Deciding the K-dimension is **PSPACE**-complete

Extended Abstract

Marcus Schaefer
School of CTI
DePaul University
243 S. Wabash Avenue
Chicago, Illinois 60604, USA
schaefer@cs.depaul.edu

January 18, 2000

Abstract

In 1988 Littlestone introduced the optimal mistake-bound learning model to learning theory. In this model the difficulty of learning a concept from a concept class is measured by the K-dimension of the concept class, which is a purely combinatorial notion. This is similar to the situation in PAC-learning, where the difficulty of learning can be measured by the Vapnik-Červonenkis dimension. We show that determining the K-dimension of a concept class is a **PSPACE**-complete problem where the concept class is given as a circuit. This also implies that any optimal learner (making the least number of mistakes) that works on all concept classes over finite universes has to be **PSPACE**-hard.

1 Introduction

In 1988 Littlestone added the Kafkaesque mistake-bound learning model to learning theory [3]. The learner (call him Mr. K) is repeatedly presented with samples from a finite (and completely meaningless) universe and has to decide whether the sample belongs to the concept he is supposed to learn. After the decision the teacher tells the learner whether his answer was correct. Every mistake Mr. K makes counts. The optimal mistake-bound is the largest number of mistakes Mr. K can be forced to make by the teacher (assuming Mr. K plays optimally).

The only knowledge the learner has in this game is the universe U from which the samples are drawn, the concept class $\mathcal{C} \subseteq \mathcal{P}(U)$ containing the concepts the teacher chooses from, and the history of the game.

The K-dimension is a combinatorial tool introduced by Littlestone [3] as a measure of the hardness of learning a concept class in the mistake-bound learning model. Littlestone showed that the largest number of mistakes a learner can be forced to make is precisely the K-dimension of $\mathcal{C}$.

For learning theorists it is of interest to determine the computational complexity of a learner that achieves the optimal mistake-bound for all concept classes over finite universes. Such an algorithm is called an *optimal MB-learning algorithm*. The work of Littlestone, and Frances and Litman [1] shows that there is an optimal MB-learning algorithm whose complexity is the same as the complexity of computing the K-dimension of concept classes (up to polynomial time reductions), and that with the help of an optimal MB-learning algorithm the K-dimension can be computed in polynomial time. Hence we can concentrate on the complexity of computing the K-dimension.

Formalizing the problem we are led to two different versions of the K-dimension and the optimal MB-learning algorithm problem depending on the way concept classes are represented as input. One philosophy represents classes as incidence matrices, the other as circuits. The question has been argued before in the case of the VC dimension problem [5]. In this paper we will concentrate on the circuit presentation, and append some remarks about the matrix case at the end. Our main result is that in the circuit representation the K-dimension is **PSPACE**-complete; the precise complexity of the K-dimension problem for matrix representations has not been settled yet, but Frances and Litman show that it lies above **LOGNP** (a class defined by Papadimitriou and Yannakakis [4]).

2 K-DIMENSION

A *concept* C is a collection of *points* from a *universe* U (that is $C \subseteq U$). A *concept class* $\mathcal{C}$ is a collection of concepts, i.e. $\mathcal{C} \subseteq \mathcal{P}(U)$. Remember that the *depth* of a rooted tree is the length (in number of edges) of the longest path from the root. In a *full binary tree* every node which is not a leaf has exactly two children (we distinguish between the left and the right child), and all leaves are on the same level. Thus a full binary tree of depth n has $2^{n+1}-1$ nodes: $2^n - 1$ internal nodes and 2^n leaves.

Definition 2.1 *A* decision tree T of depth n over U *is a full binary tree T of depth n whose internal nodes are labeled with points from U. A path p of T corresponds to a decision for each point along the path (left means out, and right means in). Thus we can call $C \subseteq U$* consistent *with a path p on T if the decisions along p are consistent with the information from C. We say that T* is a decision tree for $\mathcal{C} \subseteq \mathcal{P}(U)$ over U of depth n *if T is a decision tree over U of depth n, and each leaf is labeled by a concept in $\mathcal{C}$ which is consistent with the path the leaf is on.*

If a concept class $\mathcal{C}$ has a decision tree of depth at least n we say it has K-dimension at least n and write $K(\mathcal{C}) \geq n$.

We should point out that there is no relation of this problem to an **NP**-completeness result in the literature about decision trees due to Hyafil and Rivest [2, Problem MS15]. In their version decision trees are not full, and a collection of tests which are assigned to the inner nodes of the decision tree is part of the input.

Let us consider an example. As a universe we will take the natural numbers $\omega = \{0, 1, 2, 3, \ldots\}$. As a concept class we take the singletons: $\mathcal{C} = \{\{x\} : x \in \omega\}$. Consider the following tree: 17 at the root with two children, the left one labeled with $\{2\}$, the right one with $\{17\}$. This tree is a decision tree of depth at least one for $\mathcal{C}$, and hence the K-dimension of $\mathcal{C}$ is at least one. Moreover, we cannot have a decision tree of depth two, since this would require a concept containing at least two elements (along the rightmost path). Hence the K-dimension of $\mathcal{C}$ is one. As we mentionied earlier this is also the largest number of mistakes a learner can be forced to make on this concept class: the optimal MB learner's strategy on $\mathcal{C}$ is simply to answer "No" to all queries until the teacher claims a mistake. In that case the learner knows the concept to be learnt (the set containing the last query), and can from then on answer all queries correctly.

We now define the K-dimension problem in the Garey and Johnson style [2].

K-DIMENSION

Instance: A circuit C with $l + m$ input gates, and a positive integer k.

Question: Let $C_i := \{x \in \Sigma^m : C(i, x) = 1\}$ for $i \in \Sigma^l$, and $\mathcal{C} := \{C_i : i \in \Sigma^l\}$. Is $K(\mathcal{C}) \geq k$?

MATRIX K-DIMENSION

Instance: An $l \times m$ matrix C over $\{0, 1\}$, and a positive integer k.

Question: Let $C_i := \{x : C_{x,i} = 1, 1 \leq x \leq m\}$ for $1 \leq i \leq l$, and $\mathcal{C} := \{C_i : 1 \leq i \leq l\}$. Is $K(\mathcal{C}) \geq k$?

Given a circuit or a matrix C, let $\mathcal{C}(C)$ denote the concept class induced by it (as in the definition of K-DIMENSION). We will also write $K(C)$ for the more precise $K(\mathcal{C}(C))$.

We can now state our main result.

Theorem 2.2 K-DIMENSION *is* **PSPACE**-*complete (under m-reductions).*

Apart from its interest as a result on the computational complexity of a combinatorial notion, the theorem also allows us to draw a conclusion about the complexity of an optimal learner in the mistake-bound learning model introduced by Littlestone.

mistake-bound learning is a game in which teacher and learner move alternately. They agree on a concept class $\mathcal{C}$ over a universe U. Then the teacher starts by presenting an element of U. The learner responds with a 1 or a 0 meaning the presented elements is in or not in the concept to be learnt. The teacher then responds with the correct answer and the process is repeated. The teacher has to be honest in the sense that at each point in the game there is always a concept left that is consistent with the responses given to the learner. A wrong answer by the learner is called a *mistake*. The learner tries to minimize the number of mistakes in the worst case. The largest number of mistakes a teacher can force a learner to make for a given concept class $\mathcal{C}$ is called the *optimal mistake-bound* of $\mathcal{C}$ (this number exists if U is finite). A learner who achieves this bound for all concept classes is called an *optimal mistake-bound learner*. Note that there can be several optimal mistake-bound learners, since some points of the universe might not play a role in the concepts of $\mathcal{C}$.

Littlestone showed that the optimal mistake-bound of an optimal mistake-bound learner for a class $\mathcal{C}$ is just $K(\mathcal{C})$, hence the K-dimension is a measure of how difficult it is to learn a concept class in this model. Frances and Litman showed that if we have an optimal mistake-bound learner, then we can use this learner to compute the K-dimension of any concept class. This gives us the following corollary to the theorem.

Corollary 2.3 *An optimal mistake-bound learner is* **PSPACE***-hard.*

For the sake of completeness we include the reduction. It requires an easy combinatorial lemma.

Lemma 2.4 (Frances and Litman [1]) *Let $\mathcal{C}_0$ and $\mathcal{C}_1$ be concept classes. Then the K-dimension of $\mathcal{D} := \{C_0 \times C_1 : C_0 \in \mathcal{C}_0, C_1 \in \mathcal{C}_1\}$ is $K(\mathcal{C}_0) + K(\mathcal{C}_1)$.*

We will not prove this lemma here; its proof is straightforward, and can be found in the paper cited.

Lemma 2.5 (Frances and Litman [1]) *Using an optimal mistake-bound learner as an oracle we can compute* K-DIMENSION *in polynomial time.*

Proof Assume that we have two concept classes $\mathcal{D}_0$ and $\mathcal{D}_1$ and a point x such that $x \in D$ for all $D \in \mathcal{D}_1$, but $x \notin D$ for all $D \in \mathcal{D}_0$. Suppose an optimal MB algorithm queried on point x and universe $\mathcal{D}_0 \cup \mathcal{D}_1$ returns $i \in \{0, 1\}$. If the answer is correct, then the algorithm can be forced to make $K(\mathcal{D}_i)$ mistakes. If it is wrong $1 + K(\mathcal{D}_{1-i})$ (counting the wrong answer it just gave). Hence overall it can be forced to make $\max\{K(\mathcal{D}_i), 1 + K(\mathcal{D}_{1-i})\}$ mistakes (since the teacher chooses whether the answer is right or wrong). To minimize this term, the optimal learner has to return i if $K(\mathcal{D}_i) > K(\mathcal{D} + 1 - i)$ for $i \in \{0, 1\}$.

We will use this observation about the behavior of an optimal MB learner to tell whether $K(\mathcal{C}) \geq k$. Suppose $\mathcal{C}$ is a concept class over universe U_1. Let U_2 be a set of $2k - 1$ points disjoint from U_1 and $\mathcal{D}$ be the powerset of U_2 (hence $K(\mathcal{D}) = 2k - 1$). Look at the concept class $\mathcal{C}' := \{A \times B \cup \{x\} : A, B \in \mathcal{C}\}$ where x is a new point (not belonging to either U_1 or U_2). It follows from the previous lemma that $K(\mathcal{C}') = 2K(\mathcal{C})$. Therefore $K(\mathcal{C}) \geq k$ implies $K(\mathcal{C}') > K(\mathcal{D}) = 2k - 1$ and $K(\mathcal{C}) < k$ implies $K(\mathcal{C}') < K(\mathcal{D}) = 2k - 1$. Hence, if we present the optimal MB learner with sample x and concept class $\mathcal{C}' \cup \mathcal{D}$, the answer will be 1 if and only if $K(\mathcal{C}) \geq k$. $\diamondsuit$

The combinatorial lemma also gives us a simple nonapproximablity result for the K-dimension: computing an additive approximation to the K-dimension is as hard as computing the K-dimension itself.

Lemma 2.6 *If $|f(C) - K(C)| \leq c$ for all C, and some f and c, then $K \in \mathbf{FP}^f$.*

Proof Given a circuit C construct a new circuit C' accepting the concept class $\overbrace{C \times \cdots \times C}^{2c+1}$. Compute $f(C')$, round to the nearest multiple of $2c + 1$ and divide by $2c + 1$ to get $K(C)$. $\diamondsuit$

3 K-DIMENSION is complete for PSPACE

We begin with two combintatorial lemmas that will allow us to code existential and universal quantifiers in the **PSPACE**-completeness proof.

Lemma 3.1 (Disjoint union) *Suppose $\mathcal{C}$ is the union of concept classes $(\mathcal{C}_i)_{i\in I}$ which are defined on pairwise disjoint universes $(U_i)_{i\in I}$. Assume that for every i*

- *there is an $u_i \in U_i$ such that $u_i \in C$ for all $C \in \mathcal{C}_i$, and*
- *there is a $j \neq i$ such that $K(\mathcal{C}_i) = K(\mathcal{C}_j)$.*

Then $K(\mathcal{C}) = 1 + \max\{K(\mathcal{C}_i) : i \in I\}$.

Proof Choose $i \neq j \in I$ such that $n = K(\mathcal{C}_i) = K(\mathcal{C}_j)$ is maximal. Fix an n-decision tree T for $\mathcal{C}_i$ and T' for $\mathcal{C}_j$. Neither of them contains u_i as a label, so we can combine them by adding a root u_i and appending T to the one-side and T' to the zero-side. This gives us a $n+1$-decision tree for $\mathcal{C}$. Thus $K(\mathcal{C}) \geq 1 + \max\{K(\mathcal{C}_i) : i \in I\}$.

Consider an m-decision tree T for $\mathcal{C}$ with root x. The point x belongs to some universe U_i, and hence the one-side subtree of T is a $m-1$ decision tree for $\mathcal{C}_i$, whence $m-1 \leq K(\mathcal{C}_i)$ and therefore $K(\mathcal{C}) \leq K(\mathcal{C}_i) + 1$, implying that $K(\mathcal{C}) \leq 1 + \max\{K(\mathcal{C}_i) : i \in I\}$. ♢

Lemma 3.1 allows us to compute the maximum K-dimension of a family of concept classes. We will also need a way to compute the smallest K-dimension of a family of concept classes. Suppose we are given two concept classes $\mathcal{C}_0$ and $\mathcal{C}_1$ on disjoint universes U_0 and U_1 such that $k-1 \leq K(\mathcal{C}_i) \leq k$ for some k.

We construct a concept class $\mathcal{D}$ for which $K(\mathcal{D}) = 2 + \min\{K(\mathcal{C}_0), K(\mathcal{C}_1)\}$.

Choose a set of 3 new elements $U = \{u_i : i \in \{\varepsilon, 0, 1\}\}$. Let $S_{ab} := \{u_i : i1$ is a prefix of ab and $i \in \{0,1\}^{\leq 1}\}$ for $a, b \in \{0,1\}$. That is, $S_{00} = \{\}, S_{01} = \{u_0\}, S_{10} = \{u_\varepsilon\}$, and $S_{11} = \{u_\varepsilon, u_1\}$. With this define

$$\mathcal{D}_{ab} := \{\{(b,x) : x \in C\} \cup S_{ab} : C \in \mathcal{C}_a\},$$

for $a, b \in \{0,1\}$ and let $\mathcal{D} := \bigcup_{a,b\in\{0,1\}} \mathcal{D}_{ab}$.

Lemma 3.2 (Marked union) *Consider two concept classes $\mathcal{C}_0$ and $\mathcal{C}_1$ that are defined on pairwise disjoint universes U_0 and U_1 and for which $k-1 \leq K(\mathcal{C}_i) \leq k$ for some k. Then $K(\mathcal{D}) = 2 + \min\{K(\mathcal{C}_0), K(\mathcal{C}_1)\}$ for $\mathcal{D}$ as defined above.*

Proof Let $n = \min\{K(\mathcal{C}_0), K(\mathcal{C}_1)\}$. Then every $\mathcal{D}_{ab}$ has an n-decision tree T_{ab} none of whose nodes are labeled with elements of U. From the 3 elements of U build a tree T of 2 levels as follows: the root is labeled u_ε, the left child u_0, and the right child u_1. For every $a \in \{0,1\}$ make T_{a0} the left child of u_a and T_{a1} the right child. In this manner we get an $2+n$-decision tree for $\mathcal{D}$ which shows that $K(\mathcal{D}) \geq 2 + n = 2 + \min\{K(\mathcal{C}_i) : i \in I\}$.

Fix a decision tree T for $\mathcal{D}$. We have to show that T has depth at most $2+n$. If the root of T is labeled with some point $x \in U_0 \cup U_1$, then the right subtree of the root has depth at most $\max\{K(\mathcal{C}_0), K(\mathcal{C}_1)\} \leq k$. This implies that the whole tree has depth at most

$1+k = 2+(k-1) \leq 2+n$ and we are done. Hence we can assume that the root is labeled with a point from U. We want to show that this point is ε. Suppose it is u_a for some $a \in \{0,1\}$. Then the right subtree of the root only contains concepts from $\mathcal{D}_{a1}$ (since only S_{a1} contains u_a). Therefore the depth of that subtree is at most $K(\mathcal{D}_{a1}) = K(\mathcal{C}_a) \leq k$, and hence the whole tree has depth at most $1 + k \leq 2 + n$. Thus we can assume that the root is labeled with u_ε.

This implies that all the concepts on the left-hand side of the root are from $\mathcal{D}_{00} \cup \mathcal{D}_{01}$, and all the concepts on the right-hand side of the root are from $\mathcal{D}_{10} \cup \mathcal{D}_{11}$.

Consider $\mathcal{C}_a$ for some $a \in \{0,1\}$. If $a = 0$ let p be the left child of the root, if $a = 1$, let p be the right child with label x. Let q be the right child of p.

If $x \in U$ then $x = u_a$ and the subtree rooted in q contains elements from $\mathcal{D}_{a1}$ only and is therefore a decision tree for $\mathcal{D}_{a1}$ whence it has depth at most $K(\mathcal{D}_{a1}) = K(\mathcal{C}_a)$. Therefore the whole tree has depth at most $2 + K(\mathcal{C}_a)$.

Now if $x \notin U$ then it has to be either $(0, y)$ or $(1, y)$ for some $y \in U_a$ since the only concepts consistent with the decisions up to p are the concepts in $\mathcal{D}_{a0}$ and $\mathcal{D}_{a1}$. Hence the subtree rooted in q is a decision tree for one of these concept classes and therefore has depth at most $K(\mathcal{D}_{a0}) = K(\mathcal{D}_{a1}) = K(\mathcal{C}_a)$. Hence the whole tree has depth at most $2 + K(\mathcal{C}_a)$.

Since a was chosen arbitarily we obtain $K(\mathcal{D}) \leq 2 + n$ which is what we had to show. $\diamondsuit$

Remark. The construction of $\mathcal{D}$ generalizes to a family of 2^m concept classes with $K(\mathcal{D}) = m + 1 + \min\{K(\mathcal{C}_i) : i \in \{0,1\}^m\}$, but the proof becomes less readable. Instead we will repeatedly apply the above construction, giving us the slightly worse $K(\mathcal{D}) = 2m+\min\{K(\mathcal{C}_i) : i \in \{0,1\}^m\}$.

We are now ready to begin the proof of Theorem 2.2. First we need to show that the problem lies in **PSPACE**. The K-dimension of a concept class $\mathcal{C}$ is at most $\log |\mathcal{C}|$, since every leaf of a decision tree for $\mathcal{C}$ must be labeled with a different concept from $\mathcal{C}$. Hence $K(C) = K(\mathcal{C}(C)) \leq |C|$. Given a circuit C and a number k, we can reject if $k > |C|$. Hence we can assume that $k \leq |C|$. Now the statement $K(C) \geq k$ can be translated into a quantified Boolean formula of polynomial size in $|C|$, namely

$$(\exists x_1 \in U)(\forall b_1 \in \{0,1\}) \ldots (\exists x_k \in U)(\forall b_k \in \{0,1\})(\exists D \in \mathcal{C})(\forall 1 \leq i \leq k)[x_i \in D \leftrightarrow b_i = 1],$$

where U is the universe of $\mathcal{C}$. Hence $K(C) \geq k$ can be decided in **PSPACE**. We are left with showing **PSPACE**-hardness, a task accomplished by the following lemma.

Lemma 3.3 *For every quantified boolean formula Ψ we can compute in polynomial time a circuit C^Ψ and a number $k(\Psi)$ such that Ψ is true if and only if $K(C^\Psi) \geq k(\Psi)$.*

Proof Let Ψ be a formula in prenex normal form with n free variables. We will show by induction on the number of quantifier alternations that there is a circuit C and a number k such that for all $t \in \{0,1\}^n$:

$$\begin{aligned} \Psi(t) &\Rightarrow K(C_t) = k \\ \neg\Psi(t) &\Rightarrow K(C_t) = k-1, \end{aligned}$$

where $\Psi(t)$ is the truth-value of Ψ after the truth-values of the free variables have been fixed according to t. The circuit C takes three kind of inputs: a parameter (written as a subscript) corresponding to a truth-assignment to the free variables in the formula, an index of the concept, and a point from the universe. We will write $C_t(i, x)$ for the value of the circuit evaluated with t as the parameter, i as the index, and x as the point. Thus C_t is a circuit computing a concept class, and we can write $K(C_t)$.

In the following construction we will need to make the universes of concept classes disjoint. Given $(C_t)_{t\in\{0,1\}^n}$ we do this by prefixing the index of the class to the points:

$$C'_t(i, t'x) := \begin{cases} C_t(i, x) & \text{if } t = t', \\ 0 & \text{else.} \end{cases}$$

This construction does not change the K-dimension, hence $K(C'_t) = K(C_t)$. Note that the size of C'_t is at most the size of C_t plus a constant multiple of n.

Suppose that Ψ is a quantifier-free formula on n variables. Define a circuit C with parameter $t \in \{0, 1\}^n$ and inputs $i, x \in \{0, 1\}$ as follows:

$$C_t(i, x) := \begin{cases} x & \text{if } i = 1 \text{ and } \Psi(t) \text{ is true,} \\ 0 & \text{else.} \end{cases}$$

If $\Psi(t)$ is true then $\mathcal{C}(C_t) = \{\emptyset, \{1\}\}$ and hence $K(C_t) = 1$. If $\Psi(t)$ is false then $\mathcal{C}(C_t) = \{\emptyset\}$ and $K(C_t) = 0$. Note that the circuit can be built in polynomial time from Ψ.

For the induction step there are two cases depending on whether Ψ starts with an existential or universal quantifier.

In the first case let $\Psi(t) = (\exists y)[\Phi(y, t)]$ where $y \in \{0, 1\}^m$ and $t \in \{0, 1\}^n$. By the inductive hypothesis there is a circuit $C_{y,t}(i, x)$ and a number k such that

$$\begin{aligned} \Phi(y, t) \quad &\Rightarrow \quad K(C_{y,t}) = k \\ \neg\Phi(y, t) \quad &\Rightarrow \quad K(C_{y,t}) = k - 1, \end{aligned}$$

for all $y \in \{0, 1\}^m$ and $t \in \{0, 1\}^n$. Using the construction above we can assume that the universes of the concept classes $\mathcal{C}(C_{y,t})$ are pairwise disjoint. If we can construct a circuit C'_t that computes the disjoint union of $(\mathcal{C}(C_{y,t}))_{y\in\{0,1\}^m}$ as in Lemma 3.1, then we are done, since then $\Psi(t)$ implies $K(C_t) = k + 1$, and $\neg\Phi(t)$ implies $K(C_t) = k$. We have to ensure that each C_y occurs twice, and we have to add at least one new point for each concept class and include it in each concept of that class.

Formally we define

$$C'_t(bi, yb'cx) := \begin{cases} \max\{C_{y,t}(i, x), c\} & \text{if } b = b', \\ 0 & \text{else.} \end{cases}$$

Here b and b' take care of the duplication and c adds at least one point to each concept in a particular concept class. Now we can apply Lemma 3.1 to obtain that $\Psi(t)$ is true implies $K(C'_t) = k + 1$ and $\Psi(t)$ is false implies $K(C'_t) = k$.

In the other case $\Psi = (\forall y)[\Phi(y,t)]$ where $y \in \{0,1\}$ and $t \in \{0,1\}^n$. As above there is a circuit $C_{y,t}(i,x)$ and a number k such that

$$\begin{aligned} \Phi(y,t) &\Rightarrow K(C_{y,t}) = k \\ \neg\Phi(y,t) &\Rightarrow K(C_{y,t}) = k-1, \end{aligned}$$

for all $y \in \{0,1\}$ and $t \in \{0,1\}^n$. Again we can assume that the concept classes are pairwise disjoint.

Hence to apply Lemma 3.2 all we have left to do is show how to perform the construction of $\mathcal{D}$ as described before the lemma. The definition of the circuit is straightforward, if a bit messy.

Define a circuit

$$C'_t(abj, xcd) := \begin{cases} C_{a,t}(j,x) & \text{if } c = 1 \text{ and } b = d, \\ 1 & \text{if } c = d = 0,\ x1 \text{ is a prefix of } ab, \text{ and } 0 \le |x| \le 1, \\ 0 & \text{else.} \end{cases}$$

This circuit computes the (disjoint) union of the concept classes $\mathcal{C}(C)_{y,t}$ and marks each one of them using points ending in 00.

Fix t. Suppose that $\Psi(t)$ is true, i.e. $K(C_{y,t}) = k$ for all y. Then we can conclude that $K(C'_t) = 2 + k$ by applying Lemma 3.2. Similarly if $\Psi(t)$ is false there is a y such that $K(C_{y,t}) = k-1$ and hence $K(C'_t) = 1 + k$ by Lemma 3.2 which is what we had to show.

We have to verify that the above construction can be performed in polynomial time. In the base case, the size of the circuit is at most $|\Psi| + a_1$ for some constant a_1. In the inductive step we use three different constructions: making the universes disjoint, disjoint union, and marked union. Each of these adds at most $a_2\, n \le a_2\, |\Psi|$ gates to the circuit, where n is the length of t (equalling the number of free variables), and a_2 is a constant. Since Ψ contains at most $|\Psi|$ quantifiers, the size of the final circuit is bounded above by $a\,|\Psi|^2 + a$ for some constant a. Hence the circuit is constructible in polynomial time. $\diamondsuit$

4 Comparison between the models

Remember, that the K-dimension of a concept class $\mathcal{C}$ is bounded by $\log|\mathcal{C}|$, since a decision tree can have at most as many leaves as there are concepts. Therefore the problem MATRIX K-DIMENSION lies in **NP**, since a concept class represented by a matrix has fewer concepts than there are entries in the matrix. It is probably not **NP**-complete, however, since it lies in **DSPACE**$(\log^2 n)$ (as can be easily verified by checking the formula given before Lemma 3.3).

Frances and Litman [1] show that MATRIX VC DIMENSION reduces to MATRIX K-DIMENSION in polynomial time. Since MATRIX VC DIMENSION is complete for **LOGNP** this imples that MATRIX K-DIMENSION is **LOGNP**-hard. At this point the precise complexity of MATRIX

K-DIMENSION is unknown, but it seems unlikely that MATRIX K-DIMENSION actually lies in **LOGNP**.

One approach to this problem might be through logical characterizations of the two problems, by determining classes of formulas to which they belong. **LOGNP** does that for MATRIX VC DIMENSION, but there does not seem to be a comparable result for MATRIX K-DIMENSION. In light of such a characterization Frances and Litman's reduction of MATRIX VC DIMENSION to MATRIX K-DIMENSION would probably establish a non-trivial relationship between two logical systems.

In the circuit model we know more about the relationship between the VC-dimension and the K-dimension. VC DIMENSION reduces to K-DIMENSION, since VC DIMENSION lies in $\mathbf{\Sigma_3^p}$—is complete for this level, as a matter of fact—and K-DIMENSION is **PSPACE**-complete as we have just shown. This also implies that it is unlikely that K-DIMENSION reduces to VC DIMENSION since this would collapse **PSPACE** to $\mathbf{\Sigma_3^p}$.

Acknowledgments I'd like to thank Lance Fortnow for pointing out the paper by Frances and Litman, and several helpful discussions, and Judy Goldsmith for several valuable suggestions.

References

[1] Moti Frances and Ami Litman. Optimal Mistake Bound Learning is Hard. *Information and Computation*, pages 66–82, 1998.

[2] Michael R. Garey and David S. Johnson. *Computers and Intractability*. Freeman, San Francisco, 1979.

[3] Nick Littlestone. Learning Quickly When Irrelevant Attributes Abound: A New Linear-threshold Algorithm. *Machine Learning*, 2:285–318, 1988.

[4] Christos H. Papadimitriou and Mihalis Yannakakis. On Limited Nondeterminism and the Complexity of the V-C Dimension. In *Proceedings of the 8th Annual Conference on Structure in Complexity Theory (SCTC '93)*, pages 12–18, San Diego, CA, USA, May 1993. IEEE Computer Society Press.

[5] Marcus Schäfer. Deciding the Vapnik-Cervonenkis dimension is Σ_3^P-complete. *Journal of Computer and System Sciences*, 58:177–182, 1999.