Generalization Bounds for Decision Trees

Yishay Mansour
Computer Science Dept.
Tel-Aviv University.
mansour@math.tau.ac.il

David McAllester
AT&T Labs-Research
dmac@research.att.com

Abstract

We derive a new bound on the error rate for decision trees. The bound depends both on the structure of the tree and the specific sample (not just the size of the sample). This bound is tighter than traditional bounds for unbalanced trees and justifies "compositional" algorithms for constructing decision trees.

1 Introduction

The problem of over-fitting is central to both the theory and practice of machine learning. Intuitively, one over-fits by using too many parameters in the concept, e.g, fitting an nth order polynomial to n data points. One under-fits by using too few parameters, e.g., fitting a linear curve to clearly quadratic data. The fundamental question is how many parameters, or what concept size, should one allow for a given amount of training data. A standard theoretical approach is to prove a bound on generalization error as a function of the training error and the concept size (or VC dimension). One can then select a concept minimizing this bound, i.e., optimizing a certain tradeoff, as expressed in the bound, between training error and concept size.

Bounds on generalization error that express a tradeoff between the training error and the size of the concept are often called structural risk minimization (SRM) formulas. A variety of SRM bounds have been proved in the literature [Vap82]. The following SRM bound was proved in [McA98] and, for completeness, is proved again in Section 2. It states that with probability $1-\delta$ over the sample S we have the following.

$$\forall T \;\; \epsilon(T) \leq \hat{\epsilon}(T) + \sqrt{\frac{(\ln 2)|T| + \ln(1/\delta)}{2|S|}} \qquad (1)$$

This formula says that, for an arbitrary concept class $\mathcal{C}$ where each concept T is encoded by some bit string of length $|T|$, we have that with probability at least $1-\delta$ over the choice a sample S of size $|S|$ i.i.d. instances, all concepts have the property that their true error rate is no larger than their error rate on the training data plus a penalty that depends on $|T|$, $|S|$, and δ. A similar statement holds if we use the VC dimension of $\mathcal{C}$ rather than the encoding size of concepts [Vap82]. The right hand side of formula (1) defines a particular tradeoff between the empirical error rate $\hat{\epsilon}(T)$ and the concept size $|T|$ and we can select T so as to optimize this tradeoff.

Here we are interested in deriving bounds that are tighter than the "naive bound" expressed in (1). Note that (1) expresses a bound on $\epsilon(T) - \hat{\epsilon}(T)$ than depends only on the size of S, and not on the actual examples. To improve on (1) we construct a bound on $\epsilon(T) - \hat{\epsilon}(T)$ that depends both on the concept T and the sample.

Several approaches to the construction of tighter bounds have been taken in the literature. The most notable example is the margin bound for linear threshold functions [Vap98, AB99]. This bound depends on the threshold function chosen and the sample S, where the margin is the "separation" between positive and negative examples in the sample S.

A second approach to improving (1), more closely related to the approach taken here, is taken by Golea et. al. [GBLM97]. They give a bound for decision trees in terms of the "effective number of leaves" where unbalanced trees have a smaller number of effective leaves than balanced trees. Their proof techniques involve a margin analysis for decision trees. While their main theorem handles well the case when almost all training data reaches a single leaf, it is far less useful when a significant fraction of the training data reaches two or more leaves.

Our approach is somewhat related to the tree pruning methods developed by Kearns and Mansour [KM98]. They give an algorithm for pruning decision trees which implicitly uses a bound for subtrees of a given tree. The results in this work can be used in conjunction with the techniques of [KM98] to derive improved pruning algorithms.

Another approach that has been taken to improving (1) is to consider not only the concept T and the sample S, but also the learning algorithm that is used to generate T. This includes Freund's "self-bounding algorithms" [Fre98] and the related bounds by Langford and Blum [LB99]. The basic idea in these bounds is to measure, at each choice point in the learning algorithm, the number of alternatives that might be taken if we based the decision on a second fresh sample.

To motivate our new bound let $\mathtt{IF}(A,\ T_l,\ T_r)$ denote a decision tree with root predicate A and left and right subtrees T_l and T_r respectively. Let h be a function taking a decision tree and a sample and returning a real number. Ultimately we are interested in those functions h such that $h(T,\ S)$ is an upper bound on the error rate of T. Now consider, for a fixed predicate A, the problem of selecting T_l and T_r so as to minimize $h(\mathtt{IF}(A,\ T_l,\ T_r),\ S)$. The function h will be called *compositional* if this minimization can be done by solving independent optimization problems for T_l and T_r. More specifically, h is compositional if there exist functions h_l and h_r such that, for a given predicate A and sample S, selecting T_l and T_r so as to minimize $h(\mathtt{IF}(A,\ T_l,\ T_r),\ S)$ is equivalent to selecting T_l so as to minimize $h_l(T_l,\ S_A)$ and selecting T_r so as to minimize $h_r(T_r,\ S_{\neg A})$ where S_A and $S_{\neg A}$ are the subsets of S satisfying A and $\neg A$ respectively. The naive bound is not compositional — when selecting T_l and T_r so as to minimize the bound we have that $|T_l|$ influences the optimal choice of T_r.

To give an example of a compositional expression we first let $h_0(T,\ S)$ abbreviate the naive bound.

$$h_0(T,\ S) \equiv \hat{\epsilon}(T) + \sqrt{((\ln 2)|T| + \ln(1/\delta))/(2|S|)}$$

Now define the function $h_1(T,\ S)$ by the following equation.

$$h_1(\mathtt{IF}(A,\ T_r,\ T_l),\ S) \equiv \frac{|S_A|}{|S|}h_0(T_r,\ S_A) + \frac{|S_{\neg A}|}{|S|}h_0(T_l,\ S_{\neg A})$$

By construction the function h_1 is compositional — optimizing the choice of T_l and T_r can be done by optimizing T_l and T_r independently. It seems intuitively clear that if the sample is large then error rate of T can not be much larger than $h_1(T,\ S)$.

Our new bound is stated in terms of a "root fragment" of the tree, i.e., a set of nodes containing the root and having the property that, for any node in the set, the parent and siblings of that node must also be in the set. Let R be a root fragment and let $L(R)$ denote the leaves of R. The set $L(R)$ defines a cut of possibly varying depth through the tree T. For a node $v \in L(R)$ let T_v be the subtree of T rooted at v and let S_v be the subset of the sample reaching node v. The new bound states that, with probability at least $1-\delta$ over the choice of the sample, we have the following for all trees T and root fragments R of T.

$$\epsilon(T) - \hat{\epsilon}(T) \leq \sum_{v \in L(R)} \frac{|S_v|}{|S|}\sqrt{\frac{(\ln 2)|T_v|}{2|S_v|}} + \gamma(S,\ R,\ \delta)$$

The expression $\gamma(S,\ \ R,\ \ \delta)$ is given in section 3 and is negligible when R and $\ln(1/\delta)$ are small (as defined in section 3). Jensen's inequality implies that for $g(v) \geq 0$ we have $\sum(|S_v|/|S|)\sqrt{g(v)} \ \leq\ \sqrt{\sum(|S_v|/|S|)g(v)}$. This implies that the above bound is never larger than $\sqrt{(\ln 2)|T|/(2|S|)} + \gamma(S,\ R,\ \delta)$. So, when R and $\ln(1/\delta)$ are small, the new bound is not significantly larger than the old bound. However, the new bound can be smaller by the slack in Jensen's inequality. Section 3 shows under that, under conditions expected to hold in practice, the new bound on $\epsilon(T) - \hat{\epsilon}(T)$ will not be smaller than $(\ln 2)|T|/(2|S|)$. In summary, the new bound is compositional near the root of the tree and can potentially improve the penalty for the size of T from $\sqrt{(\ln 2)|T|/(2|S|)}$ to $(\ln 2)|T|/(2|S|)$. For a fixed tree T, section 5 gives an efficient algorithm for exactly computing the root fragment R minimizing the new bound.

2 Model and Preliminaries

Let $\mathcal{X}$ be a set of "instances". We assume some fixed but unknown distribution D on $\mathcal{X} \times \{0,\ 1\}$. Let S be a sample of m pairs $\langle x,\ y\rangle$ drawn independently from the distribution D. The notation $\forall^\delta S\ \ \Phi[S]$ is used as an abbreviation for the statement that with probability at least $1-\delta$ over the selection of S we have that $\Phi[S]$ holds. Note that $\forall x\ \ \ \forall^\delta S\ \ \ \Phi[S,\ x]$ does not imply $\forall^\delta S\ \ \forall x\ \ \Phi[S,\ x]$.

Let $\mathcal{H}$ be a set of "base predicates" each of which maps $\mathcal{X}$ to $\{0,\ 1\}$. Let $T(\mathcal{H})$ be the set of decision trees over $\mathcal{H}$, i.e., binary trees where each leaf is labeled with either 1 or 0 and each non-leaf is labeled with a predicate from $\mathcal{H}$. Each decision tree also defines a predicate on $\mathcal{X}$.

For any predicate A on $\mathcal{X}$ we define the (true) error rate of A, denoted $\epsilon(A)$, to be the probability over drawing a random pair $\langle x,\ y\rangle$ according to D that $A(x) \neq y$. When the sample S is clear from context we define the empirical error rate of A, denoted, $\hat{\epsilon}(A)$ to be $|\{\langle x,\ y\rangle \in S\ :\ \ y \neq A(x)\}|/|S|$.

We assume a prefix-free code for the predicates in $\mathcal{H}$, i.e., each predicate is named by a code string where no code string is proper prefix of any other code string. We let $|B|$ be number of bits in the code for predicate B. Prefix-free codes satisfy the Kraft inequality: $\sum_{B\in\mathcal{H}} 2^{-|B|} \leq 1$. One can devise a prefix-free code for the trees in $T(\mathcal{H})$ such that for a tree T with n internal nodes labeled with branch predicates $B_1, \ldots, B_n$ we have $|T| = 2 + 3n + \sum_{i=1}^n |B_i|$.

We now let $C(\mathcal{H})$ be the set of conjunctions of the form $(B_1 = b_1) \wedge \ldots \wedge (B_d = b_d)$ where $B_i \in \mathcal{H}$ and $b_i \in \{0, 1\}$. Each element $A \in C(\mathcal{H})$ is viewed as a predicate on $\mathcal{X}$ in the obvious way. For any node v in T we define A_v to be the predicate in $C(\mathcal{H})$ corresponding to the path from the root to node v. Note that $A_v(x) = 1$ if and only if x reaches node v. One can devise a prefix-free coding for the elements of $C(\mathcal{H})$ such that if A is the conjunction of the form $(B_1 = b_1) \wedge \ldots \wedge (B_n = b_n)$ then $|A| = 1 + 2n + \sum_{i=1}^n |B_i|$. For any predicate $A \in C(\mathcal{H})$ and sample S we define S_A to be $\{\langle x,\ y\rangle \in S\ :\ \ A(x) = 1\}$. We use S_v as an abbreviation for S_{A_v}. Let T_v be the subtree of T consisting of all nodes at or below v. We define $T(\mathcal{H},\ S)$ to be the set of decision trees $T \in T(\mathcal{H})$ such that for every node v of T we have that S_v is nonempty.

The additive Chernoff bound can be expressed as follows.

Lemma 1 *Let $X_1, \ldots, X_m$ be i.i.d. random Boolean variables, and $p = Pr[X_i = 1]$.*

$$\Pr[(1/m)\sum_{i=1}^m X_i \leq p + \gamma] \leq e^{-2m\gamma^2}$$

The naive error bound can be expressed as follows.

Lemma 2

$$\forall^\delta S\ \ \forall T \in T(\mathcal{H})\ \ \epsilon(T) \leq \hat{\epsilon}(T) + \sqrt{\frac{(\ln 2)|T| + \ln(1/\delta)}{2|S|}}$$

Proof: Consider a fixed tree T. It follows from the Chernoff bound (Lemma 1) that the probability over the choice of S that the particular tree T violates the above lemma is at most $\delta 2^{-|T|}$. By the union bound the probability that some tree violates the lemma is no larger than $\sum_{T\in T(\mathcal{H})} \delta 2^{-|T|} \leq \delta$. $\square$

3 Main Theorem

We now consider an arbitrary division of the nodes of a given tree T into shallow and deep nodes. We let R be an arbitrary set of "shallow" nodes satisfying the condition that R forms a subtree of T containing the root and such that all nodes in R are either leaves of R or have both their children in R. We denote by $L(R)$ the set of leaves of the subtree R. Our main theorem is the following.

Theorem 3

$$\forall^{\delta} S \ \ \forall T \in T(\mathcal{H},\ S) \ \ \epsilon(T) \leq \hat{\epsilon}(T) + \min_{R} f(T, S, R, \delta)$$

where

$$f(T,S,R,\delta) = \sum_{v \in L(R)} \frac{|S_v|}{|S|} \left(\begin{array}{l} \sqrt{\frac{(\ln 2)|T_v|}{2|S_v|}} \\ \\ +2\left(\sqrt{\frac{|A_v|}{|S_v|}} + \frac{|A_v|}{|S_v|}\right) \\ \\ +2\left(\sqrt{\frac{\ln(2/\delta)}{|S_v|}} + \frac{\ln(2/\delta)}{|S_v|}\right) \end{array} \right)$$

Before giving the proof in section 4, we try to clarify the bound by noting that it satisfies four rather simple properties. First, for a fixed root fragment R the bound is compositional with respect to optimizing subtrees rooted in R. In particular we have the following equation where $\hat{\epsilon}(T_v,\ S_v)$ is the error rate of T_v on the sample S_v and $root(T_v)$ denotes the root fragment of T_v consisting of just the root node of T_v.

$$\hat{\epsilon}(T) + f(T,\ S,\ R,\ \delta) = \sum_{v \in L(R)} \frac{|S_v|}{|S|} \left[\begin{array}{l} \hat{\epsilon}(T_v,\ S_v) \\ \\ +f(T_v,\ S_v,\ root(T_v),\ \delta) \end{array} \right]$$

So to select the trees T_v to minimize $\hat{\epsilon}(T) + f(T,\ S,\ R,\ \delta)$ we can optimize each subtree T_v independently.

We will call the bound on $\epsilon(T) - \hat{\epsilon}(T)$ given in formula (1) the naive penalty and we will call $\min_R f(T,\ S,\ R,\ \delta)$ the new penalty. Our second observation is that the new penalty it is not significantly larger than the naive penalty. By setting R to be the tree consisting only of the root node we get the following.

$$\min_{R} f(T,S,R,\delta) \leq \left(\begin{array}{l} \sqrt{\frac{(\ln 2)|T|}{2|S|}} \\ \\ +2\left(\sqrt{\frac{1}{|S|}} + \frac{1}{|S|}\right) \\ \\ +2\left(\sqrt{\frac{\ln(2/\delta)}{|S|}} + \frac{\ln(2/\delta)}{|S|}\right) \end{array} \right)$$

In cases where $|T|$ is large compared to both 1 and $\ln(1/\delta)$, a common occurrence in practice, both the above expression and the naive penalty will be approximately equal to $\sqrt{(\ln 2)|T|/(2|S|)}$.

It is also possible to show that the new penalty is not significantly smaller than the naive penalty for any "small" root fragment R. By Jensen's inequality we have the following.

$$f(T,S,R,\delta) \leq \left(\begin{array}{l} \sqrt{\frac{(\ln 2)|T|}{2|S|}} \\ \\ +2\left(\sqrt{\frac{\sum_{v \in L(R)} |A_v|}{|S|}} + \frac{\sum_{v \in L(R)} |A_v|}{|S|}\right) \\ \\ +2\left(\sqrt{\frac{|L(R)|\ln(2/\delta)}{|S|}} + \frac{|L(R)|\ln(2/\delta)}{|S|}\right) \end{array} \right)$$

For $\gamma > 0$, a root fragment R and will be called γ-small if $|L(R)| \leq \gamma|T|$ and for all $v \in L(R)$ we that have $|A_v| \leq \gamma|T_v|$. For γ-small R and reasonably pruned T , and assuming that $(\ln 2)|T|/(2|S|)$ is no larger than 1, the above inequality implies the following.

$$f(T,\ S,\ R,\ \delta) \leq \left(\begin{array}{l} 1 \\ \\ +2\left(\sqrt{\frac{2\gamma}{\ln 2}} + \frac{2\gamma}{\ln 2}\right) \\ \\ +2\left(\sqrt{\frac{2\gamma\ln(2/\delta)}{\ln 2}} + \frac{2\gamma\ln(2/\delta)}{\ln 2}\right) \end{array} \right) \sqrt{\frac{(\ln 2)|T|}{2|S|}}$$

We say that a tree is *reasonably pruned* if for every node v in T we have $(\ln 2)|T_v|/(2|S_v|) \leq 1$. Intuitively this condition would hold due to properties of the learning algorithm that constructed the tree. For example if each leaf has a minimal number of examples reaching it, say 3, then the tree would be reasonable pruned. Our third observation is that, for reasonably pruned trees, the new penalty is never smaller than $(\ln 2)|T|/(2|S|)$. Note that the quantity $(\ln 2)|T|/(2|S|)$ can be much smaller than $\sqrt{(\ln 2)|T|/(2|S|)}$.

For $x \in [0,\ 1]$ we have $\sqrt{x} \geq x$ so for T reasonably pruned we have the following where the third line follows from the fact that $(\ln 2)/2 < 1$.

$$\begin{array}{rcl} f(T,\ S,\ R,\ \delta) & \geq & \displaystyle\sum_{v \in L(R)} \frac{|S_v|}{|S|}\left[\sqrt{\frac{(\ln 2)|T_v|}{2|S_v|}} + \frac{|A_v|}{|S_v|}\right] \\ & \geq & \displaystyle\sum_{v \in L(R)} \frac{|S_v|}{|S|}\left[\frac{(\ln 2)|T_v|}{2|S_v|} + \frac{|A_v|}{|S_v|}\right] \\ & \geq & \displaystyle\frac{\ln 2}{2|S|}\sum_{v \in L(R)} (|T_v| + |A_v|) \\ & \geq & \displaystyle\frac{(\ln 2)|T|}{2|S|} \end{array}$$

Finally we note that even for reasonably pruned T we can have $f(T,S,R,\delta)$ arbitrarily close to $(\ln 2)|T|/(2|S|)$. Take T to be $\texttt{IF}(A,\ T_l,\ T_r)$ and take R to consist of the root plus its two children. Let S_l and S_r be the subsets of the sample reaching T_l and T_r respectively. By making $|S|$, $|T_l|$ and $|T_r|$ sufficiently large we can arrange that $|A|$ and $\ln(1/\delta)$ are both small compared to $|T_l|$ and both $\sqrt{|A|/|S_l|}$ and $\sqrt{\ln(1/\delta)/|S_l|}$ are small compared with $\sqrt{|T_l|/|S_l|}$, and similarly for T_r. Under these conditions $f(T,\ S,\ R,\delta)$ can be written as follows.

$$f(T,\ S,\ R,\ \delta) \approx \frac{|S_l|}{|S|}\sqrt{\frac{(\ln 2)|T_l|}{2|S_l|}} + \frac{|S_r|}{|S|}\sqrt{\frac{(\ln 2)|T_r|}{2|S_r|}}$$

We now formulate all other quantities to be proportional to $|S|$ so that we can scale the size of the sample to be arbitrarily large. We fix a small number $\epsilon > 0$. We take $|S_l| = \epsilon|S|$ and $|T_l| = (2/(\ln 2))\epsilon|S|$. Note that this allows T to be reasonably pruned. In particular, $(\ln 2)|T_l|/(2|S_l|) = 1$. We now take $|T_r|$ to be $\epsilon^4|S|$. This implies $|T| \approx |T_l|$ and $\epsilon = |S_l|/|S| = (\ln 2)|T_l|/(2|S|) \approx (\ln 2)|T|/(2|S|)$. Up to first order terms in ϵ we have the following.

$$\begin{aligned} f(T,S,R,\delta) &\approx \epsilon\sqrt{\frac{(\ln 2)|T_l|}{2|S_l|}} + (1-\epsilon)\sqrt{\frac{(\ln 2)|T_r|}{2|S_r|}} \\ &\approx \epsilon + \epsilon^2\sqrt{\frac{\ln 2}{2}} \\ &\approx \epsilon \approx \frac{(\ln 2)|T|}{2|S|} \end{aligned}$$

4 Proof of Main Theorem

To prove the main theorem we start with a couple preliminary lemmas. Throughout this section we let m be $|S|$. Note that we take m to be given before we select S. Now for any $A \in C(\mathcal{H})$ we define p_A to be $Pr_{\langle x,\ y\rangle \sim D}[A(x) = 1]$. For any $A \in C(\mathcal{H})$ and $T \in T(\mathcal{H})$ we define the error rate of T on the distribution induced by A, denoted $\epsilon_A(T)$, to be $Pr_{\langle x,\ y\rangle \sim D}(T(x) = y \mid A(x) = 1)$. For a given sample S we define the empirical error rate of T on the distribution induced by A, denoted $\hat{\epsilon}_A(T)$, to be $|\{\langle x,\ y\rangle \in S_A \ : T(x) \neq y\}|/|S_A|$. We now have the following lemma. (A similar lemma appears in [KM98].)

Lemma 4 $\forall^\delta S\ \ \forall A \in C(\mathcal{H})\ \ \forall T \in T(\mathcal{H})$

$$\epsilon_A(T) \leq \hat{\epsilon}_A(T) + \sqrt{\frac{(|A| + |T|)\ln 2 + \ln(1/\delta)}{2|S_A|}}$$

Proof: Consider a particular fixed predicate P in $C(\mathcal{H})$ and tree T in $T(\mathcal{H})$. We can bound the probability over the selection of S that the particular predicates A and T violate the lemma as follows.

$$\Pr\left[\epsilon_A(T) \geq \hat{\epsilon}_A(T) + \sqrt{\tfrac{(|A|+|T|)\ln 2 + \ln(1/\delta)}{|S_A|}}\right]$$

$$\begin{aligned} &= \sum_{n=0}^{\infty} \left(\begin{array}{l} \Pr[|S_A| = n] \\ \cdot \Pr\left[\begin{array}{l} \epsilon_A(T) \geq \hat{\epsilon}_A(T) + \sqrt{\frac{(|A|+|T|)\ln 2 + \ln(1/\delta)}{2n}} \\ \mid\ |S_A| = n \end{array} \right] \end{array} \right) \\ &\leq \sum_{n=0}^{\infty} \Pr[|S_A| = n]\delta 2^{-|A|}2^{-|T|} \\ &= \delta 2^{-|A|}2^{-|T|} \end{aligned}$$

By the union bound the probability that some choice of A and T violates the lemma is now bounded by the following.

$$\delta \sum_{A \in C(\mathcal{H})} 2^{-|A|} \sum_{T \in T(\mathcal{H})} 2^{-|T|} \leq \delta$$

$\square$

Our second preliminary lemma is a form of the relative Chernoff bound that is particularly well suited to machine learning applications. The relative Chernoff bound is usually stated as follows.

Lemma 5 *Let $X_1, \ldots, X_m$ be i.i.d. random Boolean variables and $p = Pr[X_i = 1]$. For $\gamma \in [0,1]$ we have the following.*

$$\Pr[\sum_{i=1}^{m} X_i \leq (1-\gamma)pm] \leq e^{-mp\gamma^2/2}$$

By setting γ equal to $\sqrt{\frac{2\ln(1/\delta)}{pm}}$ we can rephrase the relative Chernoff bound as follows,

$$\forall^\delta S\ \ p \leq \hat{p} + \sqrt{\frac{2p\ln(1/\delta)}{m}},$$

where $\hat{p} = (1/m)\sum_{i=1}^m X_i$. This bound on p is not directly useful in machine learning applications because the bound uses the unknown quantity p. We need a bound that is purely a function of the observed quantity $\hat{p}$. Such a bound is provided by the following lemma.

Lemma 6

$$\forall^\delta S\ \ p \leq \hat{p} + \sqrt{\frac{2\hat{p}\ln(1/\delta)}{m}} + \frac{2\ln(1/\delta)}{m}$$

Proof: By the relative Chernoff bound (Lemma 5) we have the following with probability at least $1-\delta$ over the choice of the sample.

$$p - \hat{p} \leq \sqrt{\frac{2p\ln(1/\delta)}{m}}$$

This implies the following.

$$m(p-\hat{p})^2 \leq 2p\ln(1/\delta)$$

or alternatively,

$$mp^2 - (2m\hat{p} + 2\ln(1/\delta))p + m\hat{p}^2 \leq 0.$$

This gives us a restrictions on the possible values of p, therefore,

$$\begin{aligned} p &\leq \frac{(2m\hat{p} + 2\ln(1/\delta)) + \sqrt{(2m\hat{p} + 2\ln(1/\delta))^2 - 4m^2\hat{p}^2}}{2m} \\ &= \hat{p} + \frac{\ln(1/\delta)}{m} + \sqrt{\frac{8m\hat{p}\ln(1/\delta) + 4\ln^2(1/\delta)}{4m^2}} \\ &= \hat{p} + \frac{\ln(1/\delta)}{m} + \sqrt{\frac{2\hat{p}\ln(1/\delta)}{m} + \frac{\ln^2(1/\delta)}{m^2}} \\ &\leq \hat{p} + \frac{2\ln(1/\delta)}{m} + \sqrt{\frac{2\hat{p}\ln(1/\delta)}{m}} \end{aligned}$$

$\square$

Lemma 6 can now be used to prove the following. (Recall that S_A is the subset of the sample satisfying the predicate A.)

Corollary 7 *For any predicate A we have*

$$\forall^\delta S \left(\begin{array}{l} (S_A = \emptyset) \\ \vee \left(p - \hat{p} \leq \hat{p}\left[\sqrt{\frac{2\ln(1/\delta)}{|S_A|}} + \frac{2\ln(1/\delta)}{|S_A|}\right]\right) \end{array} \right)$$

where $\hat{p} = |S_A|/|S|$.

Proof: It follows from Lemma 6 that with probability at least $1-\delta$ we have the following.

$$p - \hat{p} \leq \sqrt{\frac{2\hat{p}\ln(1/\delta)}{m}} + \frac{2\ln(1/\delta)}{m}.$$

We simply show that this implies the desired result. If $|S_A| = 0$ then by definition the lemma holds, so we can assume that $|S_A| > 0$. Under this assumption, and recalling that $\hat{p} = |S_A|/|S|$, we have that,

$$p - \hat{p} \leq \hat{p}\left[\sqrt{\frac{2\ln(1/\delta)}{|S_A|}} + \frac{2\ln(1/\delta)}{|S_A|}\right].$$

□

The following Corollary generalizes the result to a set of predicates $C(\mathcal{H})$.

Corollary 8 $\forall^\delta S \;\; \forall A \in C(\mathcal{H})$

$$\left(\begin{array}{l} S_A = \emptyset \\ \\ \vee p_A - \hat{p}_A \leq \hat{p}_A \left[\begin{array}{l} \sqrt{\frac{2(|A|\ln 2 + \ln(1/\delta))}{|S_A|}} \\ \\ + \frac{2(|A|\ln 2 + \ln(1/\delta))}{|S_A|} \end{array}\right] \end{array}\right)$$

Proof: Consider a fixed predicate A. Corollary 7 implies that the probability that A violates the lemma is bounded by $\delta 2^{-|A|}$. The union bound then implies that the probability that there exists a $A \in C(\mathcal{H})$ violating the lemma is no larger than δ. □

Now we are ready to prove our main theorem.

Proof of Theorem 3: By Lemma 4 we have that with probability at least $1 - \delta/2$ we have the following.

$\forall A \in C(\mathcal{H}) \;\; \forall T \in T(\mathcal{H})$

$$\epsilon_A(T) \leq \hat{\epsilon}_A(T) + \sqrt{\frac{(|A| + |T|)\ln 2 + \ln(2/\delta)}{2|S_A|}} \tag{2}$$

By Corollary 8 we have that with probability at least $1-\delta/2$ we have the following.

$\forall A \in C(\mathcal{H})$

$$p_A - \hat{p}_A \leq \hat{p}_A \left[\begin{array}{l} \sqrt{\frac{2(|A|\ln 2 + \ln(2/\delta))}{|S_A|}} \\ \\ + \frac{2(|A|\ln 2 + \ln(2/\delta))}{|S_A|} \end{array}\right] \tag{3}$$

By the union bound, with probability at least $1 - \delta$ both of these conditions hold simultaneously. Let p_v be the probability of reaching node v and $\hat{p}_v$ be $|S_v|/|S|$. Let q_v be the error probability at node v, i.e. $\epsilon_{A_v}(T_v)$, and $\hat{q}_v$ be $\hat{\epsilon}_{A_v}(T_v)$. We now rewrite the error bound as follows.

$$\begin{aligned} \epsilon(T) - \hat{\epsilon}(T) &= \sum_{v \in L(R)} (p_v q_v) - \sum_{v \in L(R)} \hat{p}_v \hat{q}_v \\ &= \sum_{v \in L(R)} [(p_v - \hat{p}_v) q_v + \hat{p}_v (q_v - \hat{q}_v)] \end{aligned}$$

The desired result now follows by bounding the first term in the sum with formula (3) and the second formula in the sum with formula (2). This gives the following.

$$\epsilon(T) - \hat{\epsilon}(T) \leq \sum_{v \in L(R)} \left(\begin{array}{l} \hat{p}_v \left[\begin{array}{l} \sqrt{\frac{2(|A_v|\ln 2 + \ln(2/\delta))}{|S_v|}} \\ \\ + \frac{2(|A_v|\ln 2 + \ln(2/\delta))}{|S_v|} \end{array}\right] q_v \\ \\ + \hat{p}_v \sqrt{\frac{(|A_v| + |T_v|)\ln 2 + \ln(1/\delta) + \ln 2}{2|S_v|}} \end{array}\right)$$

Since $q_v \leq 1$, and using $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$, we get the following.

$$\epsilon(T) - \hat{\epsilon}(T) \leq \sum_{v \in L(R)} \hat{p}_v \left(\begin{array}{l} \sqrt{\frac{(\ln 2)|T_v|}{2|S_v|}} \\ \\ + \left[\sqrt{2\ln 2} + \sqrt{(\ln 2)/2}\right] \sqrt{\frac{|A_v|}{|S_v|}} \\ \\ + [2\ln 2] \frac{|A_v|}{|S_v|} \\ \\ + 2\frac{\ln(2/\delta)}{|S_v|} \\ \\ + \left[1 + \sqrt{(\ln 2)/2}\right] \sqrt{\frac{\ln(2/\delta)}{|S_v|}} \end{array}\right)$$

This implies the theorem. □

5 Computing the optimal bound

Theorem 3 gives a bound which is a function of the best root fragment that can be found. In this section we show how, given a tree T and a sample S, one can compute the best root fragment R in an efficient way. The basic idea behind the algorithm is to use the compositional property of the bound we derive.

The main observation is that given a tree T, the best root fragment R is either the best root fragment of the right subtree combined with the best root fragment of the left subtree, or simply includes only the root. This observation will give us a simple bottom-up procedure to compute the optimal root fragment.

We define a procedure $\texttt{compute\_R}(T_v, S, |A_v|, \delta)$ that computes the best root segment of a subtree T_v. We define a simple procedure $\texttt{eval}(T_v, S_v, |A_v|, \delta)$ that evaluates the penalty of terminating the root fragment at the root of T as follows.

$$\texttt{eval}(T, S, d, \delta) = \left(\begin{array}{l} \sqrt{\frac{(\ln 2)|T|}{2|S|}} \\ \\ +2\left(\sqrt{\frac{d}{|S|}} + \frac{d}{|S|}\right) \\ \\ +2\left(\sqrt{\frac{\ln 2/\delta}{|S|}} + \frac{\ln 2/\delta}{|S|}\right) \end{array}\right)$$

We now define $\texttt{compute\_R}(T, S, d, \delta)$ with the following equations.

$$\texttt{compute\_R}(1, S, d, \delta) = \texttt{eval}(1, S, d, \delta)$$

$$\texttt{compute\_R}(0, S, d, \delta) = \texttt{eval}(0, S, d, \delta)$$

$\texttt{compute\_R}(\texttt{IF}(B, T_l, T_r), S, d, \delta)$

$$= \min \left\{ \begin{array}{l} \texttt{eval}(\texttt{IF}(B, T_l, T_r), S, d, \delta), \\ \\ \frac{|S_B|}{|S|}\texttt{compute\_R}(T_l, S_B, d + |B| + 2, \delta) \\ +\frac{|S_{\neg B}|}{|S|}\texttt{compute\_R}(T_r, S_{\neg B}, d + |B| + 2, \delta) \end{array} \right\}$$

Assuming that each predicate in T can be evaluated on a given instance in unit time, a direct implementation of these equations as a recursive procedure runs in time proportional to the sum over all nodes of the number of instances in the sample reaching that node.

Theorem 9 *For any subtree T_v of T we have*

$$\texttt{compute\_R}(T_v,\ S_v,\ |A_v|,\ \delta) = \min_R f(v,\ R)$$

where R ranges over root fragments of T_v and $f(v,\ R)$ is defined as follows.

$$f(v,\ R) = \sum_{w \in L(R)} \frac{|S_w|}{|S_v|} \left(\begin{array}{l} \sqrt{\frac{(\ln 2)|T_w|}{2|S_w|}} \\ \\ +2\left(\sqrt{\frac{|A_w|}{|S_w|}} + \frac{|A_w|}{|S_w|}\right) \\ \\ +2\left(\sqrt{\frac{\ln(2/\delta)}{|S_w|}} + \frac{\ln(2/\delta)}{|S_w|}\right) \end{array} \right)$$

Proof: Let $\texttt{root}(T_v)$ denote the root fragment of T_v containing only v. For the root fragment we have the following.

$$f(v,\ \texttt{root}(T_v)) = \texttt{eval}(T_v,\ S_v,\ |A_v|,\ \delta) \qquad (4)$$

Let $\{R_l, R_r\}$ be any root fragment of T_v consisting of the root plus non-empty left and right subtrees R_l and R_r. For root fragments of this form we have the following.

$$f(v,\ \{R_l, R_r\}) = \frac{|S_l|}{|S_v|} f(l,\ R_l) + \frac{|S_r|}{|S_v|} f(r,\ R_r) \qquad (5)$$

Given equations 4 and 5 the proof is straightforward induction on the size of T_v. If v is a leaf of T then the only choice for R is the root fragment containing only v and the result follows from equation 4. Now assume T_v is $\texttt{IF}(B,\ T_l,\ T_r)$ where l and r denote the left and right children of v respectively and where the result holds for T_l and T_r. Equation 5 and the induction hypothesis implies that the minimum over all trees of the form $\{R_l, R_r\}$ of $f(T_v,\ \{R_l, R_r\})$ equals the second argument of the $\min$ expression in the definition of $\texttt{compute\_R}$. Equation 4 implies that the first argument of the min expression handles the possibility that the minimum is just the root fragment, and the result follows. □

The above theorem immediately implies the following where f is defined as in section 3.

$$\texttt{compute\_R}(T,\ S,\ 1,\ \delta) = \min_R f(T,\ S,\ R,\ \delta)$$

The algorithm for computing $\texttt{compute\_R}$ can easily be converted to an algorithm for computing the optimal subtree R.

6 Conclusions

We have derive a new bound on the error rate for decision trees. The bounds depends both on the structure of the tree and the specific sample (and not only on the size of the sample). This bound is tighter than traditional bounds for unbalanced trees and justifies "compositional" algorithms for constructing decision trees.

References

[AB99] M. Anthony and P. Bartlett. *Neural Network Learning Theoretical Foundations*. Cambridge University Press, Cambridge, 1999.

[Fre98] Yoav Freund. Self bounding learning algorithms. In *Conference on Computational Learning Theory*, pages 247–258, 1998.

[GBLM97] Mostefa Golea, Peter Bartlett, Wee Sun Lee, and Llew Mason. Generalization in decision trees and DNF: Does size matter? In *NIPS*, 1997.

[KM98] Michael Kearns and Yishay Mansour. A fast, bottom-up decision tree pruning algorithm with near-optimal generalization. In *International Conference on Machine Learning*, pages 269–277, 1998.

[LB99] John Langford and Avrim Blum. Microchoice bounds and selfbounding learning algorithms. In *Conference on Computational Learning Theory*, pages 209–214. Morgan Kaufmann, 1999.

[McA98] David McAllester. Some PAC-Bayesian theorems. In *Conference on Computational Learning Theory*, pages 230–234, 1998.

[Vap82] V. N. Vapnik. *Estimation of Dependences Based on Empirical Data*. Springer-Verlag, New York, 1982.

[Vap98] V. N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, Inc., 1998.