Efficient Agnostic PAC-Learning with Simple Hypotheses

Wolfgang Maass

Institute for Theoretical Computer Science
Technische Universitaet Graz
Klosterwiesgasse 32/2
A-8010 Graz, Austria
e-mail: maass@igi.tu-graz.ac.at

Abstract

We exhibit efficient algorithms for agnostic PAC-learning with rectangles, unions of two rectangles, and unions of k intervals as hypotheses. These hypothesis classes are of some interest from the point of view of applied machine learning, because empirical studies show that hypotheses of this simple type (in just one or two of the attributes) provide good prediction rules for various real-world classification problems. In addition, optimal hypotheses of this type may provide valuable heuristic insight into the structure of a real-world classification problem.

The algorithms that are introduced in this paper make it feasible to compute optimal hypotheses of this type for a training set of several hundred examples. We also exhibit an approximation algorithm that can compute nearly optimal hypotheses for much larger datasets.

1 INTRODUCTION

One important goal of computational learning theory is to provide tools for the design and analysis of learning algorithms that provide satisfactory solutions for real-world learning problems. There exists already an extensive literature on empirical results regarding the performance of various heuristic learning algorithms on a number of "benchmark"-datasets for real-world classification problems (see e.g. [M], [WK 90], [WGT], [WK 91], [BN], [Ho]). However the set of learning algorithms that are examined in these applications to real-world classification problems is virtually disjoint from the set of learning algorithms that are traditionally considered in computational learning theory. The goal of this paper is to contribute tools for the design of learning algorithms that are of some interest for both theoretical and applied machine learning.

An important conceptual link between theoretical and applied machine learning has been provided by Haussler. He introduced in [H] a variation of Valiant's [V 84] well-known model for probably approximately correct learning ("PAC-learning"), that provides an adequate formal model for the generic scenario of real-world classification problems: the model for *agnostic PAC-learning*. The original PAC-learning model of Valiant relies on an assumption which is rarely met in these real-world classification problem: the assumption that the "labels" $b = C_t(z) \in \{0, 1\}$ of the training examples $\langle z, b\rangle$ arise from a target concept C_t of an a-priori known specific simple structure (such as a simple boolean formula or a geometrical object of bounded complexity). Valiant had also initiated in [V 85] the investigation of a relaxed version of the PAC-model where one allows that some labels b arise from "noise" rather than from a simple target concept. But so far one has only been able to prove positive learning results in such a model if one either assumes that the noise is of a very specific structure ([AL], [KS]), or if one assumes that the percentage η of noisy labels is only a small fraction of the desired error rate ε of the learner after training

([KL], [K], [D]). These constraints make it difficult to apply learning algorithms from this model to real-world classification problems.

Haussler's model for agnostic PAC-learning (see also [KSS]) requires no assumption at all about the labels of the examples $\langle z, b \rangle$. One assumes that some arbitrary (unknown) distribution D on $X \times \{0,1\}$ is given. For an application to a real-world learning problem this distribution D may simply reflect the distribution of data as they occur "in nature" (including "contradictions", i.e. for some $x \in X$ both $\langle x, 0 \rangle$ and $\langle x, 1 \rangle$ may occur as "examples"), without assuming that the labels are generated by some "rule". The goal of the learner is to compute for given parameters $\varepsilon, \delta > 0$ a hypothesis $H^* \in \mathcal{H}$ whose "true error"

$$\mathrm{Error}_D(H^*) := E_{\langle x,b \rangle \in D}[H^*(x) \neq b]$$

is with probability $\geq 1 - \delta$ not larger than

$$\varepsilon + \inf_{H \in \mathcal{H}} \mathrm{Error}_D(H).$$

To achieve this goal, the learner is allowed to specify a minimum size $m(\varepsilon, \delta)$ for a training set. The input for the computation of H^* by the learner is a training set (or "sample") S of $\geq m(\varepsilon, \delta)$ examples that are drawn from $X \times \{0,1\}$ according to D. One allows the learner to fail with probability $\leq \delta$, because this input S may occasionally be "untypical" for the actual distribution D.

A learner which can carry out this task for *any* distribution D over $X \times \{0,1\}$ is called an *efficient agnostic PAC-learner for hypothesis class* $\mathcal{H}$ if his sample bound $m(\varepsilon, \delta)$ (which has to be independent of D) and his number of computation steps (i.e. the number of computation steps that he needs for producing a hypothesis H^* with the property above from the sample S) can be bounded by a polynomial in the parameters involved (in particular in $1/\varepsilon$ and $1/\delta$).

Haussler [H] has shown that in order to prove a positive result for efficient agnostic PAC-learning with a specific hypothesis class $\mathcal{H} \subseteq 2^X$ of bounded VC-dimension it suffices to design an efficient algorithm for a related finite optimization problem, for which an efficient algorithmic solution is also very desirable from the point of view of applied machine learning: the *"minimizing disagreement problem for* $\mathcal{H}$*"*. This is the problem to compute for *any* given finite set $S \subseteq X \times \{0,1\}$ of labeled points from X some hypothesis $H \in \mathcal{H}$ whose "empirical error" $\mathrm{Error}_S(H)$ error is for S minimal among all hypotheses in $\mathcal{H}$ (the *empirical error* $\mathrm{Error}_S(H)$ of H for S is defined as the sum of the number of positive examples $\langle z, 1 \rangle \in S$ with $z \notin H$ and of the number of negative examples $\langle z, 0 \rangle \in S$ with $z \in H$). This reduction of the agnostic PAC-learning problem to a finite optimization problem is possible because of the following two *uniform convergence results:*

(A) Theorem 1 in [H] states that for

$$m(\varepsilon, \delta) := \frac{1}{2\varepsilon^2}(\ln|\mathcal{H}| + \ln\frac{2}{\delta})$$

one has for any sample S of $\geq m(\varepsilon, \delta)$ examples drawn with regard to some arbitrary distribution D over $X \times \{0,1\}$ that with probability $\geq 1 - \delta$ (with regard to the drawing of S) the following holds:

$$\forall H \in \mathcal{H}(|\mathrm{Error}_S(H) - \mathrm{Error}_D(H)| \leq \varepsilon).$$

(B) A recent result by Talagrand [T] (which slightly improves an earlier result by Haussler [H]) implies that under some rather harmless measurability conditions the same claim holds for

$$m(\varepsilon, \delta) := \frac{1}{\varepsilon^2}\Big(\text{VC-dimension}(\mathcal{H})(\ln K + \ln\ln(2K)) + \ln\frac{1}{\varepsilon} + \ln\frac{1}{\delta}\Big),$$

where VC-dimension($\mathcal{H}$) is the VC-dimension of $\mathcal{H}$ and K is some absolute constant that is conjectured to be not larger than 1000.

As an immediate consequence of these uniform convergence results one gets that any algorithm that solves the minimizing disagreement problem for $\mathcal{H}$ is, together with a bound for the minimal number of training examples of

$$m(\varepsilon, \delta) := \frac{1}{(\varepsilon/2)^2}\Big(\text{VC-dimension}(\mathcal{H})(\ln K + \ln\ln(2K)) + \ln\frac{1}{\varepsilon/2} + \ln 1/\delta\Big)$$

(or $m(\varepsilon, \delta) := \frac{1}{2(\varepsilon/2)^2}(\ln|\mathcal{H}| + \ln\frac{2}{\delta})$ in case that $\mathcal{H}$ is finite), an agnostic PAC-learner for hypothesis class $\mathcal{H}$. The same reasoning implies that even an efficient *approximation algorithm* for the minimizing disagreement problem for $\mathcal{H}$ is useful, because such algorithm can be used to produce with high probability a hypothesis $H \in \mathcal{H}$ whose true error $\mathrm{Error}_D(H)$ is optimal up to some fixed $\varepsilon_0 > 0$.

It should also be pointed out that it has been shown in [KSS] that for any hypothesis class $\mathcal{H}$ the existence of an efficient algorithm for the minimizing disagreement problem for $\mathcal{H}$ is a *necessary* condition for efficient agnostic PAC-learning with hypothesis class $\mathcal{H}$.

Unfortunately it has turned out that for many interesting hypothesis classes $\mathcal{H}$ the minimizing disagreement problem is computationally very hard. In particular it was shown (under the assumption that $RP \neq NP$) that there does not exist a polynomial time algorithm which solves this problem for the class of monomials ([KSS]), or for the class of halfspaces ([HSV]).

In this paper we investigate agnostic PAC-learning for some of the few interesting hypothesis classes $\mathcal{H}$ for which the minimizing disagreement problem can be solved by a polynomial time algorithm:

- the class $\mathcal{R}_2$ of axis parallel rectangles $[\ell, r] \times [y, t] \subseteq \mathbf{R}^2$ (with arbitrary $\ell, r, y, t \in \mathbf{R}$)
- the class $\mathcal{U} - \mathcal{R}_2$ of unions of two disjoint rectangles from $\mathcal{R}_2$
- the class I_k of up to k disjoint intervals over $\mathbf{R}$.

In the case of $\mathcal{R}_2$ it is obvious that the minimizing disagreement problem can be solved in time polynomial in the number m of different examples in S, since it suffices to compute the error $\text{Error}_S(R)$ of those $O(m^4)$ $R \in \mathcal{R}_2$ whose edges are occupied by positive examples from S. Hence a naive exhaustive search solves the minimizing disagreement problem for $\mathcal{R}_2$ in time $O(m^5)$. By employing a smart datastructure for orthogonal range queries (see e.g. [Me], [PS]) one can reduce this computation time to $O(m^4 \log m)$. Although this time bound is polynomial, it is so large that it makes this algorithm inapplicable for most real-world datasets S (which typically consist of several hundred examples). The goal of this paper is to provide tools for the design of more efficient algorithms for solving the minimizing disagreement problem for simple geometrical hypothesis classes such as $\mathcal{R}_2$, $\mathcal{U} - \mathcal{R}_2$, I_k, or the classes of their complements.

Simple hypothesis classes of this type have turned out to be quite interesting from the point of view of applied machine learning. Weiss et al. (see [WK 90], [WGT], [WK 91]) have shown through experiments that for many of the standard benchmark datasets a short rule that depends on only two of the numerous attributes, and which is a boolean combination of expressions of the "$x_i > c$" or "$x_i = c$" provides one of the best available prediction-rules. In particular it is reported in [WK 91] that the best prediction rule of this form for the appendicitis data turns out to be the complement of a rectangle from $\mathcal{R}_2$ (in 2 of the 8 attributes of this dataset). In addition Holte [Ho] has shown through extensive experiments that for many common datasets a hypothesis which consists simply of several intervals in just *one* of its attributes has a prediction error that comes within a few percent of the much more complex hypothesis that is generated for the same dataset (with all attributes!) by one of the most sophisticated existing learning algorithms (Quinlan's algorithm C 4.5, see [Q]). In particular this holds true for various common benchmark datasets from medical studies (for breast cancer, heart disease, hepatitis, thyroid-disease) as well as for standard datasets regarding labor negotiations, mushroom classification, and voting records.

Finally we would like to point out that optimal hypotheses of a simple type as they are considered in this paper have the additional advantage that they may provide to the human user valuable heuristic insight into the structure of a real-world learning problem.

We develop in section 2 of this paper a new datastructure ("MIN-trees"), which allows us to solve the minimizing disagreement problem for $\mathcal{R}_2$ and $\mathcal{U} - \mathcal{R}_2$ in $O(m^2 \log m)$ steps. In section 3 we use this new algorithm to design a faster approximation algorithm for the same problem, and we point out that a variation of MIN-trees gives rise to an efficient algorithm that solves the minimizing disagreement problem for the hypothesis class I_k.

2 EFFICIENT AGNOSTIC LEARNING WITH RECTANGLES AND UNIONS OF RECTANGLES

Let $S \subseteq \mathbf{R}^2 \times \{0,1\}$ be an arbitrary finite multi-set. Thus S is formally a function w_S from $\mathbf{R}^2 \times \{0,1\}$ into $\mathbf{N}$ which is nonzero for only finitely many arguments, where $w_S(\langle z, b\rangle)$ denotes the "weight" (i.e. number of occurrences) of $\langle z, b\rangle$ in S. We set $|S| := \sum_{\langle z,b\rangle \in \mathbf{R}^2 \times \{0,1\}} w_S(\langle z, b\rangle)$. We refer to an element $\langle z, 1\rangle \in S$ as a *positive example*, and to $\langle z, 0\rangle \in S$ as a *negative example*.

We consider *multi-sets* rather than sets, because in applications to real-world datasets z may consist of just two of the numerous attributes of the examples (hence different examples may give rise to the same z), and because we need the ability to assign "weights" to examples in order to apply Theorem 1 and 2 later in section 3 for the design of a very efficient approximation algorithm.

We first consider the hypothesis class

$$\mathcal{R}_2 := \{[\ell, r] \times [y, t] : \ell, r, y, t \in \mathbf{R} \text{ with } \ell \leq r \text{ and } y \leq t\} \cup \{\emptyset\}.$$

For any set $M \subseteq \mathbf{R}^2$ and any rectangle $R \in \mathcal{R}_2$ with

$R \subseteq M$ we define the *error of R in M for S* as

$$\sum_{z \in M-R} w_S(\langle z, 1\rangle) + \sum_{z \in R} w_S(\langle z, 0\rangle).$$

In the special case $M := \mathbf{R}^2$ we say "*error of R for S*" instead of "error of R in $\mathbf{R}^2$ for S".

We call a multi-set $S \subseteq \mathbf{R}^2 \times \{0,1\}$ *reduced* if there does not exist any $z \in \mathbf{R}^2$ with $\langle z, 0\rangle \in S$ and $\langle z, 1\rangle \in S$. One can easily transform any multi-set S into a reduced multi-set by removing all pairs of the form $(\langle z, 0, \rangle, \langle z, 1, \rangle)$ for any $z \in \mathbf{R}^2$, until only positive, or only negative examples for z are left in S (or neither of them). This process of transforming S into a reduced set S' has the property that it changes for any area $M \subseteq \mathbf{R}^2$ the error of all $R \in \mathcal{R}_2$ in M by *the same* additive term. Hence any $R \in \mathcal{R}_2$ has minimal error in M for S if and only if it has minimal error in M for S'. Therefore we may assume w.l.o.g. in the following that the given set S is already reduced (i.e. we really apply our algorithm to the associated reduced set S'). We would like to point out that it is easy to construct sets S of m examples such that there exist $\Omega(m^4)$ different $R \in \mathcal{R}_2$ of minimal error for S whose edges are all occupied by positive examples form S. Hence an efficient algorithm has to avoid examining all promising candidates of this type.

Theorem 1: *One can compute in $O(m^2 \log m)$ steps on a RAM for any multi-set S that contains m different points from $\mathbf{R}^2 \times \{0,1\}$ some rectangle $R \in \mathcal{R}_2$ that has minimal error for S.*

Proof: We write $(z)_1, (z)_2$ for the first respectively second coordinate of any $z \in \mathbf{R}^2$. For any $y \in \mathbf{R}$ we define a multi-set S_y by

$$S_y := \{\langle z, b\rangle \in S : (z)_2 \geq y\}.$$

Our algorithm computes for any $y \in \mathbf{R}$ with $y = (z)_2$ for some $\langle z, b\rangle \in S$ a rectangle $R \in \mathcal{R}_2$ whose bottom edge lies at y, so that R has the property that it has minimal error for S_y in $\{z \in \mathbf{R}^2 : (z)_2 \geq y\}$ among all rectangles of this type. In order to compute such rectangle R in an efficient manner, we design a special data-structure: a *MIN-tree for $\langle S_y, y\rangle$*.

A MIN-tree for $\langle S_y, y\rangle$ is a labeled binary tree. The number of leaves of such tree is equal to the number of pairwise different first coordinates $(z)_1$ of elements $\langle z, b\rangle \in S_y$. We assume that the tree is balanced (i.e. the distance from the root differs for any two leaves by at most 1).

The label of each leaf consists of two components: the first coordinate $x = (z)_1$ of some $\langle z, b\rangle \in S_y$ as first component, and as second component a list of all $\langle z, b\rangle \in S_y$ with $(z)_1 = x$ (together with their weight $w_S(\langle z, b\rangle)$ in S), ordered according to $(z)_2$. We assume that the first components of the labels of any two leafs are pairwise different, and ordered in increasing order from the leftmost to the rightmost leaf.

Assume now that ν is an internal node of a MIN-tree for $\langle S_y, y\rangle$. Let $\ell\,(r)$ be the first component of the label of the leftmost (rightmost) leaf below ν. The label of ν is a list A_ν of records, one for each $\langle z, b\rangle \in S_y$ with $(z)_1 \in [\ell, r]$. These records in A_ν are sorted according to $(z)_2$. The record for $\langle z, b\rangle$ contains the weight of $\langle z, b\rangle$ in the multi-set S_y, and five other components to which we will refer as its A-, B-, C-, D-, and E-components. These components consist of the following data:

***A*-component:** a pair $\langle [\ell', r'], e\rangle$ such that ℓ', r' occur as first components of labels of leafs below ν (hence $[\ell', r'] \subseteq [\ell, r]$), e is the error of the rectangle $[\ell', r'] \times [y, (z)_2]$ in $[\ell, r] \times [y, \infty)$ for S_y, and no rectangle $[\tilde{\ell}, \tilde{r}] \times [y, (z)_2]$ such that $\tilde{\ell}, \tilde{r}$ are first components of labels of leafs below ν has error less than e in $[\ell, r] \times [y, \infty)$ for S_y

***B*-component:** a pair $\langle [\ell, r'], e\rangle$ with the same properties as in the A-component, but with the additional constraint that $\ell' = \ell$
(in particular: no rectangle $[\ell, \tilde{r}] \times [y, (z)_2]$ where $\tilde{r}$ is the first component of the label of a leaf below ν has error less than e in $[\ell, r] \times [y, \infty)$ for S_y)

***C*-component:** a pair $\langle [\ell', r], e\rangle$ with the same properties as in the A-component, but with the additional constraint that $r' = r$
(in particular: no rectangle $[\tilde{\ell}, r] \times [y, (z)_2]$ where $\tilde{\ell}$ is the first component of the label of a leaf below ν has error less than e in $[\ell, r] \times [y, \infty)$ for S_y)

***D*-component:** the error e of $[\ell, r] \times [y, (z)_2]$ in $[\ell, r] \times [y, \infty)$ for S_y

***E*-component:** the error e of $\emptyset$ in $[\ell, r] \times [y, \infty)$ for S_y.

This completes the definition of a MIN-tree for $\langle S_y, y\rangle$.

It is obvious from this definition that one can compute in linear time from the label A_ν of the root ν of a MIN-tree for $\langle S_y, y\rangle$ a rectangle $R \in \mathcal{R}_2$ with bottom edge at y such that among all such rectangles R has minimal error in $(-\infty, \infty) \times [y, \infty)$ for S_y. To see this, one just has to note that w.l.o.g. the upper edge of such optimal R is at $(z)_2$ for some $\langle z, b\rangle \in S_y$, and that its left and right edge are given by the first coordinates $(\tilde{z})_1$ of points $\langle \tilde{z}, \tilde{b}\rangle \in S_y$. Hence $R := [\ell', r'] \times [y, (z)_2]$ has the desired property, where $[\ell', r']$ is the first part of the A-component of the record for $\langle z, b\rangle$ in A_ν.

The only purpose of the B- to E-components in the records for elements $\langle z, b\rangle$ in the lists A_ν of a MIN-tree is to facilitate the computation of labels for predecessors of ν. They allow us to compute in a recursive manner, in the direction from the leafs to the root, the labels for all nodes in a MIN-tree for $\langle S_y, y\rangle$ in altogether $O(m \log m)$ steps.

One starts such efficient computation of labels by sorting all $\langle z, b\rangle$ in S_y according to $(z)_1$, and by sorting then all $\langle z, b\rangle$ with common $(z)_1$ according to $(z)_2$. One can then construct a binary tree and assign labels to all of its leaves according to the preceding definition of a MIN-tree in altogether $O(m \log m)$ steps.

We then consider any interior node ν of that binary tree so that labels A_{ν_1}, A_{ν_2} for its sons ν_1 and ν_2 have already been computed. We consider here only the most interesting case where neither ν_1 nor ν_2 are leafs. For $i = 1, 2$ let $\ell_i(r_i)$ be the label of the leftmost (rightmost) leaf below ν_i. Let $\langle z, b\rangle$ be some element of S_y with $(z)_1 \in [\ell_1, r_2]$. We only consider the case where $(z)_1 \in [\ell_1, r_1]$ (the case where $(z)_1 \in [\ell_2, r_2]$ is handled analogously).

Computation of the A-component of the record for $\langle z, b\rangle$ in A_ν:

One compares the following three sums:

(1) (the second part e of the A-component $\langle [\ell', r'], e\rangle$ in the record for $\langle z, b\rangle$ in A_{ν_1}) + (the E-component of any record in A_{ν_2})

(2) One chooses some $\langle \tilde{z}, \tilde{b}\rangle$ with $(\tilde{z})_2 \le (z)_2$ that has a record in A_{ν_2} so that $(\tilde{z})_2$ is as large as possible. One then considers the sum
(the second part of the C-component of the record for $\langle z, b\rangle$ in A_{ν_1}) + (the second part of the B-component of the record for $\langle \tilde{z}, \tilde{b}\rangle$ in A_{ν_2}).
If there exists no $\langle \tilde{z}, \tilde{b}\rangle$ with the properties above one replaces the second summand by the E-component of any record inA_{ν_2}.

(3) This sum is only considered if there exists some $\langle \tilde{z}, \tilde{b}\rangle$ with $(\tilde{z})_2 \le (z)_2$ that has a record in A_{ν_2}. One chooses again such $\langle \tilde{z}, \tilde{b}\rangle$ so that $(\tilde{z})_2$ is as large as possible, and considers
(the E-component of any record in A_{ν_1}) + (the second part of the A-component of the record for $\langle \tilde{z}, \tilde{b}\rangle$ in A_{ν_2}).

We define the second part of the A-component of the record for $\langle z, b\rangle$ in A_ν to be the least value of these three (respectively two) sums in (1) - (3). We define the first part of the A-component for $\langle z, b\rangle$ to be the associated interval. In case (1) this is the interval $[\ell', r']$ that is mentioned in (1), and in case (2) it is an interval $[\ell', r']$ so that $[\ell', r_1]$ is the first part of the C-component of the record for $\langle z, b\rangle$ in A_{ν_1} and $[\ell_2, r']$ is the first part of the B-component of the record for $\langle \tilde{z}, \tilde{b}\rangle$ in A_{ν_2}. If there is no $\langle \tilde{z}, \tilde{b}\rangle$ with the properties that are specified in (2), one takes instead $[\ell', r_1]$ as first part of the A-component for $\langle z, b\rangle$. In case (3) one takes the first part of the A-component in the record for $\langle \tilde{z}, \tilde{b}\rangle$ in A_{ν_2}.

Computation of the B-component of the record for $\langle z, b\rangle$ in A_ν:

One compares the following three sums:

(1) (the second part e of the B-component $\langle [\ell_1, r'], e\rangle$ in the record for $\langle z, b\rangle$ in A_{ν_1}) + (the E-component of any record in A_{ν_2})

(2) One chooses some $\langle \tilde{z}, \tilde{b}\rangle$ with $(\tilde{z})_2 \le (z)_2$ that has a record in A_{ν_2} so that $(\tilde{z})_2$ is as large as possible. One then considers the sum
(D-component of the record of $\langle z, b\rangle$ in A_{ν_1}) + (the error e from the B-component $\langle [\ell_2, r'], e\rangle$ of the record of $\langle \tilde{z}, \tilde{b}\rangle$ in A_{ν_2}).
If there exists no $\langle \tilde{z}, \tilde{b}\rangle$ with the properties above one replaces the second summand by the E-component of any record inA_{ν_2}.

The second part of the B-component of the record of $\langle z, b\rangle$ in A_ν is defined as the smaller one of these two sums. One takes in either case $[\ell_1, r']$ as first part of that component.

Computation of the C-component of the record for $\langle z, b\rangle$ in A_ν:

analogous to the computation of the B-component.

Computation of the D-component of the record for $\langle z, b\rangle$ in A_ν:

Choose some $\langle \tilde{z}, \tilde{b}\rangle$ with $(\tilde{z})_2 \le (z)_2$ that has a record in A_{ν_2} so that $(\tilde{z})_2$ is as large as possible. Then one takes (the D-component of $\langle z, b\rangle$ in A_{ν_1}) + (the D-component of $\langle \tilde{z}, \tilde{b}\rangle$ in A_{ν_2}).
If there is no such $\langle \tilde{z}, \tilde{b}\rangle$ we replace the second summand by the E-component of any record in A_{ν_2}.

Computation of the E-component of the record for $\langle z, b\rangle$ in A_ν:

One takes
(the E-component of any record in A_{ν_1}) + (the E-component of any record in A_{ν_2}).

It is easy to verify that the computed data satisfy the requirements for the corresponding components in the definition of a MIN-tree. For example for the A-component one has to show in case that the sum from (2) is smallest that there exists no rectangle $[\tilde{\ell}, \tilde{r}] \times [y, (z)_2]$ with $\ell_1 \leq \tilde{\ell} \leq r_1$ and $\ell_2 \leq \tilde{r} \leq r_2$ that has a smaller error for S_y in $[\ell_1, r_2] \times [y, \infty)$.

It remains to be shown that the indicated computation of the lists A_ν for *all* interior nodes ν of the MIN-tree requires altogether only $O(m \log m)$ steps. We exploit here that each of these lists is ordered according to $(z)_2$. Hence one can find $\langle \tilde{z}, \tilde{b} \rangle$ with the properties as in the preceding definition of the algorithm (provided that it exists) in an efficient manner by essentially "merging" the lists A_{ν_1} and A_{ν_2}. In this way the number of computation steps that are needed to compute A_ν is linear in the number of records in A_ν.

Finally one exploits that each $\langle z, b \rangle \in S_y$ occurs in at most $O(\log m)$ lists A_ν in a MIN-tree for $\langle S_y, y \rangle$ to show that only $O(m \log m)$ steps are needed altogether to compute all such lists A_ν in that tree. One computes in this way in altogether $O(m^2 \log m)$ steps MIN-trees for $\langle S_y, y \rangle$ for all $y = (z)_2$ for some $\langle z, b \rangle \in S$. As indicated before, one can compute from each of these $O(m)$ MIN-trees in $O(m)$ steps some $R_y \in \mathcal{R}_2$ with bottom edge at y such that R_y has minimal error in $(-\infty, \infty) \times [y, \infty)$ for S_y, together with that error of R_y. One adds to that error of R_y for S_y the number of positive examples $\langle \tilde{z}, 1 \rangle \in S$ with $(\tilde{z})_1 < y$. This yields the error of R_y for S. The algorithm outputs one of these rectangles R_y that has minimal error for S. ∎

We define

$$\mathcal{U} - \mathcal{R}_2 \ := \ \{R_1 \cup R_2 : \ R_1, R_2 \in \mathcal{R}_2 \text{ and } R_1 \cap R_2 = \emptyset\}.$$

Remark: Dobkin and Gunopulos have independently discovered another proof of Theorem 1 (see their video [DG a] and their paper [DG b]), while exploring algorithmic questions about the bichromatic discrepancy of rectangles, that are of interest in the context of computer graphics. A forthcoming article by Dobkin, Gunopolis and Maass [in preparation] will highlight this quite interesting convergence of algorithmic problems from two seemingly unrelated areas of computer science.

Theorem 2: *One can compute on a RAM for any multi-set S with m different elements from $\boldsymbol{R}^2 \times \{0, 1\}$ in $O(m^2 \log m)$ steps a hypothesis $H \in \mathcal{U} - \mathcal{R}_2$ which has minimal error for S.*

Proof: We first search for an optimal pair R, R' of disjoint rectangles that are separated by a *horizontal* line. We compute in altogether $O(m^2 \log m)$ steps for all $\langle z, b \rangle \in S$ a MIN-tree for $\langle S_{(z)_2}, (z)_2 \rangle$, and also a dual version of such tree where one considers rectangles whose *upper* edge lies at $(z)_2$. In addition one computes in altogether $O(m^2)$ steps for any two examples $\langle z, b \rangle, \langle z', b' \rangle \in S$ with $(z)_2 > (z')_2$ the error of the empty set in $\mathbf{R} \times ((z')_2, (z)_2)$. From these data one can find in $O(m^2)$ further steps a pair R, R' of rectangles with a horizontal separating line that has minimal error for S among all such pairs of rectangles.

Finally one compares the error of this pair R, R' with that of the optimal pair of rectangles with a *vertical* separating line, which is computed in an analogous manner. ∎

3 EXTENSIONS AND APPLICATIONS

For applications of our algorithms to very large training sets S one can trade off some of the quality of the computed hypothesis against a reduction in the computation time. According to the uniform convergence results that were quoted in section 1, it suffices to produce hypotheses $H \in \mathcal{H}$ with *almost optimal* empirical error in order to prove efficient agnostic PAC-learnability for a fixed ε. Such approximation algorithms are also of independent interest from the point of view of applied machine learning.

One straightforward approach for designing an approximation algorithm exploits Haussler's uniform convergence theorem for finite hypothesis classes from [H] (see result (A) in section 1 of this article) in combination with Theorem 1 (respectively Theorem 2). Consider any fixed multi-set $S \subseteq \mathbf{R}^2 \times \{0, 1\}$ which contains m different points. Assume that one draws n times elements from S with regard to the uniform distribution over S. Let $\tilde{S}$ with $|\tilde{S}| = n$ be the resulting multi-set. Then for any $\varepsilon, \delta > 0$ and $|\tilde{S}| \geq \frac{1}{2\varepsilon^2}(4 \ln m + \ln \frac{2}{\delta})$ one has with probability $\geq 1 - \delta$ that for any $R \in \mathcal{R}_2$ whose edges are occupied by positive examples in S the error of R for $\tilde{S}$ multiplied by $|S|/|\tilde{S}|$ differs from the error for S by at most $\varepsilon \cdot |S|$. Hence by applying Theorem 1 (resp. Theorem 2) to $\tilde{S}$ one gets with probability $\geq 1 - \delta$ a rectangle (resp. union of two rectangles) whose error for S is at most by $2\varepsilon \cdot |S|$ (resp. $4\varepsilon \cdot |S|$) larger than that of the best hypothesis of this type.

The following approach, where one replaces S instead in a deterministic fashion by a suitable set S_K of weighted points ("clusters"), is a bit more complicated. However it does not require random drawings from S, and it yields a somewhat faster approximation algorithm if m is large.

Theorem 3: *One can compute in $O(m \log m)$ steps on a RAM for any natural number K and any multi-set S that contains m different points from $\boldsymbol{R}^2 \times \{0,1\}$ a multi-set S_K of $\leq K^2$ different points from $\boldsymbol{R}^2 \times \{0,1\}$ such that for any rectangle $R \in \mathcal{R}_2$ the error of R for S and the error of R for S_K differ by at most $4 \cdot \frac{|S|}{K}$.*

Proof: The construction of S_K is more delicate than one might expect because of difficulties caused by points in S of weight > 1, by possible occurrences of several points $\langle z, b\rangle \in S$ with a common $(z)_1$ (respectively a common $(z)_2$), and by our desire to keep the constant factor in the estimate for the error difference of S and S_K small.

We start the construction of S_K by partitioning $\mathbf{R}^2$ from left to right into $\leq K$ vertical strips $(\ell, r] \times (-\infty, \infty)$ with

$$r := \inf\{r' \in \mathbf{R} : \sum\{w_S(\langle z, b\rangle) : \langle z, b\rangle \in (\ell, r'] \times (-\infty, \infty)\} \geq \frac{|S|}{K}\}.$$

One should note that the total weight of points from S in $(\ell, r] \times (-\infty, \infty)$ may be substantially larger than $\frac{|S|}{K}$ (because of several points $\langle z, b\rangle \in S$ with $(z)_1 = r$, or because of a point $\langle z, b\rangle$ with $(z)_1 = r$ of weight > 1). In fact this total weight of points in a strip may be arbitrarily large (i.e. as large as $|S|$). Therefore one has to argue rather carefully in the subsequent error estimate.

Each of these strips $(\ell, r] \times (-\infty, \infty)$ is partitioned independently from top to bottom into boxes $(\ell, r] \times [u, v)$ with

$$u := \sup\{u' \in \mathbf{R} : \sum\{w_S(\langle z, b\rangle) : \langle z, b\rangle \in (\ell, r] \times [u', v)\} \geq \frac{|S|}{K^2}\}.$$

One should note that such box may contain points with total weight much larger than $\frac{|S|}{K^2}$, and that a vertical strip may become partitioned into many more than K boxes.

For each box $B = (\ell, r] \times [u, v)$ we define the weight $w(B)$ of B by

$$w(B) := \sum\{w_S(\langle z, 1\rangle) : z \in B\} - \sum\{w_S(\langle z, 0\rangle) : z \in B\},$$

and a new point $p(B)$ by

$$p(B) := \begin{cases} \langle\langle r, u\rangle, 1\rangle\ , & \text{if } w(B) \geq 0 \\ \langle\langle r, u\rangle, 0\rangle\ , & \text{if } w(B) < 0. \end{cases}$$

The multi-set S_K is defined as the set of all points $p(B)$ for boxes B that arise in the preceding construction, with the weight $w_{S_K}(p(B))$ of $p(B)$ defined as $|w(B)|$. It is easy to see that S_K can be computed in $O(m \log m)$ steps.

We consider now an arbitrary rectangle $R = [\alpha, \beta] \times [\gamma, \delta]$ from $\mathcal{R}_2$. A difference in the error of R for S and S_K can only arise from boxes B that are cut by R (i.e. both $R \cap B$ and $(\mathbf{R}^2 - R) \cap B$ are nonempty). Unfortunately we neither have a good bound for the number of such boxes B, nor for their individual weights $w(B)$.

Nevertheless one can argue as follows. Assume that the right edge $x = \beta$ of R runs strictly within the strip $(\ell, r] \times (-\infty, \infty)$ from our partition (i.e. $\ell < \beta < r$). Then R does not contain $p(B)$ for any box B in this strip. Furthermore the total weight of all points $\langle z, b\rangle \in S$ with $(z)_1 \leq \beta$ that lie in this strip is less than $\frac{|S|}{K}$ (by the definition of r). Therefore one can bound the total deviation between S and S_K of the error of R which arises in theses boxes B by $\frac{|S|}{K}$.

One can also show that the total weight of all points from S that lie in boxes $B = (\ell, r] \times [u, v)$ that are intersected by the upper edge $y = \delta$ of R and which lie strictly above the line $y = \delta$ can be bounded by $\frac{|S|}{K}$. This holds in spite of the fact that we have no bound on the total weight of points from S in any such box B. One exploits here the definition of u in order to bound the total weight of points from S above the line $y = \delta$ in any such box B by $\frac{|S|}{K^2}$. The definition of $p(B)$ implies that the deviation of the error of R between S and S_K that is caused by points in B can also be bounded by $\frac{|S|}{K^2}$.

The argument for boxes that are intersected by the left respectively bottom edge of R is similar. ∎

By applying the algorithms from Theorem 1 and Theorem 2 to the multi-set S_K from Theorem 3 one gets the following result.

Theorem 4: *One can compute in $O(m \log m + K^2 \log K)$ steps on a RAM for any multi-set S that contains m different points from $\boldsymbol{R}^2 \times \{0,1\}$ and for any given $K \in \boldsymbol{N}$ a rectangle from $\mathcal{R}_2$ (or a union of two disjoint rectangles from $\mathcal{R}_2$) whose error for S is by at most $8\frac{|S|}{K}$ larger than the minimal error of any hypothesis of this type.* ∎

Holte [Ho] has shown that for many of the common benchmark datasets one gets already very good prediction rules by using as hypotheses simply unions of k intervals in *one* of the attributes. He used for that a heuristic algorithm that computed a hypothesis without any *guarantee* for its quality. However with the help of a simple variation of the MIN-tree one can actually compute quite fast the *best* hypothesis of this type (for any given k):

Theorem 5: *One can compute in $O(m(\log m + k^2))$ steps on a RAM for any multi-set $S \subseteq \boldsymbol{R} \times \{0,1\}$ that contains m different points and any given $k \in \boldsymbol{N}$ a union of $\leq k$ intervals that has minimal error for S (in comparison with all other hypotheses of this type).*

Proof: One builds a balanced binary tree whose leafs are labeled from left to right by the ordered points $x \in \mathbf{R}$ with $\langle x, b\rangle \in S$ for some $b \in \{0,1\}$ (together with their multiplicities). For any interior node ν where $\ell\,(r)$ is the label of the leftmost (rightmost) leaf below ν the label of ν contains for each $j \in \{0,\dots,k\}$ the following 4 components:

***A*-component:** a pair $\langle\langle I_1,\dots,I_j\rangle, e\rangle$ such that $I_1,\dots,I_j$ are disjoint closed intervals with endpoints in $[\ell,r] \cap \{x \in \mathbf{R} : \langle x,1\rangle \in S\}$ and e is the error of $I_1 \cup \ldots \cup I_j$ in $[\ell,r]$ for S, such that there are no intervals $I'_1,\dots,I'_j$ with these properties so that $I'_1 \cup \ldots \cup I'_j$ has error less than e in $[\ell,r]$ for S

***B*-component:** a pair $\langle\langle I_1,\dots,I_j\rangle, e\rangle$ with analogous properties as in the A-component, but with the additional constraint that ℓ is the left endpoint of I_1

***C*-component:** a pair $\langle\langle I_1,\dots,I_j\rangle, e\rangle$ with analogous properties as in the A-component, but with the additional constraint that r is the right endpoint of I_j

***D*-component:** a pair $\langle\langle I_1,\dots,I_j\rangle, e\rangle$ with analogous properties as in the A-component, but with the additional constraint that ℓ is the left endpoint of I_1 *and* r is the right endpoint of I_j.

One can construct this datastructure from S in $O(m \log m + mk^2)$ steps on a RAM, since only $O(k^2)$ steps are needed to compute the label of an internal node in the tree. A union of $\leq k$ intervals $\subseteq \mathbf{R}$ with minimal error for S in $\mathbf{R}$ can be trivially computed from the label of the root of the tree. ∎

4 CONCLUSION

We have introduced in this paper tools for the design of more efficient learning algorithms for agnostic PAC-learning with very simple hypothesis classes such as rectangles and unions of two rectangles in two dimensions, and unions of k intervals in one dimension. One automatically gets from that also efficient learning algorithms for the class of complements of sets in $\mathcal{R}_2$ resp. $\mathcal{U} - \mathcal{R}_2$ (just apply the algorithm to a variation of S where positive and negative examples have been interchanged).

Preceding empirical studies suggest that for various real-world learning problems there exist good prediction rules of this type ([WGT], [WK 90], [WK 91], [Ho]). The algorithms that are exhibited in this paper may potentially provide better hypotheses from these classes as the heuristic algorithms that were used in these studies, since our algorithms come together with a rigorous performance guarantee (their hypothesis has the least possible empirical error, i.e. the smallest number of classification errors on the training set). Furthermore the almost quadratic computation time of the algorithms from Theorem 1 and Theorem 2 make them applicable to common datasets that consist of several hundred examples (even in combination with an exhaustive search over all pairs of attributes; there are usually around 15 or 20 attributes).

We have also exhibited in Theorem 4 a very fast approximation algorithm that can be used to compute almost optimal hypotheses of the considered type even for very large datasets.

ACKNOWLEDGEMENT

I would like to thank Alok Aggarwal, Franz Aurenhammer, Bernard Chazelle, Herbert Edelsbrunner, Dimitrios Gunopulos, David Haussler, Haym Hirsh, Sholom Weiss, and Lorenz Wernisch for helpful communications regarding the research for this paper.

References

[AL] D. Angluin, P. Laird, *Learning from noisy examples*, Machine Learning, vol. 2, 1988, 343 - 370

[BN] W. Buntine, T. Niblett, *A further comparison of splitting rules for decision-tree induction*, Machine Learning, vol. 8, 1992, 75 - 82

[D] S. E. Decatur, *Statistical queries and faulty PAC oracles*, Proc. of the 6th ACM Conference on Computational Learning Theory, 1993, 262 - 268

[DG a] D. P. Dobkin, D. Gunopulos, *Computing the rectangle discrepancy (video)*, 3rd Annual Video Review of Computational Geometry

[DG b] D. P. Dobkin, D. Gunopulos, *Computing the rectangle discrepancy*, Tech Report 443-94, Princeton University

[H] D. Haussler, *Decision theoretic generalizations of the PAC-model for neural nets and other learning applications*, Inf. and Comp., vol. 100, 1992, 78 - 150

[HSV] K. U. Hoeffgen, H. U. Simon, K. S. Van Horn, *Robust trainability of single neurons*, preprint (1993)

[Ho] R. C. Holte, *Very simple classification rules perform well on most commonly used datasets*, Machine Learning, vol. 11, 1993, 63 - 91

[K] M. Kearns, *Efficient noise-tolerant learning from statistical queries*, Proc. of the 25th ACM Symp. on the Theory of Computing, 1993, 392 - 401

[KL] M. Kearns, M. Li, *Learning in the presence of malicious errors*, SIAM J. Comput., vol. 22, 1993, 807 - 837

[KS] M. J. Kearns, R. E. Schapire, *Efficient distribution-free learning of probabilistic concepts*, Proc. of the 31st Annual IEEE Symp. on Foundations of Computer Science, 1990, 382 - 391

[KSS] M. J. Kearns, R. E. Schapire, L. M. Sellie, *Toward efficient agnostic learning*, Proc. of the 5th ACM Workshop on Computational Learning Theory, 1992, 341 - 352

[M] J. Mingers, *An empirical comparision of pruning methodes for decision tree induction*, Machine Learning, vol. 4, 1989, 227 - 243

[Me] K. Mehlhorn, *Multi-Dimensional Searching and Computational Geometry*, Springer, 1984

[PS] F. P. Preparata, M. I. Shamos, *Computational Geometry*, Springer, 1985

[Q] J. R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann, 1992

[T] M. Talagrand, *Sharper bounds for empirical processes*, to appear in Annals of Probability and its Applications

[V 84] L. G. Valiant, *A theory of the learnable*, Comm. of the ACM, vol. 27, 1984, 1134 - 1142

[V 85] L. G. Valiant, *Learning disjunctions of conjunctions*, Proc. of the 9th Intern. Joint Conf. on Art. Int., 1985, 560 - 566

[WGT] S. M. Weiss, R. Galen, P. V. Tadepalli, *Maximizing the predictive value of production rules*, Art. Int., vol. 45, 1990, 47 - 71

[WK 90] S. M. Weiss, I. Kapouleas, *An empirical comparison of pattern recognition, neural nets, and machine learning classification methods*, Proc. of the 11th Int. Joint Conf. on Art. Int. 1990, Morgan Kaufmann, 781 - 787

[WK 91] S. M. Weiss, C. A. Kulikowski, *Computer Systems that Learn*, 1991, Morgan Kaufmann