Agnostic PAC Learning of k-juntas Using L_2-Polynomial Regression

Mohsen Heidari
Indiana University, Bloomington

Wojciech Szpankowski
Purdue University, West Lafayette

Abstract

Many conventional learning algorithms rely on loss functions other than the natural 0-1 loss for computational efficiency and theoretical tractability. Among them are approaches based on absolute loss ($\mathcal{L}_1$ regression) and square loss ($\mathcal{L}_2$ regression). The first is proved to be an *agnostic* PAC learner for various important concept classes such as *juntas*, and *half-spaces*. On the other hand, the second is preferable because of its computational efficiency which is linear in the sample size. However, PAC learnability is still unknown as guarantees have been proved only under distributional restrictions. The question of whether $\mathcal{L}_2$ regression is an agnostic PAC learner for 0-1 loss has been open since 1993 and yet has to be answered.

This paper resolves this problem for the junta class on the Boolean cube — proving agnostic PAC learning of k-juntas using $\mathcal{L}_2$ polynomial regression. Moreover, we present a new PAC learning algorithm based on the Boolean Fourier expansion with lower computational complexity. Fourier-based algorithms, such as Linial et al. (1993), have been used under distributional restrictions, such as uniform distribution. We show that with an appropriate change, one can apply those algorithms in agnostic settings without any distributional assumption. We prove our results by connecting the PAC learning with 0-1 loss to the minimum mean square estimation (MMSE) problem. We derive an elegant upper bound on the 0-1 loss in terms of the MMSE error. Based on that, we show that the sign of the MMSE is a PAC learner for any concept class containing it.

Proceedings of the 26[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

1 Introduction

To gain computational efficiency or analytic tractability, many conventional learning methods such as support-vector machine (SVM) rely on intermediate loss functions other than the natural 0-1 loss. Absolute difference ($\mathcal{L}_1$ distance) is an example. It is known that polynomial regression under $\mathcal{L}_1$ distance leads to *agnostic probably approximately correct* (PAC) learners (Kalai et al., 2008) for various hypothesis classes such as k-juntas, *polynomial-approximated* predictors, and *half-spaces*. However, the running time of computing $\mathcal{L}_1$ distance is quadratic in sample size and hence prohibitive for large data sets.

Square loss ($\mathcal{L}_2$ distance), on the other hand, is an alternative with computational complexity linear in the size of the data. This has been an incentive to use learning algorithms such as the *low-degree* algorithm (Linial et al., 1993) and LS-SVM (Suykens and Vandewalle, 1999). From the learning theoretic perspective, PAC learning using $\mathcal{L}_2$-based approaches has been studied for the aforementioned concept classes, but with distributional assumptions (Linial et al., 1993; Kalai et al., 2008; Jackson, 2006).

For instance, under the *realizability* assumption, where zero generalization loss is possible ($opt = 0$), the $\mathcal{L}_2$-polynomial regression is a PAC learner. In addition to the realizability assumption, under the uniform input distribution, the low-degree algorithm is also a PAC learner (Mossel et al., 2004, 2003; Blais et al., 2010). Under the distribution-free (*agnostic)* setting, PAC bounds of the form $c(opt)$ with *opt* being the minimum loss of the class and c a constant as high as $c = 8$, have been proved so far for various concept classes (Kalai et al., 2008; Kearns et al., 1994; Jackson, 2006). Therefore, *agnostic* PAC learnability of $\mathcal{L}_2$-based approaches is still open and yet to be determined.

This paper resolves this problem for learning k-juntas on the Boolean cube, i.e., Boolean functions over d inputs whose output depends on at most $k < d$ variables, where k is typically a constant much smaller than d. Learning juntas has been studied extensively in the literature with various motivations such as feature selection in machine learning (Guyon and Elisseeff, 2003; Blais et al., 2010; Heidari et al., 2021b; Kalai et al., 2008; Klivans et al., 2009; Birnbaum and Shwartz, 2012; Diakonikolas et al., 2019).

We prove that agnostic PAC learning is possible using $\mathcal{L}_2$-polynomial regression for k-juntas. Moreover, we present a more efficient variant of $\mathcal{L}_2$ regression using a Boolean Fourier expansion. We show that this algorithm is also an agnostic PAC learner with respect to k-juntas. This result implies that Fourier algorithms such as the low-degree algorithm of Linial et al. (1993) that were initially designed for uniform distribution also apply to agnostic settings.

1.1 Summary of the Contributions

Learning k-juntas with least square regression: The focus of this paper is PAC learning of k-junta class, on Boolean inputs, using $\mathcal{L}_2$-regression and with the usual 0-1 loss. Following the standard PAC learning model, the training set contains n samples $\{(\mathbf{x}(i), y(i))\}_{i=1}^{n}$ with feature-vectors $\mathbf{x}(i) \in \{-1,1\}^d$ and binary labels $y(i) \in \{-1,1\}$. The objective of the $\mathcal{L}_2$-polynomial regression is to minimize the empirical square loss between the target label y and a polynomial $p(\mathbf{x})$ of degree up to k. Given such a polynomial, a predictor g is created by simply taking the sign of this polynomial as $g(\mathbf{x}) = \text{sign}[p(x)]$.

The first main result of this paper shows that $\mathcal{L}_2$ polynomial regression agnostically PAC learns k-juntas. More precisely, with probability at least $(1-\delta)$, the generalization loss of the predictor g is within a small deviation of the optimal loss among all k-juntas, i.e., $\mathbb{P}\{Y \neq g(\mathbf{X})\} \leq opt+\epsilon$, with *opt* being the optimal loss in k-junta class. More formally, we prove the following theorem.

Theorem 1 (abbreviated). *Given $k \leq d$, there is an algorithm based on $\mathcal{L}_2$-polynomial regression with degree limit k (Algorithm 1) that agnostically PAC learns k-juntas with sample complexity n up to $O(\frac{k2^k}{\epsilon^2}\log\frac{d}{\delta\epsilon^2})$ and computational complexity $O(nd^{\Theta(k)})$.*

We note the computational complexity of learning k-juntas with the $\mathcal{L}_1$-polynomial regression is $O(n^2 d^{(3+\omega)3k})$ which is worse for large n.

One of the main technical challenges in proving PAC bounds with $\mathcal{L}_1$ or $\mathcal{L}_2$ regression is analyzing the connections between the 0-1 loss and the square or absolute loss. Conventional results for $\mathcal{L}_2$ rely on the inequality $\mathbb{1}\{y \neq \text{sign}[p(x)]\} \leq (y-p(x))^2$ that holds for $y \in \{-1, 1\}$. Based on this bound, the PAC bound $8opt$ is derived (Linial et al., 1993). Hence, this raises the question as to whether taking the sign is optimal in $\mathcal{L}_2$-based PAC learning. When $x \in \{-1,1\}^d$, Blum *et al.* (Blum et al., 1994) and Jackson (Jackson, 2006) proposed a clever idea of randomized rounding instead of taking the sign. As a result, they improved the factor from $8opt$ to $2opt$. In Section 2.1, we argue that these bounds are loose, at least for binary inputs. We prove new bounds connecting the 0-1 loss and the square loss (Lemma 3 for binary input and Lemma 6 for real-valued inputs). Using these results, we show that for k-junta class, taking the sign is not problematic and gives *opt*, hence $\mathcal{L}_2$-based agnostic PAC learnability. Moreover, we improve the factor $8opt$ to $2opt$ for more general classes with $x \in \mathbb{R}^d$.

Our approach relies on a framework using vector spaces equipped with probability measures as a proxy to derive PAC learning bounds. Among others, we consider a joint vector space for functions on the feature-label set $\mathcal{X} \times \mathcal{Y}$, incorporating the sample-label relation and the underlying joint distribution D. This approach establishes our results by connecting the PAC learning model and powerful tools for analyzing vector spaces. Notably, we prove an elegant upper bound on the 0-1 loss based on amenable quantities such as 1-norm and 2-norm (see Corollary 1 and 3 in Section 3). A notable feature of our approach is that the expressions are quite compact and insightful.

Learning with Fourier algorithm: In addition, we present another more efficient algorithm for binary-valued samples. This algorithm's running time is linear in n and scales with d^k which is asymptotically better than the two other approaches as they grow with $d^{O(k)}$. Our result relies on the Boolean Fourier expansion defined for the uniform distribution (Wolf, 2008; O'Donnell, 2014). We prove a counter-intuitive result by showing that the uniform Boolean Fourier is in fact applicable to agnostic distribution-free settings. Motivated by Linial's low-degree algorithm (Linial et al., 1993) on uniform distribution, we develop a Fourier algorithm that performs $\mathcal{L}_2$ polynomial regression more efficiently and without any distributional assumption. We then show that this algorithm also agnostically PAC learns the k-junta class. More formally, we prove the following statement.

Theorem 2 (abbreviated). *Given $k < d$, the Fourier algorithm (Algorithm 2) agnostically PAC-learns k-juntas with sample complexity $O(\frac{k2^k}{\epsilon^2}\log\frac{d}{\delta})$ and computational complexity $O(\frac{nkd^k}{(k-1)!})$.*

Table 1 compares various PAC learning algorithms in terms of their sample complexity, running time, and PAC loss. The $\mathcal{L}_2$ and Fourier algorithms have lower sample and computational complexities when compared to the other methods. When compared to (Kalai et al., 2008) using $\mathcal{L}_1$-polynomial regression, we obtain a lower sample complexity and computational complexity. Note that the running time of $\mathcal{L}_1$ regression grows with $O(n^2 d^{O(k)})$, which is quadratic in sample size n and hence prohibitive in large data sets. The running time of $\mathcal{L}_2$ regression is $O(nd^{O(k)})$ which is linear in n. Lastly, the running time of the Fourier algorithm grows with $O(nd^k)$, which is a better exponent than $\mathcal{L}_2$. Overall, given that k is typically a constant independent of d, the $\mathcal{L}_2$ regression and the Fourier algorithm are suitable for large data sets. Lastly, we present a lower bound on the sample complexity of the k-junta class. Based on the standard VC-dimension argument

Table 1: Comparison of the PAC-learning algorithms for k-juntas.

Algorithm	Sample Cmplx.	Comp. Cmplx.	PAC Error
Brute force ERM	$O(\frac{k2^k}{\epsilon^2}\log\frac{d}{\delta})$	$O(nd^k 2^{2^k})$	$opt+\epsilon$
$\mathcal{L}_1$-Poly. Reg. (Kalai et al., 2008)	$O(\frac{1}{\epsilon}k^{\Theta(k)}\log\frac{d}{\delta})$	$O(n^2 d^{(3+\omega)3k})$	$opt+\epsilon$
$\mathcal{L}_2$-Poly. Reg.	$O(\frac{k2^k}{\epsilon^2}\log\frac{d}{\epsilon^2\delta})$	$O(nd^{\Theta(k)})$	• $2opt+\epsilon$ (Jackson, 2006) • $opt+\epsilon$, [**Thm. 1**]
Low-degree Alg. (*uniform dist.*) (Linial et al., 1993)	$O(k2^k\log\frac{d}{\delta})$	$O(\frac{nkd^k}{(k-1)!})$	• $8opt+\epsilon$ (Linial et al., 1993) • $2opt+\epsilon$ (Jackson, 2006) • $\frac{1}{4}+opt(1-opt)+\epsilon$ (Kearns et al., 1994) • $opt+\epsilon$, [**Thm. 2**]
Stochastic Fourier (**Algorithm 2**)	$O(\frac{k2^k}{\epsilon^2}\log\frac{d}{\delta})$	$O(\frac{nkd^k}{(k-1)!})$	$opt+\epsilon$, [**Thm. 2**]

which gives $O(\frac{1}{\epsilon^2}(VC+\log(\frac{1}{\delta}))$. The exact expression for the VC dimension of the k-junta class is unknown, but it is between 2^k and $2^k + k\log d$.

1.2 Related Works

The problem of learning juntas is a classical problem in machine learning. There is a large body of work on learning and testing of juntas (Mossel et al., 2004; Bshouty and Costa, 2016; Liu et al., 2019; Arpe and Mossel, 2008; Fischer et al., 2004; Servedio et al., 2015; De et al., 2019; Vempala and Xiao, 2011; Chen et al., 2021; Iyer et al., 2021). Juntas are of significant interest in learning theory as they are connected to other fundamental problems such as learning with feature selection (Guyon and Elisseeff, 2003), DNF formulas, and decision trees (Mossel et al., 2004). Particularly, learning with feature selection can be expressed as learning k-juntas (with k out of d features). Additionally, every k-junta is implemented by a decision tree or DNF formula of size 2^k and conversely, any size-k decision tree is also a k-junta, and any k-term DNF is ϵ-approximated by a $k\log(\frac{k}{\epsilon})$-junta. Hence, obtaining efficient algorithms for these problems is closely related to learning juntas (Mossel et al., 2004). PAC learning with respect to k-juntas has been studied using various approaches. We briefly review the approaches for learning these concept classes below and summarize them in Table 1.

Naive Empirical Risk Minimization (ERM): This is the usual exhaustive search over all predictors to minimize the empirical loss. For k-juntas, ERM is an agnostic PAC learning algorithm with sample complexity $O(\frac{k2^k}{\epsilon^2}\log\frac{d}{\delta})$ and computational complexity $O(nd^k 2^{2^k})$ (Shalev-Shwartz and Ben-David, 2014). With the computational complexity of doubly exponential with respect to k, ERM is prohibitive even for small values of k.

Learning with $\mathcal{L}_1$ Regression. Kalai *et al.* (Kalai et al., 2008) introduced polynomial regression as an approach for PAC learning with the $0-1$ loss function. They showed that $\mathcal{L}_1$-Polynomial regression agnostically PAC learns with respect to (k,ϵ)-concentrated hypothesis class which includes k-juntas. Adopting this algorithm to k-juntas requires a sample complexity $O(\frac{1}{\epsilon}d^{\Theta(k)})$. With a *linear programming* implementation, the computational complexity of this algorithm is $O(n^2 d^{(3+\omega)3k})$, where $\omega < 2.4$ is the matrix-multiplication exponent. The quadratic growth of the computational complexity of this approach makes it expensive for large sample sizes. This motivates us to study $\mathcal{L}_2$ based approaches.

Learning with $\mathcal{L}_2$ Polynomial Regression. This approach is similar to its $\mathcal{L}_1$ counterpart with absolute error replaced by the square loss. Fast implementations of $\mathcal{L}_2$ regression with linear complexity in sample size have been studied (Drineas et al., 2006, 2010). PAC learning using this approach has been studied in (Kalai et al., 2008; Jackson, 2006). In the agnostic setting, it is shown that this approach is a *weak learner* with error $8opt$. With the use of a nondeterministic rounding proposed in (Blum et al., 1994; Jackson, 2006), the PAC bound can be reduced to $2opt$. This paper shows that for k-juntas opt is obtained without any randomized rounding. For other non-binary classes, in Section 4, we prove the bound $2opt$.

Fourier Algorithms. This approach is viewed as a special solution for $\mathcal{L}_2$ regression. Linial *et al.* (Linial et al., 1993) investigated PAC learning from an alternative perspective and introduced the well-known "Low-Degree Algorithm". They provide theoretical guarantees under the *uniform* and *known* distribution on $\{-1,1\}^d$ of the samples. The low-degree is based on the Fourier expansion on the Boolean cube. Although computationally efficient, this algorithm has limited practical applications due to its distributional restrictions — uniform (and known) distribution is unrealistic in many applications. Furst *et al.* (Furst et al., 1991) relaxed such a distributional restriction by adopting a low-degree algorithm for learning AC^0 functions under the product probability distributions. The Fourier expansion has been used to analyze Boolean functions (Wolf, 2008; O'Donnell, 2014) with a wide range of applications,

Algorithm 1: $\mathcal{L}_2$-Algorithm

Input: Training samples $\mathcal{S}_n = \{(\mathbf{x}(i), y(i))\}_{i=1}^n$, degree parameter k.

for *each subset* $\mathcal{J} \subseteq [d]$ *with* $|\mathcal{J}| = k$ **do**

  Find a polynomial $\hat{p}_{\mathcal{J}}$ of degree up to k that minimizes $\frac{1}{n}\sum_i \big(y(i) - p(\mathbf{x}(i))\big)^2$.

Select $\hat{p}$ as the $\hat{p}_{\mathcal{J}}$ that has the smallest square loss.

return $\hat{g} \equiv \text{sign}[\hat{p}]$.

namely computational learning (Linial et al., 1993; Mossel et al., 2004), noise sensitivity (O'Donnell, 2014; Kalai, 2005; Li and Médard, 2018; Heidari et al., 2019), approximation (Blais et al., 2010), feature selection (Heidari et al., 2021b), and other information-theoretic problems (Courtade and Kumar, 2014; Weinberger and Shayevitz, 2017, 2018; Heidari et al., 2021a). In this work, we also generalize this approach for *agnostic* PAC learning — hence, removing the distributional assumptions.

2 Formulations and Main Results

Model: We use the usual formulation of *agnostic* PAC learning model (Valiant, 1984; Kearns et al., 1994). The focus of this paper is on binary classification with the 0-1 loss. Available is a set of n labeled samples $\mathcal{S}_n = \{(\mathbf{x}(i), y(i))\}_{i=1}^n$ generated independent and identically distributed (i.i.d.) from an unknown but fixed probability distribution D. The generalization loss of a predictor g is given by $\mathcal{L}_D(g) := \mathbb{P}_D\{Y \neq g(\mathbf{X})\}$. An algorithm agnostically PAC learns a hypothesis class $\mathcal{H}$, if, for any ϵ, $\delta \in (0,1)$, and given $n > n(\epsilon, \delta)$ training samples drawn from any distribution D, it outputs with probability $(1-\delta)$ a predictor g whose expected loss is at most $opt + \epsilon$, where opt is the minimum loss in $\mathcal{H}$.

Notation: For any natural number d, the set $\{1, 2, \cdots, d\}$ is denoted by $[d]$. For a pair of functions f, g on $\mathcal{X}$, the notation $f \equiv g$ means that $f(x) = g(x)$ for all $x \in \mathcal{X}$. For any function $h : \mathcal{Z} \to \mathbb{R}$ and input distribution D, the 1-norm and 2-norm are defined as $\|h\|_{1,D} := \mathbb{E}_D[|h(Z)|]$ and $\|h\|_{2,D} := \sqrt{\mathbb{E}_D[h(Z)^2]}$, respectively.

2.1 Warm-Up

We start with highlighting one of the main difficulties in proving PAC bounds with $\mathcal{L}_1$ or $\mathcal{L}_2$ regression. The main challenge is the analysis of the 0-1 loss after taking the sign of the resulting polynomial. For that, one needs to study the relations between the 0-1 loss and the square or absolute loss. To see this, let p be the polynomial minimizing the square loss. Then, it is not difficult to see that $\mathbb{1}\{y \neq \text{sign}[p(x)]\} \leq (y - p(x))^2$, where $y \in \{-1, 1\}$. As a result, the 0-1 loss of sign$[p]$ is bounded as $\mathbb{P}\{Y \neq \text{sign}[p(\mathbf{X})]\} \leq \mathbb{E}[(Y - p(\mathbf{X}))^2]$. This is a loose bound that leads to a PAC bound of $8opt$. To see the argument, let f be the optimal predictor with the 0-1 loss opt. Additionally, suppose f is approximated by a polynomial $\tilde{p}$ with the square error less than ϵ^2. Then, we can write

$$\begin{aligned}\mathbb{P}\{Y \neq \text{sign}[p(\mathbf{X})]\} &\leq \mathbb{E}[(Y - p(\mathbf{X}))^2] \\ &\overset{(a)}{\leq} \mathbb{E}[(Y - \tilde{p}(\mathbf{X}))^2] \\ &\overset{(b)}{\leq} 2\mathbb{E}[(Y - f(\mathbf{X}))^2 + (f(\mathbf{X}) - \tilde{p}(\mathbf{X}))^2] \\ &\overset{(c)}{\leq} 8opt + 2\epsilon^2,\end{aligned}$$

where (a) follows as p is the optimal polynomial, (b) holds from the AM-GM inequality, and (c) holds as $\mathbb{1}\{y \neq f(x)\} = \frac{1}{4}(y - f)^2$ for any $f : \mathcal{X} \to \{-1, 1\}$.

These observations raise the question of whether taking the sign is optimal in PAC learning. When $x \in \{-1, 1\}^d$, Blum *et al.* (Blum et al., 1994) and Jackson (Jackson, 2006) proposed a clever idea of randomized rounding instead of taking the sign. As a result, they improved the factor from $8opt$ to $2opt$.

In Lemma 3, we prove a tighter bound between the 0-1 loss and the square loss. Using this lemma, we prove in Theorem 1 that for k-junta class taking the sign is optimal and results in opt (i.e., agnostic PAC learnability). Moreover, we develop a more general analysis and show that sign of the MMSE of Y given the observation $\mathbf{X}$ give a PAC learner, see Theorem 3.

2.2 Learning with $\mathcal{L}_2$-Polynomial Regression

We employ a PAC learning algorithm using $\mathcal{L}_2$-polynomial regression. Given a training set, the objective of the polynomial regression is to minimize the empirical square loss over all polynomials of degrees up to k. This process can be implemented by stochastic gradient descent or solving a linear equations system. Based on this regression, one can study PAC learning of various concept classes. In this paper, we consider k-juntas.

k-junta class: A k-junta is a Boolean function $h : \{-1, 1\}^d \to \{-1, 1\}$ with d input variables whose output depends on at most k out of d inputs.

For k-junta classes, we use a variant of $\mathcal{L}_2$-polynomial regression (see Algorithm 1) that has the same computational complexity as compared to the vanilla $\mathcal{L}_2$ polynomial regression. With this approach, we establish the following theorem.

Theorem 1. *Algorithm 1, with a degree limit of $k \leq d$, agnostically PAC learns k-juntas. More precisely, given $\delta \in [0, 1]$, with probability $(1 - \delta)$, its generalization loss*

does not exceed the following

$$\text{opt} + O\Big(\sqrt{\frac{2^k + k\log d}{n}\log\frac{n}{2^k + k\log d}}\Big) + \sqrt{\frac{\log(1/\delta)}{2n}},$$

where n is the number of samples. Furthermore, the resulting computational complexity is $O(nd^{\Theta(k)})$.

By simplifying the above expression, we get a sample complexity bound of $n(\delta,\epsilon) = O(\frac{k2^k}{\epsilon^2}\log\frac{d}{\epsilon^2\delta})$. The proof is presented in Section 3.3.

The polynomial regression procedure in Algorithm 1 can be implemented via a linear L_2 regression in $\mathbb{R}^D$, where $D = d^k$. The factor d^k is because a polynomial of degree up to k is a linear combination of monomials of the form $\prod_{j=1}^{k} X_{i_j}$, where $i_j \in [d]$. Linear regression can be implemented via Moore-Penrose (generalized) inverse. The generalized inverse is computed using classical methods in $O(nD^2 + D^3)$. Hence, given that $n > D$, we can perform polynomial regression in $O(nD^2) = O(nd^{2k})$. However, we note that under special cases (e.g., $d^k = \lceil n^r \rceil$) the regression can be done in $O(nd^{k\omega(r)})$, where $\omega(r)$ is a constant given in (Gall and Urrutia, 2018). Hence, the computational complexity of this algorithm is $O(nd^{\Theta(k)})$ as noted in Table 1.

2.3 Fourier-Based Learning Algorithm

We present another $\mathcal{L}_2$-based approach that is computationally more efficient than the $\mathcal{L}_2$-polynomial regression. The computational cost of the $\mathcal{L}_2$ regression grows as $O(nd^{\Theta(k)})$ which is more efficient than its $\mathcal{L}_1$ variant with complexity $O(n^2d^{(3+\omega)3k})$. This leads to the question as to whether the factor $d^{\Theta(k)}$ can be further reduced. We answer this question using a Fourier analysis on the Boolean cube. Particularly, we present an algorithm with the complexity of $O(\frac{nkd^k}{(k-1)!})$.

Our solution is based on the Boolean Fourier expansion applied to the uniform distribution on the Boolean cube (O'Donnell, 2014; Wolf, 2008). Surprisingly, we plan to use this Fourier for agnostic settings. Let us briefly explain the standard Boolean Fourier expansion.

Fact 1 (Boolean Fourier)**.** *Any (bounded) function $f : \{-1,1\}^d \to \mathbb{R}$ admits the following decomposition*

$$f(\mathbf{x}) = \sum_{\mathcal{S}\subseteq[d]} \mathsf{f}_{\mathcal{S}}\,\chi_{\mathcal{S}}(\mathbf{x}), \quad \forall \mathbf{x} \in \{-1,1\}^d,$$

where $\chi_{\mathcal{S}}(\mathbf{x})$ is the monomial corresponding to the subset $\mathcal{S} \subseteq [d]$ and is defined as $\chi_{\mathcal{S}}(\mathbf{x}) = \prod_{j\in\mathcal{S}} x_j$. Further, the coefficients $\mathsf{f}_{\mathcal{S}} \in \mathbb{R}$ are called the Fourier coefficients of f and are calculated as

$$\mathsf{f}_{\mathcal{S}} = \frac{1}{2^d}\sum_{\mathbf{x}} f(\mathbf{x})\chi_{\mathcal{S}}(\mathbf{x}), \quad \forall \mathcal{S} \in [d]$$

This expansion relies on the restriction that the input variables are uniformly distributed over the Boolean cube. This limits the applications of Fourier-based algorithms such as (Linial et al., 1993) to *agnostic* learning problems without any distributional assumptions. This issue can be resolved via a Gram-Schmidt-type orthogonalization process that yields a generalized Boolean Fourier expansion (Heidari et al., 2021a).

However, in this paper, we take a slightly different path and propose a simple adjustment to the standard Boolean Fourier that applies to certain agnostic problems. Hence, we get PAC learnability together with computational efficiency. In what follows, we describe this adjustment.

Let D_X be any probability distribution on $\{-1,1\}^d$ and f be a Boolean function. Define

$$f_{\mathcal{S}} := \frac{1}{2^d}\sum_{\mathbf{x}} f(\mathbf{x})D_X(\mathbf{x})\chi_{\mathcal{S}}(\mathbf{x}).$$

From Fact 1, $f_{\mathcal{S}}$ is the Fourier coefficient of the real-valued function $f(\mathbf{x})D_X(\mathbf{x})$. Note that under the uniform D_X, $f_{\mathcal{S}} = \frac{1}{2^d}\mathsf{f}_{\mathcal{S}}$, where $\mathsf{f}_{\mathcal{S}}$ is the Fourier coefficient of $f(\mathbf{x})$ as in Fact 1. In agnostic settings where D_X is unknown, $f_{\mathcal{S}}$ is not accessible. However, we can estimate it empirically.

Before explaining the estimation, let us introduce another extension. In agnostic settings, the label y is not necessarily a function of the features $\mathbf{x}$. Hence, to make the Fourier expansion applicable to agnostic PAC, we expand it, beyond deterministic function, to stochastic mappings:

Consider a random vector $\mathbf{X}$ and a labeling variable Y. Let $(\mathbf{X}, Y) \sim D$ where D is a probability distribution over $\{-1,1\}^d \times \{-1,1\}$. Then the stochastic Fourier coefficients are defined as

$$a_{\mathcal{S}} := \frac{1}{2^d}\mathbb{E}[Y\chi_{\mathcal{S}}(\mathbf{X})], \tag{1}$$

for all $\mathcal{S} \subseteq [d]$. If $Y = f(\mathbf{X})$, then $a_{\mathcal{S}} = f_{\mathcal{S}}$. Given the i.i.d. samples $\{x(i), y(i)\}_{i=1}^n$, the empirical estimation of $a_{\mathcal{S}}$ is

$$\hat{a}_{\mathcal{S}} := \frac{1}{2^d}\frac{1}{n}\sum_{i=1}^{n} y(i)\chi_{\mathcal{S}}(\mathbf{x}(i)). \tag{2}$$

Note that the estimation is agnostic to the underlying distribution D, but we show that it converges to $a_{\mathcal{S}}$.

Lemma 1. *Let D be any probability distribution on $\{-1,1\}^{d+1}$. Let $\mathcal{S}_1, \mathcal{S}_2, \cdots, \mathcal{S}_m$ be m subsets of $[d]$. Given $\delta \in (0,1)$ and n samples drawn i.i.d. from D, the inequality*

$$\sup_{1\le j\le m} |\hat{a}_{\mathcal{S}_j} - a_{\mathcal{S}_j}| \le \frac{1}{2^d}\sqrt{\frac{1}{2n}\log\frac{2m}{\delta}}$$

holds with probability at least $(1-\delta)$, where $a_{\mathcal{S}}$ and $\hat{a}_{\mathcal{S}}$ are given in (1) and (2), respectively.

Algorithm 2: Stochastic Fourier

Input: Training samples $\mathcal{S}_n = \{(\mathbf{x}(i), y(i))\}_{i=1}^n$, degree parameter k.

Output: Predictor $\hat{g}$

1 For each $\mathcal{S} \subseteq [d]$ with at most k elements compute the empirical Fourier coefficients as $\hat{a}_{\mathcal{S}} = \frac{1}{2^d}\frac{1}{n}\sum_{i=1}^n y(i)\prod_{j\in\mathcal{S}} x_j(i)$.

2 For each $\mathcal{J} \subseteq [d]$ with k elements construct the function $\hat{f}^{\mathcal{J}}(\mathbf{x}) = \sum_{\mathcal{S}\subseteq\mathcal{J}} \hat{a}_{\mathcal{S}}\chi_{\mathcal{S}}(\mathbf{x})$.

3 Find $\hat{\mathcal{J}}$ with the minimum empirical loss of $\text{sign}[\hat{f}^{\mathcal{J}}]$.

4 **return** $\hat{g} \equiv \text{sign}[\hat{f}^{\hat{\mathcal{J}}}]$.

Proof. Observe that for any $\mathcal{S} \subseteq [d]$

$$\begin{aligned}\mathbb{E}_{\mathcal{S}_n\sim D^n}[\hat{a}_{\mathcal{S}}] &= \frac{1}{2^d}\mathbb{E}[Y(1)\chi_{\mathcal{S}}(\mathbf{X}(1))] \\ &= \frac{1}{2^d}\sum_{\mathbf{x},y} D(\mathbf{x},y)y\,\chi_{\mathcal{S}}(\mathbf{x}) = a_{\mathcal{S}}.\end{aligned}$$

By taking the factor $\frac{1}{2^d}$ in the definition of $a_{\mathcal{S}}$ and $\hat{a}_{\mathcal{S}}$, we have that

$$|\hat{a}_{\mathcal{S}} - a_{\mathcal{S}}| = \frac{1}{2^d}\Big|\frac{1}{n}\sum_{i=1}^n y(i)\chi_{\mathcal{S}}(\mathbf{x}(i)) - \mathbb{E}[Y\chi_{\mathcal{S}}(X)\Big|.$$

We apply McDiarmid inequality to bound the right-hand side term. It is not difficult to check that

$$\mathbb{P}\Big\{\Big|\frac{1}{n}\sum_{i=1}^n y(i)\chi_{\mathcal{S}}(\mathbf{x}(i)) - \mathbb{E}[Y\chi_{\mathcal{S}}(X)\Big| \geq \epsilon\Big\} \leq 2e^{-\frac{n\epsilon^2}{2}}.$$

Therefore, by considering the factor $\frac{1}{2^d}$, from the union bound, and by equating the right-hand side to δ, we establish the lemma. □

With this approach, our Fourier algorithm (See Algorithm 2) performs a polynomial regression in the Fourier domain by estimating the Fourier coefficients of the label from the training samples. In the following theorem, we present a PAC bound for learning k-juntas using this approach.

Theorem 2. *The Fourier algorithm agnostically learns k-juntas for $k \leq d/2$ and with error less than*

$$\text{opt} + O\Big(\sqrt{\frac{2^k}{n}\log\frac{d^k}{(k-1)!\delta}}\Big),$$

with probability at least $(1-\delta)$, where n is the number of samples. Moreover, the resulted computational complexity is $O(\frac{nkd^k}{(k-1)!})$.

In the next section, we discuss our main ideas. The proof of this theorem is given in Section 3.5.

3 Main Technical Results

The main results of this paper rely on a fundamental connection between square loss and the 0-1 loss presented as Corollary 1 and 3 in Section 3.2. In this section, we present this connection and describe the steps in proving Theorem 1 and 2.

3.1 A Vector Space Representation

We introduce a vector representation incorporating the feature-label distribution. Such representation is a proxy to use powerful algebraic tools developed for vector spaces. In what follows, we describe this representation.

Let $\mathcal{X}$ denote the input set, $\mathcal{Y} = \{-1, 1\}$ be the label set, and D be the underlying distribution on $\mathcal{X}\times\mathcal{Y}$. Consider the vector space of all functions $h : \mathcal{X}\times\mathcal{Y} \mapsto \mathbb{R}$ for which $\mathbb{E}_D[h(\mathbf{X}, Y)^2]$ is finite[1]. Naturally, the inner product between two functions h_1, and h_2 is defined as

$$\langle h_1, h_2\rangle_D \triangleq \mathbb{E}_D[h_1(\mathbf{X}, Y)h_2(\mathbf{X}, Y)].$$

With this formulation, the true labeling is simply the function $(\mathbf{x}, y) \mapsto y$. Note that this complies with the agnostic setting, where the label is not necessarily a function of $\mathbf{x}$. In addition, a predictor g in the learning model is viewed as the mapping $(\mathbf{x}, y) \mapsto g(\mathbf{x})$. Since $\mathcal{Y} = \{-1, 1\}$, then the 0-1 loss of any predictor g can be written as

$$\mathcal{L}_D(g) = \frac{1}{2} - \frac{1}{2}\langle Y, g\rangle_D = \frac{1}{4}\|Y - g\|_{2,D}^2, \qquad (3)$$

where, with slight abuse of notation, Y and g are understood as the mappings $(\mathbf{x}, y) \mapsto y$ and $(\mathbf{x}, y) \mapsto g(\mathbf{x})$, respectively. This first equality in (3) is because of the identity $\mathbb{1}\{a \neq b\} = \frac{1}{2}(1 - ab)$ for any $a, b \in \{-1, 1\}$. The second equality is from the definition of 2-norm and the fact that $\|Y\|_{2,D} = \|g\|_{2,D} = 1$.

One benefit of this representation is that the theoretical results under the known distribution D can be easily translated to the agnostic setting. This is easily done by replacing D with the empirical distribution $\hat{D}$ that is uniform on the training set and zero outside of it. For instance, the empirical loss of g immediately satisfies the same type of relationship as in (3):

$$\frac{1}{n}\sum_i \mathbb{1}\{y_i \neq g(\mathbf{x}_i)\} = \frac{1}{2} - \frac{1}{2}\langle Y, g\rangle_{\hat{D}} = \frac{1}{4}\|Y - g\|_{2,\hat{D}}^2.$$

3.2 PAC and MMSE

In what follows, we derive bounds on the expected and empirical loss and prove the main theorems. The main ingredient in the proof of the main results (Theorem 1 and 2) is a connection between the MMSE and the PAC learning loss.

[1] A zero function in this space is a function that maps $(\mathbf{x}, y) \mapsto 0$ for all $\mathbf{x}, y$ except a zero-probability subset.

Consider a general problem in which Z is the observations and the goal is to predict Y. Here Z takes values from a generic set $\mathcal{Z}$ and Y from $\{-1,1\}$. Let Y_{MMSE} be the MMSE of Y given Z. It is known that $Y_{MMSE} = \mathbb{E}[Y|Z]$. In the following lemmas, we establish the connection between MMSE and PAC. The proofs are provided in Appendix A and B.

Lemma 2. *Suppose $(Y,Z) \sim D$ is a pair of random variables, where Y takes values from $\{-1,1\}$ and Z from some set $\mathcal{Z}$. Suppose $g : \mathcal{Z} \to \{-1,1\}$ is any predictor of Y from Z. Then,*

$$\mathbb{P}\Big\{Y \neq g(Z)\Big\} = \frac{1}{2} - \frac{1}{2}\langle Y_{MMSE}, g\rangle.$$

Moreover, let opt_Z be the minimum 0-1 loss among all predictors of Y given Z. Then,

$$opt_Z = \frac{1}{2} - \frac{1}{2}\mathbb{E}\Big[\big|\mathbb{E}[Y|Z]\big|\Big]. \tag{4}$$

Lastly, $g^* \equiv \text{sign}[Y_{MMSE}]$ is the optimal predictor.

Lemma 3. *Let $\mathcal{Z}$ be any set and $h : \mathcal{Z} \to \mathbb{R}$ be any bounded function. Suppose $(Y,Z) \sim D$ be a pair of random variables, where Y take values from $\{-1,1\}$ and Z from $\mathcal{Z}$. Then,*

$$\mathbb{P}\Big\{Y \neq \text{sign}[h(Z)]\Big\} \leq opt_Z + U\big(\mathbb{E}\Big[\big(Y_{MMSE} - h(Z)\big)^2\Big]\big),$$

where U is a polynomial defined as $U(x) = x^3 + \frac{3}{2}x^2 + \frac{3}{2}x$.

Connections to learning k-juntas: Given the above results, we can derive bounds on the error in learning many classes such as k-juntas. Let $\mathcal{J}$ be a subset of $[d]$ with k elements. Set $Z = X^{\mathcal{J}}$ as our observation variable. Consider all polynomials on the coordinates of $\mathcal{J}$ as the input variables. The polynomial that minimizes the square loss is defined as the projection of Y onto the subset $\mathcal{J}$. This polynomial is formally defined as

$$\Pi_Y^{\mathcal{J}} := \arg\min_{p \in \mathcal{P}_k} \|Y - p(X^{\mathcal{J}})\|_{2,D} \tag{5}$$

where *opt* is the set of polynomials of degree at most k. Note that $\Pi_Y^{\mathcal{J}}$ is the MMSE of Y from the observation $Z = X^{\mathcal{J}}$. Then we immediately get the following result from Lemma 2.

Corollary 1. *Let* opt *be the minimum 0-1 among all the k-juntas for a fixed $k \leq d$. Then,*

$$\text{opt} = \frac{1}{2} - \frac{1}{2}\max_{\mathcal{J}\subseteq[d],\,|\mathcal{J}|=k} \|\Pi_Y^{\mathcal{J}}\|_{1,D}. \tag{6}$$

Based on these results, we are ready to prove Theorem 1 on PAC learning of k-juntas using $\mathcal{L}_2$ regression.

3.3 Proof of Theorem 1

For any $\mathcal{J}$, let $\hat{p}_{\mathcal{J}}$ be the output of the empirical polynomial regression, that is $\hat{p}_{\mathcal{J}} = \arg\min_{p\in\mathcal{P}_k} \|Y - p_{\mathcal{J}}\|_{2,\hat{D}}$, where $\hat{D}$ is the empirical distribution. Note that the selected predictor is of the form $\text{sign}[\hat{p}_{\mathcal{J}}]$, as in Algorithm 1. As a result, from Corollary 3 with D replaced with $\hat{D}$ and $Z = \mathbf{X}^{\mathcal{J}}$, the empirical loss of $\text{sign}[\hat{p}_{\mathcal{J}}]$ is bounded as $\mathcal{L}_{\hat{D}}(\text{sign}[\hat{p}_{\mathcal{J}}]) \leq \frac{1}{2} - \frac{1}{2}\|\hat{p}_{\mathcal{J}}\|_{1,\hat{D}}$, where the $U(\cdot)$ term in Lemma 3 is zero, as $\hat{p}_{\mathcal{J}}$ is the MMSE of Y under $\hat{D}$. Next, we minimize both sides over all k-element subsets $\mathcal{J}$. From Corollary 1, with D replaced by $\hat{D}$, the right-hand side of the above inequality minimized over $\mathcal{J}$ is the minimum empirical loss $\widehat{opt}$. This implies that $\min_{\mathcal{J}:|\mathcal{J}|=k} \mathcal{L}_{\hat{D}}(\text{sign}[\hat{p}_{\mathcal{J}}]) = \widehat{opt}$. Hence, we proved that the minimum empirical loss is achieved using the $\mathcal{L}_2$ polynomial regression. Naturally, the next step is to extend this result to the generalization loss. This part follows from the standard arguments in VC theory (See Corollary 3.19 in (Mohri et al., 2018)) and the fact that the VC dimension of the k-junta class is less than $2^k + O(k\log d)$. Particularly, given $\delta \in (0,1)$, with probability $(1-\delta)$, the generalization loss is less than $opt + O\Big(\sqrt{\frac{2^k + k\log d}{n}\log\frac{n}{2^k+k\log d}}\Big) + \sqrt{\frac{\log(1/\delta)}{2n}}$, where n is the number of samples. With this inequality, the theorem is proved.

3.4 PAC Learning in Fourier Domain

Next, we analyze the Fourier algorithm and prove Theorem 2. We study the PAC learning problem in the Fourier domain. For that, we start with the following lemma connecting the prediction loss to the Fourier coefficients.

Lemma 4. *Let $(\mathbf{X}, Y) \sim D$ where D is a distribution on $\{-1,1\}^{d+1}$. Then the prediction loss of any $g(\mathbf{x})$ equals to*

$$\mathcal{L}_D(g) = \frac{1}{2} - 2^{d-1}\sum_{\mathcal{S}\subseteq[d]} a_{\mathcal{S}} g_{\mathcal{S}},$$

where $g_{\mathcal{S}}$ is the (uniform) Fourier coefficient of g corresponding to $\mathcal{S}$, as in Fact 1, and $a_{\mathcal{S}}$ is the stochastic Fourier coefficient of Y as in (1).

Proof. Recall from (3) that $\mathcal{L}_D(g) = \frac{1}{2} - \frac{1}{2}\mathbb{E}[Yg(\mathbf{X})]$. Then, from the definition of $a_{\mathcal{S}}$ in (1), we have that

$$\begin{aligned}
\mathbb{E}[Yg(\mathbf{X})] &= \sum_{y,\mathbf{x}} D(x,y) yg(\mathbf{x}) \\
&= \sum_{y,\mathbf{x}} yD(x,y)\big(\sum_{\mathcal{S}} g_{\mathcal{S}}\chi_{\mathcal{S}}(\mathbf{x})\big) \\
&= \sum_{\mathcal{S}} g_{\mathcal{S}} \sum_{y,\mathbf{x}} yD(x,y)\chi_{\mathcal{S}}(\mathbf{x}) \\
&= \sum_{\mathcal{S}\subseteq[d]} g_{\mathcal{S}}(2^d a_{\mathcal{S}}),
\end{aligned}$$

as needed. □

Interestingly, with this lemma, the prediction loss under any distribution D can be written in terms of $g_{\mathcal{S}}$'s which are the Fourier coefficient of g under the uniform distribution. We use this intuition and prove the following lemma in Appendix C.

Lemma 5. *Let $(\mathbf{X}, Y) \sim D$ where D is a distribution on $\{-1,1\}^{d+1}$. Given any subset coordinate $\mathcal{J}$, let $f^{\mathcal{J}}(\mathbf{x}) = \sum_{\mathcal{S}\subseteq\mathcal{J}} a_{\mathcal{S}}\chi_{\mathcal{S}}(\mathbf{x})$, with $a_{\mathcal{S}}$'s being the stochastic Fourier coefficients of Y. Let $h_{\mathcal{J}}$ be any real-valued function on coordinate $\mathcal{J}$, then the prediction loss of $g \equiv \text{sign}[h_{\mathcal{J}}]$ is bounded as*

$$\mathcal{L}_D(g) \le \frac{1}{2}(1 - \|f^{\mathcal{J}}\|_{1,\text{unif}}) + U(\|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\text{unif}}),$$

where the norm is computed on the uniform distribution and $U(x) = x^3 + \frac{3}{2}x^2 + \frac{3}{2}x$.

This lemma is different from Lemma 3 in that $f^{\mathcal{J}}$ is not the MMSE estimate of Y as it is defined based on the uniform Fourier expansion.However, it gives a different characterization of the optimal loss *opt*.

Corollary 2. *The optimal loss among k-juntas under any distribution D satisfies the following equation*

$$\text{opt} = \frac{1}{2} - \frac{1}{2}\max_{\mathcal{J}\subseteq[d],\,|\mathcal{J}|=k} \|f^{\mathcal{J}}\|_{1,\text{unif}}.$$

Based on these results, we prove Theorem 2 on the PAC learning of the Fourier algorithm.

3.5 Proof of Theorem 2

We prove the theorem by showing that $\hat{g}$ in Algorithm 2 achieves *opt* of k-juntas. Recall that $\hat{g} \equiv \text{sign}[\hat{f}^{\hat{\mathcal{J}}}]$, where $\hat{f}^{\hat{\mathcal{J}}}$ is the constructed for the selected subset $\hat{\mathcal{J}}$. Thus, from Lemma 5, the prediction loss of $\hat{g}$ is bounded as

$$\mathcal{L}_D(\hat{g}) \le \frac{1}{2}(1 - \|f^{\hat{\mathcal{J}}}\|_{1,unif}) + U(\|f^{\hat{\mathcal{J}}} - \hat{f}^{\hat{\mathcal{J}}}\|_{2,unif}).$$

Next, we bound the second term on the right-hand side. Note that $\hat{f}^{\hat{\mathcal{J}}} \equiv \sum_{\mathcal{S}\subseteq\hat{\mathcal{J}}} \hat{a}_{\mathcal{S}}\chi_{\mathcal{S}}$. Parseval identity gives

$$\|f^{\hat{\mathcal{J}}} - \hat{f}^{\hat{\mathcal{J}}}\|^2_{2,unif} = \sum_{\mathcal{S}\subseteq\hat{\mathcal{J}}} (a_{\mathcal{S}} - \hat{a}_{\mathcal{S}})^2. \tag{7}$$

Consider all $\mathcal{S} \subseteq [d]$ with at most k elements. Let K be the number of such subsets. Then, as $|\hat{\mathcal{J}}| = k$, using Lemma 1 the above summation is bounded as,

$$\|f^{\mathcal{J}} - \hat{f}^{\mathcal{J}}\|^2_{2,unif} \le 2^k \sup_{\mathcal{S}:|\mathcal{S}|\le k} (a_{\mathcal{S}} - \hat{a}_{\mathcal{S}})^2 \le \frac{2^k}{2n}\log\frac{2K}{\delta},$$

where the second inequality holds with probability at least $(1-\delta)$. As a result, the prediction loss satisfies

$$\mathcal{L}_D(\hat{g}) \le \frac{1}{2}(1 - \|f^{\hat{\mathcal{J}}}\|_{1,unif}) + O\Big(\sqrt{\frac{2^k}{n}\log\frac{K}{\delta}}\Big),$$

where we used the fact that $U(x) \le 4x$ for $x \in [0,1]$. Next, we minimize the right-hand side over the choice of $\hat{\mathcal{J}}$ by considering all k-element coordinates $\mathcal{J}$. Let $\mathcal{J}^*$ be the optimal set. Then, from Corollary 2, we obtain that

$$\mathcal{L}_D(\text{sign}[\hat{f}^{\mathcal{J}^*}]) \le opt + O\Big(\sqrt{\frac{2^k}{n}\log\frac{K}{\delta}}\Big).$$

Note that $\mathcal{J}^*$ is not necessarily the same as the algorithm's choice $\hat{\mathcal{J}}$. However, as $\hat{\mathcal{J}}$ is the k-element coordinate that minimizes the empirical loss, then $\mathcal{L}_{\hat{D}}(\text{sign}[\hat{f}^{\hat{\mathcal{J}}}]) \le \mathcal{L}_{\hat{D}}(\text{sign}[\hat{f}^{\mathcal{J}^*}])$. Therefore, from McDiarmid's inequality with probability $(1-\delta)$ we obtain that

$$\mathcal{L}_{\hat{D}}(\text{sign}[\hat{f}^{\mathcal{J}^*}]) \le \mathcal{L}_D(\text{sign}[\hat{f}^{\mathcal{J}^*}]) + \sqrt{\frac{k}{2n}\log\frac{2}{\delta}},$$

where we used the fact that there are at most 2^k Boolean functions on coordinate $\mathcal{J}^*$. To sum up, we proved that

$$\mathcal{L}_{\hat{D}}(\text{sign}[\hat{f}^{\hat{\mathcal{J}}}]) \le opt + O\Big(\sqrt{\frac{2^k}{n}\log\frac{K}{\delta}}\Big).$$

Assuming that $k \le d/2$, we bound K as

$$K \le \sum_{\ell=0}^{k}\binom{d}{\ell} \le 1 + k\binom{d}{k} = 1 + \frac{d^k}{(k-1)!}.$$

The rest of the argument follows from VC theory for replacing $\hat{D}$ with D in the left-hand side.

The computational complexity of Algorithm 2 is dominated by the procedure for estimating all the K Fourier coefficients. Each estimation takes $O(nk)$. Hence, the overall computational complexity of the algorithm is $O(nkK) = O(nk\frac{d^k}{(k-1)!})$ as given in Table 1.

3.6 PAC Learning with MMSE

Lastly, we discuss a more general indication of our result about the PAC learnability of MMSE.

Theorem 3. *Suppose $\mathcal{A}$ is an algorithm that outputs $\text{sign}[\hat{Y}_{MMSE}]$, where $\hat{Y}_{MMSE}$ is the empirical MMSE of Y given the observation samples $\mathcal{S}_n$. Then, $\mathcal{A}$ agnostically PAC learns any concept class $\mathcal{C}$ containing $\text{sign}[\hat{Y}_{MMSE}]$ with error up to*

$$opt_{\mathcal{C}} + O\Big(\sqrt{\frac{VC}{n}\log(\frac{n}{\delta VC})}\Big),$$

where VC is the VC dimension of $\mathcal{C}$.

This result is a consequence of Lemma 3 applied to empirical loss followed by VC theory.

Algorithm 3: Learning with $\mathcal{L}_2$-Polynomial Regression

Input: Training samples $\mathcal{S}_n = \{(\mathbf{x}(i), y(i))\}_{i=1}^n$, degree parameter k.

1 Find a polynomial $\hat{p}$ of degree up to k that minimizes $\frac{1}{n}\sum_i \big(y(i) - p(\mathbf{x}(i))\big)^2$.

2 Find $\theta \in [-1, 1]$ such that the empirical error of $\text{sign}[\hat{p}(\mathbf{x}) - \theta]$ is minimized.

3 **return** $\hat{g} \equiv \text{sign}[\hat{p} - \theta]$.

4 Learning Other Hypothesis Classes

In this section, we study learning more general concept classes using the vanilla $\mathcal{L}_2$ polynomial regression (see Algorithm 3). An important concept class is the set of predictors that are approximated by fixed-degree polynomials as studied in (Kalai et al., 2008; Blais et al., 2010).

(ϵ, k)-approximated concept class: Given $\epsilon \in [0, 1]$, $k \in \mathbb{N}$ and any probability distribution $D_{\mathbf{X}}$ on $\mathcal{X}$, a concept class $\mathcal{C}$ of functions $c : \mathbb{R}^d \mapsto \{-1, 1\}$ is (ϵ, k)-approximated if

$$\sup_{c\in\mathcal{C}} \inf_{p\in\mathcal{P}_k} \mathbb{E}\big[\big(c(\mathbf{X}) - p(\mathbf{X})\big)^2\big] \leq \epsilon^2,$$

where $\mathcal{P}_k$ is the set of all polynomials of degree up to k.

We prove in Appendix D that the $\mathcal{L}_2$ polynomial regression learns the approximated concept class with error up to $2opt + \epsilon$. This is an improvement compared to the best known bound $8opt$ in (Linial et al., 1993).

Theorem 4. *Given $\epsilon > 0$ and $k \in \mathbb{N}$, the degree k $\mathcal{L}_2$ polynomial regression (Algorithm 3) learns any (ϵ, k)-approximated concept class, with probability greater than $(1 - \delta)$, and error up to*

$$2\text{opt} + 3\epsilon + O\Big(\sqrt{\frac{2\, d^{k+1}}{n} \log \frac{en}{d^{k+1}}}\Big) + \sqrt{\frac{1}{2n} \log \frac{1}{\delta}}$$

where d is the input dimension and n is the sample size.

Note that when changing the inputs from binary to non-binary, the $\mathcal{L}_2$ polynomial regression is not necessarily agnostic PAC learner as the scalar increases to $2opt$.

This result is derived using the following lemma proved in Appendix D.1, eliminating the need for randomized rounding.

Lemma 6. *Suppose θ is a random variable with the probability density function $f_\theta(t) = 1 - |t|$, for $t \in [-1, 1]$. Then, the following bound holds for any polynomial p*

$$\mathbb{E}_\theta\Big[\mathcal{L}_{\hat{D}}(\text{sign}[p(\mathbf{X}) - \theta])\Big] \leq \frac{1}{2}\|Y - p\|^2_{2,\hat{D}}.$$

Conclusion

This paper studies PAC learning using algorithms based on $\mathcal{L}_2$ polynomial regression. Mainly, we show that $\mathcal{L}_2$ based algorithms are PAC learners for the k-junta class. Moreover, we present a more efficient PAC learning algorithm based on the (uniform) Boolean Fourier expansion. Our approach relies on two frameworks, one connecting MMSE and PAC and the other connecting PAC and the Boolean Fourier expansion. With this approach and powerful tools for analyzing vector spaces, we derive tighter bounds between the 0-1 loss and the square loss.

Acknowledgments

This work was partially supported by the NSF Center for Science of Information (CSoI) Grant CCF-0939370, and also by NSF Grants CCF-2006440, CCF-2007238, CCF-2211423, and Google Research Award.

References

J. Arpe and E. Mossel. Agnostically learning juntas from random walks. June 2008.

A. Birnbaum and S. S. Shwartz. Learning halfspaces with the zero-one loss: time-accuracy tradeoffs. In *Advances in Neural Information Processing Systems*, pages 926–934, 2012.

E. Blais, R. O'Donnell, and K. Wimmer. Polynomial regression under arbitrary product distributions. *Machine learning*, 80(2-3):273–294, 2010.

A. Blum, M. Furst, J. Jackson, M. Kearns, Y. Mansour, and S. Rudich. Weakly learning DNF and characterizing statistical query learning using fourier analysis. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing - STOC 94*. ACM Press, 1994. doi: 10.1145/195058.195147.

N. H. Bshouty and A. Costa. Exact learning of juntas from membership queries. In *Algorithmic Learning Theory (ALT)*, pages 115–129. Springer International Publishing, 2016. doi: 10.1007/978-3-319-46379-7_8.

X. Chen, R. Jayaram, A. Levi, and E. Waingarten. Learning and testing junta distributions with sub cube conditioning. In M. Belkin and S. Kpotufe, editors, *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pages 1060–1113. PMLR, 15–19 Aug 2021.

T. A. Courtade and G. R. Kumar. Which Boolean functions maximize mutual information on noisy inputs? *IEEE Trans. Inf. Theory*, 60(8):4515–4525, 2014.

A. De, E. Mossel, and J. Neeman. Junta correlation is testable. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, Nov. 2019.

I. Diakonikolas, T. Gouleakis, and C. Tzamos. Distribution-independent pac learning of halfspaces with massart noise. In *Advances in Neural Information Processing Systems*, pages 4749–4760, 2019.

P. Drineas, M. W. Mahoney, and S. Muthukrishnan. Sampling algorithms for l2 regression and applications. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm*, SODA '06, page 1127–1136, USA, 2006. Society for Industrial and Applied Mathematics. ISBN 0898716055.

P. Drineas, M. W. Mahoney, S. Muthukrishnan, and T. Sarlos. Faster least squares approximation. *Numerische Mathematik*, 117(2):219–249, oct 2010. doi: 10.1007/s00211-010-0331-6.

E. Fischer, G. Kindler, D. Ron, S. Safra, and A. Samorodnitsky. Testing juntas. *Journal of Computer and System Sciences*, 68:753–787, 2004.

M. L. Furst, J. C. Jackson, and S. W. Smith. Improved learning of AC^0 functions. In *COLT*, volume 91, pages 317–325, 1991.

F. L. Gall and F. Urrutia. Improved rectangular matrix multiplication using powers of the coppersmith-winograd tensor. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1029–1046. Society for Industrial and Applied Mathematics, jan 2018. doi: 10.1137/1.9781611975031.67.

I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

M. Heidari, S. S. Pradhan, and R. Venkataramanan. Boolean functions with biased inputs: Approximation and noise sensitivity. In *Proc. IEEE Int. Symp. Information Theory (ISIT)*, pages 1192–1196, July 2019. doi: 10.1109/ISIT.2019.8849233.

M. Heidari, J. Sreedharan, G. I. Shamir, and W. Szpankowski. Information sufficiency via fourier expansion. In *Proc. IEEE Int. Symp. Information Theory (ISIT)*, July 2021a.

M. Heidari, J. Sreedharan, G. I. Shamir, and W. Szpankowski. Finding relevant information via a discrete fourier expansion. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 4181–4191. PMLR, 18–24 Jul 2021b.

V. Iyer, A. Tal, and M. Whitmeyer. Junta distance approximation with sub-exponential queries. June 2021.

J. C. Jackson. Uniform-distribution learnability of noisy linear threshold functions with restricted focus of attention. In G. Lugosi and H. U. Simon, editors, *Learning Theory*, pages 304–318, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. ISBN 978-3-540-35296-9.

A. T. Kalai, A. R. Klivans, Y. Mansour, and R. A. Servedio. Agnostically learning halfspaces. *SIAM Journal on Computing*, 37(6):1777–1805, jan 2008. doi: 10.1137/060649057.

G. Kalai. Noise sensitivity and chaos in social choice theory. Technical report, Hebrew University, 2005.

M. J. Kearns, R. E. Schapire, and L. M. Sellie. Toward efficient agnostic learning. *Machine Learning*, 17(2-3):115–141, 1994. doi: 10.1007/bf00993468.

A. R. Klivans, P. M. Long, and R. A. Servedio. Learning halfspaces with malicious noise. *Journal of Machine Learning Research*, 10(12), 2009.

J. Li and M. Médard. Boolean functions: Noise stability, non-interactive correlation, and mutual information. In *Proc. IEEE ISIT*, 2018.

N. Linial, Y. Mansour, and N. Nisan. Constant depth circuits, Fourier transform, and learnability. *J. ACM*, 40(3):607–620, 1993.

Z. Liu, X. Chen, R. A. Servedio, Y. Sheng, and J. Xie. Distribution-free junta testing. *ACM Transactions on Algorithms*, 15(1):1–23, jan 2019. doi: 10.1145/3264434.

M. N. Y. U. Mohri, A. (Google, I. Rostamizadeh, A. U. of California, and B. Talwalkar. *Foundations of Machine Learning*. MIT Press Ltd, 2018. ISBN 0262039400.

E. Mossel, R. O'Donnell, and R. P. Servedio. Learning juntas. In *Proc. ACM Symp. on Theory of Computing*, pages 206–212, 2003.

E. Mossel, R. O'Donnell, and R. A. Servedio. Learning functions of k relevant variables. *J. Comput. Syst. Sci*, 69(3):421–434, 2004.

R. O'Donnell. *Analysis of boolean functions*. Cambridge University Press, 2014.

R. A. Servedio, L.-Y. Tan, and J. Wright. Adaptivity helps for testing juntas. In *Proceedings of the 30th Conference on Computational Complexity*, CCC '15, page 264–279, Dagstuhl, DEU, 2015. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 9783939897811.

S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, New York, NY, USA, 2014. ISBN 1107057132, 9781107057135.

J. Suykens and J. Vandewalle. *Neural Processing Letters*, 9(3):293–300, 1999. doi: 10.1023/a:1018628609742.

L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, nov 1984. doi: 10.1145/1968.1972.

S. S. Vempala and Y. Xiao. Structure from local optima: Learning subspace juntas via higher order pca. Aug. 2011.

N. Weinberger and O. Shayevitz. On the optimal Boolean function for prediction under quadratic loss. *IEEE Trans. Inf. Theory*, 63(7):4202–4217, 2017.

N. Weinberger and O. Shayevitz. Self-predicting Boolean functions. In *Proc. IEEE ISIT*, 2018.

R. d. Wolf. *A Brief Introduction to Fourier Analysis on the Boolean Cube*. Number 1 in Graduate Surveys. Theory of Computing Library, 2008. doi: 10.4086/toc.gs.2008.001.

A Proof of Lemma 2

Lemma 2. *Suppose $(Y, Z) \sim D$ is a pair of random variables, where Y takes values from $\{-1, 1\}$ and Z from some set $\mathcal{Z}$. Suppose $g : \mathcal{Z} \to \{-1, 1\}$ is any predictor of Y from Z. Then,*

$$\mathbb{P}\Big\{Y \neq g(Z)\Big\} = \frac{1}{2} - \frac{1}{2}\langle Y_{MMSE}, g\rangle.$$

Moreover, let opt_Z be the minimum 0-1 loss among all predictors of Y given Z. Then,

$$opt_Z = \frac{1}{2} - \frac{1}{2}\mathbb{E}\Big[\big|\mathbb{E}[Y|Z]\big|\Big]. \tag{8}$$

Lastly, $g^* \equiv \text{sign}[Y_{MMSE}]$ is the optimal predictor.

Proof. From (3) in the main text, the generalization error of g can be written as $\frac{1}{2} - \frac{1}{2}\langle Y, g\rangle$. This inner product equals to the following

$$\begin{aligned}\langle Y, g\rangle &= \mathbb{E}\big[Yg(Z)\big] = \mathbb{E}_Z\Big[\mathbb{E}_{Y|Z}\big[Yg(Z)\big] \mid Z\Big]\Big] \\ &= \mathbb{E}_Z\Big[\mathbb{E}_{Y|Z}\big[Y|Z\big]g(Z)\Big].\end{aligned}$$

Let $Y_{MMSE} = \mathbb{E}[Y|Z]$. Hence, we obtain that

$$\mathbb{P}\Big\{Y \neq g(Z)\Big\} = \frac{1}{2} - \frac{1}{2}\langle Y_{MMSE}, g\rangle \tag{9}$$

Note that

$$\mathbb{P}\Big\{Y \neq g(\mathbf{X})\Big\} = \frac{1}{2} - \frac{1}{2}\langle Y_{MMSE}, g\rangle_D \geq \frac{1}{2} - \frac{1}{2}\langle |Y_{MMSE}|, |g|\rangle_D \geq \frac{1}{2} - \frac{1}{2}\|Y_{MMSE}\|_{1,D},$$

where the last inequality follows as $|g(Z)| = 1$. Therefore, we get the bound $opt_Z \geq \frac{1}{2} - \frac{1}{2}\|Y_{MMSE}\|_{1,D}$. Hence, we established a lower-bound on opt_Z. Next, we show that this bound is achievable. For that construct a predictor as $g^* = \text{sign}[Y_{MMSE}]$. Then, from the above argument, the generalization error of such g equals

$$\mathbb{P}\Big\{Y \neq \text{sign}[Y_{MMSE}]\Big\} = \frac{1}{2} - \frac{1}{2}\langle Y_{MMSE}, \text{sign}[Y_{MMSE}]\rangle_D = \frac{1}{2} - \frac{1}{2}\|Y_{MMSE}\|_{1,D},$$

where the last equality follows due to the identity $\langle h, \text{sign}[h]\rangle = \|h\|_1$ for any function h. Therefore, we showed that the lower bound is achievable which implies that $opt_Z = \frac{1}{2} - \frac{1}{2}\|Y_{MMSE}\|_{1,D}$ and that $g^* = \text{sign}[Y_{MMSE}]$ is the optimal predictor. □

B Proof of Lemma 3

Lemma 3. *Let $\mathcal{Z}$ be any set and $h : \mathcal{Z} \to \mathbb{R}$ be any bounded function. Suppose $(Y, Z) \sim D$ be a pair of random variables, where Y take values from $\{-1, 1\}$ and Z from $\mathcal{Z}$. Then,*

$$\mathbb{P}\Big\{Y \neq \text{sign}[h(Z)]\Big\} \leq opt_Z + U\Big(\sqrt{\mathbb{E}\Big[\big(Y_{MMSE} - h(Z)\big)^2\Big]}\Big),$$

where U is a polynomial defined as $U(x) = x^3 + \frac{3}{2}x^2 + \frac{3}{2}x$.[2]

Proof. For shorthand, let $f(z) = \mathbb{E}[Y|z]$ for any $z \in \mathcal{Z}$. Hence, $f(Z) = Y_{MMSE}$. From (9) in the proof of Lemma 2, the generalization error of $\text{sign}[h]$ can be written Hence, we obtain that

$$\mathbb{P}\Big\{Y \neq \text{sign}[h(Z)]\Big\} = \frac{1}{2} - \frac{1}{2}\langle f, \text{sign}[h]\rangle$$

[2]There is a typo in the original statement of the lemma in the main text. The square root is missing.

Recall that $\|f\|_{2,D} := \sqrt{\mathbb{E}_D[f(X)^2]}$. Hence, $\|a-b\|_{2,D}^2 = \|a\|_{2,D}^2 + \|b\|_{2,D}^2 - 2\langle a, b\rangle$. Therefore,

$$\begin{aligned}\langle f, \operatorname{sign}[h]\rangle &= \frac{1}{2}\big(\|f\|_{2,D}^2 + \|\operatorname{sign}[h]\|_{2,D}^2 - \|f - \operatorname{sign}[h]\|_{2,D}^2\big)\\ &= \frac{1}{2}\big(\|f\|_{2,D}^2 + 1 - \|f - \operatorname{sign}[h]\|_{2,D}^2\big),\end{aligned}$$

where we used the fact that $|\operatorname{sign}[h]| = 1$. As a result,

$$\mathbb{P}\Big\{Y \neq \operatorname{sign}[h(Z)]\Big\} = \frac{1}{4}\big(1 - \|f\|_{2,D}^2 + \|f - \operatorname{sign}[h_{\mathcal{J}}]\|_{2,D}^2\big). \tag{10}$$

In what follows, we bound the $\|f - \operatorname{sign}[h_{\mathcal{J}}]\|_{2,D}^2$. By adding and subtracting h, we have that

$$\begin{aligned}\|f - \operatorname{sign}[h]\|_{2,D}^2 &\overset{(a)}{\leq} \Big(\|f-h\|_{2,D} + \|h - \operatorname{sign}[h]\|_{2,D}\Big)^2\\ &= \Big(\|f-h\|_{2,D}^2 + \underbrace{\|h - \operatorname{sign}[h]\|_{2,D}^2}_{\text{(I)}} + 2\|f-h\|_{2,D}\underbrace{\|h - \operatorname{sign}[h]\|_{2,D}}_{\text{(II)}}\Big),\end{aligned} \tag{11}$$

where (a) follows from the Minkowski's inequality for 2-norm. Next, we provide separate bounds for the terms (I) and (II):

Bounding (I): Note that $|h - \operatorname{sign}[h]| = |1 - |h||$. Therefore,

$$\begin{aligned}\text{(I)} = \|h - \operatorname{sign}[h]\|_{2,D}^2 &= \mathbb{E}\left[(1 - |h(Z)|)^2\right]\\ &= 1 + \|h\|_{2,D}^2 - 2\|h\|_{1,D}.\end{aligned} \tag{12}$$

Bounding (II): From (12), we have

$$\begin{aligned}\|h - \operatorname{sign}[h]\|_{2,D}^2 &= 1 + \|h\|_{2,D}^2 - 2\|h\|_{1,D}\\ &\overset{(a)}{\leq} 1 + 2(\|f\|_{2,D}^2 + \|f-h\|_{2,D}^2) - 2\|h\|_{1,D}\\ &\overset{(b)}{=} 1 + 2(\|f\|_{2,D}^2 + \|f-h\|_{2,D}^2) - 2\big(\|f\|_{1,D} + (\|h\|_{1,D} - \|f\|_{1,D})\big)\\ &= 1 + 2(\|f\|_{2,D}^2 - \|f\|_{1,D}) + 2\|f-h\|_{2,D}^2 - 2\big(\|h\|_{1,D} - \|f\|_{1,D}\big)\\ &\overset{(c)}{\leq} 1 + 2\|f-h\|_{2,D}^2 - 2\big(\|h\|_{1,D} - \|f\|_{1,D}\big)\\ &\overset{(d)}{\leq} 1 + 2\|f-h\|_{2,D}^2 + 2\|f-h\|_{2,D},\end{aligned} \tag{13}$$

where (a) follows from the Minkowski's inequality for 2-norm and the inequality $(x+y)^2 \leq 2(x^2+y^2)$. Equality (b) follows by adding and subtracting $\|f\|_1$. Inequality (c) holds as $|f(x)| \leq 1$ implying that $\|f\|_2^2 \leq \|f\|_1$. Lastly, (d) holds because of the following chain of inequalities

$$\Big|\|f\|_{1,D} - \|h\|_{1,D}\Big| \leq \|f-h\|_{1,D} \leq \|f-h\|_{2,D}, \tag{14}$$

where the first is due to the Minkowski's inequality for 1-norm and the second is due to Holder's.

Next, we show that the quantity $\big\|h - \operatorname{sign}[h_{\mathcal{J}}]\big\|_{2,D}$ without the square is upper bounded by the same term as in the right-hand side of (13). That is

$$\text{(II)} = \big\|h - \operatorname{sign}[h_{\mathcal{J}}]\big\|_{2,D} \leq \lambda_1 \triangleq 1 + 2\|f-h\|_{2,D}^2 + 2\|f-h\|_{2,D}. \tag{15}$$

The argument is as follows: if $\big\|h - \operatorname{sign}[h_{\mathcal{J}}]\big\|_{2,D}$ is less than one, then the upper bound holds trivially as $\lambda_1 \geq 1$; otherwise, this quantity is less than its squared and, hence, the upper-bound holds.

Now combining (15), (12) and (11) gives

$$\|f - \operatorname{sign}[h]\|_{2,D}^2 \leq \|f-h\|_{2,D}^2 + 1 + \|h\|_{2,D}^2 - 2\|h\|_1 + 2\lambda_1\|f-h\|_{2,D}. \tag{16}$$

From this bound and (10), the error probability satisfies:

$$4\mathbb{P}\Big\{Y \neq \text{sign}[h(Z)]\Big\} \leq 2 - 2\|h\|_{1,D} + \underbrace{\|h\|_{2,D}^2 - \|f\|_{2,D}^2}_{\text{(III)}} + \|f-h\|_{2,D}^2 + 2\lambda_1\|f-h\|_{2,D}. \tag{17}$$

In what follows, we bound the term denoted by (III).

Bounding (III): From the Minkowski's inequality for 2-norm, we have

$$\begin{aligned}\|h\|_{2,D}^2 &\leq \Big(\|f\|_{2,D} + \|h-f\|_{2,D}\Big)^2 \\ &= \|f\|_{2,D}^2 + \|h-f\|_{2,D}^2 + 2\|f\|_{2,D}\|h-f\|_{2,D} \\ &\leq \|f\|_{2,D}^2 + \|h-f\|_{2,D}^2 + 2\|h-f\|_{2,D}\end{aligned}$$

where the second inequality is due Bessel's inequality implying that $\|f\|_{2,D} \leq 1$. Hence, the term (III) in (17) is upper bounded as

$$\text{(III)} \leq \lambda_2 \triangleq \|h-f\|_{2,D}^2 + 2\|h-f\|_{2,D}. \tag{18}$$

As a result of the bounds in (17), (18), we obtain that

$$\begin{aligned}4\mathbb{P}\Big\{Y \neq \text{sign}[h(Z)]\Big\} &\leq 2 - 2\|h\|_{1,D} + \lambda_2 + \|f-h\|_{2,D}^2 + 2\lambda_1\|f-h\|_{2,D} \\ &= 2 - 2\|f\|_{1,D} + 2\Big(\|f\|_{1,D} - \|h\|_{1,D}\Big) + \lambda_2 + \|f-h\|_{2,D}^2 + 2\lambda_1\|f-h\|_{2,D} \\ &\leq 2 - 2\|f\|_{1,D} + 2\|f-h\|_{2,D} + \lambda_2 + \|f-h\|_{2,D}^2 + 2\lambda_1\|f-h\|_{2,D},\end{aligned}$$

where the last inequality is due to (14). Therefore, from the definition of λ_1 and λ_2, and the function U in the statement of the lemma, we obtain

$$4\mathbb{P}\Big\{Y \neq \text{sign}[h(Z)]\Big\} \leq 2 - 2\|f\|_{1,D} + 4U(\|f-h\|_{2,D}).$$

This completes the proof by recalling that $f(z) = \mathbb{E}[Y|z]$ and that from Lemma 2, $opt_Z = \frac{1}{2} - \frac{1}{2}\|f\|_{1,D}$. □

C Proof of Lemma 5

Lemma 5. *Let $(\mathbf{X}, Y) \sim D$ where D is a distribution on $\{-1,1\}^{d+1}$. Given any subset coordinate $\mathcal{J}$, let $f^{\mathcal{J}}(\mathbf{x}) = \sum_{\mathcal{S}\subseteq\mathcal{J}} a_{\mathcal{S}}\chi_{\mathcal{S}}(\mathbf{x})$, with $a_{\mathcal{S}}$'s being the stochastic Fourier coefficients of Y. Let $h_{\mathcal{J}}$ be any real-valued function on coordinate $\mathcal{J}$, then the prediction loss of $g \equiv \text{sign}[h_{\mathcal{J}}]$ is bounded as*

$$\mathcal{L}_D(g) \leq \frac{1}{2}(1 - \|f^{\mathcal{J}}\|_1) + U(2^d\|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\text{unif}}),$$

where the norm is computed on the uniform distribution and $U(x) = x^3 + \frac{3}{2}x^2 + \frac{3}{2}x$.[3]

Proof. From Lemma 4 in the main text, the generalization error of $g = \text{sign}[h_{\mathcal{J}}]$ can be written as

$$\mathcal{L}_D(g) = \frac{1}{2} - 2^{d-1}\sum_{\mathcal{S}\subseteq[d]} a_{\mathcal{S}}g_{\mathcal{S}},$$

Note that since g depends only on the coordinates $\mathcal{J}$, then $g_{\mathcal{S}} = 0$ for any $\mathcal{S} \nsubseteq \mathcal{J}$. Hence, the above equation simplifies to

$$\mathcal{L}_D(g) = \frac{1}{2} - 2^{d-1}\sum_{\mathcal{S}\subseteq\mathcal{J}} a_{\mathcal{S}}g_{\mathcal{S}}.$$

[3]There is a typo in the original statement of the lemma in the main text. The factor 2^d is missing and the first norm does not have *unif*.

Note that $\chi_{\mathcal{S}}$'s are orthogonal for different $\mathcal{S}$'s and $\sum_{\mathbf{x}} \chi_{\mathcal{S}}(\mathbf{x})^2 = 2^d$. Hence,

$$\begin{aligned}\mathcal{L}_D(g) &= \frac{1}{2} - \frac{1}{2}\sum_{\mathbf{x}} \Big(\sum_{\mathcal{S}\subseteq\mathcal{J}} a_{\mathcal{S}}\chi_{\mathcal{S}}(\mathbf{x})\Big)\Big(\sum_{\mathcal{S}\subseteq\mathcal{J}} g_{\mathcal{S}}\chi_{\mathcal{S}}(\mathbf{x})\Big) \\ &= \frac{1}{2} - \frac{1}{2}\sum_{\mathbf{x}} f^{\mathcal{J}}(\mathbf{x})g(\mathbf{x}),\end{aligned}$$

where $f^{\mathcal{J}} \equiv \sum_{\mathcal{S}\subseteq\mathcal{J}} a_{\mathcal{S}}\chi_{\mathcal{S}}$. By multiplying and dividing 2^d, the above summation equals to the inner product on the uniform distribution as

$$\sum_{\mathbf{x}} f^{\mathcal{J}}(\mathbf{x})g(\mathbf{x}) = 2^d \langle f^{\mathcal{J}}, g\rangle_{\mathit{unif}}.$$

Hence, with the definition of g, we obtain that

$$\mathcal{L}_D(g) = \frac{1}{2} - \frac{2^d}{2}\langle f^{\mathcal{J}}, \mathrm{sign}[h_{\mathcal{J}}]\rangle$$

Using a similar argument in deriving (10), we can show that

$$\mathcal{L}_D(g) = \frac{1}{2} - \frac{2^d}{4}\big(1 + \|f^{\mathcal{J}}\|_{2,\mathit{unif}}^2 - \|f^{\mathcal{J}} - \mathrm{sign}[h_{\mathcal{J}}]\|_{2,\mathit{unif}}^2\big). \tag{19}$$

Notice that this equation is different from (10) because of the factor 2^d and that the norm quantities are taken with respect to the uniform distribution. We proceed with bounding the 2-norm quantities. Note that we can apply exactly the same argument used to derive in (16), as it holds for any underlying distribution. The 2-norm quantity above is upper-bounded as follows

$$\|f^{\mathcal{J}} - \mathrm{sign}[h_{\mathcal{J}}]\|_{2,\mathit{unif}}^2 \le \|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}}^2 + 1 + \|h_{\mathcal{J}}\|_{2,\mathit{unif}}^2 - 2\|h_{\mathcal{J}}\|_{1,\mathit{unif}} + 2\lambda_1\|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}},$$

where $\lambda_1 = 1 + 2\|f^{\mathcal{J}} - h\|_{2,\mathit{unif}}^2 + 2\|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}}$. As a result, the loss of g satisfies

$$\mathcal{L}_D(g) = \frac{1}{2} - \frac{2^d}{4}\Big(2\|h_{\mathcal{J}}\|_{1,\mathit{unif}} + \underbrace{\big(\|f^{\mathcal{J}}\|_{2,\mathit{unif}}^2 - \|h_{\mathcal{J}}\|_{2,\mathit{unif}}^2\big)}_{\text{(I)}} - \|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}}^2 - 2\lambda_1\|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}}\Big).$$

Note that (I) $\ge -\lambda_2$ with $\lambda_2 \triangleq \|h_{\mathcal{J}} - f^{\mathcal{J}}\|_{2,\mathit{unif}}^2 + 2\|h_{\mathcal{J}} - f^{\mathcal{J}}\|_{2,\mathit{unif}}$ as in (18). Next, by adding and subtracting $2\|f^{\mathcal{J}}\|_{1,\mathit{unif}}$, we have that

$$\begin{aligned}\mathcal{L}_D(g) &\le \frac{1}{2} - \frac{2^d}{4}\Big(2\|f^{\mathcal{J}}\|_{1,\mathit{unif}} + 2\underbrace{\big(\|h_{\mathcal{J}}\|_{1,\mathit{unif}} - \|f^{\mathcal{J}}\|_{1,\mathit{unif}}\big)} - \lambda_2 - \|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}}^2 - 2\lambda_1\|f - h\|_{2,\mathit{unif}}\Big) \\ &\le \frac{1}{2} - \frac{2^d}{4}\Big(2\|f^{\mathcal{J}}\|_{1,\mathit{unif}} - 2\|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}} - \lambda_2 - \|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}}^2 - 2\lambda_1\|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}}\Big),\end{aligned}$$

where we used (14) to derive the inequality. Therefore, from the definition of λ_1, λ_2 and $U(x) = x^3 + \frac{3}{2}x^2 + \frac{3}{2}x$ we have that

$$\mathcal{L}_D(g) \le \frac{1}{2} - \frac{2^d}{2}\|f^{\mathcal{J}}\|_{1,\mathit{unif}} + 2^d U(\|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}}).$$

Lastly, we further bound this expression. Note that $\|\cdot\|_1 = 2^d\|\cdot\|_{1,\mathit{unif}}$. Then, we have that

$$\mathcal{L}_D(g) = \frac{1}{2} - \frac{1}{2}\|f^{\mathcal{J}}\|_1 + 2^d U(\|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}}) \le \frac{1}{2} - \frac{1}{2}\|f^{\mathcal{J}}\|_1 + U(2^d\|f^{\mathcal{J}} - h_{\mathcal{J}}\|_{2,\mathit{unif}}),$$

where the last inequality holds by bringing 2^d inside $U(\cdot)$. This completes the proof of the lemma. □

D Proof of Theorem 4

To derive an upper bound on the empirical error of $\hat{g}$, we first consider a weaker version of the algorithm. The idea is to select θ randomly instead of optimizing it as in the algorithm. For that, we use Lemma 6 in Section 4. Consequently, from the lemma and due the fact that θ in the algorithm is selected to minimize the empirical error, we obtain that

$$\mathbb{P}_{\hat{D}}\Big\{Y \neq \hat{g}(\mathbf{X})\Big\} \leq \frac{1}{2}\|Y - \hat{p}\|^2_{2,\hat{D}}, \tag{20}$$

where $\hat{p}$ is the output of $\mathcal{L}_2$-polynomial regression and $\hat{g} \equiv \text{sign}[\hat{p} - \theta]$, as in Algorithm 1. Let c^* be the predictor with minimum generalization error in the (ϵ, k)-approximated concept class. Let p be a degree k polynomial such that $\|c^* - p\|_2 \leq \epsilon$. Since $\hat{p}$ minimizes the empirical 2-norm, then the right-hand side of (20) satisfies

$$\frac{1}{2}\|Y - \hat{p}\|^2_{2,\hat{D}} \leq \frac{1}{2}\|Y - p^*\|^2_{2,\hat{D}}. \tag{21}$$

We proceed by taking the expected error of the empirical error with respect to the random training samples. From (20) and (21) we obtain the following inequalities

$$\begin{aligned}
\mathbb{E}\Big[\mathbb{P}_{\hat{D}}\Big\{Y \neq \hat{g}(\mathbf{X})\Big\}\Big] &\leq \frac{1}{2}\mathbb{E}\Big[\|Y - p^*\|^2_{2,\hat{D}}\Big] = \frac{1}{2}\|Y - p^*\|^2_{2,D} \\
&\overset{(a)}{\leq} \frac{1}{2}\Big(\|Y - c^*\|_{2,D} + \|p^* - c^*\|_{2,D}\Big)^2 \\
&\leq \frac{1}{2}\Big(\|Y - c^*\|_{2,D} + \epsilon\Big)^2 \\
&\overset{(b)}{\leq} \frac{1}{2}\Big(\|Y - c^*\|^2_{2,D} + 4\epsilon + \epsilon^2\Big) \\
&\overset{(c)}{\leq} 2opt + \frac{5}{2}\epsilon,
\end{aligned} \tag{22}$$

where (a) holds from Minkowski's inequality for 2-norm, (b) holds as $\|Y - c^*\|_{2,D} \leq 2$, and (c) holds because of the second equality in (3) and that $opt = \mathbb{P}\{Y \neq c^*(\mathbf{X})\}$.

Next, we connect the empirical error of $\hat{g}$ to its generalization error. Note that the Vapnik–Chervonenkis (VC) dimension of all functions of the form $\text{sign}[p]$ for some polynomial of degree upto k does not exceed d^{k+1}. Therefore, from VC theory (See Corollary 3.19 in (Mohri et al., 2018)) for any δ, with probability at least $(1 - \delta)$, the following inequality holds

$$\mathbb{P}\Big\{Y \neq \hat{g}(\mathbf{X})\Big\} \leq \mathbb{P}_{\hat{D}}\Big\{Y \neq \hat{g}(\mathbf{X})\Big\} + \sqrt{\frac{2\, d^{k+1}}{n}\log\frac{en}{d^{k+1}}} + \sqrt{\frac{\log\frac{1}{\delta}}{2n}}. \tag{23}$$

Therefore, the proof is complete by taking the expectation and combining it with the last bound in (22).

D.1 Proof of Lemma 6

Note that $y \neq \text{sign}(p(\mathbf{x}) - \theta)$, if θ is between y and $p(\mathbf{x})$. Hence, the expected empirical error of $\text{sign}[p(\mathbf{X}) - \theta]$ with respect to the random θ equals to

$$\begin{aligned}
&\mathbb{E}_\theta\Big[\mathbb{P}_{\hat{D}}\Big\{Y \neq \text{sign}[p(\mathbf{X}) - \theta]\Big\}\Big] \\
&\quad= \frac{1}{n}\sum_i \mathbb{E}_\theta\Big[\mathbb{1}\big\{y_i \neq \text{sign}(p(\mathbf{x}_i) - \theta)\big\}\Big] \\
&\quad= \frac{1}{n}\sum_i \underbrace{\mathbb{P}\Big\{\theta \in [p(\mathbf{x}_i), y_i] \bigcup [y_i, p(\mathbf{x}_i)]\Big\}}_{\mathbb{P}_i}.
\end{aligned} \tag{24}$$

Next, we show that $\mathbb{P}_i \leq \frac{1}{2}(y_i - p(\mathbf{x}_i))^2$ for all $(\mathbf{x}_i, y_i)$'s. Suppose $y_i = 1$. If $p(\mathbf{x}_i) > 1$, then $\mathbb{P}_i = 0$ as $\theta \leq 1$. If $p(\mathbf{x}_i) \in [0, 1]$, then

$$\begin{aligned}\mathbb{P}_i &= \mathbb{P}\Big\{\theta \in [p(\mathbf{x}_i), 1]\Big\} = \int_{p(\mathbf{x}_i)}^{1} (1-t)dt \\ &= \frac{1}{2}\big(1 - p(\mathbf{x}_i)\big)^2 = \frac{1}{2}\big(y_i - p(\mathbf{x}_i)\big)^2.\end{aligned}$$

If $p(\mathbf{x}_i) \in [-1, 0]$, then

$$\begin{aligned}\mathbb{P}_i &= \mathbb{P}\Big\{\theta \in [p(\mathbf{x}_i), 1]\Big\} = \int_{p(\mathbf{x}_i)}^{1} 1 - |t| dt \\ &= \frac{1}{2} + \int_{p(\mathbf{x}_i)}^{0} (1+t)dt \\ &= \frac{1}{2} - p(\mathbf{x}_i) - \frac{1}{2}(p(\mathbf{x}_i))^2 \\ &\leq \frac{1}{2}(1 + |p(\mathbf{x}_i)|)^2 = \frac{1}{2}(y_i - p(\mathbf{x}_i))^2.\end{aligned}$$

Lastly, if $p(\mathbf{x}_i) < -1$, then $\mathbb{P}_i = 1$ because $\theta \geq -1$. In this case also $\mathbb{P}_i \leq \frac{1}{2}(y_i - p(\mathbf{x}_i))^2$. The case for $y_i = -1$ follows by symmetricity. Hence, we obtain the following inequality

$$\mathbb{E}_\theta\Big[\mathbb{P}_{\hat{D}}\Big\{Y \neq \hat{g}(\mathbf{X})\Big\}\Big] \leq \frac{1}{n}\sum_i \frac{1}{2}\big(y_i - p(\mathbf{x}_i)\big)^2.$$

The proof is complete by noting that the right-hand side equals to $\frac{1}{2}\|Y - p\|^2_{2,\hat{D}}$.