On Agnostic PAC Learning using $\mathcal{L}_2$-polynomial Regression and Fourier-based Algorithms

Mohsen Heidari
Purdue University
mheidari@purdue.edu

Wojciech Szpankowski
Purdue University
szpan@purdue.edu

Abstract—We develop a framework using Hilbert spaces as a proxy to analyze PAC learning problems with structural properties. We consider a joint Hilbert space incorporating the relation between the true label and the predictor under a joint distribution $D$. We demonstrate that agnostic PAC learning with 0-1 loss is equivalent to an optimization in the Hilbert space domain. With our model, we revisit the PAC learning problem using methods based on least-squares such as $\mathcal{L}_2$ polynomial regression and [1]'s low-degree algorithm. We study learning with respect to several hypothesis classes such as half-spaces and polynomial-approximated classes (i.e., functions approximated by a fixed-degree polynomial). We prove that (under some distributional assumptions) such methods obtain generalization error up to $2P_{\text{opt}}$ with $P_{\text{opt}}$ being the optimal error of the class. Hence, we show the tightest bound on generalization error when $P_{\text{opt}} \leq 0.2$.

I. INTRODUCTION

We study binary classification using polynomial regression from the agnostic PAC learning perspective [2], [3]. In this problem, multiple training instances are generated IID according to an underlying distribution $D$ on the feature-label sets $\mathcal{X} \times \{-1, 1\}$. In addition, we are given a hypothesis class with respect to which the learning process takes place. If $P_{\text{opt}}$ is the minimum error attained using the given class, then the objective of the learning algorithm is to output, with high probability, a classifier whose generalization error is not greater than $P_{\text{opt}} + \epsilon$.

To gain computational efficiency or analytical tractability, many conventional learning methods such as support-vector machine (SVM) rely on intermediate loss functions other than the natural 0–1 loss. Square loss is an example that is a basis for $\mathcal{L}_2$-polynomial regression or another variant of SVM known as LS-SVM [4]. The well-known “low-degree” algorithm [1] is also known to be in this category of algorithms [5]. Such methods have been analyzed for many PAC learning classes with appropriate geometrical properties. We consider a joint Hilbert space incorporating the relation between the true label and the predictor under the joint distribution $D$. This is unlike conventional analysis using Hilbert spaces that focus only on the predictors with marginal $D_X$ on the features. As a byproduct, we improve the above mentioned bounds and show that the generalization error of $\mathcal{L}_2$-polynomial regression and the low-degree algorithm is less than $2P_{\text{opt}}$. This bound the improves upon the previous bounds when $P_{\text{opt}} \leq 0.2$. We show that methods based on square loss are suitable for learning classes with appropriate geometrical properties.

A. Our approach

We develop our framework by constructing two Hilbert spaces one with respect to the true underlying distribution $D$ and the other with respect to the empirical one. The first one is $\mathcal{L}_2(D)$, that is all real-valued functions $f$ on $\mathcal{X} \times \mathcal{Y}$ such that $\mathbb{E}[f(X,Y)^2] < \infty$. The second one is $\mathcal{L}_2(\hat{D})$ with $\hat{D}$ being the empirical distribution of the training set. With this formulation, the true label $Y$ and the training labels are understood as a member of these spaces. With this formulation, the generalization error of any classifier $c$ equals $\frac{1}{n}\|Y-c\|_2^2$. Similarly, when the distance is calculated in the second Hilbert space, we obtain a characterization of the empirical error. Hence, minimizing the generalization (or empirical) error is equivalent to minimizing the distance between $Y$ and the classifier $c$ in the first (or second) Hilbert space. We argue that the mentioned hypothesis classes have appropriate structures using that allows us to drive lower bounds on its minimum error $P_{\text{opt}}$. For instance, given $k$, the polynomial-approximated class is characterized by the subspace of $\mathcal{L}_2(D)$ spanned by polynomials of degree up to $k$. With this structure, finding $P_{\text{opt}}$ is equivalent to finding the minimum distance of $Y$ to the subspace spanned by polynomials of degree up to $k$. As for the learning algorithms, we argue the low-degree algorithm and $\mathcal{L}_2$-polynomial regression have suitable structures using which we drive our upper bounds on their generalization errors. For instance, in the case of $\mathcal{L}_2$-polynomial regression, the error of any classifier of the form $\text{sign}(p(x) - \theta)$, with $\theta$ chosen appropriately, is bounded from above by $\frac{1}{n}\|Y-p\|_2^2$. Hence, minimizing the squares-loss as in $\mathcal{L}_2$-regression yields an error less than $2P_{\text{opt}}$. 

---

Mohsen Heidari
Purdue University
mheidari@purdue.edu

Wojciech Szpankowski
Purdue University
szpan@purdue.edu

---
B. Summary of the Results

In this work, we first present a more general version of the low-degree algorithm incorporating non-uniform but product probability distributions. We refer to this generalization as Fourier algorithm. With our framework, we study learning with respect to three well-known hypothesis classes. The first class is half-spaces consisting of all the Boolean-valued functions of the form $c(x) = \text{sign}(\sum_{j=1}^{d} w_j x_j - \theta)$. The second class is called polynomial-approximated functions. Given a positive integer $k$ and $\epsilon > 0$, it consists of Boolean-valued functions that are approximated by a degree $k$ polynomial with square error up to $\epsilon^2$. The third class is a generalization of the second. We use our framework to analyze learning these hypothesis classes using $L_2$-polynomial regression and the Fourier algorithm. Below, is the summary of our results:

1) The $L_2$ polynomial regression with degree $k$ outputs a hypothesis $\hat{g}$ whose generalization error has the following properties:
   - For learning polynomial-approximated classes, it is less than $2P_{\text{opt}} + 3\epsilon$ (Theorem 1).
   - For learning half-spaces, when the marginal $D_x$ is uniform over the unit ball in $\mathbb{R}^d$, it is less than $2P_{\text{opt}} + 3\epsilon$ (Theorem 3).
   - For learning generalized concentrated classes, under any distribution, it is less than $2P_{\text{opt}} + \epsilon$ (Theorem 4).

2) If the marginal $D_x$ is a product probability distribution on $\{-1, 1\}^d$, then with probability $(1 - \delta)$, the Fourier algorithm outputs a hypothesis such that its generalization error is less than $2P_{\text{opt}} + 2\epsilon$ for learning polynomial-approximated classes.

C. Related Works

The low-degree algorithm is introduced by [1] with PAC learning guarantees under the uniform distribution over $\{-1, 1\}^d$. This algorithm which is based on the Fourier expansion on the Boolean cube has been used for various problems [6], [8], [9]. The $L_2$ polynomial regression along with its $L_1$ counterpart is introduced by [5] for learning with respect to polynomial-approximated classes, $k$-juntas, and half-spaces. Learning with respect to such classes has been studied extensively in the literature [5], [10]–[12]. Among such classes, learning with respect to half-spaces is the most challenging. In the case of proper agnostic PAC learning, where the algorithm’s predictor must be a half-space, it is an NP-hard problem [13], [14]. Even without the proper restriction, the problem is NP-hard. That said, under distributional assumptions, polynomial time algorithms are introduced [5], [15], [16]. Among them are the improper learning algorithms based on regression methods such as $L_1$ or $L_2$ polynomial regression [1], [5]. In particular, [5] proved that $L_1$ polynomial regression learns a range of hypothesis classes such as half-spaces (under distributional assumptions) and polynomial-approximated classes.

II. PRELIMINARIES

Notation: The input set is denoted by $\mathcal{X}$ which is a subset of $\mathbb{R}^d$ for some positive integer $d$. The output set is denoted by $\mathcal{Y}$ which is a subset of $\mathbb{R}$. In binary classification $\mathcal{Y} = \{-1, 1\}$. For shorthand, the random vectors in $\mathbb{R}^d$ are denoted by $\mathbf{X} = (X_1, X_2, ..., X_d)$. Further, for any ordered subset $\mathcal{J} = \{j_1, j_2, \cdots, j_m\}$, by $\mathbf{X}^\mathcal{J}$ denote the random vector $(X_{j_1}, X_{j_2}, \cdots, X_{j_m})$. Similarly, by $x^\mathcal{J}$ denote the vector $(x_{j_1}, x_{j_2}, \cdots, x_{j_m})$. For a pair of functions $f, g$ on $\mathcal{X}$, the notation $f \equiv g$ means that $f(x) = g(x)$ for all $x \in \mathcal{X}$. Lastly, for any natural number $\ell$, the set $\{1, 2, \cdots, \ell\}$ is denoted by $[\ell]$.

A. A Hilbert Space Representation

We first develop a Hilbert Space formulation for the binary classification problem. Let $D$ be a joint probability distribution on the input-output set $\mathcal{X} \times \mathcal{Y}$. In this paper, it is assumed that the marginal $D_x$ of any joint distribution $D$ on $\mathcal{X} \times \mathcal{Y}$ has finite moments. Consider a Hilbert space of all real-valued functions $f : \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}$ which are $L_2(D)$, that is $\mathbb{E}_D[f(\mathbf{X}, \mathbf{Y})]^2 < \infty$. The inner product between two members $f, g$ is defined as $\langle f, g \rangle \triangleq \mathbb{E}_D[f(\mathbf{X}, \mathbf{Y})g(\mathbf{X}, \mathbf{Y})]$. Given any integer $p > 0$ and distribution $D$, the $p$-norm of a function $f$ is defined as $||f||_{p, D} \triangleq \left(\mathbb{E}_D[|f(\mathbf{X}, \mathbf{Y})|^p]\right)^{1/p}$. Given any training sample $\mathcal{S} = \{(x_i, y_i) : i = 1, 2, ..., n\}$, let $\hat{D}$ denote its empirical distribution, that is a uniform distribution on $\mathcal{S}$ and zero outside of it. Associated with this distribution, we consider the Hilbert space $L_2(\hat{D})$ with the inner product and norms defined based on the empirical distribution $\hat{D}$. We use this formulation to study the binary classification problem where $\mathcal{Y} = \{-1, 1\}$. Therefore, the generalization error of any predictor $c : \mathcal{X} \mapsto \{-1, 1\}$ can be written in terms of the inner products as $\mathbb{P}_D\left\{Y \neq c(\mathbf{X})\right\} = \frac{1}{2} - \frac{1}{2}\langle \mathbf{Y}, c \rangle_D = \frac{1}{4}|\mathbf{Y} - c|^2_{2, D}.$ (1)

where, with slight abuse of notation, $Y$ is understood as the mapping $(x, y) \mapsto y$ and $c$ is understood as a mapping on $\mathcal{X} \times \mathcal{Y}$ which depends only on $\mathcal{X}$. Similarly, the empirical error of $c$ is equal to $\hat{\mathbb{P}}_{\hat{D}}\left\{Y \neq c(\mathbf{X})\right\} = \frac{1}{2} - \frac{1}{2}\langle \hat{\mathbf{Y}}, c \rangle_{\hat{D}} = \frac{1}{4}|\hat{\mathbf{Y}} - c|^2_{2, \hat{D}}.$

The goal now is to derive bounds on the minimum generalization error when learning with respect to various hypothesis classes. In Section III we describe $L_2$-polynomial regression and the Fourier algorithm, in Section IV we study polynomial-approximated classes, and finally in Section V we discuss half-spaces, and more general hypothesis classes that have structural properties.

III. PAC LEARNING WITH $L_2$-POLYNOMIAL REGRESSION

We employ a PAC learning algorithm using $L_2$-polynomial regression. Given a training set, the objective of the polynomial regression is to minimize the empirical square loss over all polynomials of degree up to $k$. This process can be implemented by stochastic gradient descent or by solving a linear system of equations. We describe how this polynomial
regression can be used for PAC learning. Let \( \hat{p} \) be the output of the polynomial regression. The idea is to shift the polynomial \( \hat{p} \) by a threshold \( \theta \) and take its sign. This process is demonstrated as Algorithm 1.

Algorithm 1 PAC Learning with \( L_2 \)-Polynomial Regression

Input: Degree parameter \( k \), and training samples \( \{(x(i), y(i)), i \in [n]\} \).

1: Find a polynomial \( \hat{p} \) of degree up to \( k \) that minimizes
\[
\frac{1}{n} \sum_{i} (y(i) - p(x(i)))^2.
\]

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

3: return \( \hat{g} = \text{sign}(\hat{p} - \theta) \).

A. Fourier-Based Learning Algorithm

We present another variant of \( L_2 \) polynomial regression, known as the low-degree (Fourier) algorithm [1]. Although this algorithm is more efficient than the polynomial regression, it requires binary input set \( X = \{-1, 1\}^d \). The low-degree algorithm was originally designed for uniform distribution on the Boolean cube. In this paper, we present a more general version of it for incorporating non-uniform but product probability distributions on \( \{-1, 1\}^d \) [17]. In this approach, the objective is to find an estimate of the \( p^* \) polynomial that minimizes the square loss \( \|Y - p^*\|_{2,D} \) under the true distribution. This method is based on the Fourier expansion on the Boolean cube [18] and is summarized in the following.

Under product probability distribution on \( \{-1, 1\}^d \), any bounded real-valued functions can be written as
\[
f(x) = \sum_{S \subseteq [d]} f_S \psi_S(x),
\]
where \( f_S \)'s are the Fourier coefficients and calculated as \( f_S \hat{=} \langle f, \psi_S \rangle \) for every subset \( S \subseteq [d] \). Further, the parity \( \psi_S \) is a monomial defined as
\[
\psi_S(x) = \prod_{j \in S} \frac{x_j - \mu_j}{\sigma_j},
\]
with \( \mu_j \) and \( \sigma_j \) being the mean and standard deviation of the \( X_j \), respectively. As the distribution is unknown, these quantities are estimated in the algorithm.

As a result, we can write the Fourier decomposition of the optimal polynomial \( p^* \). For that, we have the following statement:

Fact 1. Let \( D \) be a probability distribution with the marginal \( D_X \) that is a product probability distribution on \( \{-1, 1\}^d \). Then, the optimal polynomial \( p^* \) admits the following Fourier decomposition
\[
p^* = \sum_{S \subseteq [d]: |S| \leq k} \langle Y, \psi_S \rangle \psi_S.
\]

With that decomposition, the idea behind the Fourier algorithm is to compute an empirical estimate of \( \langle Y, \psi_S \rangle \). This is demonstrated as Algorithm 2.

Algorithm 2 Fourier-Based Learning

Input: Training samples \( \{(x(i), y(i)), i \in [n]\} \).

1: Compute the empirical mean \( \hat{\mu}_j \) and standard deviation \( \hat{\sigma}_j \) of each feature.

2: For every \( S \subseteq [d] \) with \( |S| \leq k \), construct the empirical parity as \( \psi_S(x) = \prod_{j \in S} \frac{x_j - \hat{\mu}_j}{\hat{\sigma}_j} \).

3: Compute the empirical Fourier coefficients \( a_S \), for every \( S \) with at most \( k \) elements, as
\[
a_S = \frac{1}{n} \sum_{i=1}^{n} y(i) \psi_S(x(i)).
\]

4: Construct and return the function \( \hat{\Pi}_Y \) as
\[
\hat{\Pi}_Y(x) \hat{=} \sum_{S: |S| \leq k} a_S \psi_S(x).
\]

IV. POLYNOMIALLY APPROXIMATED CLASS

In this section, we study agnostic PAC learning with respect to concept classes whose members are approximated by fixed-degree polynomials. We adopt the Hilbert space representation in Section II-A to analyze PAC learning using Algorithm 1 and 2. We start with the following formulation:

Definition 1. Given \( \epsilon \in [0, 1] \), \( k \in \mathbb{N} \) and any probability distribution \( D_X \) on \( X \), a concept class \( C \) of functions \( c : X \mapsto \{-1, 1\} \) is \( (\epsilon, k) \)-approximated if
\[
\sup_{c \in C} \inf_{p \in P_k} \mathbb{E}[ (c(X) - p(X))^2 ] \leq \epsilon^2,
\]
where \( P_k \) is the set of all polynomials of degree up to \( k \).

We consider agnostic PAC learning with respect to \( C \) and the 0 \(- 1 \) loss function. The minimum generalization error and empirical error of \( C \) are, respectively, defined as
\[
P_{opt} \hat{=} \min_{c \in C} \mathbb{P}_D \{ Y \neq c(X) \},
\]
\[
\hat{P}_{opt} \hat{=} \min_{p \in P_k} \mathbb{P}_D \{ Y \neq c(X) \}.
\]
We use the Hilbert space representation in Section II-A and provide a lower bound on $P_{\text{opt}}$.

**Lemma 2.** The minimum generalization error attainable by any $(\epsilon, k)$ concept class $C$ is bounded from below as

$$P_{\text{opt}} \geq \frac{1}{2} - 2\|p^*\|_1, \quad D - \epsilon,$$

where $p^* = \arg\min_{p \in P_k} E_D[(Y - p(X))^2]$.

We show in Section III-A that the lower-bound in Lemma 2 helps to prove our results for the low-degree algorithm.

### A. PAC Learning Bounds

Next, we analyze Algorithm 1 and 2 for this class and prove the first main result of the paper.

**Theorem 1.** Given $\epsilon > 0$ and $k \in \mathbb{N}$, the degree $k$ $L_2$ polynomial regression agnostically PAC learns any $(\epsilon, k)$-approximated concept class with expected error up to

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

where $d$ is the number of input variables and $n$ is the sample size.

**Proof.** To derive an upper bound on the empirical error of $\hat{g}$, we consider a weaker version of the algorithm. The idea is to select $\theta$ randomly instead of optimizing it as in the algorithm. For that, we establish the following lemma.

**Lemma 3.** Suppose $\theta$ 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 $p$,

$$E_\theta \left[ \hat{P} \left\{ Y \neq \text{sign}[p(X) - \theta] \right\} \right] \leq \frac{1}{2} \|Y - p\|_{2, \hat{D}}^2.$$

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

$$E_\theta \left[ \hat{P} \left\{ Y \neq \text{sign}[p(X) - \theta] \right\} \right] = \frac{1}{n} \sum_i E_\theta \left[ \mathbb{I}\{y_i \neq \text{sign}[p(x_i) - \theta]\} \right] = \frac{1}{n} \sum_i \hat{P} \left\{ \theta \in [p(x_i), y_i] \cup [y_i, p(x_i)] \right\}.$$  \hspace{1cm} (3)

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

$$P_{\epsilon} = \hat{P} \left\{ \theta \in [p(x_i), 1] \right\} = \int_{p(x_i)}^1 (1 - t) dt = \frac{1}{2}(1 - p(x_i))^2 = \frac{1}{2}(y_i - p(x_i))^2.$$  \hspace{1cm} (4)

If $p(x_i) \in [-1, 0]$, then

$$\hat{P} \left\{ \theta \in [p(x_i), 1] \right\} = \int_{p(x_i)}^0 (1 - t) dt = \frac{1}{2}(1 - p(x_i))^2 = \frac{1}{2}(y_i - p(x_i))^2.$$  \hspace{1cm} (5)

Lastly, if $p(x_i) < -1$, then $P_{\epsilon} = 1$ because $\theta \geq -1$. In this case also $P_{\epsilon} \leq \frac{1}{2}(y_i - p(x_i))^2$. The case for $y_i = -1$ follows by symmetry. Hence, we obtain the following inequality

$$E_\theta \left[ \hat{P} \left\{ Y \neq \hat{g}(X) \right\} \right] \leq \frac{1}{n} \sum_i \frac{1}{2}(y_i - p(x_i))^2.$$  \hspace{1cm} (6)

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

Consequently, from the lemma and due the fact that $\theta$ in the algorithm is selected to minimize the empirical error, we obtain that

$$\hat{P} \left\{ Y \neq \hat{g}(X) \right\} \leq \frac{1}{2} \|Y - \hat{p}\|_{2, \hat{D}}^2,$$  \hspace{1cm} (7)

where $\hat{p}$ is the output of $L_2$-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 $(\epsilon, k)$-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 (7) satisfies

$$\frac{1}{2} \|Y - \hat{p}\|_{2, \hat{D}}^2 \leq \frac{1}{2} \|Y - p^*\|_{2, \hat{D}}^2.$$  \hspace{1cm} (8)

We proceed by taking the expected error of the empirical error with respect to the random training samples. From (4) and (5) we obtain the following inequalities

$$E \left[ \hat{P} \left\{ Y \neq \hat{g}(X) \right\} \right] \leq \frac{1}{2} E \left[ \|Y - p^*\|_{2, \hat{D}}^2 \right] = \frac{1}{2} \|Y - p^*\|_{2, \hat{D}}^2 \leq \frac{1}{2} \left( \|Y - c^*\|_{2, \hat{D}} + \|p^* - c^*\|_{2, \hat{D}} \right)^2 \leq \frac{1}{2} \left( \|Y - c^*\|_{2, \hat{D}} + \epsilon \right)^2 \leq \frac{1}{2} \left( \|Y - c^*\|_{2, \hat{D}} + 4\epsilon + \epsilon^2 \right),$$  \hspace{1cm} (9)

where (a) holds from Minkowski’s inequality for 2-norm, (b) holds as $\|Y - c^*\|_{2, \hat{D}} \leq 2$, and (c) holds because of the second equality in (1) and that $P_{\text{opt}} = P\{Y \neq c^*(X)\}$.

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 up to $k$ does not exceed $d^{k+1}$. Therefore, from VC
theory (See Corollary 3.19 in [19]) for any $\delta$, with probability at least $(1 - \delta)$, the following inequality holds

$$
\mathbb{P}\{Y \neq \hat{g}(X)\} \leq \mathbb{P}\{Y \neq \hat{g}(X)\} + \left( \sqrt{\frac{2}{n}} \log \frac{en}{\delta} + \frac{\log \frac{1}{\delta}}{2n} \right) \cdot (7)
$$

Set $\delta = \exp\left[-\frac{1}{2}n\epsilon^2\right]$. Therefore, the proof is complete by taking the expectation and combining it with the last bound in (6).

**PAC bounds for the Fourier algorithm:** Next, we employ a low-degree (Fourier) algorithm (Algorithm 2) for PAC learning with respect to the polynomially approximated hypothesis class.

**Theorem 2.** Let $D$ be a joint probability distribution with marginal $D_X$ that is a product probability distribution on $\{-1, 1\}^d$. Then, for any $\delta \in [0, 1]$, with probability at least $1 - \delta$, the Fourier-based algorithm agnostically PAC learns any $(\epsilon, k)$-approximated concept class with generalization error up to

$$
2P_{opt} + 2\epsilon + O\left( \frac{d^k c_k}{(k - 1)!} \log \frac{4d^k}{(k - 1)!} \right),
$$

where $c_k \triangleq \max_{S \subseteq [d], |S| \leq k} \|\psi_S\|_2^2$.

**Corollary 1.** If the expected value of each $X_j$ satisfies $|\mu_j| \leq 1 - \frac{1}{k}$, then the generalization error of the Fourier algorithm is upper bounded by

$$
2P_{opt} + 2\epsilon + O\left( \frac{\sqrt{k}(cd)^k}{n} \log \frac{ed}{k} + \log \frac{2\sqrt{k}}{\delta} \right).
$$

**V. Learning Other Hypothesis Classes**

In this section, we extend our results to two other type of concept classes. The first one is called half-spaces and the other one is a generalized version of the concentrated hypothesis classes.

**A. Half-spaces**

In this section, we consider learning another class of functions called half-spaces. More precisely, a half-space is a Boolean-valued function of the form

$$
c(x) = \text{sign}\left[ a_0 + \sum_{j=1}^{d} a_j x_j \right], \quad \forall x \in \mathbb{R}^d
$$

where $a_j \in \mathbb{R}$. We start with a lower-bound on the optimal classification error of the class.

**Lemma 4.** Let $D$ be any joint probability distribution on $\mathbb{R}^d \times \{-1, 1\}$ with marginal $D_x$ that is the uniform distribution on $\mathbb{S}^{d-1}$ or jointly Gaussian on $\mathbb{R}^d$. Then, for any $\epsilon > 0$, the minimum generalization error of learning with respect to half-spaces satisfy the following lower bound

$$
P_{opt} \geq \frac{1}{2} - \frac{1}{2} \|p^*\|_{1, D_X} - \epsilon,
$$

where $p^*$ is a polynomial of degree up to $O(\frac{1}{\epsilon})$ minimizing $\|Y - p\|_{2, D}$ among all such polynomials.

The proof of the lemma follows from Lemma 2 and [5]’s result (Theorem 6) on the sign function. This result is stated as

**Lemma 5 ([5]).** Let $X$ be a random variable with uniform distribution on $\mathbb{S}^{d-1}$ or jointly Gaussian on $\mathbb{R}^d$. Then, for any $\epsilon > 0$, there exists a polynomial $p$ of degree $O(\frac{1}{\epsilon})$ such that $\mathbb{E}\left[ (p(X) - \text{sign}(X))^2 \right] \leq \epsilon^2$.

This lemma makes a connection between half-spaces and the polynomial-approximated class. That said, in the following theorems we show our results for PAC learning using Algorithm 1.

**Theorem 3.** Let $D$ be any joint probability distribution on $\mathbb{R}^d \times \{-1, 1\}$, with marginal $D_X$ that is uniform on the unit sphere or jointly Gaussian. Then, $L_2$-polynomial regression PAC learns half-spaces with expected generalization error up to

$$
2P_{opt} + 3\epsilon + O\left( \frac{dO(\frac{1}{\epsilon})}{n} \log \frac{n}{dO(\frac{1}{\epsilon})} \right).
$$

**B. Generalized approximated class**

Lastly, we finish this paper by extending our results to a more general hypothesis class. Fix a set of functions $e_1(x), e_2(x), ..., e_n(x)$ and let $\mathcal{H}$ be a Hilbert space spanned by these functions. Let $C$ be a class of functions each of which approximated by elements of $\mathcal{H}$ with square error up to $\epsilon$, that is,

$$
\inf_{h \in \mathcal{H}} \|c - h\|_{2, D} \leq \epsilon,
$$

for any $c \in C$. As a special case, suppose $e_i$’s are all the functions of the form $e(x) = \prod_{j=1}^{d} x_j^{\alpha_j}$ where $\alpha_j$’s are non-negative integers adding up to $k$. Then $C$ is a $(k, \epsilon)$-approximated class as in Section IV.

**Theorem 4.** Suppose $A$ is any algorithm that given $n$ training instances finds a function $h \in \mathcal{H}$ so that the empirical loss $\|Y - h\|_{2, D}$ is minimized. Then, the predictor $\text{sign}[\hat{h}]$ learns $C$ with expected generalization error up to

$$
2P_{opt} + 3\epsilon + O\left( \frac{\text{VC}(C)}{n} \log \frac{n}{\text{VC}(C)} \right),
$$

where $\text{VC}(C)$ is the VC dimension of $C$.

**ACKNOWLEDGEMENT**

This work was supported in part by NSF Center on Science of Information Grants CCF-0939370 and NSF Grants CCF-1524312, CCF-2006440, CCF-2007238, and Google Research Award.
REFERENCES


