האוניברסיטה העברית בירושלים
THE HEBREW UNIVERSITY OF JERUSALEM

LEARNING THE DECISIONS OF SMALL COMMITTEES

by

YUVAL SALANT

Discussion Paper # 332 September 2003

מרכז לחקר הרציונליות

CENTER FOR THE STUDY OF RATIONALITY

Feldman Building, Givat-Ram, 91904 Jerusalem, Israel
PHONE: [972]-2-6584135 FAX: [972]-2-6513681
E-MAIL: ratio@math.huji.ac.il
URL: http://www.ratio.huji.ac.il/

Learning the Decisions of Small Committees*

Yuval Salant
The School of Computer Science and Engineering, and
The Center for the Study of Rationality
The Hebrew University, Jerusalem †

August 16, 2003

Abstract

A committee is a collection of members, where every member has a linear ordering on the alternatives of a finite ground set X. The committee chooses between pairs of alternatives drawn from X by a simple majority vote. The committee's choices induce a preference relation on X. In this paper, we study the possibility of learning preference relations of small committees from examples. We prove that it is impossible to *precisely* learn the preference relation of a committee before seeing all its choices, even if a teacher guides the learner through the learning process. We also prove that a learner can *approximately* learn the preference relation of a committee from a relatively few random examples.

*I am most grateful to Gil Kalai for his devoted guidance, encouragement, and many important comments. I thank Elchanan Ben-Porath, Ariel Rubinstein, and Ron Siegel for fruitful discussions and helpful remarks, and Michael Borns for editing the text.

†Email: yuvals@pob.huji.ac.il.

1 Introduction

Consider a committee of three members, where every member has a linear ordering on the N elements of a finite ground set X. The committee chooses between pairs of alternatives drawn from X by a simple majority vote: a is chosen over b, if a is ranked higher than b by at least two members. The choices of the committee induce a preference relation (i.e., an asymmetric binary relation) on the elements of X. Condorcet's famous "paradox" asserts that preference relations of three-member committees might violate the most basic property of rational choice; namely, they can be cyclic. Nevertheless, they constitute a very basic example of non-rational (i.e., intransitive) social choice, and it could be of interest to examine what properties of rational choice extend to the case of small committees.

In this paper, we explore this question in the framework of learning. Learning refers to the process of observing the choices of an agent for a while, and then formulating a hypothesis that will enable us to predict future choices of the agent. One of the most basic properties of learning is the number of examples a learner needs to see in order to formulate a good hypothesis, where an example is a pair of alternatives and the choice from the pair. Another basic property is the quality of the hypothesis, and we distinguish between two types of learning accordingly: *exact learning* in which the learner should predict future choices based on past examples without making mistakes (i.e., the learner's task is to fully discover the choices of the agent); and *approximate learning* in which the learner should predict future choices with high accuracy.

Learning the choices of a single decision-maker (DM), who has a linear ordering on the elements of X, is widely studied in the literature. Rubinstein [9] asks how many examples a teacher has to communicate to a student in order to fully describe a linear ordering to the student. He shows that $N-1$ examples are enough (where $N = |X|$), and that fewer examples do not suffice. Computer scientists are usually interested in the related *sorting* problem: How many questions does an algorithm (i.e., an independent learner) need to ask in order to precisely discover a linear ordering? There are sorting algorithms that need to ask at most $N \log N$ questions for every ordering, and there are various results that this is optimal (see [1, 6]). While the above results refer to exact learning of linear orderings, Kalai [4] investigates the problem of approximate learning, and proves that a learner needs to see $O(N)$ random examples in order to approximately discover the structure of a linear ordering.[1] This paper explores whether the relative

[1] We use the following notation throughout the paper. Let $f, g : \mathbb{N} \longrightarrow \mathbb{R}_+$. We write $f(n) = O(g(n))$ if there is a constant $c_1 > 0$ such that $f(n) \leq c_1 g(n)$ for every n. We write $f(n) = \Omega(g(n))$ if there is a constant $c_2 > 0$ such that $f(n) \geq c_2 g(n)$ for every n.

ease of learning rational choice extends to simple forms of social choice. Particularly, we examine this issue in the context of three-member committees, and extend the results to larger committees, as well as to decisive majority decisions of large societies.

The first notion we examine is *exact learning*. For that purpose, we use the *describability* measure introduced by Rubinstein [9]. Given a family of binary relations, describability is defined as the minimal number of examples a teacher needs to communicate to a student in order to precisely describe any binary relation in the family to the student.

Theorem 1. The *describability* of the family of binary relations induced by the choices of three-member committees is $\binom{N}{2}$. That is, in the worst case a teacher cannot describe the decisions of a committee to a student without telling the student all its choices.

An interesting algorithmic implication of Theorem 1 concerns the sorting problem mentioned earlier. In the context of committees, the sorting problem can be formulated as follows: How many questions does an independent learner need to ask in order to learn a preference relation of a three-member committee?

Theorem 2. Any independent learner who seeks to discover the content of a binary relation induced by the choices of a three-member committee needs to ask $\binom{N}{2}$ questions in the worst case.

Thus, exact learning of the preferences of a three-member committee is already very difficult to carry out; i.e., one cannot guarantee to precisely learn the choices of a three-member committee without knowing all its choices. Note that learning is much easier to carry out, if the learner can query individual members of the committee. In that case, the number of examples needed for learning is $3(N-1)$ in the teacher-student scenario or $3N \log N$ in the independent learner scenario.

The second notion we investigate replaces *exact* learning with *approximate* learning. For that purpose, we use the basic statistical model of *PAC-Learning* (PAC stands for Probably Approximately Correct; see [5, 11]). This model assumes that examples of the committee's choices are revealed to the learner randomly and independently according to some fixed probability distribution. After seeing the sample set, the learner has to formulate a hypothesis that will enable him to predict future choices of the committee with high probability (with respect to the same distribution). Note the two differences from the describability notion. On the one hand, in the PAC-learning model the examples are drawn *at random* instead of *optimally*, but on the other hand, the learner has to predict *most* of the future choices instead of *all* of them (see Kalai [4] for an interesting discussion about PAC-learnability and describability). We are interested in the number of examples needed to formulate such a hypothesis in the case of three-member committees.

Theorem 3. Preference relations of three-member committees are learnable with high probability in the PAC model from a number of examples, which is linear in the number of alternatives N.

Theorems 1 and 3 imply the main result of the paper. While approximate learning of a preference relation of a small committee from a relatively "few" examples is possible, it is impossible (in the worst case) to concisely describe or exactly learn the relation.

Throughout the paper we generalize this result to committees of r members, where r is an arbitrary odd integer. As any three-member committee can be viewed as a committee of r members, the theorem about exact learning continues to hold: It is impossible (in the worst case) to concisely describe or learn the preference relation of an r-member committee. As for approximate learning, we have the following theorem.

Theorem 4. Preference relations of r-member committees are learnable with high probability in the PAC model from $f(r, N) \cdot N$ examples, where $f(r, N) = O(\min\{r^2 \log r, r \log N\})$.

While the above results are phrased in terms of committees, they can easily be applied to individual choice. Indeed, instead of an r-member committee, we can think of a single decision-maker with r considerations according to which she ranks the alternatives. The decision-maker chooses a over b, if a is ranked higher than b in more than half of the considerations. Thus, the reader might want to change the phrase "r-member committee" with the phrase "single decision-maker with r considerations" in the above theorems.

The paper also explores the learnability of the choices of a decisive society. An α-*decisive society* is a society of m members, where every member has a linear ordering on the alternatives of X and one vote. Every choice of the society is α-decisive; i.e., at least $(\frac{1}{2} + \alpha)m$ of the members agree with every decision, where $0 < \alpha < 1/2$ is a small real number. Note that the choices of an α-decisive society are stable in the following sense: If less than αm members of the society change their minds about some choice, it has no influence on the preferences of the society.

Theorem 5. The choices of α-decisive societies are learnable with high probability in the PAC model from at most $f(\alpha) \cdot N$ examples, where $f(\alpha) = O((\frac{1}{\alpha})^2 \log \frac{1}{\alpha})$.

There are various results (e.g., see McGarvey [7]) that any asymmetric binary relation can be induced by a majority vote of a "large" society. Consequently, PAC-learning of the family of binary relations induced by large societies is difficult. However, according to Theorem 5, if we know that a society is decisive, it becomes much easier to PAC-learn its choices regardless of the number of members in the society.

2 Preliminaries

The model of choice discussed in this paper is a simple one. Let $X = \{x_1, x_2, \ldots, x_N\}$ be a finite set of N alternatives. Let $Y = \{(x_i, x_j) : i < j\}$ be the collection of all ordered pairs of elements in X. We confine ourselves to choice functions, which are defined on pairs of alternatives.

Definition 2.1 *A choice function is a mapping $c : Y \longrightarrow X$, which assigns to every choice problem $(y_1, y_2) \in Y$ an element $c(y_1, y_2) \in (y_1, y_2)$. In other words, a choice function is a complete asymmetric binary relation (i.e., a preference relation) on the elements of X.*

Often, it is convenient to think of choice functions as tournaments. Generally, a *tournament* is a directed graph in which every two vertices are connected by an edge (see Moon [8] for a general discussion of tournaments). In our case, a choice function c can be represented by a tournament, whose vertices are the elements of X, and whose edges respond to the choices of c; i.e., there is a directed edge from a to b, if and only if $c(a, b) = a$. Otherwise, there is a directed edge from b to a.

We denote the set of all choice functions on X by $\mathcal{C}$. A *class* of choice functions is a subset of $\mathcal{C}$. We are interested in *learning* a specific choice function in a given class of choice functions. Knowledge about a given choice function is acquired via examples. An *example* is a vector $((a,\ b), c(a, b))$, where $(a,\ b) \in Y$. As producing examples, processing them, and remembering them is costly, we wish to explore how many examples are needed in order to learn a choice function in different models of learning.

Throughout this paper, we mainly explore the learnability of classes of functions induced by majority vote of committees. A committee is a collection of $2r + 1$ members, where r is a positive integer (the number of members is odd to avoid ties). Every member has a strict linear ordering on the alternatives of X (no indifference is allowed). When the committee has to choose between two alternatives, $a, b \in X$, it chooses a over b if at least $r + 1$ members prefer a over b. Otherwise, b is chosen. We denote the class of all choice functions realized by committees of $2r + 1$ members by *(2r+1)Maj*.

3 Exact learning of committees' choices

Consider a teacher, called Alice, who wants to communicate her knowledge about a choice function to a student, called Bob. Both Alice and Bob know the nature (the class) of the choice function, but Bob does not know the real structure of the function. For example, Bob knows that Alice decides according to several considerations, but he does not know her actual choices.

Knowledge is communicated via examples. Alice selects examples and communicates them to Bob. Bob has to deduce the real structure of the choice function from the examples supplied by Alice. As Bob needs to learn the whole choice function, Alice should supply him with enough examples. As generating examples as well as communicating them and deducing from them is costly, both Alice and Bob would want the number of examples to be as small as possible. The minimal number of examples needed to deduce the entire choice function is called the describability of the choice function.

3.1 Definitions

The notion of describability was introduced by Rubinstein [9], who seeks "to explain the fact that certain properties of binary relations are frequently observed in natural language." One of the features he investigates is the describability of a relation, i.e., the ease with which the relation can be described by means of examples. We find this notion appealing for two reasons. Firstly, describability is an intuitive measure of "supervised-exact" learning, in which an instructor guides a student through the learning process in order to obtain optimal results. Secondly, describability serves as a lower bound for "independent-exact" learning, in which a learner seeks to independently discover the structure of the function.

In our setting, describability is defined as follows. Let C be a class of tournaments. The describability of C is the minimal integer k such that every tournament in C is uniquely determined by k examples or less. Formally, for every $c \in C$, denote by $d_C(c)$ the minimal integer m such that there exist m pairs, $(a_1, b_1), (a_2, b_2), \ldots, (a_m, b_m) \in Y$, which obey the following: if $c' \in C$ and $c'(a_i, b_i) = c(a_i, b_i)$, for all $i = 1, 2, \ldots, m$, then $c' = c$. Then,

Definition 3.1 *The describability of the class of tournaments C is*

$$desc(C) = \max_{c \,\in\, C} \{ \ d_C(c) \ \}.$$

The definition implies that for every two classes of choice functions, C_1 and C_2, if $C_1 \subseteq C_2$, then $desc(C_1) \leq desc(C_2)$. Moreover, the describability of the class of all tournaments on N alternatives is $\binom{N}{2}$, because if even one example is missing we can always find two tournaments that agree on all the examples given, but disagree on the missing example. These two observations imply that for any class of tournaments C, $desc(C) \leq \binom{N}{2}$.

Rubinstein [9] shows that the describability of the class of linear orderings is $N-1$. Indeed, any linear ordering of the form $x_1 \succ x_2 \succ x_3 \succ \ldots \succ x_{N-1} \succ x_N$ can be described by the examples $x_1 \succ x_2,\ x_2 \succ x_3,\ \ldots\ ,\ x_{N-1} \succ x_N$. On the other hand, a linear ordering cannot be described by less than $N-1$ examples, because we cannot deduce the order between two elements that are not chosen. Rubinstein conjectures that apart from a few small examples every symmetric class of tournaments (i.e., a class which is closed under relabelling of the alternatives) requires at least $N-1$ examples to be described. He suggests that the ease of description of linear orderings gives them an evolutionary advantage over other structures in natural language. In the context of decision-making, this result could be interpreted as giving an evolutionary advantage to transitive preferences.

3.2 Committees of three members (or more)

We would like to explore how difficult it is to describe decisions of committees. We start with the class of three-member committees, denoted by *3Maj*. Note that *3Maj* contains a relatively small number of tournaments (at most $(N!)^3$) in comparison to the total number of tournaments on the elements of X, which is $2^{\binom{N}{2}}$. One may expect the describability of *3Maj* to be approximately similar to the describability of the class of linear orderings. However,

Proposition 3.2 *The describability of 3Maj is* $\binom{N}{2}$.

Proof. Consider the class $C = C_1 \cup C_2$, where C_1 is the class of all tournaments realized by linear orderings, and C_2 is the class of all tournaments that deviate from some linear ordering in exactly one pair.

The describability of C is $\binom{N}{2}$. Indeed, any tournament $c_1 \in C_1$ cannot be described by less than $\binom{N}{2}$ examples, because for every set of $\binom{N}{2}-1$ examples used to describe c_1, there is a tournament $c_2 \in C_2$ that agrees with this set of examples and still disagrees with c_1 on the missing example.

Moreover, $C \subset 3Maj$. Indeed, $C_1 \subset 3Maj$, because we can replicate any linear ordering three times and receive a tournament in $3Maj$. $C_2 \subset 3Maj$, because we can generate any tournament with one deviation from a linear ordering as a majority vote of three linear orderings. Without loss of generality, we illustrate this for a tournament which is consistent with the linear ordering

$$1 \succ \ldots \succ i \succ \ldots \succ j \succ \ldots \succ n$$

except for one deviation $j \succ i$. This tournament can be obtained as a majority vote of the following three linear orderings:

$$1 \succ \ldots \succ i \succ \ldots \succ j \succ \ldots \succ n$$

$$j \succ i \succ 1 \succ \ldots \succ n$$

$$1 \succ \ldots \succ n \succ j \succ i \ .$$

Consequently, we get that

$$\binom{N}{2} = desc(C) \leq desc(3Maj) \leq \binom{N}{2};$$

that is, $desc(3Maj) = \binom{N}{2}$. □

There is an easy generalization of Proposition 3.2 to the case of larger committees.

Conclusion 3.3 *Let r be a positive integer. Then,*

$$desc((2r+1)Maj) = \binom{N}{2} \ .$$

Proof. As every function in *3Maj* is based on 3 linear orderings, we can replicate the first and the second orderings r times, leave the third ordering as it is, and receive the exact same function as a function in *(2r+1)Maj*. This implies that $3Maj \subseteq (2r+1)Maj$, which results in the conclusion. □

Thus, there is a significant gap between describing the preference relation of a single decision-maker who uses a linear ordering on the alternatives, and the preference relation of a three-member committee, where every member has a linear ordering on the alternatives. While the preferences of a single DM can be described by $N-1$ examples, describing the preferences of a three-member committee requires $\binom{N}{2}$ examples in the worst case.

3.3 Two algorithmic implications

The traditional comparison-based *sorting* problem in computer science can be formulated as follows: Given a set X of N objects, how many questions does an algorithm (i.e., an independent learner) need to ask an oracle in order to discover a linear ordering on the objects of X? (A question is a pair of objects submitted to the oracle, and the oracle's answer is the higher object according to the linear ordering.) Note the difference from the describability notion; Sorting refers to *independent* exact learning while describability refers to *supervised* exact learning.

It can be shown (see [1, 6]) that any sorting algorithm needs to ask $\Omega(N \log N)$ questions in the worst case, and $\Omega(N \log N)$ questions in the average case (assuming a uniform probability distribution on all the orderings). There are sorting algorithms, like *merge-sort*, that can discover any given ordering by asking $N \log N$ questions, and are therefore asymptotically optimal.

One might consider the following modified sorting problem: How many questions does an algorithm need to ask in order to discover a tournament induced by a majority vote of three linear orderings? Proposition 3.2 suggests that any such algorithm needs to ask $\binom{N}{2}$ questions in the worst case. Consequently, we draw the following corollary.

Corollary 3.4 *Any independent learner who seeks to discover the content of a binary relation induced by a three-member committee needs to ask $\binom{N}{2}$ questions in the worst case.*

Knuth [6] devotes the first part of his book to the sorting problem, and the second part to the complementary problem of *searching*. A somewhat modified version of the searching problem in the context of linear orderings asks: Suppose that the N objects of the set X are already sorted (i.e., linearly ordered). Given a new object y, how many questions does an algorithm have to ask in order to find the location of y within the ordering, assuming y has no influence on the relations between the elements of X? An algorithm called *binary search* adds y to the linear ordering after asking $O(\log N)$ questions (see [1, 6]).

In the context of three-member committees, the searching problem can be stated as follows: Suppose that a preference relation induced by a three-member committee on the alternatives of X is already known. Given a new alternative y, how many questions does an algorithm have to ask in order to discover the relation between y and the alternatives of X?

Corollary 3.5 *Any searching algorithm, which seeks to add a new element y to a tournament induced by a three-member committee on the elements of X, must ask N questions in the worst case.*

To see why this is so, assume that the committee has a linear ordering on the elements of X, and that y is located somewhere within the ordering except for possibly one mistake (as in the proof of Proposition 3.2). In that case, any algorithm has to ask N questions in order to discover whether the mistake exists and where. The exact construction of the three orderings is similar to that in Proposition 3.2, and is left to the reader.

3.4 Economic interpretations

The results about exact learning suggest that in learning "aggregated choice" (i.e., choices of committees) there is a large gap between a situation where learning is based only on observing the committee choices and a situation where learning is based also on observing the choices of the committee members. Namely, if we (as learners) can query individual members of the committee about their preferences, then it is possible to learn the choices of the committee from $3(N-1)$ examples or $3N \log N$ examples, depending on the model. However, if we have access only to the choices of the committee, then in the worst case we cannot learn the preference relation of a committee before seeing all its choices.

Corollary 3.5 suggests another observation along the same lines. Suppose that the committee has already formulated its preference relation on X. A new alternative y is now available to the committee members. Then, every member will have to ask himself $O(\log N)$ questions in order to add the new element to his linear ordering. However, the committee will have to discuss N questions in the worst case in order to formulate the relation between y and the elements of X. Thus, in the worst case there are no shortcuts, and the committee will have to discuss the relation between the new element and all the other elements.

Proposition 3.2 and Corollary 3.4 imply another nice result. Suppose that an observer does not care about learning the committee choices. He only wishes to examine whether the choices of the committee are transitive or not. As can be inferred from the proof of Proposition 3.2, he has no way of doing so without seeing all the choices of the committee.

As we briefly mentioned earlier, our results apply to individual choice as well. Indeed, let us consider the "single decision-maker with several criteria" interpretation instead of a committee with several members. In this interpretation, Proposition 3.2 and Corollary 3.4 suggest that it is much easier to exactly learn the decision-maker's choices if we can identify the different criteria according to which the DM decides, and query about them, as opposed to a situation where we can observe only the choices of the DM.

4 Approximate learning of committees' choices

Consider a scenario where a sample set of the choices of a three-member committee is revealed to us randomly according to some probability distribution on all the choice pairs (which is known or unknown to us). We would like to approximately learn the choices of the committee. That is, given the sample set, we would like to predict future decisions of the committee with high accuracy. Note the difference from the describability measure discussed in the previous section, where the sample set is chosen optimally (and not according to some probability distribution), and where we demand exact prediction of future choices (and not prediction with high accuracy).

In order to analyze approximate learning, we use the basic model of PAC-Learning. The definitions we use are mostly based on [5]. More general definitions can be found in [11].

4.1 Definitions

Let C be a family of boolean functions from an instance space I to $\{0,1\}$. We assume that C is known, and we want to learn a specific target function $c \in C$. (Note that choice functions are boolean functions, if we interpret $c(a,b) = 0$ as implying that a is chosen over b.) Let $\mathcal{D}$ be a fixed probability distribution over I. If $h \in C$ is any function, then the distribution $\mathcal{D}$ provides a natural measure of error between h and c. Namely, we define

$$error(h) = Pr[x \in I : c(x) \neq h(x)].$$

(In the case of choice functions $error(h) = \sum_{x \in Y \text{ s.t. } c(x) \neq h(x)} Pr(x)$.)

Let EX(c, $\mathcal{D}$) be a procedure that on each call returns an example $(y, c(y))$, where $y \in I$ is drawn randomly and independently according to $\mathcal{D}$.

Definition 4.1 *We say that a class of boolean functions C is PAC-learnable if there exists an algorithm L with the following property: For every function $c \in C$, for every distribution $\mathcal{D}$ on I, and for all $0 < \epsilon, \delta < 1/2$, if L is given access to EX(c, $\mathcal{D}$) and inputs ϵ and δ, then with probability at least $1-\delta$, L will output a function $h \in C$ satisfying $error(h) \leq \epsilon$.*

The hypothesis $h \in C$ is thus "approximately correct" (i.e., $error(h) \leq \epsilon$) with high probability (the confidence level is at least $1 - \delta$), hence the name Probably Approximately Correct learning.

A fundamental aspect of PAC-learnability is the number of examples needed to learn a class of functions C. In order to evaluate this number, we use the combinatorial measure of Vapnik-Chervonenkis Dimension, a measure that assigns to every class C a single number that characterizes the sample size needed to learn C.

Let $S = \{s_1, s_2, \ldots, s_m\} \subseteq I$, and denote by

$$\Pi_C(S) = \{(c(s_1), c(s_2), \ldots, c(s_m)) : c \in C\} \subseteq \{0, 1\}^m$$

the set of all behaviors on S that are realized by C. If $\Pi_C(S) = \{0, 1\}^m$, we say that S is *shattered* by C. Thus, S is shattered by C, if C realizes all the possible behaviors of S.

Definition 4.2 *The Vapnik-Chervonenkis Dimension of C, denoted as $VCD(C)$, is the cardinality d of the largest set $S = \{s_1, s_2, \ldots, s_d\}$ shattered by C. If arbitrarily large finite sets can be shattered by C, then $VCD(C) = \infty$.*

The definition implies three important things. Firstly, it follows from the definition that $s_1, s_2, \ldots, s_d$ must be distinct. Secondly, in order to prove that *VCD(C)* is at least d, we have to find *some* shattered set of size d. Thirdly, in order to show that *VCD(C)* is at most d, we have to show that no set of size $d + 1$ is shattered. For example, consider I to be the real line and C to be the class of indicator functions of closed intervals of I; i.e., $c_{[a,b]} \in C$ assigns the value 1 to points inside the interval $[a, b]$. Then, any set of two points can be shattered by C, yet no set of three points (x_1, x_2, x_3) where $x_1 < x_2 < x_3$ can be shattered, because the configuration $\{1, 0, 1\}$ cannot be realized. Therefore, $VCD(C) = 2$. The following theorem suggests that we can use the *VCD* as our principal measure of learnability.

Theorem 4.3 *For fixed values of ϵ and δ, the number of examples needed to PAC-learn a class of boolean functions is bounded above and below by a linear function of the VC-dimension.*

The proof of the upper bound implies a stronger result. It shows that any algorithm that takes as input a set of $m(\cdot)$ examples (where $m(\cdot)$ is a linear function of the *VCD*), and produces a hypothesis that is consistent with the set is a PAC-learning algorithm.

For further details about the connection between the *VCD* and PAC-learning, and the dependence of the number of examples on ϵ and δ, see [5], Chapter 3.

Thus, in order to evaluate how many examples are needed to learn a class of functions in the PAC model, it is enough to investigate the *VCD* of the class. An easy observation is that if the *VCD* of a class C is d, then C must contain at least 2^d functions (otherwise, it will be impossible to shatter a set of size d). Therefore, we obtain the next proposition.

Proposition 4.4 *Let C be a family of functions. Then, $VCD(C) \leq \log_2 |C|$.*

The following theorem, which was proved independently by Sauer [10], Shelah, and Perles, gives a more general connection between the *VCD* of a class C, and the number of functions in C.

Theorem 4.5 *Let C be a family of boolean functions from an instance space of n elements to $\{0,\ 1\}$ (alternatively, we can think of C as a collection of vectors in $\{0,\ 1\}^n$). If $VCD(C) \leq d$, then the number of functions in C is at most $g_d(n) = \sum_{i=0}^{d} \binom{n}{i}$.*

Corollary 4.6 *If the number of functions in C is at least $g_d(n) + 1$, then $VCD(C) \geq d + 1$.*

There is an interesting connection between the VC-dimension and the PAC-learnability of a class of functions C, and between the *testability* of the class. Indeed, fundamental results from statistical learning theory imply that the size of the sample set needed to examine whether an arbitrary function h belongs to C or not, and to evaluate how "far" h is from C, is at most proportional to the VC-dimension of the class. Kalai [4] discusses these results and their implications in the context of individual choice.

4.2 The PAC-learnability of *3Maj*

We return to the class of three-member committees, and show that for fixed values of ϵ and δ this class can be PAC-learned from $O(N)$ examples. To do so, we will use the result of Theorem 4.3 to prove that *VCD(3Maj)*, and therefore the number of examples needed to learn this class, are linear in N. Note that by using the result of Proposition 4.4 along with the fact that $|3Maj| < (N!)^3$, we get that $VCD(3Maj) < 3N \log N$. We will asymptotically improve this upper bound in Proposition 4.9. We start by proving a lower bound on the VC-dimension.

Proposition 4.7 *The VC-dimension of 3Maj is at least* $3(N-2)$.

Proof. Let $X = \{x_1, x_2, \ldots, x_N\}$. In order to prove that $VCD(3Maj) \geq 3(N-2)$, we have to introduce a set of $3(N-2)$ choice pairs that can be shattered by functions in *3Maj*. First, we introduce a set of $3(N-3)$ pairs that can be shattered, and then we add three other pairs.

The set of $3(N-3)$ pairs is separated into three types:

$$\begin{array}{lll} T_1: & \forall\, 4 \leq i \leq N & (x_1, x_i) \\ T_2: & \forall\, 4 \leq i \leq N & (x_2, x_i) \\ T_3: & \forall\, 4 \leq i \leq N & (x_3, x_i) \end{array}$$

Given a configuration of choices from these pairs (i.e., a vector in $\{0,1\}^{3(N-3)}$), we construct a function c in *3Maj* that realizes this configuration by introducing three orderings O_1, O_2, O_3, where O_i "takes care" of pairs of type T_i. Indeed, in O_i we place x_i above all the elements x_j, $4 \leq j \leq N$, such that $c(x_i, x_j) = 0$, and below all the elements x_j such that $c(x_i, x_j) = 1$. In the other two orderings, we place x_i once below all the other elements and once above all the other elements. Therefore, O_i determines the realization of the T_i examples, and this realization is consistent with the given configuration.

We now add three additional pairs $(x_1, x_2), (x_1, x_3), (x_2, x_3)$. Note that the construction of the orderings O_i still leaves one "degree of freedom" that allows us to realize all the configurations of the additional pairs. We distinguish between two cases.

Case 1: $c(x_1, x_3) = 0$. Then, the orderings are

$$\begin{array}{ll} O_1: & \ldots \succ x_1 \succ \ldots \succ x_2, x_3 \\ O_2: & x_3 \succ \ldots \succ x_2 \succ \ldots \succ x_1 \\ O_3: & x_1, x_2 \succ \ldots \succ x_3 \succ \ldots \end{array}$$

Changing the order between x_2 and x_3 in O_1 and between x_1 and x_2 in O_3 allows us to realize all the configurations in which $c(x_1, x_3) = 0$.

Case 2: $c(x_1, x_3) = 1$. Then, the orderings are

$$\begin{array}{ll} O_1: & x_2, x_3 \succ \ldots \succ x_1 \succ \ldots \\ O_2: & x_1 \succ \ldots \succ x_2 \succ \ldots \succ x_3 \\ O_3: & \ldots \succ x_3 \succ \ldots \succ x_1, x_2 \end{array}$$

Changing the order between x_2 and x_3 in O_1 and between x_1 and x_2 in O_3 allows us to realize all the configurations in which $c(x_1, x_3) = 1$.

This gives us a set of $3(N-3)+3 = 3(N-2)$ pairs, which can be shattered by functions in *3Maj*, as required. □

We now turn to prove an upper bound on $VCD(3Maj)$. We will use the following proposition in the course of the proof.

Proposition 4.8 *(Kalai [4]) The VC-dimension of the class of order relations (i.e., linear orderings) is* $N-1$.

Note that Proposition 4.8 and Theorem 4.3 imply that for fixed values of ϵ and δ, the number of examples needed to PAC-learn the class of order relations is linear in the number of alternatives N.

Proposition 4.9 *The VC-dimension of 3Maj is less than* $99N$.

Proof. The idea of the proof is as follows. Suppose that the VC-dimension is M. Then, there are M pairs of alternatives that are shattered by tournaments in *3Maj* (i.e., *3Maj* realizes all the possible choices from these pairs). Given a configuration of choices from the M pairs and the three linear orderings that realize it, there must be one ordering that "agrees" with at least $\frac{2M}{3}$ choices of the configuration. In that case, we say that the ordering "covers" the configuration. In order to "cover" all the configurations by linear orderings, we need at least

$$U = \frac{\text{number of configurations}}{\text{max number of configurations covered by one ordering}}$$

different orderings. On the other hand, the number of orderings should not exceed $\binom{M}{N}$. Otherwise, according to Corollary 4.6, there are N pairs among the M pairs that are shattered by the linear orderings, in contradiction to Proposition 4.8. Therefore, we receive the inequality $U \leq \binom{M}{N}$, which implies that $M < 99N$.

Let us go into details. Denote the VC-dimension by M. Then, there are M choice pairs, $A_1, A_2, \ldots, A_m \in Y$, such that every configuration of choices from these pairs is realized by a tournament in *3Maj*. Thus, given a configuration of choices from the M pairs (i.e., a vector in $\{0,\ 1\}^M$), we can find 3 linear orderings, such that for every coordinate (or choice) of the configuration at least 2 of the 3 orderings "agree" with it. Consequently, there is one ordering (or more) that agrees with at least $\frac{2M}{3}$ coordinates of the configuration. In that case, we say that the ordering "covers" the configuration. What is the minimal number of different orderings needed in order to cover all the possible configurations of the M pairs? A single ordering can agree with $\binom{M}{i}$ configurations on $M-i$ coordinates (we take the configuration induced by the ordering, and we have

$\binom{M}{i}$ options to choose the i coordinates that disagree with the ordering). Consequently, a single ordering can cover at most $\binom{M}{0}+\binom{M}{1}+\cdots+\binom{M}{M-\frac{2M}{3}}$ configurations. Therefore, as the total number of configurations is 2^M, the number of different orderings needed is at least

$$U = \frac{2^M}{\binom{M}{0}+\binom{M}{1}+\cdots+\binom{M}{M-\frac{2M}{3}}} \geq 2^{M(1-H(\frac{1}{3}))},$$

where $H(\lambda) = -\lambda \log \lambda - (1-\lambda)\log(1-\lambda)$, $0<\lambda<1$, is the binary entropy function, and the inequality is derived from Conclusion A.3 in Appendix A.

Let us think of these U orderings as U different vectors in $\{0,\ 1\}^M$, where we identify each ordering with the configuration it realizes. According to Corollary 4.6, if the number of vectors exceeds $g_{N-1}(M) = \sum_{i=0}^{N-1}\binom{M}{i}$, the VC-dimension of these linear orderings (and, consequently, the VC-dimension of all linear orderings) is at least N. This is impossible due to Proposition 4.8. Therefore, we get that

$$2^{M(1-H(\frac{1}{3}))} \leq U \leq \sum_{i=0}^{N-1}\binom{M}{i} \leq 2^{MH(\frac{N}{M})},$$

where the right inequality is derived from Conclusion A.3. Taking $\log_2$ of both sides and dividing by M, we get that

$$1 - H(\frac{1}{3}) \leq H(\frac{N}{M}),$$

which implies that M must obey $M < 99N$. □

Remark. The proof of Proposition 4.9 can be applied to societies of potentially many members—every member with a linear ordering on the alternatives and one vote—in which every choice is supported by two thirds or more of the votes.

Propositions 4.7 and 4.9 suggest that *VCD(3Maj)* is linear in N. Consequently, we get the following theorem.

Theorem 4.10 *For fixed values of ϵ and δ, the number of examples needed to learn the class of three-member committees in the PAC model is linear in the number of alternatives N.*

5 PAC-learnability of large committees and decisive societies

The results of the previous section can be generalized to larger committees. Consider the class *(2r+1)Maj* for an arbitrary integer r. The number of functions in *(2r+1)Maj* depends on both the number of alternatives N and the number of members $2r+1$. If $2r+1$ is very large, then any tournament on N alternatives can be realized by a committee of $2r+1$ members, and consequently $VCD((2r+1)Maj) = \binom{N}{2}$. In fact, Erdős and Moser [2] show that every tournament on N alternatives can be realized by a majority vote of $O(\frac{N}{\log N})$ orderings. Therefore, we limit ourselves to committees of at most $2r+1 << \frac{N}{\log N}$ members.

According to Proposition 4.4, $VCD((2r+1)Maj) < (2r+1)N \log N$, because the number of functions in *(2r+1)Maj* is less than $(N!)^{2r+1}$. The same line of argument used in the proofs of Propositions 4.7 and 4.9 can be used to obtain the following result (see Appendix B for a detailed proof).

Theorem 5.1 *Let r be a positive integer. Then,*

1. *The VC-dimension of (2r+1)Maj is at least $N \cdot (2r+1) - O(r^2)$.*
2. *The VC-dimension of (2r+1)Maj is at most $N \cdot f(2r+1, N)$, where $f(2r+1, N) = \min\{10(2r+1)^2 \log(2r+1), (2r+1)\log N\}$.*

Consequently, for fixed values of ϵ and δ, the number of examples needed to PAC-learn (2r+1)Maj is linear in the above numbers.

Another interesting class of choice functions is the class of *α-decisive societies*. An *α-decisive society* is a society of m members, where m is potentially very large. Every member of the society has a linear ordering on the N alternatives and one vote. Every choice of the society is decisive; i.e., at least $(\frac{1}{2} + \alpha)m$ of the members agree with every decision, where $0 < \alpha < 1/2$ is a small real number. In other words, the choices of an α-decisive society are not sensitive to a small fraction of people changing their minds. It is easy to verify that any committee of $2r+1$ members is a decisive society for $\alpha = \frac{1}{4r+2}$. In the other direction (which is more difficult), if we randomly sample a committee of $\frac{\ln N}{\alpha^2}$ members from the society, then with probability $> \frac{1}{2}$, the choices of the committee will coincide with the choices of the society. While the result of Erdős and Moser suggests that large committees induce any asymmetric binary relation, and therefore PAC-learning their decisions is difficult, it turns out that if we know that a society is decisive, it becomes much easier to PAC-learn its choices regardless of the number of members in the society.

Proposition 5.2 *For fixed values of ϵ and δ, the class of choice functions induced by α-decisive societies can be PAC-learned from at most $f(\alpha) \cdot N$ examples, where $f(\alpha) = O((\frac{1}{\alpha})^2 \log \frac{1}{\alpha})$.*

The proof of Proposition 5.2 is similar to that of Proposition B.2 in Appendix B, and is left to the reader.

6 Concluding remarks

This paper has explored whether it is possible to learn preference relations of small committees from examples. The first part of the paper discussed exact learning. We showed that in the worst case $\binom{N}{2}$ examples are needed in order to describe a preference relation of a three-member committee. Consequently, in the worst case we cannot avoid making all the $\binom{N}{2}$ queries in the analogue algorithmic question of "sorting." It is still an open problem whether a smaller number of examples (or queries) suffices in the average case, when we assume that the linear orderings of the members are uniformly and independently distributed.

One might raise the question of whether the teacher-student scenario (or the independent learner scenario) is realistic, and suggest to consider other models as well. Firstly, we should keep in mind that the results of the paper apply not only to committees but also to individual choice. In the case of individual choice, these scenarios seem plausible. Secondly, the describability notion is appealing because it serves as a lower bound to the number of examples needed for exact learning in any model, where learning is based only on examples. For example, suppose that the learner "observes" the behavior of the committee; i.e., he sees various examples of the committee choices. Then, the results about describability suggest that in the worst case there will be decisions of the committee that he will not be able to predict (unless the examples cover all the choice pairs).

The second part of the paper discussed approximate learning. We proved that approximate learning of preference relations of three-member committees is easier than exact learning, and that a learner can approximately learn the choices of a three-member committee after seeing a number of examples, which is linear in the number of alternatives. Combined with the results about exact learning, we get a nice observation about committees: We can approximately predict the choices of a committee after observing it for a short while. However, we may still find ourselves surprised by a decision, even if we have already seen many decisions of the committee.

Our results in the PAC model are asymptotic in nature. Namely, we regard situations where the number of alternatives N is large. This follows the basic paradigm of theoretical computer science, which draws the main insights on the behavior of algorithms from their asymptotic behavior. For example, we proved that $VCD(3Maj) < 3N \log N$

and that $VCD(3Maj) < 99N$. Of course, it might be the case that the constant 99 in the second inequality can be significantly improved, but as it stands the first inequality is stronger when $N < 2^{33}$, i.e., for all practical purposes. Nevertheless, the second inequality provides an insight that cannot be deduced from the first inequality: The number of examples needed for PAC-learning the class of three-member committees is asymptotically similar to the number of examples needed for PAC-learning the class of order relations; i.e., both classes can be learned from $O(N)$ examples.

The results about PAC-learning leave open a complementary algorithmic question: Given a sample set of the choices of a three-member committee, what can be deduced about the other choices of the committee, and how? We argued that there exists an algorithm that can deduce most of the committee choices after seeing only a "few" of them. However, we did not present such an efficient algorithm; i.e., an algorithm that produces a "good" hypothesis after a number of steps which is polynomial in the number of alternatives N.

Preference relations of three-member committees are a basic extension of order relations. It would be an interesting challenge to examine which properties of order relations extend to this case, and which do not. Fishburn [3] contains some results in this direction. He shows that every tournament on 6 or less alternatives can be realized by a three-member committee, and he exhibits a tournament on 8 alternatives that cannot. He also shows that there is a tournament in *3Maj* that maximizes the number of cyclic triangles attainable by a tournament on N alternatives. We conclude with a thought in the same direction. We know that there are efficient ways to "test" whether a given binary relation is an order relation. Is there an efficient way to "test" whether a binary relation is induced by a committee of three members? We also know that a binary relation is an order relation, if it is an order relation when it is restricted to every subset of three alternatives. Is there a similar characterization (with three replaced by a larger constant number) for preference relations of three-member committees? A positive answer to the latter question will provide a positive answer to the "testing" question.

Appendix

A Combinatorial approximations

The main combinatorial result we will use throughout this section is Stirling's approximation:

$$\sqrt{2\pi n}(\frac{n}{e})^n \leq n! \leq \sqrt{2\pi n}(\frac{n}{e})^n e^{\frac{1}{12n}}.$$

Proposition A.1 *Let* $0 < \lambda < 1$. *Then,*

$$\binom{n}{\lambda n} \le \frac{2}{\sqrt{n}} 2^{nH(\lambda)},$$

where $H(\lambda) = -\lambda \log \lambda - (1-\lambda)\log(1-\lambda)$ *is the binary entropy function.*

Proof. Using Stirling's approximation, we have

$$\begin{aligned}
\binom{n}{\lambda n} &\le \frac{2\sqrt{\pi n}(\frac{n}{e})^n}{\sqrt{2\pi\lambda n}(\frac{\lambda n}{e})^{\lambda n}\sqrt{2\pi(1-\lambda)n}(\frac{(1-\lambda)n}{e})^{(1-\lambda)n}} \\
&\le \frac{\sqrt{n}}{\sqrt{\lambda n}\sqrt{(1-\lambda)n}} \cdot \frac{n^n}{(\lambda n)^{\lambda n}((1-\lambda)n)^{(1-\lambda)n}} \\
&= \frac{1}{\sqrt{\lambda(1-\lambda)n}} \cdot \left(\frac{1}{\lambda^{\lambda}(1-\lambda)^{(1-\lambda)}}\right)^n \\
&= \frac{1}{\sqrt{\lambda(1-\lambda)n}} 2^{nH(\lambda)} \\
&\le \frac{2}{\sqrt{n}} 2^{nH(\lambda)}
\end{aligned}$$

□

Proposition A.2 *Let* $0 \le k < \frac{1}{2}n$. *Then,* $\sum_{i=0}^{k} \binom{n}{i} \le \binom{n}{k}\frac{n-k}{n-2k}$.

Proof. It is easy to verify that $\binom{n}{k-i} \le (\frac{k}{n-k+1})^i \binom{n}{k}$. Indeed, the inequality holds for $i = 0, 1$, and for $i > 1$ we receive by induction on i that

$$\binom{n}{k-i} = \frac{k-i+1}{n-k+i}\binom{n}{k-(i-1)} \le \frac{k}{n-k+1}\binom{n}{k-(i-1)} \le (\frac{k}{n-k+1})^i \binom{n}{k}.$$

Thus,

$$\sum_{i=0}^{k}\binom{n}{i} \le \binom{n}{k}\sum_{i=0}^{k}(\frac{k}{n-k+1})^i \le \binom{n}{k}\sum_{i=0}^{k}(\frac{k}{n-k})^i \le \binom{n}{k}\frac{1}{1-\frac{k}{n-k}} = \binom{n}{k}\frac{n-k}{n-2k}.$$

□

Conclusion A.3 *Let* $0 < \lambda \le \frac{\sqrt{n}-2}{2\sqrt{n}-2}$. *Then,*

$$\sum_{i=0}^{\lambda n}\binom{n}{i} \le 2^{nH(\lambda)}.$$

Proof. According to Proposition A.2,

$$\sum_{i=0}^{\lambda n}\binom{n}{i} \le \frac{1-\lambda}{1-2\lambda}\binom{n}{\lambda n}.$$

Using Proposition A.1 and the inequality $\lambda \le \frac{\sqrt{n}-2}{2\sqrt{n}-2}$, we obtain the conclusion. □

B Proof of Theorem 5.1

The result of Theorem 5.1 is immediately implied from the following two propositions.

Proposition B.1 $VDC((2r+1)Maj) \geq N(2r+1) - O(r^2)$.

Proof. The proof is a generalization of Proposition 4.7. We will introduce a group of $(2r+1)N - \binom{r+1}{2} - (r+1)(2r+1)$ choice pairs that are shattered by tournaments in *(2r+1)Maj*. The pairs are of two types, T1 and T2, as follows.

$$\begin{array}{llll} T1: & 1. & \forall\ 2 \leq i \leq N & (x_1, x_i) \\ & 2. & \forall\ 3 \leq i \leq N & (x_2, x_i) \\ & \vdots & & \\ & r. & \forall\ r+1 \leq i \leq N & (x_r, x_i) \\ T2: & r+1. & \forall\ 2r+2 \leq i \leq N & (x_{r+1}, x_i) \\ & r+2. & \forall\ 2r+2 \leq i \leq N & (x_{r+2}, x_i) \\ & \vdots & & \\ & 2r+1. & \forall\ 2r+2 \leq i \leq N & (x_{2r+1}, x_i) \end{array}$$

There are $Nr - \binom{r+1}{2}$ pairs of type T1, and $(r+1)(N-(2r+1))$ pairs of type T2. Consequently, the total number of pairs is

$$Nr - \binom{r+1}{2} + (r+1)(N-(2r+1)) = (2r+1)N - \binom{r+1}{2} - (r+1)(2r+1)\ .$$

Given a configuration of the pairs, we construct the $2r+1$ orderings, denoted by $O_1, \ldots, O_{2r+1}$, as follows. Generally, every ordering O_i has four regions:

$$\underbrace{\text{A}} \succ \underbrace{\text{B}} \succ x_i \succ \underbrace{\text{C}} \succ \underbrace{\text{D}}\ .$$

Regions A and D are "balance" regions, which assure that every x_i, $1 \leq i \leq 2r+1$, appears r times as a "small" element (that is, in region D) and r times as a "big" element (that is, in region A) in the $2r$ orderings except for O_i. This implies that the ordering O_i, by manipulating elements in regions B and C, exclusively determines the realization of the pairs in which x_i appears first. More specifically, the orderings are divided into two types and constructed as follows.

Type 1. There are r orderings of this type, denoted by $O_1, O_2 \ldots O_r$. The ordering O_i is "responsible" for the pairs of type T1–i, that is, pairs in which x_i appears first. The ordering O_i has four regions as described above. Regions A and D of O_i are "balance" regions, which include the elements $x_1, \ldots, x_{i-1}$. Together with the "balance" regions of the remaining orderings, they provide a tool to place every element out of $x_1, \ldots, x_{i-1}$, exactly r times in region A and r times in region D. Regions B and C include elements out of $x_{i+1}, \ldots, x_N$. Region B includes all the elements out of $x_{i+1}, \ldots, x_N$ such that $c(x_i, x_j) = 1$. Region C includes all the elements out of $x_{i+1}, \ldots, x_N$ such that $c(x_i, x_j) = 0$.

The ordering of the elements within the regions obeys the following rule: The smaller the index of the element is, the further it is located from x_i. For example, if both x_j and $x_{j'}$ appear in region A(or D), and $j < j'$, then $x_j \succ x_{j'}$ (or $x_{j'} \succ x_j$).

Type 2. There are $r+1$ orderings of this type, denoted by $O_{r+1} \ldots O_{2r+1}$. The ordering O_i is "responsible" for the pairs of type T2–i, and has four regions, as described above. Regions A and D are the "balance" regions, which include elements out of $x_1, \ldots, x_{2r+1}$ (one may notice that more elements appear in these regions with respect to the orderings of type 1). Regions B and C include elements out of $x_{2r+2}, \ldots, x_N$. Region B includes all the elements out of $x_{2r+2}, \ldots, x_N$ such that $c(x_i, x_j) = 1$. Region C includes all the elements out of $x_{2r+2}, \ldots, x_N$ such that $c(x_i, x_j) = 0$. The ordering of the elements within the regions obeys the same rule as in type 1 orderings.

It still remains to describe the balancing process, i.e., how to position the elements in regions A and D in the $2r+1$ orderings. It can be deduced from the construction of the orderings that x_i, $1 \leq i \leq 2r+1$, may appear in regions B and C only in the orderings $O_1, \ldots, O_r$; that is, x_i can appear in regions B and C at most r times. We have to "balance" these appearances with appearances in regions A and D. For example, if x_i appeared 2 times in region B and 4 times in region C (which implies that x_i is x_7), then it will be located in the remaining orderings (except for the ordering O_7) $r-2$ times in region A and $r-4$ times in region D. The ordering of the elements within regions combined with the fact that for $j > i$, x_j "joins" regions A and D not before x_i joins these regions, implies that $x_i \succ x_j$ in orderings (except for O_i) in which x_i appears in either region A or B, and that $x_j \succ x_i$ whenever x_i appears in regions C or D. Therefore, the "balancing" process assures that for every $j > i$, $x_i \succ x_j$ in r orderings (not including O_i), and $x_j \succ x_i$ in r orderings (not including O_i). Consequently, O_i alone realizes the examples in which x_i appears first, according to the location of the elements in regions B and C in this ordering. As every O_i agrees with the given configuration on the pairs in which x_i appears first, we conclude that the $2r+1$ orderings realize the configuration. □

Proposition B.2 *Let $r \geq 2$ be a positive integer. Then,*

$$VDC((2r+1)Maj) \leq \min\{(2r+1)N \log N, 10N(2r+1)^2 \log(2r+1)\}.$$

Proof. We already showed earlier that

$$VCD((2r+1)Maj) < (2r+1)N\log N.$$

It is left to show that $VDC((2r+1)Maj) \leq 10N(2r+1)^2\log(2r+1)$. Indeed, denote the VC-dimension by M. The same line of arguments we used in the proof of Proposition 4.9 can be applied here to get the inequality:

$$1 \leq H(\frac{r}{2r+1}) + H(\frac{N}{M}).$$

A series of algebraic manipulations will allow us to derive the result of the proposition.

Step 1: Approximate $H(\frac{r}{2r+1})$.

For a small x, we can approximate $H(\frac{1}{2}+x)$ using Taylor's formula around $x_0 = \frac{1}{2}$:

$$H(\frac{1}{2}+x) = H(\frac{1}{2}) + H'(\frac{1}{2})x + H''(\frac{1}{2})\frac{x^2}{2} + H^{(3)}(\frac{1}{2})\frac{x^3}{6} + R_4(x),$$

where $R_4(x)$ is the remainder in Taylor's formula. In our case, $R_4(x)$ is negative, because every odd derivative of H in the point $x_0 = \frac{1}{2}$ is zero and every even derivative is negative. Substituting for numbers in the above formula and using the fact that $R_4(x)$ is negative, we have:

$$\begin{array}{ll}
H(\frac{1}{2}+x) \leq 1 - 2x^2\log_2 e & \Longrightarrow_{\text{substitute } x \text{ with } -\frac{1}{4r+2}} \\
H(\frac{r}{2r+1}) \leq 1 - 2\log_2 e\frac{1}{(4r+2)^2} & \Longrightarrow_{\text{use the fact that } 1\leq H(\frac{r}{2r+1})+H(\frac{N}{M})} \\
1 \leq 1 - 2\log_2 e\frac{1}{(4r+2)^2} + H(\frac{N}{M}) & \Longrightarrow \\
2\log_2 e\frac{1}{(4r+2)^2} \leq H(\frac{N}{M}) & \Longrightarrow_{\log_2 e>1} \\
2\frac{1}{(4r+2)^2} \leq H(\frac{N}{M}) & \Longrightarrow \\
\frac{1}{2(2r+1)^2} \leq H(\frac{N}{M}) &
\end{array}$$

Step 2: Approximate $H(\frac{N}{M})$.

Let $0 < t < 1$. The sum of the geometric progression with multiplier t is:

$$\begin{array}{ll}
\frac{1}{1-t} = 1 + t + t^2 + t^3 + \ldots & \Longrightarrow_{\text{use integration on both sides}} \\
-\log(1-t) = t + \frac{1}{2}t^2 + \frac{1}{3}t^3 + \ldots & \Longrightarrow_{\text{multiply by } (1-t)} \\
-(1-t)\log(1-t) = t - (\frac{1}{2}t^2 + \frac{1}{6}t^3 + \ldots) \leq t & \Longrightarrow_{H(t)=-t\log t-(1-t)\log(1-t)} \\
H(t) \leq -t\log t + t & \Longrightarrow_{\frac{1}{2(2r+1)^2}\leq H(\frac{N}{M})} \\
\frac{1}{2(2r+1)^2} \leq -\frac{N}{M}\log\frac{N}{M} + \frac{N}{M} &
\end{array}$$

Denote $M = cN$. Then, the last inequality can be written as

$$\frac{1}{2(2r+1)^2} \leq \frac{1}{c}\log c + \frac{1}{c}\ .$$

This inequality implies that $c = O((2r+1)^2 \log(2r+1))$. We still want to determine the constant in the $O()$.

Indeed, denote $c = 2d(2r+1)^2 \log(2r+1)$. Then,

$$\begin{array}{ll}
\frac{1}{2(2r+1)^2} \leq \frac{1}{c} \log c + \frac{1}{c} & \Longrightarrow \\
\frac{1}{2(2r+1)^2} \leq \frac{1}{c} (\log c + 1) & \Longrightarrow_{c=2d(2r+1)^2 \log(2r+1)} \\
\frac{1}{2(2r+1)^2} \leq \frac{1}{2d(2r+1)^2 \log(2r+1)} (\log 2d(2r+1)^2 \log(2r+1) + 1) & \Longrightarrow \\
1 \leq \frac{1}{d \log(2r+1)} (\log 2d + 2\log(2r+1) + \log\log(2r+1) + 1) & \Longrightarrow_{\text{multiply by } d} \\
d \leq 2 + \frac{\log 2d + \log\log(2r+1) + 1}{\log(2r+1)} \leq 2 + 3 = 5 &
\end{array}$$

Consequently, the VC-dimension is at most $10N(2r+1)^2 \log(2r+1)$. □

References

[1] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to algorithms*, The MIT Press, Cambridge, Massachusetts, 1990.

[2] P. Erdős and L. Moser, *On the representation of directed graphs as unions of orderings*, Magyar Tud. Akad. Mat. Kutató Int Közl **9** (1964), 125–132.

[3] P. Fishburn, *Tournaments and majorities*, Unpublished manuscript, 2002.

[4] G. Kalai, *Learnability and rationality of choice*, Journal of Economic Theory (2003), forthcoming.

[5] M. J. Kearns and U. V. Vazirani, *An introduction to computational learning theory*, The MIT Press, Cambridge, Massachusetts, 1994.

[6] D. E. Knuth, *Sorting and searching*, volume 3 of *The Art of Computer Programming*, Addison-Welsey, 1973.

[7] D. C. McGarvey, *A theorem on the construction of voting paradoxes*, Econometrica **21** (1953), 608–610.

[8] J. W. Moon, *Topics on tournaments*, Holt, Rinehart and Winston, New York, 1968.

[9] A. Rubinstein, *Why are certain properties of binary relations relatively more common in natural language*, Econometrica **64** (1996), 343–356.

[10] N. Sauer, *On the density of families of sets*, Journal of Combinatorial Theory Series A **13** (1972), 145–147.

[11] M. Vidyasagar, *A theory of learning and generalization*, Communications and control engineering series, Springer, London, 1997.