Learning Nested Halfspaces and Uphill Decision Trees

Adam Tauman Kalai[*]

Georgia Tech
http://www.cc.gatech.edu/~atk

Abstract. Predicting class probabilities and other real-valued quantities is often more useful than binary classification, but comparatively little work in PAC-style learning addresses this issue. We show that two rich classes of real-valued functions are learnable in the probabilistic-concept framework of Kearns and Schapire.

Let X be a subset of Euclidean space and f be a real-valued function on X. We say f is a *nested halfspace function* if, for each real threshold t, the set $\{x \in X | f(x) \leq t\}$, is a halfspace. This broad class of functions includes binary halfspaces with a margin (e.g., SVMs) as a special case. We give an efficient algorithm that provably learns (Lipschitz-continuous) nested halfspace functions on the unit ball. The sample complexity is independent of the number of dimensions.

We also introduce the class of *uphill decision trees*, which are real-valued decision trees (sometimes called *regression trees*) in which the sequence of leaf values is non-decreasing. We give an efficient algorithm for provably learning uphill decision trees whose sample complexity is polynomial in the number of dimensions but independent of the size of the tree (which may be exponential). Both of our algorithms employ a real-valued extension of Mansour and McAllester's boosting algorithm.

1 Introduction

Consider the problem of predicting whether a patient will develop diabetes ($y \in \{0, 1\}$) given n real valued attributes ($x \in \mathbb{R}^n$). A real prediction of $\Pr[y = 1|x]$ is much more informative than the binary prediction of whether $\Pr[y = 1|x] > 1/2$ or not. Hence, learning probabilities and, more generally, real-valued functions has become a central problem in machine learning.

This paper introduces algorithms for learning two classes of real-valued functions. The first is the class of *nested halfspace functions* (NHFs), and the second is that of *uphill decision trees*. These are real-valued classes of functions which naturally generalize halfspaces and decision lists, respectively. We believe that these classes of functions are much richer than their binary classification counterparts.

[*] Supported in part by NSF award SES-527656. Part of this work was done while the author was visiting the Weizmann Institute.

Kearns and Schapire give a rigorous definition of learning such probabilistic concepts [5] in which there is a set X, and a distribution $\mathcal{D}$ over $(x, y) \in X \times \{0, 1\}$. The learner's goal is to predict $f(x) = \Pr_{\mathcal{D}}[y = 1|x]$ as accurately as possible. Roughly speaking, a learning algorithm *learns* a family $\mathcal{C}$ of *concepts* $c : X \to [0, 1]$ if, for **any** distribution $\mathcal{D}$ such that $f(x) = \Pr_{(x,y)\sim\mathcal{D}}[y = 1|x] \in \mathcal{C}$, with high probability it outputs an *hypothesis* $h : X \to [0, 1]$ such that,

$$\mathrm{E}_{\mathcal{D}}[(h(x) - f(x))^2] \leq \epsilon. \tag{1}$$

The algorithm should be computationally efficient and use only $\mathrm{poly}(1/\epsilon)$ independent samples from $\mathcal{D}$.

Remark 1. Two remarks from [5] elucidate the power of their probabilistic learning model. First of all, without loss of generality one can allow $y \in [0, 1]$ (the generalization to any interval $[a, b]$ is straightforward) as long as $f \in \mathcal{C}$ where now $f(x) = \mathrm{E}[y|x]$. The reason is that, one can *randomly round* any example in $(x, y) \in X \times [0, 1]$ to be in $X \times \{0, 1\}$ by choosing $(x, 1)$ with probability y and $(x, 0)$ with probability $1 - y$. This does not change $\mathrm{E}[y|x]$ and converts a distribution over $X \times [0, 1]$ to be over $X \times \{0, 1\}$. Such a setting models real-valued prediction, e.g., estimating the value of a used car from attributes.[1]

Remark 2. In expanding $\mathrm{E}[(h(x) - f(x) + f(x) - y)^2]$, for $f(x) = \mathrm{E}[y|x]$ and any hypothesis h, the cross-term $\mathrm{E}[(h(x) - f(x))(f(x) - y)] = 0$ cancels. So,

$$\mathrm{E}_{(x,y)\sim\mathcal{D}}[(h(x) - y)^2] = \mathrm{E}[(h(x) - f(x))^2] + \mathrm{E}[(f(x) - y)^2]. \tag{2}$$

Hence, a hypothesis meeting (1) not only makes predictions $h(x)$ that are close to the *truth* $f(x)$, but also has expected *squared error* $\mathrm{E}[(h(x) - y)^2]$ within ϵ of the minimal squared error that could achieve knowing f.

1.1 Nested Halfspace Functions and Uphill Decision Trees

Let $X \subseteq \mathbb{R}^n$ and define a function $f : X \to \mathbb{R}$ to be an NHF if for every $t \in \mathbb{R}$, the set of x such that $f(x) \leq t$ is a halfspace. More formally, for all $t \in \mathbb{R}$, there must exist $w \in \mathbb{R}^n, \theta \in \mathbb{R}$ such that

$$\{x \in X \mid f(x) \leq t\} = \{x \in X \mid x \cdot w \leq \theta\}.$$

We call this a *nested* halfspace function because for thresholds $t < t'$, the set $H_t = \{x \in X \mid f(x) \leq t\}$ must be contained in $H_{t'} = \{x \in X \mid f(x) \leq t'\}$ and both must be halfspaces.

When $X = \mathbb{R}^n$, the NHFs reduce[2] to the class of *generalized linear models* where it is required that $f(x) = u(w \cdot x)$ where $w \in \mathbb{R}^n$ and $u : \mathbb{R} \to \mathbb{R}$ is a nondecreasing function. Generalized linear models include halfspaces as well as linear and logistic regression as special cases.

[1] Note that the learner does not directly get samples $(x, f(x))$. If $y = f(x)$ always was the case (a special distribution), then learning nested halfspaces would be trivial, as the learner could learn $f(x) \leq t$ for each t himself by thresholding the data to see which $y \geq t$ and then running a standard halfspace learning algorithm.

[2] Technically we need to also permit the set $\{x \in X \mid f(X) \leq t\}$ to be an open halfspace $\{x \in X \mid x \cdot w < \theta\}$, but for simplicity, we consider only closed halfspaces.

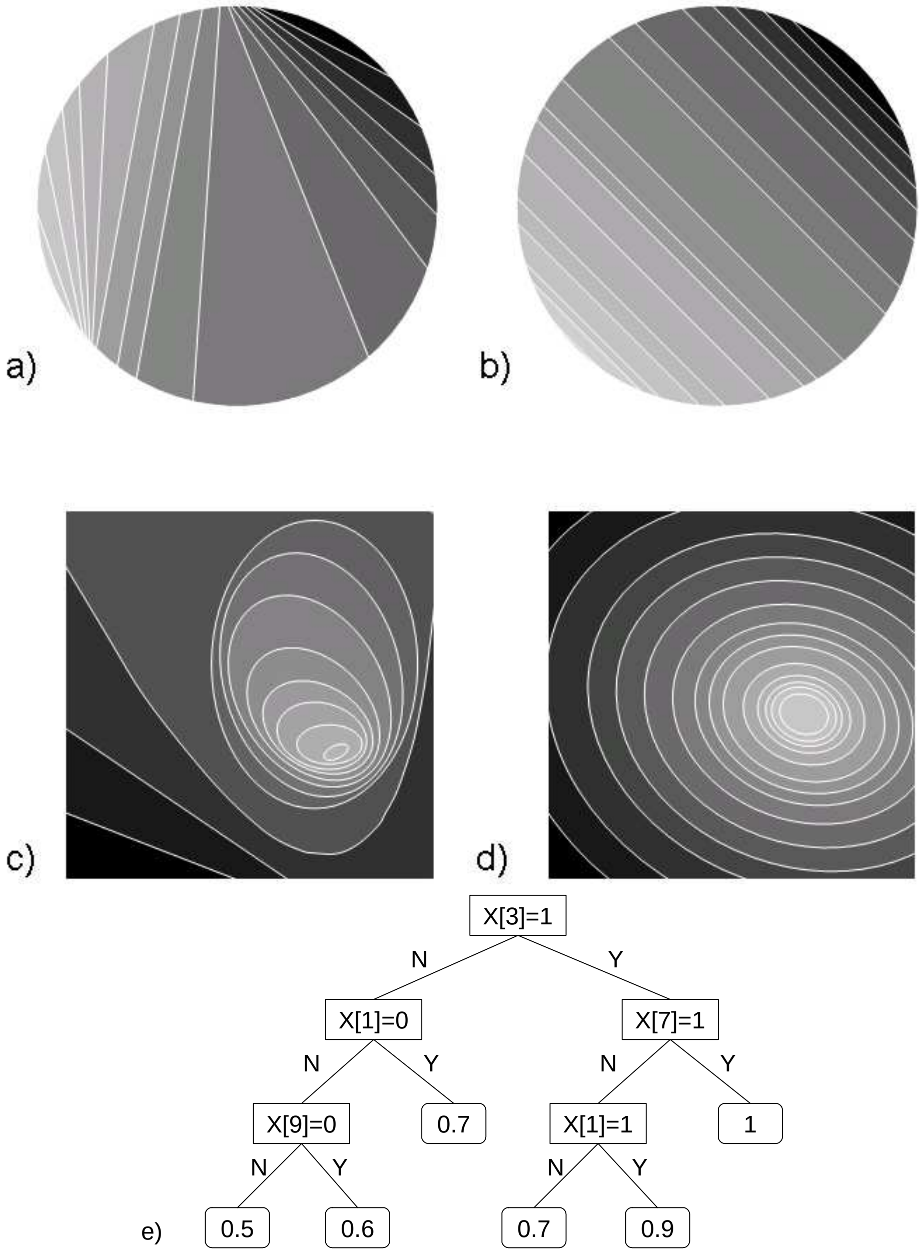

Fig. 1. a-d) Illustrations of NHFs. Whiter=larger function value. a) An NHF on a ball. b) A generalized linear model, where all halfspaces must be parallel. c) An NHF with a degree-2 polynomial kernel. The function can be a potpourri of ellipses, parabolas, etc., at varying orientations. d) A generalized linear model with a degree-2 polynomial kernel. The objects must be concentric, of the same type and same orientation. e) An uphill decision tree.

For general $X \subseteq \mathbb{R}^n$, the NHFs have much more flexibility. Figure 1 illustrates NHFs overs the unit ball, and kernelized NHFs with a degree 2 polynomial kernel. There are several additional characterizations of NHFs. For example, an NHF has the property that the restriction of the function to any line is either nondecreasing or nonincreasing.

In Section 3, we give an algorithm for learning continuous NHFs over the unit ball $B = \{x \in \mathbb{R}^n \mid \|x\| \leq 1\}$. A function $f : X \to \mathbb{R}$ is said to be L-*Lipschitz* continuous if,

$$\forall x, x' \in X \quad |f(x) - f(x')| \leq L\|x - x'\|.$$

(For differentiable f and convex X, this is equivalent to requiring $\|\nabla f(x)\| \leq L$ for all $x \in X$). Formally, the class of functions we learn is,

$$\mathrm{NHF}_L = \{f : B \to [0, 1] \mid f \text{ is } L\text{-Lipschitz and an NHF}\}.$$

Result 1 *The class* NHF_L *for any dimension* n *can be learned to error* $\leq \epsilon$ *using a computationally efficient algorithm and poly*$(1/\epsilon, L)$ *data.*

Notice that the amount of data does not depend on n, and the algorithm can be naturally Kernelized. In this sense, the parameter L plays the analogous role to a margin parameter of $1/L$.

1.2 Uphill decision trees

In this case, let $X = \{0, 1\}^n$. A decision tree with real values at the leaves (sometimes called a regression tree) is called an uphill decision tree if the values on the leaves are nondecreasing in order from left to right. Formally, this means that there should be some depth-first traversal of the tree in which the values encountered at the leaves occur in nondecreasing order. An example of an uphill decision tree is given in Figure 1e). Note that uphill (or downhill) decision trees differ from the notion of *monotonic decision trees* (see, e.g., [10]), which require that changing any single attribute from 0 to 1 can only increase (or decrease) the function. The example in Figure 1e) demonstrates that an uphill decision tree need not be monotonic: changing $x[1]$ from 0 to 1 may increase or decrease the value, depending on the other attributes.

Result 2 *The class of uphill decision trees (of any size) in any dimension* n *can be learned to error* $\leq \epsilon$ *using a computationally efficient algorithm and poly*$(n, 1/\epsilon)$ *data.*

We note that the set of uphill decision trees with $\{0, 1\}$ leaf values is exactly the same as the set of binary decision lists. Similarly, thresholding an uphill decision tree gives a binary decision list. Hence uphill decision trees are in fact a special case of NHFs. However, we cannot use our result (or algorithm) from the previous section here because it has incompatible conditions.

It is also worth defining Kearns and Schapire's notion of a Probabilistic decision list [5]. This is a decision list with real values in the leaves. However, there is the additional restriction that the real values have to be an interleaving of two sequences: one nonincreasing and the other nondecreasing up until some common value θ. This seemingly strange condition actually makes them a special case of uphill decision trees.

2 Preliminaries: Real-Valued Learning

Concept class of real functions	Binary $\{0,1\}$ special case
Monotonic functions of a single variable [5]	1-dimensional threshold functions
Functions of (const.) k relevant variables [5]	Binary functions of k relevant variables
Probabilistic decision lists [5]	Decision lists
(Lipschitz) Generalized Linear Models [3, 8]	Halfspaces(+margin)
(Lipschitz) Generalized Additive Models [3, 2]	Additive threshold functions(+margin)
(Lipschitz) NHFs	Halfspaces(+margin)
Uphill decision trees	Decision lists

Following Kearns and Schapire [5], we assume that we have a set X, a probability distribution $\mathcal{D}$ over $X \times [0,1]$, and a family $\mathcal{C}$ of *concepts* $f : X \rightarrow [0,1]$ such that $f(x) = \mathrm{E}_{(x,y)\sim\mathcal{D}}[y|x] \in \mathcal{C}$.[3] An example oracle $EX = EX_{\mathcal{D}}$ is an oracle that, each time called, returns an independent draw (x,y) from distribution $\mathcal{D}$. That is, if the algorithm calls the oracle m times, it receives samples $(x_1,y_1),\ldots,(x_m,y_m)$ which are i.i.d. from $\mathcal{D}$. In the case where $X \subseteq \mathbb{R}^n$, we denote the ith attribute $x[i] \in \mathbb{R}$.

Definition 1 (Polynomial learning in the real-valued setting). *A (possibly randomized) learning algorithm L takes as input $\epsilon, \delta > 0$ and EX and outputs a function $h : X \rightarrow [0,1]$. L* polynomially *learns $(X, \mathcal{C})$ if there exists a polynomial $p(\cdot,\cdot)$ such that: for any $\epsilon, \delta > 0$ and any distribution $\mathcal{D}$ over $X \times [0,1]$ whose $f(x) = \mathrm{E}_{(x,y)\sim\mathcal{D}}[y|x] \in \mathcal{C}$, with probability $1-\delta$ over the samples returned by EX (and its internal randomization), it outputs $h : X \rightarrow [0,1]$ such that, $\mathrm{E}_{(x,y)\sim\mathcal{D}}[(h(x)-f(x))^2] \leq \epsilon$* and *with probability 1, the runtime of the algorithm (hence number of calls to EX as well) and the runtime of h on any $x \in X$, are at most $p(1/\epsilon, 1/\delta)$.*

Remark. Often times we are interested in the asymptotic behavior of an algorithm on sequence of learning problems $(X_n, \mathcal{C}_n)$ for $n = 1, 2, \ldots$. In this case, n is an input to the learning algorithm as well, and the above requirement must hold for any $n \geq 1$ and learning problem $(X_n, \mathcal{C}_n)$, and the runtime may grow with n but must be polynomial in n as well. In this case we say the algorithm polynomially learns $\{(X_n, \mathcal{C}_n)\}$.

[3] This means $f(z) = E[y|x=z]$ if we think of (x,y) as joint random variables drawn according to $\mathcal{D}$. For x not in the support of $\mathcal{D}$, $f(x) \in [0,1]$ may be chosen arbitrarily so that $f \in \mathcal{C}$.

2.1 Covariance and Correlation

While $\{0, 1\}$ error rate is often the most useful metric in designing binary classification algorithms, in the real-valued setting notions of variance, covariance and correlation often prove useful.

For random variables $A, B \in \mathbb{R}$, define the *covariance* to be,

$$\text{cov}(A, B) = \mathrm{E}[AB] - \mathrm{E}[A]\mathrm{E}[B] = \mathrm{E}[(A - \mu_A)(B - \mu_B)] = \mathrm{E}[(A - \mu_A)B],$$

where $\mu_A = \mathrm{E}[A]$ denotes the expectation of random variable A. We note that for any constant $c \in \mathbb{R}$, $\text{cov}(A+c, B) = \text{cov}(A, B)$. Also, covariance is symmetric ($\text{cov}(A, B) = \text{cov}(B, A)$) and bilinear (for any random variables $A_1, A_2, B \in \mathbb{R}$ and constants $c_1, c_2 \in \mathbb{R}$, $\text{cov}(c_1 A_1 + c_2 A_2, B) = c_1 \text{cov}(A_1, B) + c_2 \text{cov}(A_2, B)$). Define the *variance* of A to be

$$\text{var}(A) = \text{cov}(A, A) = \mathrm{E}[A^2] - \mathrm{E}[A]^2 = \mathrm{E}[(A - \mu_A)^2].$$

We assume that we have some distribution $\mathcal{D}$ over $X \times [0, 1]$. For functions $g, h : X \to \mathbb{R}$, we define

$$\text{cov}(g(x), h(x)) = \mathrm{E}_{(x,y)\sim\mathcal{D}}[g(x)h(x)] - \mathrm{E}[g(x)]\mathrm{E}[h(x)] = \mathrm{E}[(g(x) - \mu_g)h(x)],$$

where $\mu_g = \mathrm{E}[g(x)]$. Similarly, define $\text{var}(g(x)) = \mathrm{E}[g^2(x)] - \mathrm{E}[g(x)]^2$. Note that $\text{var}(f(x)) = 0$ has special meaning. It means that $f(x) = \mathrm{E}[y|x]$ is constant for all x in the support of $\mathcal{D}$, hence the most accurate hypothesis to output is this constant function and no better learning is possible.

Note that for f and any $h : X \to \mathbb{R}$ we also have

$$\text{cov}(f(x), h(x)) = \mathrm{E}[f(x)h(x)] - \mathrm{E}[f(x)]\mathrm{E}[h(x)] = \mathrm{E}[yh(x)] - \mu_f \mathrm{E}[h(x)].$$

Hence we have the useful relation, for any $h : X \to \mathbb{R}$:

$$\text{cov}(f(x), h(x)) = \mathrm{E}[yh(x)] - \mathrm{E}[y]\mathrm{E}[h(x)] = \text{cov}(y, h(x)). \qquad (3)$$

We also refer to $\text{cov}(y, h(x))$ as the *true* covariance of h in analogy to the *true error* (also called *generalization error*) of an algorithm outputting $h : X \to \mathbb{R}$. For such an algorithm, we refer to the *expected covariance* $\mathrm{E}[\text{cov}(y, h(x))]$, where the expectation is over draws of the training set $\mathcal{Z}_m = (x_1, y_1), \ldots, (x_m, y_m)$ drawn i.i.d. from $\mathcal{D}$, and we are talking about the expectation, over datasets of the true covariance. In particular, this is, $\mathrm{E}_{\mathcal{Z}_m \sim \mathcal{D}^m}[\text{cov}(y, h(x))]$ and is not to be confused with the *empirical covariance* $\widehat{\text{cov}}(y, h(x))$ defined as follows.

$$\widehat{\text{cov}}(y, h(x)) = \frac{1}{m}\sum_{i=1}^{m} y_i h(x_i) - \frac{1}{m}\sum_{i=1}^{m} y_i \frac{1}{m}\sum_{i=1}^{m} h(x_i) = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{\mu}_f)h(x_i),$$

where we define $\hat{\mu}_f = \frac{1}{m}\sum_{i=1}^{m} y_i$.

Finally, for random variables A, B, we define the *correlation coefficient*,

$$\text{cor}(A, B) = \frac{\text{cov}(A, B)}{\sqrt{\text{var}(A)\text{var}(B)}} \in [-1, 1].$$

Note that $\text{cor}(c_1 A + c_2, B) = \text{cor}(A, B)$ for constants $c_1 > 0, c_2 \in \mathbb{R}$. Similarly for $g, h : X \to \mathbb{R}$, we define $\text{cor}(g(x), h(x)) = \text{cov}(g(x), h(x))/\sqrt{\text{var}(h(x))\text{var}(g(x))}$.

2.2 Real boosting

Classification boosting [11] is an extremely useful tool for designing provably efficient learning algorithms. In order to learn real-valued functions, we need to use the real-valued analog of boosting. In [3], it was shown that the boosting by branching programs algorithm of Mansour and McAllester [7] (building on work of Kearns and Mansour [4]) can be adapted to the real-valued setting.

In classification boosting, a weak learner was defined [6] to be an algorithm whose output had error strictly less than 1/2 ($\leq 1/2 - \gamma$ for $\gamma > 0$). In the real-valued setting, this definition does not make sense. Instead, the definition we use requires positive correlation rather than error less than 1/2. Note that in the real-valued setting, our definition of a weak learner is complicated by the fact that when $\mathrm{var}(f(x)) = 0$, it is impossible to have positive covariance (or positive correlation), i.e., $\mathrm{cov}(h(x), y) = 0$ for all $h : X \to \mathrm{I\!R}$.

Definition 2 (Weak correlator [3]). *Let $\rho : [0,1] \to [0,1]$ be a nondecreasing function. A ρ-weak correlator for $(X, \mathcal{C})$ is a learning algorithm that takes input $\epsilon, \delta > 0$ and EX such that, for any $\epsilon, \delta > 0$, and any distribution $\mathcal{D}$ over $X \times [0,1]$ where $f(x) = \mathrm{E}[y|x] \in \mathcal{C}$ and $\mathrm{var}(f(x)) \geq \epsilon$, with probability $1 - \delta$, it outputs $h : X \to \mathrm{I\!R}$ such that $\mathrm{cor}(h(x), f(x)) \geq \rho(\epsilon)$.*

A weak correlator is said to be *efficient* if its runtime (hence number of calls to EX) and the runtime of evaluating h, are polynomial in $1/\epsilon, 1/\delta$ and if $1/\rho(\epsilon)$ is polynomial in $1/\epsilon$ as well. In the case where we consider a sequence of learning problems $\{(X_n, \mathcal{C}_n)\}$, the runtime and $1/\rho$ must grow polynomially in n as well.

The following is shown:

Theorem 1 (Real-valued boosting [3]). *There is a boosting algorithm that, given any black-box efficient ρ-correlator for $(X, \mathcal{C})$, polynomially learns $(X, \mathcal{C})$ in the real-valued setting.*

A somewhat simpler notion of weak learner in the real-valued setting can be given, making analysis simpler. We define a simplified weak learner as follows.

Definition 3 (Simplified real weak learner). *Let $\sigma : [0,1] \to [0,1]$ be a nondecreasing function such that $1/\sigma(\epsilon)$ is polynomial in $1/\epsilon$ and let $q(\cdot)$ be a polynomial. The simplified real weak learner for $(X, \mathcal{C})$ is a learning algorithm that takes input $m \geq 1$ and training set $\mathcal{Z}_m = (x_1, y_1), \ldots, (x_m, y_m)$ drawn i.i.d. from $\mathcal{D}$ such that, for any $\epsilon > 0$ and any $m \geq q(1/\epsilon)$, and any distribution $\mathcal{D}$ over $X \times [0,1]$ where $f(x) = \mathrm{E}[y|x] \in \mathcal{C}$ and $\mathrm{var}(f(x)) \geq \epsilon$, it outputs $h : X \to [-1,1]$ such that $\mathrm{E}_{\mathcal{Z}_m \sim \mathcal{D}^m}\left[|\mathrm{cov}(h(x), y)|\right] \geq \sigma(\epsilon)$ and the runtime of the weak learner and h on any inputs in $(X \times [0,1])^m$ must be polynomial in m.*

We call this definition "simplified," because it involves covariance rather than correlation, and because it is arguably more natural to view a learning algorithm as taking a training set as input rather than desired accuracy and confidence parameters. In this way, we also avoid explicit dependence on $1/\delta$.

Again, in the case of a sequence of learning problems $\{(X_n, \mathcal{C}_n)\}$, the above guarantee must hold for any $n \geq 1$, but $1/\rho$, p, and the runtimes are allowed to be polynomial in n as well. Using standard techniques, given a simplified real weak learner for $\{X_n, \mathcal{C}_n\}_{n\geq 1}$, we can construct a efficient ρ-weak learner and hence polynomially learn the family.

Lemma 1. *Given a simplified weak learner for $(X, \mathcal{C})$, one can construct an efficient weak correlator (and hence a polynomial learner) for $(X, \mathcal{C})$.*

Proof (Sketch). Notice that for any $h : X \to [-1, 1]$, $\mathrm{var}(h(x)) \leq 1$. Since $f : X \to [0, 1]$, we have $\mathrm{var}(f(x)) \leq 1/4$. Hence,

$$\mathrm{cor}(h(x), f(x)) = \frac{\mathrm{cov}(h(x), y)}{\sqrt{\mathrm{var}(h(x))\mathrm{var}(f(x))}} \geq 2\mathrm{cov}(h(x), y).$$

Hence, to achieve $\geq \rho$ correlation, it suffices to output hypothesis $h : X \to [-1, 1]$ with $\mathrm{cov}(h(x), y) \geq \rho/2$.

We take $\rho(\epsilon) = \sigma(\epsilon)$. Given $\epsilon, \delta > 0$, we run the simplified weak learner $T = O(\log(1/\delta)/\epsilon)$ times on fresh data. For each run $t = 1, \ldots, T$, we have an output $h_t : X \to [0, 1]$. We use $O(\log(1/\delta)/\sigma^2(\epsilon))$ fresh random samples to estimate the covariance on a fresh set of held-out data set, and return $h(x) = h_t(x)$ or $h(x) = -h_t(x)$ of maximal empirical covariance. Since $\mathrm{cov}(-h_t(x)) = -\mathrm{cov}(h_t(x))$ and we are considering both possibilities for each t, WLOG we can assume that $cov(h_t(x), y) \geq 0$.

Now, for each $1 \leq t \leq T$, we have $\mathrm{E}_{\mathcal{Z}_m}[1 - \mathrm{cov}(h_t(x), y)] \leq 1 - \sigma(\epsilon)$ and also $\mathrm{cov}(h_t(x), y) \in [0, 1]$ since $h_t(x), f(x) \leq 1$. By Markov's inequality on $1 - \mathrm{cov}(h_t(x), y) \geq 0$, we have,

$$\Pr_{\mathcal{Z}_m \sim \mathcal{D}^m}[1 - \mathrm{cov}(h_t(x), y) \geq 1 - (3/4)\sigma(\epsilon)] \leq \frac{1 - \sigma(\epsilon)}{1 - (3/4)\sigma(\epsilon)} \leq 1 - \sigma(\epsilon)/4.$$

In other words, with probability $\geq \sigma(\epsilon)/4$, $\mathrm{cov}(h_t(x), y) \geq (3/4)\sigma(\epsilon)$. Thus, after $T = O(\log(1/\delta)/\sigma(\epsilon))$ repetitions of the algorithm, with probability $\geq 1 - \delta/2$, at least one of them will have $\mathrm{cov}(h_t(x), y) = \mathrm{cov}(h_t(x), f(x)) \geq (3/4)\sigma(\epsilon)$. If we measure the empirical covariance $\widehat{\mathrm{cov}}(h_t(x), y)$ of each on a test set of size $O(\log(1/\delta)/\sigma^2(\epsilon))$, with probability $\geq 1 - \delta/2$, all of them (including both $h_t(x)$ and $-h_t(x)$ will have empirical covariance within $\sigma(\epsilon)/8$ of their true covariance. Hence, by the union bound, with probability $\geq 1 - \delta$, we will output $h : X \to [-1, 1]$ with $\mathrm{cov}(h(x), y) \geq \sigma(\epsilon)/2 = \rho(\epsilon)/2$. □

The same lemma holds for $\{(X_n, \mathcal{C}_n)\}$ as the polynomial dependence on n trivially carries through the above reduction.

3 Learning Continuous NHFs

In this section we take $X = B = \{x \in \mathbb{R}^n \mid \|x\| \leq 1\}$ to be the unit ball, and we consider NHFs $f : B \to [0, 1]$ that are L-Lipschitz-continuous, for some value $L > 0$.

The idea is to use a linear $h(x) = w \cdot x$ weak learner that maximizes the empirical covariance with y on the data. This is easy to compute, as follows.

Define $\hat{\mu}_f = \frac{1}{m}\sum_{i=1}^m y_i$, and $v, \hat{v} \in \mathbb{R}^n$ by,

$$v = \mathrm{E}_{(x,y)\sim\mathcal{D}}[(y - \mu_f)x]$$
$$\hat{v} = \frac{1}{m}\sum_{i=1}^m (y_i - \hat{\mu}_f)x_i.$$

For any $w \in \mathbb{R}^n$, we have,

$$\mathrm{cov}(y, w \cdot x) = \mathrm{E}[(y - \mu_f)(w \cdot x)] = w \cdot \mathrm{E}[(y - \mu_f)x] = w \cdot v. \tag{4}$$

Similarly,

$$\widehat{\mathrm{cov}}(y, w \cdot x) = \frac{1}{m}\sum_{i=1}^m (y_i - \hat{\mu}_f)(w \cdot x_i) = w \cdot \frac{1}{m}\sum_{i=1}^m (y_i - \hat{\mu}_f)x_i = w \cdot \hat{v}. \tag{5}$$

Thus the vectors $u, \hat{u} \in B$ that maximize $\mathrm{cov}(y, u \cdot x)$ and $\widehat{\mathrm{cov}}(y, \hat{u} \cdot x)$ are $u = \frac{v}{\|v\|}$ and $\hat{u} = \frac{\hat{v}}{\|\hat{v}\|}$, respectively[4].

The main result of this section is that the (trivially efficient) algorithm that outputs hypothesis $h(x) = \hat{u} \cdot x$ is a simplified real weak learner. This directly implies Result 1, through Lemma 1.

Theorem 2. *For any $\epsilon > 0$ distribution $\mathcal{D}$ over $B \times \{0, 1\}$ such that $f \in \mathrm{NHF}_L$ and $\mathrm{var}(f) \geq \epsilon$, given $m \geq 100L^2/\epsilon^4$ examples, the vector $\hat{u} = \frac{\hat{v}}{\|\hat{v}\|}$ defined by $\hat{v} = \frac{1}{m}\sum_{i=1}^m (y_i - \hat{\mu}_f)x_i$ yields,*

$$\mathrm{E}_{\mathcal{Z}_m}[\mathrm{cov}(y, u \cdot x)] \geq \frac{1}{5L}\epsilon^2.$$

To prove this, we begin by claiming that there is some vector $w \in B$ such that $(w \cdot x)$ has relatively large covariance with respect to y.

Lemma 2. *Suppose $f \in \mathrm{NHF}_L$. Then there exists some vector $w \in B$ such that,*

$$\mathrm{cov}(y, w \cdot x) = \mathrm{cov}(f(x), w \cdot x) \geq \frac{4}{5L}\big(\mathrm{var}(f(x))\big)^2.$$

We will prove this lemma in Section 3.1. Finally, we also use the following generalization bound for covariance.

Lemma 3. *For any distribution $\mathcal{D}$ over $B \times [0, 1]$, any $\epsilon, \delta > 0$, and m samples iid from $\mathcal{D}$,*

$$\mathrm{E}\left[\sup_{w \in B} \left|\widehat{\mathrm{cov}}(y, w \cdot x) - \mathrm{cov}(y, w \cdot x)\right|\right] \leq \frac{3}{\sqrt{m}}.$$

[4] If v (or $\hat{v}$) is 0, we take $u = 0$ (resp. $\hat{u} = 0$).

Proof. By equations (4) and (5), we have for any $w \in B$,

$$|\widehat{\mathrm{cov}}(y, w \cdot x) - \mathrm{cov}(y, w \cdot x)| = |w \cdot (\hat{v} - v)| \leq \|\hat{v} - v\|.$$

Thus it suffices to show that $\mathrm{E}\left[\|\hat{v} - v\|^2\right] \leq 9/m$ because $\mathrm{E}[|Z|] \leq \sqrt{\mathrm{E}[Z^2]}$ for any random variable $Z \in \mathrm{I\!R}$.

Note that $\mathrm{E}[\hat{v}] = v$. Also note that $\hat{v}$, which is a function of the training data $(x_1, y_1), \ldots, (x_m, y_m)$ is *stable* in the following sense. If we change only one training example $(x_i, y_i) \in B \times [0, 1]$, this can move $\hat{v}$ by a vector of magnitude at most $3/m$. To see this, note that $(1/m)(y_i - \hat{\mu}_f)x_i$ is a vector of magnitude $\leq 1/m$ and hence changing (x_i, y_i) changes this by a vector of magnitude at most $2/m$. Also, changing (x_i, y_i) moves $(1/m)(y_j - \hat{\mu}_f)x_j$ (for $j \neq i$) by a vector of at most $1/m^2$ because $\hat{\mu}_f$ changes by at most $1/m$ and $(x_j, y_j) \in B \times [0, 1]$ do not change. Hence the magnitude of the total change is at most $2/m + (m-1)/m^2 \leq 3/m$. (For those who are familiar with McDiarmid's inequality [9], we remark that we do something similar for the vector $\hat{v}$, though it is much simpler since we are only looking for a bound in expectation and not with high probability.)

Define vector-valued random variables $V_1, V_2, \ldots, V_m \in \mathrm{I\!R}^n$ to be,

$$V_i = V_i(x_1, y_1, \ldots, x_i, y_i) = \mathrm{E}\left[\hat{v}|x_1, y_1, \ldots, x_i, y_i\right] - \mathrm{E}\left[\hat{v}|x_1, y_1, \ldots, x_{i-1}, y_{i-1}\right].$$

Hence, we have

$$\hat{v} - v = \sum_{i=1}^{m} V_i(x_1, y_1, \ldots, x_i, y_i).$$

It is also not difficult to see that $\mathrm{E}[V_i] = 0$ and even $\mathrm{E}[V_i|x_1, y_1, \ldots, x_{i-1}, y_{i-1}] = 0$, and hence $\mathrm{E}[V_i|V_j] = 0$ for $i > j$. Thus we also have $\mathrm{E}[V_i \cdot V_j] = 0$ for $i \neq j$. Also, note that $\|V_i\| \leq 3/m$ since changing (or fixing) (x_i, y_i) changes $\hat{v}$ by a vector of magnitude at most $3/m$. Finally,

$$\mathrm{E}\left[(\hat{v} - v)^2\right] = \mathrm{E}\left[(V_1 + \ldots + V_m)^2\right] = \sum_i \mathrm{E}\left[V_i^2\right] + 2\sum_{i>j} \mathrm{E}[V_i \cdot V_j].$$

The above is $\leq m\left(\frac{3}{m}\right)^2 = \frac{9}{m}$. □

We can now prove Theorem 2.

Proof (of Theorem 2). The proof is straightforward given what we have shown already. We know by Lemma 2 that there is some vector $w \in B$ of covariance at least $\frac{4}{5L}\epsilon^2$. If every vector has true covariance within δ of its empirical covariance, then by outputting the vector of maximal empirical covariance, we achieve a true covariance $\geq \frac{4}{5L}\epsilon^2 - 2\delta$. By Lemma 3, we have $\mathrm{E}[\delta] \leq \frac{3}{\sqrt{m}}$. By our choice of $m = 100L^2/\epsilon^4$, the expected true covariance is $\geq \frac{4}{5L}\epsilon^2 - 2(3\epsilon^2/10L) = \frac{1}{5L}\epsilon^2$. □

3.1 Proof of Lemma 2

In order to prove Lemma 2, the following geometric lemma is helpful:

Lemma 4. *Suppose $f : B \to \mathbb{R}$ is an L-Lipschitz function, $w \in \mathbb{R}^n$, $\|w\| = 1$, $t \in \mathbb{R}$, and $\{x \in B \mid f(x) \le t\} = \{x \in B \mid w \cdot x \le \theta\}$. (a) If $\theta \ge 0$ then $|f(x) - t| \le L|w \cdot x - \theta|$ for all $x \in B$ such that $w \cdot x > \theta$. (b) If $\theta \le 0$ then $|f(x) - t| \le L|w \cdot x - \theta|$ for all $x \in B$ such that $w \cdot x < \theta$.*

In other words, we have a Lipschitz-bound on $f(x)$ based on the projection onto the vector w, but it only holds on side of the hyperplane $w \cdot x = \theta$ (the side that has the smaller intersection with the ball).

Proof. For any $x \in B$ such that $(w \cdot x - \theta)\theta \ge 0$ (note that this includes both cases $\theta \ge 0 \wedge w \cdot x - \theta > 0$ and $\theta < 0 \wedge w \cdot x - \theta < 0$), consider the point $x' = x - w(w \cdot x - \theta)$. Note that we have chosen x' so that $w \cdot x' = \theta$ and $x' \in B$. To see that $x' \in B$, notice that $\|w\| = 1$ implies,

$$\begin{aligned}
\|x\|^2 - \|x'\|^2 &= 2x \cdot w(w \cdot x - \theta) - w^2(w \cdot x - \theta)^2 \\
&= (w \cdot x - \theta) \cdot (2w \cdot x - (w \cdot x - \theta)) \\
&= (w \cdot x - \theta) \cdot (w \cdot x + \theta) \\
&= (w \cdot x - \theta)^2 + 2(w \cdot x - \theta)\theta \ge 0
\end{aligned}$$

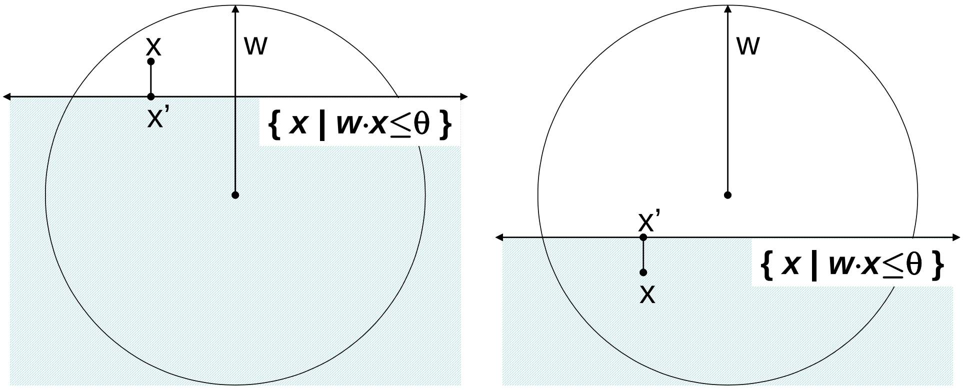

Fig. 2. On the left, we illustrate the case $\theta > 0$ and, on the right, $\theta < 0$. In either case, the lemma only applies to the part of the ball that has less than 1/2 the volume. Not shown are the cases $\theta \ge 1$ ($\{x \in B | w \cdot x \le \theta\} = B$) and $\theta < -1$ ($\{x \in B | w \cdot x \le \theta\} = \emptyset$).

Hence $\|x'\| \le \|x\|$, so $x' \in B$. This is more easily seen geometrically from Figure 2. Now, using the Lipschitz property with the points x, x', we have

$$|f(x) - f(x')| \le L\|x - x'\| = L|w \cdot x - \theta|.$$

Case 1: $|\theta| \ge 1$. In this case, the lemma holds vacuously because there can be no point $x \in B$ meeting the conditions of the lemma. **Case 2:** $\theta \in (-1, 1)$. Since

$x \cdot w = \theta$, we have $f(x') \leq t$. On the other hand, the continuity of f and the fact that $\{x \in B | w \cdot x > \theta\} = \{x \in B | f(x) > t\}$ is non-empty implies that $f(x') = t$. This combined with the displayed equation above gives the lemma. □

We also use the following probabilistic lemma. It states that the variance of a random variable can be broken into two parts, based on whether the variable is greater or smaller than its mean, and both parts can be lower-bounded with respect to the variance of the original variable.

Lemma 5. *Let $X \in [0,1]$ be a random variable with expectation μ and variance $V = \mathrm{E}\left[(X-\mu)^2\right]$. Then $\mathrm{E}\left[(X-\mu)^2 I(X > \mu)\right] \geq \frac{4}{5}V^2$ and similarly $\mathrm{E}\left[(X-\mu)^2 I(X < \mu)\right] \geq \frac{4}{5}V^2$.*

Here the indicator function $I(P) = 1$ if predicate P holds and 0, otherwise. The proof of this lemma is deferred to the appendix.

Proof (of Lemma 2). Take $w \in \mathbb{R}^n$, $\theta \in \mathbb{R}$ such that $\{x \in B \mid f(x) \leq \mu_f\} = \{x \in B \mid w \cdot x \leq \theta\}$. WLOG we assume $\|w\| = 1$ (if $w = 0$ then $\mathrm{var}(f(x)) = 0$ and the lemma holds trivially). Note that $(f(x) - \mu_f)(w \cdot x - \theta) \geq 0$ for all x, be we would like to lower-bound the expectation of this quantity, i.e., $\mathrm{cov}(f(x), w \cdot x)$. **Case 1:** $\theta \geq 0$. Lemma 5 implies that

$$\mathrm{E}[(f(x) - \mu_f)^2 I(w \cdot x > \theta)] \geq \frac{4}{5}\mathrm{var}^2(f).$$

However, whenever $I(w \cdot x > \theta) = 1$, we have $f(x) > \mu_f$ and $f(x) - \mu_f \leq L(w \cdot x - \theta)$. Hence, we have,

$$\begin{aligned}
\frac{4}{5}\mathrm{var}^2(f) &\leq \mathrm{E}[(f(x) - \mu_f)^2 I(w \cdot x > \theta)] \\
&\leq \mathrm{E}[(f(x) - \mu_f) L(w \cdot x - \theta) I(w \cdot x \geq \theta)] \\
&\leq \mathrm{E}[(f(x) - \mu_f) L(w \cdot x - \theta)] \\
&= \mathrm{cov}(f(x), w \cdot x) L.
\end{aligned}$$

The last equality holds by the definition of covariance and the fact that it remains unchanged under additive shifts, i.e., $\mathrm{cov}(A, B) = \mathrm{cov}(A, B + \theta)$ for any constant $\theta \in \mathbb{R}$. **Case 2:** $\theta \leq 0$. This follows in an entirely similar manner to case 1. □

4 Learning Uphill Decision Trees

In this case $X = \{0,1\}^n$. Our algorithm here again uses boosting. The weak learner here is quite similar to the (strong) learner used by Kearns and Schapire for learning Probabilistic Decision Lists.

The following simple probabilistic lemma will be helpful.

Lemma 6. *Let $\mathcal{G}$ be a finite family of binary functions $g : X \to \{0,1\}$ and let $\mathcal{D}$ be an arbitrary probability distribution over $X \times [0,1]$. Then, for any $\epsilon > 0$*

```
1. Let dataset $\mathcal{Z} := \mathcal{Z}_m$.
2. Let $P := \emptyset, N := \emptyset$.
3. If there is an attribute $x[j]$ such that the number of examples in $\mathcal{Z}$ with $x[j] = 0$
   is $\leq 6m^{7/8}\ln(n+3)$, then:
   (a) Let $P := P \cup \{j\}$.
   (b) Remove all examples from $\mathcal{Z}$ such that $x[j] = 0$.
   (c) Goto 3.
4. If there is an attribute $x[j]$ such that the number of examples in $\mathcal{Z}$ with $x[j] = 1$
   is $\leq 6m^{7/8}\ln(n+3)$, then:
   (a) Let $N := N \cup \{j\}$.
   (b) Remove all examples from $\mathcal{Z}$ such that $x[j] = 1$.
   (c) Goto 3.
5. OUTPUT $h(x) := (x[j^*] = b^*) \bigwedge_{j \in P}(x[j] = 1) \bigwedge_{j \in N}(x[j] = 0)$ where $j^* \in [n], b^* \in$
   $\{0,1\}$ are chosen to maximize $|\widehat{\text{cov}}(h(x), y)|$ (over the original $\mathcal{Z}_m$).
```

Fig. 3. A weak learner for uphill decision trees.

and $m \geq 1$, for a random dataset $\mathcal{Z}_m$ of m examples,

$$\Pr_{\mathcal{Z}_m \sim \mathcal{D}^m}\left[\max_{g \in \mathcal{G}} |\hat{\mu}_g - \mu_g| \geq \sqrt{\frac{\ln(2|\mathcal{G}|/\delta)}{2m}}\right] \leq \delta$$

$$\Pr_{\mathcal{Z}_m \sim \mathcal{D}^m}\left[\max_{g \in \mathcal{G}} |\widehat{\text{cov}}(g(x), y) - \text{cov}(g(x), y)| \geq \sqrt{\frac{2\ln(4|\mathcal{G}|/\delta)}{m}}\right] \leq \delta.$$

The proof of this lemma is in the appendix. The following lemma shows that the algorithm of Figure 3 is a weak learner for Uphill decision trees. This implies Result 2, through Lemma 1.

Lemma 7. *For any $\epsilon > 0$ and distribution μ over $\{0,1\}^n \times \{0,1\}$ such* $\text{var}(f(x)) \geq \epsilon$*, given $m \geq (12n\ln(n+3)/\epsilon)^4$ examples, the algorithm of Figure 3 returns $h(x)$ such that,*

$$\mathrm{E}\big[|cov(h(x), y)|\big] \geq \frac{\epsilon^4}{250}.$$

Proof. Let $R(x) = \bigwedge_{j \in P}(x[j] = 1) \bigwedge_{j \in N}(x[j] = 0)$. Let $b = 6m^{7/8}\ln(n+3)$. Consider the family of all conjunctions of subsets of variables

$$\mathcal{G} = \{g(x) = \bigwedge j \in S_1(x[j] = 1) \bigwedge j \in S_1(x[j] = 0) | S_1, S_2 \subseteq [n], S_1 \cap S_2 = \emptyset\}.$$

We have $|\mathcal{G}| \leq 4^n$. Let $\varDelta = 2\sqrt{\ln(n+3)/m}$. By Lemma 6 with a value of $\delta = 1/16$,

$$\Pr\big[\forall g \in \mathcal{G}\ \ |\hat{\mu}_g - \mu_g|, |\widehat{\text{cov}}(g(x), y) - \text{cov}(g(x), y)| \leq \varDelta\big] \geq 7/8.$$

Let us suppose that $|\hat{\mu}_g - \mu_g| \leq \varDelta$ and $|\widehat{\text{cov}}(g(x), y) - \text{cov}(g(x), y)| \leq \varDelta$ for all $g \in \mathcal{G}$. In particular, this directly implies that $\Pr[R(x) = 0] \leq \frac{b}{m}n + \varDelta$, because on the training data we have thrown out at most bn examples. It also implies that for every $j \notin P \cup N$, $\Pr[x[j] = 0 \wedge R(x) = 1] \geq \frac{b}{m} - \Delta$. This follows from the fact that we did not put j in P in step 3, so there must have been $\geq b$ examples such that $x[j] = 0$ and $R(x) = 1$. Similarly, $\Pr[x[j] = 1 \wedge R(x) = 1] \geq \frac{b}{m} - \Delta$.

Now, the idea of the proof is to argue that there is some $g'(x) = (x[j'] = b') \wedge R(x)$ for some $j' \in [n]$ and $b' \in \{0, 1\}$ such that

$$|\mathrm{cov}(g'(x), y)| \geq 128n^2 \ln^4(n + 3)/\sqrt{m}. \tag{6}$$

This means that by outputting $h(x) = (x[j^*] = b^*) \wedge R(x)$ of maximal empirical covariance, we will output $h(x)$ with $|\mathrm{cov}(h(x), y)| \geq 128n^2 \ln^4(n + 3)/\sqrt{m} - 2\Delta \geq 100n^2 \ln^4(n + 3)/\sqrt{m}$, since we have assumed that all empirical and true covariances differ by at most Δ. By our choice of $m \geq (12n \ln(n + 3)/\epsilon)^4$, we have $100n^2 \ln^4(n+3)/\sqrt{m} \geq \epsilon^4/(215)$. The lemma follows from the fact that this happens with probability $\geq 7/8$, so $\mathrm{E}[|\mathrm{cov}(h(x), y)|] \geq (7/8)\epsilon^4/(215) \geq \epsilon^4/(250)$. So it remains to prove the existence of such a g' satisfying (6).

Now, let $x[j^+] = b^+$ be the first test in the tree such that $j^+ \notin P \cup N$. By this, we mean, if we assume that $R(x) = 1$, then we may be able to traverse part of the tree until we get to a test on $j^+ \notin P \cup N$. Another way to put this would be to say that, if we restrict $R(x) = 1$, then we can simplify the tree by pruning off irrelevant tests and we will be left with a potentially smaller tree. In this case, $x[j^+] = b^+$ would be the test at the root of the tree. If the tree reduces to a constant c (single leaf=root), then we can take any test $x[j^+] = b^+$ such that $j^+ \notin P \cup N$, with value c at the two child leaves. Now define,

$$\begin{aligned}
A_1 &= \{x \in \{0, 1\}^n \mid (x[j^+] \neq b^+) \wedge (R(x) = 1)\} \\
A_2 &= \{x \in \{0, 1\}^n \mid (x[j^+] = b^+) \wedge (R(x) = 1)\} \\
A_3 &= \{x \in \{0, 1\}^n \mid R(x) = 0\} \\
p_i &= \Pr[x \in A_i] \quad \text{(for } i = 1, 2, 3) \\
\mu_i &= \mathrm{E}[y | x \in A_i]
\end{aligned}$$

We have that $\mu_2 \geq \mu_1$ and that $A_1 \cup A_2 \cup A_3 = \{0, 1\}^n$ partition $\{0, 1\}^n$, hence $p_1 + p_2 + p_3 = 1$ and $\mu_f = p_1\mu_1 + p_2\mu_2 + p_3\mu_3$. We have argued already that

$$p_3 \leq \frac{b}{m}n + \Delta \leq \frac{4bn}{3m}. \tag{7}$$

$$p_1, p_2 \geq \frac{b}{m} - \Delta \geq \frac{2b}{3m}. \tag{8}$$

In the above we have used the fact that $\Delta \leq b/(3m)$ for our choice of Δ and b. Now we define the function, $g^+(x) = \mu_i$ for i such that $x \in A_i$. We first argue that $g^+(x)$ has large positive covariance with y. Then we will argue that this implies that there is an appropriate binary $g'(x) = (x[j'] = b') \wedge R(x)$ with large absolute value covariance.

We have that

$$\begin{aligned}
\mathrm{cov}(g^+(x), y) &= \mathrm{E}[g^+(x)y] - \mathrm{E}[g^+(x)]\mathrm{E}[y] \\
&= \sum_{i=1}^{3} p_i \mathrm{E}[g^+(x)y | x \in A_i] - \left(\sum_{i=1}^{3} p_i \mu_i\right)^2 \\
&= p_1\mu_1^2 + p_2\mu_2^2 + p_3\mu_3^2 - (p_1\mu_1 + p_2\mu_2 + p_3\mu_3)^2 \\
&= p_1p_2(\mu_1 - \mu_2)^2 + p_1p_3(\mu_1 - \mu_3)^2 + p_2p_3(\mu_2 - \mu_3)^2 \\
&\geq p_1p_2(\mu_1 - \mu_2)^2. \qquad (9)
\end{aligned}$$

We will use the fact that the $\mathrm{var}(f(x)) \geq \epsilon$ to lower-bound the above quantity. In particular, we can bound $\mathrm{var}(f(x))$ in terms of p_i and μ_i. To do this, take threshold $\theta \in [\mu_1, \mu_2]$ to be the value at the rightmost leaf of the left subtree under the test $x[j^+] = b^+$, i.e., such that $f(x) \leq \theta$ for all $x \in A_1$ and $f(x) \geq \theta$ for all $x \in A_2$. Then we have that,

$$\mathrm{var}(f(x)) = \mathrm{E}[(f(x) - \mu_f)^2] \leq \mathrm{E}[(f(x) - \theta)^2] \leq \mathrm{E}[|f(x) - \theta|] \leq \mu_2 - \mu_1 + p_3.$$

The first inequality hold for any $\theta \in \mathbb{R}$ because $\mu_f = \mathrm{E}[f(x)]$ and the second one holds because $|f(x) - \theta| \leq 1$. The last inequality holds by a case analysis: conditioned on $x \in A_1$, we have $|f(x) - \theta| = \theta - f(x)$ which has expected value $\theta - \mu_1 \leq \mu_2 - \mu_1$, similarly for $x \in A_2$, and for $x \in A_3$ we have simply $|f(x) - \theta| \leq 1$. By the above and (7), we have,

$$\mu_2 - \mu_1 \geq \epsilon - \frac{4bn}{3m} \geq \frac{2bn}{3m}.$$

The latter inequality above follows by our choice of b and restriction on ϵ. Finally, by (9), (8), and the above, we have

$$\mathrm{cov}(g^+(x), y) \geq p_1p_2(\mu_1 - \mu_2)^2 \geq \left(\frac{2b}{3m}\right)^2 \left(\frac{2bn}{3m}\right)^2 = \frac{2^4b^4n^2}{3^4m^4}.$$

Again, as mentioned above, this is only a bound on $\mathrm{cov}(g^+(x), y)$. But letting $g_1(x) = (x[j^+] \neq b^+) \wedge R(x)$ and $g_2(x) = (x[j^+] = b^+) \wedge R(x)$, we have,

$$g^+(x) = (\mu_1 - \mu_3)g_1(x) + (\mu_2 - \mu_3)g_2(x) + \mu_3.$$

This means, by linearity of covariance,

$$\begin{aligned}
\mathrm{cov}(g^+(x), y) &= (\mu_1 - \mu_3)\mathrm{cov}(g_1(x), y) + (\mu_2 - \mu_3)\mathrm{cov}(g_2(x), y) \\
&\leq |\mathrm{cov}(g_1(x), y)| + |\mathrm{cov}(g_2(x), y)|.
\end{aligned}$$

The last inequality follows from $\mu_1 - \mu_3, \mu_2 - \mu_3 \in [-1, 1]$. Hence, we have that, for either $i = 1$ or $i = 2$,

$$|\mathrm{cov}(g_i(x), y)| \geq \frac{1}{2}\mathrm{cov}(g^+(x), y) \geq \frac{2^3b^4n^2}{3^4m^4} \geq \frac{128n^2 \ln^4(n+3)}{\sqrt{m}}.$$

Since both g_1, g_2 are of the desired form, we have met the conditions of (6).

5 Conclusions and future work

We have introduced NHFs, a natural generalization of generalized linear models, halfspaces, and decision lists. We have given computationally and statistically efficient learning algorithms for two classes of real valued functions that are special cases of NHFs. Our algorithms are efficient in the sense that their runtime and sample complexity are polynomial in the sizes of the problems. In one case the size corresponds to the Lipschitz constant, analogous to a margin. In the second, discrete case, the size corresponds to the number of variables (interestingly with no dependence on the size of the tree).

Our algorithms and analyses are almost certainly not the best possible. It would be very interesting to improve on the present results. Also, it would be interesting to generalize the types of NHFs that one can learn. It seems like a difficult problem to remove the Lipschitz requirement for NHFs over a ball in n dimensions. It does not seem that one can easily generalize the techniques used by Blum et al. [1] for removing the margin constraint in learning halfspaces with random classification noise, which are another special case of NHFs.

Lastly, classification Boosting has received much attention in the machine learning community, and elegant characterizations are known about what is possible to learn theoretically via boosting. Real-valued boosting seems, at present, much more complex. It would be interesting to come up with simpler models and a deeper understanding of what is provably possible to learn using real-valued boosting.

References

1. A. Blum, A. Frieze, R. Kannan, and S. Vempala. A polynomial time algorithm for learning noisy linear threshold functions. *Algorithmica*, 22(1/2):35–52, 1997.
2. Hastie, T.J., and Tibshirani, R.J. (1990). *Generalized Additive Models,* London: Chapman and Hall.
3. Kalai, A. (2004). Learning Monotonic Linear Functions. In *Lecture Notes in Computer Science: Proceedings of the 17th Annual Conference on Learning Theory*, 3120, 487–501.
4. Kearns, M., and Mansour, Y. (1999). On the boosting ability of top-down decision tree learning algorithms. *Journal of Computer and System Sciences*, 58, 109–128.
5. Kearns, M., and Schapire, R. (1994). Efficient distribution-free learning of probabilistic concepts. *Journal of Computer and Systems Sciences*, 48, 464–497.
6. Kearns, M., and Valiant, L. (1988). Learning boolean formulae or finite automata is as hard as factoring. Technical Report TR-14-88, Harvard University Aiken Computation Laboratory.
7. Mansour, Y., and McAllester, D. (2002). Boosting using branching programs. *Journal of Computer and System Sciences*, 64, 103–112.
8. P. McCullagh and J. Nelder. *Generalized Linear Models,* Chapman and Hall, London, 1989.
9. McDiarmid, C. (1989). On the method of bounded differences. In J Siemons, editor, *Surveys in Combinatorics.* London Math Society.

10. O'Donnell, R. and Servedio, R. (2006). Learning Monotone Decision Trees in Polynomial Time. In *Proceedings of the 21st Annual Conference on Computational Complexity* (CCC), 213–225.
11. Schapire, R. (1990). The strength of weak learnability. *Machine Learning*, 5, 197–227.

A Additional Proofs

Proof (of Lemma 5). Let $p = \Pr[X \geq \mu]$ and $q = \Pr[X < \mu] = 1 - p$. Let $a = \mathrm{E}[X - \mu | X \geq \mu]$ and $b = \mathrm{E}[\mu - X | X < \mu]$. Since $\mu = \mathrm{E}[X]$, $pa = qb$. Finally, let $V_1 = \mathrm{E}[(X - \mu)^2 | X \geq \mu]$ and $V_2 = \mathrm{E}[(X - \mu)^2 | X < \mu]$ so $V = pV_1 + qV_2$. We assume $V \in (0, 1/4]$ (variance $V \leq 1/4$ for any random variable $X \in [0, 1]$ and if $V = 0$ the lemma follows trivially) which implies that $0 < p, q, a, b, V_1, V_2 < 1$.

Since $\mathrm{E}[Y^2] \geq \mathrm{E}[Y]^2$ for any random variable Y, we have that that $V_1 \geq a^2$ (by letting $Y = X - \mu$ conditioned on $X \geq \mu$). We can upper-bound V_2 by noting,

$$V_2 = \mathrm{E}[(\mu - X)^2 | X < \mu] \leq \mathrm{E}[\mu - X | X < \mu] = b.$$

In the above we have used the fact that $x^2 \leq x$ for any real $x \in [0, 1]$.

Now, since $V \leq 1/4$, in our notation it suffices to show the stronger inequality that $\mathrm{E}[(\mu - X)^2 | X > \mu] = pV_1 \geq V^2/(1 + V)$ (the case $X < \mu$ follows by symmetry). In order to complete the lemma, it thus suffices to show that,

$$\begin{aligned}
&pV_1 \geq \frac{V^2}{1 + V} \Leftrightarrow \\
&pV_1 \geq (V - pV_1)V = qV_2(pV_1 + qV_2) \Leftrightarrow \\
&pV_1 \geq \frac{(qV_2)^2}{1 - qV_2}
\end{aligned}$$

However, we have already shown that $V_1 \geq a^2$ and $V_2 \leq b$. This implies that $pV_1 \geq pa^2$ and $(qV_2)^2 \leq (qb)^2 = (pa)^2$. We also have $1 - qV_2 \geq 1 - qb \geq 1 - q = p$, using $b \in [0, 1]$ since $X \in [0, 1]$. Hence $(qV_2)^2/(1 - qV_2) \leq (pa)^2/p = pa^2 \leq pV_1$, which is what we needed for the last displayed equation. □

Proof (of Lemma 6). For the first part of the lemma, we have by Hoeffding bounds that for any single g, $\Pr[|\hat{\mu}_g - \mu_g| \geq \epsilon] \leq 2e^{-2m\epsilon^2}$. By the union bound, this happens for any $g \in \mathcal{G}$ with probability $\leq 2e^{-2m\epsilon^2}|\mathcal{G}|$. For the stated value of ϵ in the first part of the theorem, this probability is $\leq \delta$.

The second part follows from the fact that, for any $g : X \rightarrow reals$,

$$|\widehat{\mathrm{cov}}(g(x), y) - \mathrm{cov}(g(x), y)| = \left| \frac{1}{m} \sum_{i=1}^{m} (y_i - \hat{\mu}_f) g(x_i) - \mathrm{E}[(y - \mu_f) g(x)] \right|.$$

The above is at most

$$\leq \left| \frac{1}{m} \sum_{i=1}^{m} (y_i - \mu_f) g(x_i) - \mathrm{E}[(y - \mu_f) g(x)] \right| + \left| \frac{1}{m} \sum_{i=1}^{m} (\mu_f - \hat{\mu}_f) g(x_i) \right|.$$

By Chernoff bounds, for any $g : X \to \{0,1\}$, the probability that the term on the left is $\geq \epsilon/2$ is at most $2e^{-m\epsilon^2/2}$. Similarly, $\Pr[|\mu_f - \hat{\mu}_f| \geq \epsilon] \leq 2e^{-m\epsilon^2/2}$. (Note that $\frac{1}{m}|\sum(\mu_f - \hat{\mu}_f)g(x_i)| \leq |\mu_f - \hat{\mu}_f|$.) The probability any of these events happen for any $g \in \mathcal{G}$ or f is $\leq 2e^{-m\epsilon^2/2}(|\mathcal{G}|+1) \leq 4e^{-m\epsilon^2/2}|\mathcal{G}|$, which is $\leq \delta$ for the value of ϵ used in the second part of the lemma. If none of these events happens, then we have that the left and right terms are at most $\epsilon/2$ for all $g \in \mathcal{G}$ and hence the empirical and true covariance differ by at most ϵ.