedings of Thirty Eighth Conference on Learning Theory*, volume 291 of *Proceedings of Machine Learning Research*, pages 4831–4837. PMLR, 30 Jun–04 Jul 2025. 3

[Ros58] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958. 3

[Ser06] Rocco A. Servedio. Every linear threshold function has a low-weight approximator. In *Proceedings of the 21st Annual IEEE Conference on Computational Complexity*, CCC '06, pages 18–32, USA, 2006. IEEE Computer Society. 4, 6, 31

[SSSS10] Shai Shalev-Shwartz, Ohad Shamir, and Karthik Sridharan. Learning kernel-based halfspaces with the zero-one loss. *ArXiv*, abs/1005.3681, 2010. 3

[Vem10] Santosh S Vempala. A random-sampling-based algorithm for learning intersections of halfspaces. *Journal of the ACM (JACM)*, 57(6):1–14, 2010. 1, 2

[WZDD23] Puqian Wang, Nikos Zarifis, Ilias Diakonikolas, and Jelena Diakonikolas. Robustly learning a single neuron via sharpness. In *International conference on machine learning*, pages 36541–36577. PMLR, 2023. 3

[WZDD24] Puqian Wang, Nikos Zarifis, Ilias Diakonikolas, and Jelena Diakonikolas. Sample and computationally efficient robust learning of gaussian single-index models. *Advances in Neural Information Processing Systems*, 37:58376–58422, 2024. 3

[WZDD25] Puqian Wang, Nikos Zarifis, Ilias Diakonikolas, and Jelena Diakonikolas. Robustly learning monotone single-index models. *ArXiv*, abs/2508.04670, 2025. 3

[YZ17] Songbai Yan and Chicheng Zhang. Revisiting perceptron: Efficient and label-optimal learning of halfspaces. *Advances in Neural Information Processing Systems*, 30, 2017. 2

[ZLC17] Yuchen Zhang, Percy Liang, and Moses Charikar. A hitting time analysis of stochastic gradient langevin dynamics. In *Conference on Learning Theory*, pages 1980–2022. PMLR, 2017. 2

[ZWDD24] Nikos Zarifis, Puqian Wang, Ilias Diakonikolas, and Jelena Diakonikolas. Robustly learning single-index models via alignment sharpness. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 58197–58243. PMLR, 21–27 Jul 2024. 3

A Auxiliary Tools

Lemma 33 (Berry–Esseen). *Let $X_1, \dots, X_d$ be independent random bariables such that $\mathbb{E}[X_i] = 0$, $\sum_{i\in[d]} X_i^2 \le \sigma^2$, and $\sum_{i\in[d]} |X_i|^3 \le \beta$. Let $S = \frac{1}{\sigma}\sum_{i\in[d]} X_i$ and denote with D_S the distribution of S. Then we have the following:*

$$\left| \Pr_{S\sim D_S}[S \ge \tau] - \Pr_{z\in\mathcal{N}(0,1)}[z \ge \tau] \right| \le \frac{\beta}{\sigma^3}, \text{ for all } \tau \in \mathbb{R}$$

Corollary 34. Let $\boldsymbol{v} \in \mathbb{R}^d$ such that $\|\boldsymbol{v}\|_4^2 \le \gamma \|\boldsymbol{v}\|_2^2$. Then, for every $\tau \in \mathbb{R}$ we have:

$$\left| \Pr_{\boldsymbol{x}\sim\{\pm1\}^d}\left[\frac{\boldsymbol{x}\cdot\boldsymbol{v}}{\|\boldsymbol{v}\|_2} \ge \tau\right] - \Pr_{z\sim\mathcal{N}(0,1)}[z \ge \tau] \right| \le \gamma$$

Proof of Corollary 34. Let $X_i = v_i x_i$. We have that $\sum_i X_i^2 = \|\boldsymbol{v}\|_2^2 =: \sigma^2$. Moreover, by applying the Cauchy-Schwarz inequality, we obtain the following:

$$\sum_i |X_i|^3 \le \sqrt{\sum_i X_i^2} \cdot \sqrt{\sum_i X_i^4} = \|\boldsymbol{v}\|_2 \cdot \|\boldsymbol{v}\|_4^2 \le \gamma \|\boldsymbol{v}\|_2^3 = \gamma\sigma^3 =: \beta$$

We may now apply Lemma 33 to obtain the desired result for $S = \frac{\boldsymbol{x}\cdot\boldsymbol{v}}{\|\boldsymbol{v}\|_2}$. □

B Theorems on GLM learning

This section contains all the theorems and proofs relevant to GLM learning that were omitted in Section 5.

B.1 Realizable GLM with truncated activation

We show a weaker form of Theorem 17 which has no misspecification. This will be a helpful warmup to read before Theorem 17 and contains many of the ideas that lead to the improved dependence on $w_{\max}$. Another feature is that this weaker theorem is sufficient for proving our result for learning sigmoidal GLMs (Theorem 1). We are now ready to state the theorem on learning realizable truncated GLMs.

Theorem 35. *Let $\alpha \in \mathbb{R}_{\ge 1}$ and suppose σ is a an activation function $\mathbb{R} \to [-1, 1]$ that satisfies the following three conditions:*

- *σ is monotone non-decreasing.*
- *σ is 1-Lipschitz.*
- *$\sigma(z) = \operatorname{sign}(z)$ whenever $|z| \ge \alpha$.*

Then, there exists an algorithm $\mathcal{A}^{\sigma}_{\mathrm{GLM}}(\varepsilon, w_{\max}, \alpha, D)$ that

- *Draws $O\left(\frac{\Delta\alpha^2}{\varepsilon^2} \ln \frac{\Delta\alpha}{\varepsilon}\right)$ samples $\{x_i\}$ from a distribution D over $\left\{\boldsymbol{x} \in \mathbb{R}^\Delta : \|\boldsymbol{x}\|_2 \le \sqrt{\Delta}\right\} \times \{\pm 1\}$ and outputs a vector $\boldsymbol{v} \in \mathbb{R}^\Delta$.*
- *Runs in time $\operatorname{poly}\left(\frac{\alpha\Delta}{\varepsilon} \log w_{\max}\right)$.*

- *When D is a GLM with an arbitrary $\mathbb{R}^\Delta$-marginal which is supported on $\{\boldsymbol{x} \in \mathbb{R}^\Delta : \|\boldsymbol{x}\|_2 \le \sqrt{\Delta}\}$, and an activation $\sigma(\boldsymbol{v}^* \cdot \boldsymbol{x})$, where $\|\boldsymbol{v}^*\|_2 \le w_{\max}$, then with probability at least 0.99 the algorithm $\mathcal{A}^\sigma_{\mathrm{GLM}}(\varepsilon, w_{\max}, \alpha, D)$ will output a vector $\boldsymbol{v}_0$ satisfying*

$$\mathop{\mathbb{E}}_{\boldsymbol{x} \sim D_X}\left[\left(\sigma(\boldsymbol{x} \cdot \boldsymbol{v}^*) - \sigma(\boldsymbol{x} \cdot \boldsymbol{v}_0)\right)^2\right] \le \varepsilon \tag{50}$$

where D_X is the $\mathbb{R}^\Delta$-marginal of D. (Note that no assumption is made on D_X.)

Proof. We will follow closely the Varun Kanade's notes [Kan18]. Define the surrogate loss ℓ^σ as follows:

$$\ell^\sigma(\boldsymbol{v}; \boldsymbol{x}, y) = \int_0^{\boldsymbol{v}\cdot\boldsymbol{x}} (\sigma(z) - y)\, dz.$$

The surrogate loss is convex because σ is monotone. The algorithm $\mathcal{A}^\sigma_{\mathrm{GLM}}(\varepsilon, w_{\max}, D)$ does the following:

- Draw a collection S of $C\frac{\Delta\alpha^2}{\varepsilon^2}\ln\frac{\Delta\alpha}{\varepsilon} + C$ samples from D, where C is a sufficiently large absolute constant.
- Using the Ellipsoid algorithm, find

$$\boldsymbol{v}_0 \leftarrow \mathop{\arg\min}_{\boldsymbol{v}:\ \|\boldsymbol{v}\| \le w_{\max}} \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\in S} [\ell^\sigma(\boldsymbol{v}; \boldsymbol{x}, y)]$$

- Output $\boldsymbol{v}_0$.

The run-time is dominated by the second step, which runs in time $\mathrm{poly}\left(\frac{\alpha\Delta}{\varepsilon}\log w_{\max}\right)$.

To analyze the algorithm, we define the following auxilliary functions:

$$\mathrm{trunc}(z) := \min(\alpha, \max(-\alpha, z))$$

$$\ell^\sigma_{\mathrm{trunc}}(\boldsymbol{v}; \boldsymbol{x}, y) = \int_0^{\mathrm{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})} (\sigma(z) - y)\, dz.$$

Since $\sigma(z) = \mathrm{sign}(z)$ whenever $|z| \ge \alpha$, for every $z \in \mathbb{R}$ we have

$$\sigma(z) = \sigma(\mathrm{trunc}(z)). \tag{51}$$

We now make the following observation:

Claim 36. For every $(\boldsymbol{x}, y)$ in the support of D we have

$$\ell^\sigma(\boldsymbol{v}^*; \boldsymbol{x}, y) = \ell^\sigma_{\mathrm{trunc}}(\boldsymbol{v}^*; \boldsymbol{x}, y) \in [-2\alpha, 2\alpha]$$

Proof. Since $\sigma(z) = \mathrm{sign}(x)$ whenever $|x| \ge \alpha$, for every $(\boldsymbol{x}, y)$ in the support of D, if $|\boldsymbol{x} \cdot \boldsymbol{v}^*| > \alpha$ we have $y = \mathrm{sign}(\alpha)$. Therefore,

- If $\boldsymbol{x} \cdot \boldsymbol{v}^* > \alpha$, it has to be that $y = \mathrm{sign}(\boldsymbol{x} \cdot \boldsymbol{v}^*) = 1$ and therefore

$$\ell^\sigma(\boldsymbol{v}^*; \boldsymbol{x}, y) = \int_0^{\boldsymbol{v}^*\cdot\boldsymbol{x}} (\sigma(z)-1)\, dz = \int_0^{\alpha} (\sigma(z)-1)\, dz = \int_0^{\mathrm{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})} (\sigma(z)-1)\, dz = \ell^\sigma_{\mathrm{trunc}}(\boldsymbol{v}^*; \boldsymbol{x}, y).$$

  We also see that $\int_0^\alpha (\sigma(z) - 1)\, dz$ is in $[-2\alpha, 0]$ since $\sigma(z) \in [-1, 1]$ for all z.

- Analogously, if $\boldsymbol{x} \cdot \boldsymbol{v}^* < -\alpha$, then analogously $y = -1$ and therefore

$$\ell^\sigma(\boldsymbol{v}^*; \boldsymbol{x}, y) = \int_0^{\boldsymbol{v}^* \cdot \boldsymbol{x}} (\sigma(z)+1)\, dz = \int_0^{-\alpha} (\sigma(z)+1)\, dz = \int_0^{\text{trunc}(\boldsymbol{v}^* \cdot \boldsymbol{x})} (\sigma(z)+1)\, dz = \ell^\sigma_{\text{trunc}}(\boldsymbol{v}^*; \boldsymbol{x}, y).$$

  We also see that $\int_0^{-\alpha}(\sigma(z)+1)\, dz = -\int_{-\alpha}^{0}(\sigma(z)+1)\, dz$ is in $[-2\alpha, 0]$ since $\sigma(z) \in [-1, 1]$ for all z.

- In the remaining case $\boldsymbol{x} \cdot \boldsymbol{v}^* \in [-\alpha, \alpha]$ we have

$$\ell^\sigma(\boldsymbol{v}^*; \boldsymbol{x}, y) = \int_0^{\boldsymbol{v}^* \cdot \boldsymbol{x}} (\sigma(z) - y)\, dz = \int_0^{\text{trunc}(\boldsymbol{v}^* \cdot \boldsymbol{x})} (\sigma(z) - 1)\, dz = \ell^\sigma_{\text{trunc}}(\boldsymbol{v}^*; \boldsymbol{x}, y).$$

  And again, we see that since $\boldsymbol{x} \cdot \boldsymbol{v}^* \in [-\alpha, \alpha]$ we have

$$\left| \int_0^{\boldsymbol{v}^* \cdot \boldsymbol{x}} (\sigma(z) - y)\, dz \right| \leq \int_0^{\boldsymbol{v}^* \cdot \boldsymbol{x}} |\sigma(z) - y|)\, dz \leq 2\alpha.$$

□

Now, since $|\ell^\sigma(\boldsymbol{v}^*; \boldsymbol{x}, y)| \leq 2\alpha$, we can use the Hoeffding bound to conclude that w.p. 0.99 over the choice of our sample set S satisfies

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}[\ell^\sigma(\boldsymbol{v}^*; \boldsymbol{x}, y)] = \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}[\ell^\sigma(\boldsymbol{v}^*; \boldsymbol{x}, y)] \pm O\left(\frac{\alpha}{\sqrt{|S|}}\right),$$

Since $\boldsymbol{v}_0$ minimizes the empirical surrogate loss, among all vectors whose norm is at most $w_{\max}$, and $\|\boldsymbol{v}^*\|_2 \leq w_{\max}$ we also have

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}[\ell^\sigma(\boldsymbol{v}_0; \boldsymbol{x}, y)] \leq \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}[\ell^\sigma(\boldsymbol{v}^*; \boldsymbol{x}, y)] = \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}[\ell^\sigma(\boldsymbol{v}^*; \boldsymbol{x}, y)] \pm O\left(\frac{\alpha}{\sqrt{|S|}}\right) = . \\ \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}[\ell^\sigma(\boldsymbol{v}^*; \boldsymbol{x}, y)] \pm \frac{\varepsilon}{4}, \quad (52)$$

where in the last step we substituted $|S| = C\frac{\Delta\alpha^2}{\varepsilon^2} \ln \frac{\Delta\alpha}{\varepsilon} + C$ and took C to be a sufficiently large absolute constant.

We make the following further observations regarding the truncated loss $\ell^\sigma_{\text{trunc}}(\boldsymbol{v}; \boldsymbol{x}, y)$:

- For every $\boldsymbol{v}, \boldsymbol{x}$ and y we have

$$|\ell^\sigma_{\text{trunc}}(\boldsymbol{v}; \boldsymbol{x}, y)| \leq \int_0^{\text{trunc}(\boldsymbol{v} \cdot \boldsymbol{x})} |\sigma(z) - y|\ dz \leq 2\alpha.$$

- For every $\boldsymbol{v}, \boldsymbol{x}$ and y we have

$$\ell^\sigma_{\text{trunc}}(\boldsymbol{v}; \boldsymbol{x}, y) \leq \ell^\sigma(\boldsymbol{v}; \boldsymbol{x}, y) \quad (53)$$

  because

– If $\boldsymbol{v}\cdot\boldsymbol{x}\in[-\alpha,\alpha]$, we have $\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})=\boldsymbol{v}\cdot\boldsymbol{x}$ and we have $\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v};\boldsymbol{x},y)=\ell^{\sigma}(\boldsymbol{v};\boldsymbol{x},y)$.

– If $\boldsymbol{v}\cdot\boldsymbol{x}>\alpha$ we have

$$\ell^{\sigma}(\boldsymbol{v};\boldsymbol{x},y)=\int_0^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}(\sigma(z)-y)\,dz+\int_{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}^{\boldsymbol{v}\cdot\boldsymbol{x}}(1-y)\,dz\geq\int_0^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}(\sigma(z)-y)\,dz$$

– If $\boldsymbol{v}\cdot\boldsymbol{x}<-\alpha$ we have

$$\ell^{\sigma}(\boldsymbol{v};\boldsymbol{x},y)=\int_0^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}(\sigma(z)-y)\,dz+\int_{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}^{\boldsymbol{v}\cdot\boldsymbol{x}}(-1-y)\,dz\geq\int_0^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}(\sigma(z)-y)\,dz$$

We also need the following claim:

Claim 37. For a sufficiently large value of the absolute constant C, with probability at least 0.999 over the choice of the set S, we have the following uniform convergence bound for $\ell^{\sigma}_{\text{trunc}}$:

$$\max_{\boldsymbol{v}'\in\mathbb{R}^d}\left|\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v}';\boldsymbol{x},y)\right]-\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}\left[\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v}';\boldsymbol{x},y)\right]\right|\leq\tilde{O}\left(\alpha\sqrt{\frac{\Delta}{|S|}}\right)\leq\frac{\varepsilon}{4}\tag{54}$$

Proof. We have

$$\begin{aligned}\int_0^{\text{trunc}(\boldsymbol{v}'\cdot\boldsymbol{x})}(\sigma(z)-y)\,dz&=\int_0^{\alpha}(\sigma(z)-y)\mathbb{1}_{z\leq\text{trunc}(\boldsymbol{v}'\cdot\boldsymbol{x})}\,dz-\int_{-\alpha}^{0}(\sigma(z)-y)\mathbb{1}_{z>\text{trunc}(\boldsymbol{v}'\cdot\boldsymbol{x})}\,dz\\&=\int_0^{\alpha}(\sigma(z)-y)\mathbb{1}_{z\leq\boldsymbol{v}'\cdot\boldsymbol{x}}\,dz-\int_{-\alpha}^{0}(\sigma(z)-y)\mathbb{1}_{z>\boldsymbol{v}'\cdot\boldsymbol{x}}\,dz\end{aligned}$$

which implies

$$\begin{aligned}&\left|\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\int_0^{\text{trunc}(\boldsymbol{v}'\cdot\boldsymbol{x})}(\sigma(z)-y)\,dz\right]-\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}\left[\int_0^{\text{trunc}(\boldsymbol{v}'\cdot\boldsymbol{x})}(\sigma(z)-y)\,dz\right]\right|\\\leq&\alpha\max_{\boldsymbol{v}'\in\mathbb{R}^d,z\in\mathbb{R}}\left[\left|\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[(\sigma(z)-y)\mathbb{1}_{z\leq\boldsymbol{v}'\cdot\boldsymbol{x}}\right]-\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}\left[(\sigma(z)-y)\mathbb{1}_{z\leq\boldsymbol{v}'\cdot\boldsymbol{x}}\right]\right|\right]\\&+\alpha\max_{\boldsymbol{v}'\in\mathbb{R}^d,z\in\mathbb{R}}\left[\left|\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[(\sigma(z)-y)\mathbb{1}_{z>\boldsymbol{v}'\cdot\boldsymbol{x}}\right]-\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}\left[(\sigma(z)-y)\mathbb{1}_{z>\boldsymbol{v}'\cdot\boldsymbol{x}}\right]\right|\right].\end{aligned}\tag{55}$$

Standard VC-dimension bounds for linear classifiers tell us that for a sufficiently large absolute constant C, if $|S|\geq C\frac{\Delta\alpha^2}{\varepsilon^2}\ln\frac{\Delta\alpha}{\varepsilon}+C$ then with probability 0.999 for all $\boldsymbol{v}'$ and z we have

1. $\left|\mathbb{E}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{z\leq\boldsymbol{v}'\cdot\boldsymbol{x}}\right]-\mathbb{E}_{(\boldsymbol{x},y)\sim S}\left[\mathbb{1}_{z\leq\boldsymbol{v}'\cdot\boldsymbol{x}}\right]\right|\leq\frac{\varepsilon}{16\alpha}$,
2. $\left|\mathbb{E}_{(\boldsymbol{x},y)\sim D}\left[\mathbb{1}_{z>\boldsymbol{v}'\cdot\boldsymbol{x}}\right]-\mathbb{E}_{(\boldsymbol{x},y)\sim S}\left[\mathbb{1}_{z>\boldsymbol{v}'\cdot\boldsymbol{x}}\right]\right|\leq\frac{\varepsilon}{16\alpha}$,
3. $\left|\mathbb{E}_{(\boldsymbol{x},y)\sim D}\left[y\cdot\mathbb{1}_{z\leq\boldsymbol{v}'\cdot\boldsymbol{x}}\right]-\mathbb{E}_{(\boldsymbol{x},y)\sim S}\left[y\cdot\mathbb{1}_{z\leq\boldsymbol{v}'\cdot\boldsymbol{x}}\right]\right|\leq\frac{\varepsilon}{16\alpha}$, and
4. $\left|\mathbb{E}_{(\boldsymbol{x},y)\sim D}\left[y\cdot\mathbb{1}_{z>\boldsymbol{v}'\cdot\boldsymbol{x}}\right]-\mathbb{E}_{(\boldsymbol{x},y)\sim S}\left[y\cdot\mathbb{1}_{z>\boldsymbol{v}'\cdot\boldsymbol{x}}\right]\right|\leq\frac{\varepsilon}{16\alpha}$.

Therefore, combining the bounds above with Equation 55, using a triangle inequality and recalling $|\sigma(z)|\leq 1$ for every z, we obtain Equation 54 . □

Thus, we have

$$\begin{aligned}
\mathop{\mathbb{E}}_{y\sim D_{y|\boldsymbol{x}}}\left[\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v};\boldsymbol{x},y)\right] - \mathop{\mathbb{E}}_{y\sim D_{y|\boldsymbol{x}}}\left[\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v}^*;\boldsymbol{x},y)\right] &= \mathop{\mathbb{E}}_{y\sim D_{y|\boldsymbol{x}}}\left[\int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}(\sigma(z)-y)\,dz\right] \\
&= \int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}(\sigma(z)-\sigma(\boldsymbol{v}^*\cdot\boldsymbol{x}))\,dz = \int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}(\sigma(z)-\sigma(\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})))\,dz \\
&\geq \frac{1}{2}\left(\sigma(\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x}))-\sigma(\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x}))\right)^2 = \frac{1}{2}\left(\sigma(\boldsymbol{v}\cdot\boldsymbol{x})-\sigma(\boldsymbol{v}^*\cdot\boldsymbol{x})\right)^2. \qquad (56)
\end{aligned}$$

The first equality is due to linearity of expectation and the definition of $\ell^{\sigma}_{\text{trunc}}$. The second equality is because D_x is a GLM with vector $\boldsymbol{v}^*$ and activation σ. The third equality is due to Equation 51. The inequality $\int_{\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})}^{\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x})}(\sigma(z)-\sigma(\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x})))\,dz \geq \frac{1}{2}\left(\sigma(\text{trunc}(\boldsymbol{v}\cdot\boldsymbol{x}))-\sigma(\text{trunc}(\boldsymbol{v}^*\cdot\boldsymbol{x}))\right)^2$ is due to σ being montonically increasing and 1-Lipschitz (same as in page 5 of [Kan18]). The last equality is due to Equation 51. We therefore have for any vector $v\in\mathbb{R}^d$

$$\begin{aligned}
\frac{1}{2}\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\left(\sigma(\boldsymbol{v}\cdot\boldsymbol{x})-\sigma(\boldsymbol{v}^*\cdot\boldsymbol{x})\right)^2\right] &\leq \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v};\boldsymbol{x},y)-\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v}^*;\boldsymbol{x},y)\right] \\
&= \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v};\boldsymbol{x},y)-\ell^{\sigma}(\boldsymbol{v}^*;\boldsymbol{x},y)\right] \\
&\leq \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}\left[\ell^{\sigma}_{\text{trunc}}(\boldsymbol{v};\boldsymbol{x},y)\right] - \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\ell^{\sigma}(\boldsymbol{v}^*;\boldsymbol{x},y)\right]+\varepsilon/4 \\
&\leq \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim S}\left[\ell^{\sigma}(\boldsymbol{v};\boldsymbol{x},y)\right] - \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\ell^{\sigma}(\boldsymbol{v}^*;\boldsymbol{x},y)\right]+\varepsilon/4.
\end{aligned}$$

The first step uses Equation 56. The second step uses Claim 36. The third step uses Equation 54. The fourth step uses Equation 53. Finally, combining with Equation 52, we get the desired Equation 50. □

B.2 Learning Sigmoidal Activations

We are now ready to prove Theorem 1 on the learnability of sigmoidal activations.

Proof of Theorem 1. The proof proceeds by reducing to the task of learning from GLMs with clipped activation. Let $\rho = C(\log\frac{\Delta}{\epsilon})$ be some parameter for a sufficiently large constant C. Let D_X be the marginal distribution of D. We define the "clipped" activation $\sigma^{\text{sig}}_{\text{clipped}}$ as

$$\sigma^{\text{sig}}_{\text{clipped}}(z) = \begin{cases} \sigma^{\text{sig}}(z) & \text{if } |z|\leq\rho-1 \\ \sigma^{\text{sig}}(\rho-1)+(1-\sigma^{\text{sig}}(\rho-1))\cdot(z-(\rho-1)) & \text{if } z\in(\rho-1,\rho] \\ \sigma^{\text{sig}}(-(\rho-1))-(1+\sigma^{\text{sig}}(-(\rho-1)))\cdot(-z-(\rho-1)) & \text{if } z\in[-\rho,-\rho+1) \\ \text{sign}(z) & \text{if } |z|>\rho \end{cases} \qquad (57)$$

Observe that for all $t\in\mathbb{R}$

$$|\sigma^{\text{sig}}(t)-\sigma^{\text{sig}}_{\text{clipped}}(t)| \leq |1-(1-e^{-\rho})/(1+e^{-\rho})| \leq e^{-\rho}/2. \qquad (58)$$

Let D_{clipped} be a GLM on $\mathbb{R}^{\Delta+1}\times\{\pm1\}$ defined as follows:

- The marginal of D_{clipped} is sampled as follows: Sample $\boldsymbol{x}$ from the marginal of D and output $(\boldsymbol{x}, 1)$.
- The activation function is $\sigma^{\text{sig}}_{\text{clipped}}$.

Claim 38. Let $\widehat{\boldsymbol{v}} = (\widehat{\boldsymbol{w}}, \widehat{\tau})$ be obtained by running $\mathcal{A}^{\sigma^{\text{sig}}_{\text{clipped}}}_{\text{GLM}}(4\varepsilon, w_{\max}, D_{\text{clipped}})$. Then, with probability at least 0.99, $(\widehat{\boldsymbol{w}}, \widehat{\tau})$ satisfies the equation

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim D}\left[\left(\sigma^{\text{sig}}(\boldsymbol{w}^* \cdot \boldsymbol{x} + \tau^*) - \sigma^{\text{sig}}(\widehat{\boldsymbol{w}} \cdot \boldsymbol{x} + \widehat{\tau})\right)^2\right] \leq C'\varepsilon\,. \tag{59}$$

for universal constant C'.

Proof. We see that the choice of $\sigma = \sigma^{\text{sig}}_{\text{clipped}}$ satisfies the requirements on σ in Theorem 35. Indeed, it holds that : (1) $\sigma^{\text{sig}}_{\text{clipped}}$ is monotone non-decreasing, (2) the function $\sigma^{\text{sig}}_{\text{clipped}}$ is 1-Lipschitz, and (3) by construction, we have $\sigma^{\text{sig}}_{\text{clipped}}(z) = \text{sign}(z)$ whenever z is not in $[-\rho, \rho]$. Thus, we have that

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim D_X}\left[\left(\sigma^{\text{sig}}_{\text{clipped}}(\boldsymbol{w}^* \cdot \boldsymbol{x} + \tau^*) - \sigma^{\text{sig}}_{\text{clipped}}(\widehat{\boldsymbol{w}} \cdot \boldsymbol{x} + \widehat{\tau})\right)^2\right].$$

Recall from Equation (58) that $\max_{t\in\mathbb{R}} |\sigma^{\text{sig}}(t) - \sigma^{\text{sig}}_{\text{clipped}}(t)| \leq e^{-\rho/2}$. Thus, we have

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim D_X}\left[\left(\sigma^{\text{sig}}(\boldsymbol{w}^* \cdot \boldsymbol{x} + \tau^*) - \sigma^{\text{sig}}(\widehat{\boldsymbol{w}} \cdot \boldsymbol{x} + \widehat{\tau})\right)^2\right] \leq 2\epsilon + 2e^{-\rho} \leq C'\epsilon$$

for $\rho = C\log(\Delta/\epsilon)$. □

Let $\tilde{D}$ be the distribution formed from D by concatenating one to the marginal. A sample from $\tilde{D}$ is generated as: sample $(\boldsymbol{x}, y)$ from D and output $((\boldsymbol{x}, 1), y)$. We compare the distributions $\tilde{D}$ and D_{clipped}. First, for every specific z, the TV distance between a pair of $\{\pm 1\}$-valued random variables with expectations $\sigma^{\text{sig}}_{\text{clipped}}(z)$ and $\sigma^{\text{sig}}(z)$ respectively is bounded by $\left|\sigma^{\text{sig}}(z) - \sigma^{\text{sig}}_{\text{clipped}}(z)\right|/2$, which is at most $e^{-\rho}$ by Equation 58. Comparing this with the definitions of $\tilde{D}$ and D_{clipped} we see that

$$\text{dist}_{\text{TV}}(\tilde{D}, D_{\text{clipped}}) \leq \frac{1}{2}\max_{z\in\mathbb{R}}\left|\sigma^{\text{sig}}_{\text{clipped}}(z) - \sigma^{\text{sig}}(z)\right| \leq e^{-\rho}.$$

Theorem 35 tells us that the algorithm $\mathcal{A}^{\sigma^{\text{sig}}_{\text{clipped}}}_{\text{GLM}}$ uses at most $O\left(\frac{\Delta\rho^2}{\varepsilon^2}\ln\frac{\Delta\rho}{\varepsilon}\right)$ samples from the distribution it accesses. This, together with the data-processing inequality and the bound above, lets us conclude that

$$\begin{aligned}\text{dist}_{\text{TV}}\left(\mathcal{A}^{\sigma^{\text{sig}}_{\text{clipped}}}_{\text{GLM}}(\varepsilon, w_{\max}, \rho, \tilde{D}), \mathcal{A}^{\sigma^{\text{sig}}_{\text{clipped}}}_{\text{GLM}}(\varepsilon, w_{\max}, \rho, D_{\text{clipped}})\right) &\leq O\left(\frac{\Delta\rho^2}{\varepsilon^2}\ln\frac{\Delta\rho}{\varepsilon}\right)\text{dist}_{\text{TV}}(\tilde{D}, D_{\text{clipped}}) \\ &\leq O\left(\frac{\Delta\rho^2}{\varepsilon^2}\ln\frac{\Delta\rho}{\varepsilon}\right)\cdot e^{-\rho}.\end{aligned}$$

Substituting $\rho = C(\log\frac{\Delta}{\varepsilon} + 1)$, we see that for sufficiently large absolute constant C the expression above is at most 0.05. Combining this with Claim 38, we see that the vector $(\widehat{\boldsymbol{w}}, \widehat{\tau})$ given by $\mathcal{A}^{\sigma^{\text{sig}}_{\text{clipped}}}_{\text{GLM}}(\varepsilon, \rho, w_{\max}, \tilde{D})$ with probability at least 0.95 satisfies Equation (59). To complete the proof, we observe that samples from $\tilde{D}$ can be easy generated given samples from D. □