On the Cut-off Point for Combinatorial Group Testing

Paul Fischer
Norbert Klasner
Ingo Wegener

Lehrstuhl Informatik II
Universität Dortmund
D-44221 Dortmund

Forschungsbericht Nr. 649, 1997

ON THE CUT-OFF POINT FOR COMBINATORIAL GROUP TESTING

Paul Fischer, Norbert Klasner, and Ingo Wegener *
FB Informatik, LS II, Univ. Dortmund, 44221 Dortmund, Germany
`fischer,klasner,wegener@ls2.informatik.uni-dortmund.de`

Abstract

The following problem is known as group testing problem for n objects. Each object can be essential (defective) or non-essential (intact). The problem is to determine the set of essential objects by asking queries adaptively. A query can be identified with a set Q of objects and the query Q is answered by 1 if Q contains at least one essential object and by 0 otherwise. In the statistical setting the objects are essential, independently of each other, with a given probability $p < 1$ while in the combinatorial setting the number $k < n$ of essential objects is known. The cut-off point of statistical group testing is equal to $p^* = \frac{1}{2}(3 - \sqrt{5})$, i.e., the strategy of testing each object individually minimizes the average number of queries iff $p \geq p^*$ or $n = 1$. In the combinatorial setting the worst case number of queries is of interest. Here the conjecture is established that the cut-off point of combinatorial group testing is equal to $\alpha^* = \frac{1}{3}$, i.e., the strategy of testing $n - 1$ objects individually minimizes the worst case number of queries iff $k/n \geq \alpha^*$. Some results in favor of this conjecture are proved.

Keywords: Combinatorics, Group testing, Cut-off point, Learning, Query learning.

*Supported in part by DFG grant We 1066/8.

1. INTRODUCTION

We start with the history of the group testing problem. In World War II, all persons who were drafted into the United States Army were given medical examinations by the United States Public Health Service and the Selective Service System. The Wassermann test was used to detect all those who had syphilis. The test was discovered in 1906 by August von Wassermann and detects syphilis in its early stages. The test is a serum reaction which identifies antibodies in the blood of the person who has syphilis. First, blood must be drawn from every subject. Then, in the conventional methods, each blood sample is subjected to the Wassermann test. Hence, one test was conducted for each test subject. It was soon discovered that tests could be saved as follows: Portions of the blood samples from several persons can first be mixed together and this mixture subjected to the Wassermann test. If the reaction is negative, then we know that all persons whose blood is in the mixture do not have syphilis. In the case of a positive reaction, we only know that at least one of these persons has syphilis.

In this scenario it is natural to assume that the persons are independently infected with a known (or estimated) probability $p < 1$. One is interested in a test sequence minimizing the average number of tests. The worst case scenario is trivial in this setting, since in the case that all persons are infected we have to test all persons individually. There is a list of papers describing good search strategies, e.g. Dorfman (1943), Sterrett (1957), Sobel and Groll (1959), and Sobel (1960).

Ungar (1960) determined the so-called optimal cut-off point for statistical group testing. For $p^* = \frac{1}{2}(3 - \sqrt{5})$ he proved that the strategy of testing the n persons individually is optimal iff $p \geq p^*$. More general problems have been considered by Sobel (1968) and Kumar (1970). In the hypergeometric group testing problem the number k of infected persons is known and it is assumed that all $\binom{n}{k}$ subsets of size k have the same probability of being the set of infected persons and in the trinomial group testing problem each person is in one of three possible states.

In the scenario introduced by Angluin (1988) as concept learning similar problems are considered with respect to worst case complexity. We only introduce the class of concept learning problems we are interested in. The concept is an unknown Boolean function f from some concept class $\mathcal{F}$ of Boolean functions on n inputs. The learner may ask for each $a \in \{0,1\}^n$ for the value of $f(a)$ and the teacher has to tell the right value. The learner may choose the queries adaptively, i.e., the i-th query may depend on the answers to the first $i-1$ queries. The combinatorial group testing problem is the counterpart of the statistical group testing problem considered above. In this new setting the concept class $\mathrm{OR}(k)$ contains the $\binom{n}{k}$ functions which are the disjunctions (OR) of k of the n possible variables. An input a is equivalent to the query $Q = \{i | a_i = 1\}$. The answer $f(a)$ equals 1 iff the set Q contains at least one of the k so-called essential variables or attributes. This is the same as testing a mixture of the blood of all persons $i \in Q$. The answer also is 1 iff at least one of the persons in Q is infected. In learning theory the case of small k is of particular interest. Attribute-efficient learning is defined as learning in the worst case with $\mathrm{poly}(k, \log n)$ queries. A lot of problems of this kind, even with more general concept classes and query types, have been solved, e.g., by Blum, Hellerstein, and Littlestone (1995), Bshouty and Hellerstein (1996), and Hegedűs and Indyk (1997). Uehara, Tsuchida, and Wegener (1997) consider also the case of arbitrary large k and look for learning algorithms whose worst case number of queries is only by a constant factor larger than some known lower bound which often is the trivial information theoretical lower bound $\lceil \log |\mathcal{F}| \rceil$.

Already the monographs of Picard (1965) and Ahlswede and Wegener (1987) have collected among many problems all the problems described above under the notion of search problems. Therefore, it seems to be natural to ask for the cut-off point for combinatorial group testing. The trivial strategy of asking query sets of size 1 needs $n-1$ queries in the worst case, since we know k and always can determine whether the last object is essential without testing it. A number α^* is called cut-off point of the combinatorial group testing problem if the strategy of asking queries of size 1 is optimal iff $k/n \geq \alpha^*$ and $k < n$.

Our conjecture is that $\alpha^* = 1/3$ is the cut-off point of the combinatorial group testing problem but we cannot even prove the existence of a cut-off point. We present results in favor of our conjecture. In Section 2 we present a learning strategy which needs in the worst case less than $n-1$ queries iff $k/n < 1/3$. In Section 3 we prove that $n-1$ queries are necessary if $k/n \geq 1/2$ and $k < n$. In Section 4 the problem where the queries are restricted to sets of at most two objects is investigated. Then $\alpha^* = 1/3$ is the cut-off point. Finally, we discuss why we cannot prove our more general conjecture.

2. A SIMPLE LEARNING STRATEGY

The information theoretic lower bound for the combinatorial group testing problem equals $\lceil \log \binom{n}{k} \rceil \geq k \log(\frac{n}{k})$. The best known upper bound due to Uehara, Tsuchida, and Wegener (1997) equals $k \lceil \log(\frac{n}{k}) \rceil + 2k - 2$. This bound is smaller than $n-1$ only if $k/n = 1/4$ or $k/n \leq 1/5$. Moreover, this strategy uses the fact that k is known. In the following we present and analyze a simple learning strategy with the following additional properties:

- the size of all queries is bounded by 2,
- the algorithm does not use the knowledge of k.

Algorithm 1: At the beginning all objects are unclassified.

a) If $n = 0$, stop.

b) If $n = 1$, solve the problem with the query consisting of the unclassified object, stop.

c) If $n \geq 2$, ask a query $Q = \{x, y\}$ containing two unclassified objects.

> **Case 1 :** The answer is 0. Then x and y are non-essential, $n := n - 2$.
>
> **Case 2 :** The answer is 1. Then ask the query $\{x\}$.

Case 2.1 : The answer is 1. Then x is essential and y unclassified, $n := n-1$.

Case 2.2 : The answer is 0. Then x is non-essential and it can be concluded that y is essential, $n := n-2$.

Theorem 1: i) Algorithm 1 uses at most $2k + \lceil (n-k)/2 \rceil$ queries.

ii) The combinatorial group testing problem can be solved with at most $2k + \lfloor (n-k)/2 \rfloor - 1$ queries, if $n \geq 2$.

iii) The combinatorial group testing problem can be solved with less than $n-1$ queries if $k/n < 1/3$.

Proof: i) Obviously, the learner only takes advantage from Case 2.2 of Algorithm 1. Therefore, we assume that Case 2 always leads to Case 2.1. If $n \geq 2$, we either identify 2 non-essential objects with 1 query or 1 essential object with 2 queries. Let k describe the number of unclassified essential objects. Then, $k := k$ in Case 1 and $k := k-1$ in Case 2.1. Hence, $n-k$ remains even (or odd) if it is even (resp. odd) in the beginning. If $n-k$ is even, then $n = 1$ implies $k = 1$ and we identify the last essential object with only one query, and $n = 0$ implies $k = 0$. If $n-k$ is odd, then $n = 1$ implies $k = 0$ and the identification of the last non-essential object costs one query. Altogether the number of queries is bounded by $2k + \lceil (n-k)/2 \rceil$.

ii) At the end of Algorithm 1 we can save queries, if we know k. Since n decreases at most by 2, we reach a subproblem where $n \in \{2,3\}$. By a simple case inspection we obtain the entries of Table 1 where *upper* is the value of the upper bound $2k + \lceil (n-k)/2 \rceil$ and *opt* the value of the optimal solution.

k	n	$n-k$	*upper*	*opt*	*upper*$-$*opt*
0	2	even	1	0	1
1	2	odd	3	1	2
2	2	even	4	0	4
0	3	odd	2	0	2
1	3	even	3	2	1
2	3	odd	5	2	3
3	3	even	6	0	6

Table 1: Bounds and optimal solutions for $n \in \{2,3\}$.

The upper bound $2k + \lceil (n-k)/2 \rceil$ assumes that we always reach Case 2.1. If we reach Case 2.2, we may ignore the information that x is non-essential. Then the upper bound still holds. If $n-k$ is odd in the beginning, we reach a subproblem (k', n') in this case where $n'-k'$ is odd and $n' \in \{2,3\}$. Hence, we can save 2 queries, if $n-k$ is odd, and 1 query in any case. This leads to the upper bound $2k + \lfloor (n-k)/2 \rfloor - 1$, if $n \geq 2$.

iii) The upper bound $2k + \lfloor (n-k)/2 \rfloor - 1$ equals $\frac{1}{2}n + \frac{3}{2}k - 1$ for even $n-k$. This is smaller than $n-1$ iff $k < \frac{1}{3}n$. The upper bound equals $\frac{1}{2}n + \frac{3}{2}k - \frac{3}{2}$ for odd $n-k$. This is smaller than $n-1$ iff $k < \frac{1}{3}n + \frac{1}{3}$. This last inequality is for integers k and n and $n-k$ odd equivalent to $k < \frac{1}{3}n$. (The only integer case where $\frac{1}{3}n \le k < \frac{1}{3}n + \frac{1}{3}$ is $k = \frac{1}{3}n$ implying that $n-k$ is even.) $\square$

Using the learning strategy with queries of size 1 we pay one unit per element (except the last one). Our simple strategy works with pairs and pays two units for essential and half a unit for non-essential objects (except the last one). This is cheaper iff there are at least twice as many non-essential objects than essential ones.

3. A GENERAL ADVERSARY STRATEGY

Lower bounds can be proved by adversary strategies. An adversary gives consistent answers to the queries of the learner in order to force the learner to ask many queries. We assume w.l.o.g. that the learner does not ask a query whose answer already is known. Then each answer of the adversary is consistent. This assumption implies that no query contains an object known to be essential. We also assume w.l.o.g. that the learner does not include an object known to be non-essential into a query. In order to simplify the analysis of the adversary strategy the adversary may give some further information to the learner.

Theorem 2: If $\lfloor n/2 \rfloor \le k \le n-1$, $n-1$ queries are necessary for the combinatorial group testing problem.

Proof: The statement is obvious for $n \in \{1,2\}$. If $k = n-1$, the result is easy to prove. Queries containing at least two objects are useless, since the answer 1 is known. Hence, the learner only asks queries of size 1. If the adversary always answers with 1, then the learner is left after $n-2$ queries with the problem of identifying the non-essential object among two objects. Hence, the learner needs a further query.

If $k < n-1$, the adversary works as follows. As long as the query sets are disjoint and at most $n-k-1$ queries of size 1 have been asked, the adversary answers 0 for queries of size 1 and 1 for queries of larger size. These answers are possible, since there are $n-k$ non-essential objects and since, if $k \ge \lfloor n/2 \rfloor$, at most $\lfloor n/2 \rfloor$ disjoint queries with at least two objects can be asked.

If the learner has asked $n-k-1$ queries of size 1, the learner has classified $n-k-1$ non-essential objects with $n-k-1$ queries. By the considerations above the learner needs k further queries for the remaining problem (k', n'), where $n' = k+1$ and $k' = k$.

Otherwise, we consider the first point in time where the learner asks a query Q which is not disjoint from a former query Q'. By our assumptions on the learner Q' is of size at least 2. The adversary reveals more information by choosing an object $x \in Q \cap Q'$ and deciding that x is essential and by choosing an object $y \in Q' \setminus \{x\}$ and deciding that y is non-essential. Then the answers 1 to Q and Q' are of no further value. The learner has used two queries to identify an essential and a non-essential object. With the remaining queries the learner has to solve the problem (k', n') where $n' = n-2$ and $k' = k-1$.

Since the cases $n = 1$ and $n = 2$ can serve as base of induction, we conclude by induction that the learner needs $n-3$ further queries for the remaining problem. Hence, the learner needs $n-1$ queries altogether. □

4. THE CUT-OFF POINT FOR QUERIES OF SIZE BOUNDED BY 2

If the size of the query sets is bounded by 2, we can prove that 1/3 is the cut-off point of combinatorial group testing.

Theorem 3: If the size of the query sets is bounded by 2, the cut-off point is 1/3, i.e., $n-1$ queries are necessary iff $n/3 \leq k \leq n-1$.

Proof: It follows from the results of Section 2 that the combinatorial group testing problem can be solved with less than $n-1$ queries whose size is bounded by 2 if $k/n < 1/3$. Therefore, it is sufficient to prove that $n-1$ queries are necessary if $n/3 \leq k \leq n-1$.

We again work with an adversary strategy and work with the same assumptions as in Section 3. The adversary works with the notions *free*, *bound*, and *classified* for the objects. At the beginning all objects are free. With a query of size 1 a free or bound object becomes classified (as essential or non-essential). If a query of two free objects is answered by 0, the free objects become classified as non-essential. If the answer is 1, the free objects become bound and are called partners. It is known that at least one of the partners is essential. The adversary guarantees that no object is simultaneously bound by two queries and that the first of the two partners which becomes classified is classified as essential. Then the former query gives no information about the other partner which becomes free again.

Before we can describe the adversary strategy in detail, we introduce some notations used by the adversary. Let $Q_1, \ldots, Q_t$ be the first t queries of the learner and $A_1, \ldots, A_t$ the answers of the adversary. The adversary describes object i by f_i, if it is free, by c_i, if it is classified, and by b_i^s, if it is currently bound because of query Q_s. In the last case there exists the partner b_j^s, $j \neq i$. The query Q_r is alive at t, if it contains at least one object which is bound after Q_t because of Q_r.

The adversary chooses the answers and the additional information (which is for free) in such a way that as long as the job of the learner is not finished the number of classified objects is not larger than the number of queries asked and that, in the end, the number of queries is at least $n-1$. Moreover, the adversary takes care that the queries which are alive are pairwise disjoint. More precisely the adversary evaluates the situation with the following parameters:

- γ_t, the number of queries which are alive,
- σ_t, the number of classified objects,
- $\sigma_{1,t}$, the number of objects classified as essential,
- $n_t := n - \gamma_t - \sigma_t$,

- $k_t := k - \gamma_t - \sigma_{1,t}$, and
- $d_t := 3k_t - n_t$.

Intuitively, n_t describes the number of "free" decisions and k_t the number of free decisions which have to give the result "essential". Only if $1 \le k_t \le n_t - 1$, the "free" decisions are really free. Moreover, d_t describes whether the number of "free essential" objects is large with respect to the number of "free" objects.

Obviously, $\gamma_0 = \sigma_0 = \sigma_{1,0} = 0$, $n_0 = n$, and $k_0 = k$. Moreover, $d_0 \ge 0$, since $n/3 \le k$. Now we describe how the adversary answers the query Q_t. Let $i \ne j$.

Case 1 $Q_t = \{f_i\}$, $d_{t-1} \le 1$.

Answer 0, $f_i \to c_i$.

Case 2 $Q_t = \{f_i\}$, $d_{t-1} \ge 2$.

Answer 1, $f_i \to c_i$.

Case 3 $Q_t = \{b_i^s\}$.

Answer 1, $b_i^s \to c_i$, $b_j^s \to f_j$ for the partner b_j^s of b_i^s.

Case 4 $Q_t = \{f_i, f_j\}$, $d_{t-1} \le -1$.

Answer 0, $f_i \to c_i$, $f_j \to c_j$.

Case 5 $Q_t = \{f_i, f_j\}$, $d_{t-1} \ge 0$.

Answer 1, $f_i \to b_i^t$, $f_j \to b_j^t$.

Case 6 $Q_t = \{b_i^s, x\}$, x free or bound.

Answer 1, $b_i^s \to c_i$, $b_j^s \to f_j$ for the partner b_j^s of b_i^s, the status of x is not changed.

Table 2 describes how the six essential parameters change their value by Q_t and the answer of the adversary. The contents of the table follow from the descriptions of the adversary strategy.

Case	Answer	$\gamma_t - \gamma_{t-1}$	$\sigma_t - \sigma_{t-1}$	$\sigma_{1,t} - \sigma_{1,t-1}$	$n_t - n_{t-1}$	$k_t - k_{t-1}$	$d_t - d_{t-1}$
1	0	0	+1	0	−1	0	+1
2	1	0	+1	+1	−1	−1	−2
3	1	−1	+1	+1	0	0	0
4	0	0	+2	0	−2	0	+2
5	1	+1	0	0	−1	−1	−2
6	1	−1	+1	+1	0	0	0

Table 2: Change of the values of the six essential parameters.

We know that $d_0 \geq 0$. If $d_0 > 2$, the Cases 1 and 4 are impossible and the d-value does not increase. Moreover, the answer to all queries is 1 until $d_t \in \{0, +1, +2\}$ for some t. Since Case 4 is impossible, the number of classified objects is bounded above by the number of queries. This even holds for the number of queries which are not alive. Only in Case 5 a query which is alive is created. But in Case 5 no object is classified.

The Cases 3 and 6 do not change the d-value. The assumptions of the other cases ensure that $d_r \in \{-2, -1, 0, +1, +2\}$ for all $r > t$ if $d_t \in \{-2, -1, 0, +1, +2\}$. The change of the d-values is illustrated in Figure 1. The node labels are the current d-values. For each node only one of the Cases 1 and 2 is possible. Also only one of the Cases 4 and 5 is possible. The corresponding edge is drawn as a thick line. The Cases 3 and 6 lead to self-loops. The edge labels describe the change of the σ-values.

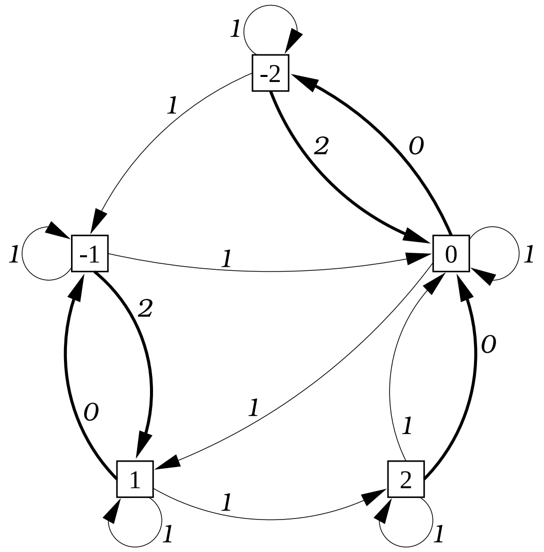

Figure 1: The changes of d- and σ-values.

Let us investigate the case $d_0 \in \{0, +1, +2\}$. Each learning strategy together with our adversary strategy leads to a path in the graph of Figure 1 starting at d_0. It is easy to see that the sum of the edge labels of each path starting at d_0 is bounded above by the length of the path which is equal to the number of queries. The simple reason is that edges with label 2 only start at the nodes -1 and -2. In order to reach these nodes we have to choose an edge with label 0. Together with our investigations of the case $d_0 > 2$ we have proved that the number of classified objects is not larger than the number of queries as long as $1 \leq k_t \leq n_t - 1$. We still have to investigate what happens if $k_t = 0$ or $k_t = n_t$. If $d_t \geq 3$, $3k_t - n_t \geq 3$. Hence, $k_t > 0$. As long as $d_t \geq 3$, all queries are answered by 1 and $n_t - k_t$ remains unchanged. Hence, $n_t - k_t = n - k \geq 1$. We still have to investigate the cases where $d_t \in \{-2, -1, 0, +1, +2\}$. If $k_t = n_t$, then $d_t = 3k_t - n_t = 2k_t$ and $n_t = k_t = d_t/2$. This is only possible if $d_t = 2$ and $n_t = k_t = 1$. In this situation we save one query. If $k_t = 0$, then $d_t = -n_t$. This implies that $d_t = -1$ and $n_t = 1$ or $d_t = -2$ and $n_t = 2$. In the last case we even save two queries. But in this case our path ends in the node -2 and the number of queries was larger than the number of classified objects. In each case the number of queries is at least $n-1$. $\square$

5. AN ADVERSARY STRATEGY FOR QUERIES OF SIZE BOUNDED BY l

The adversary strategy of the previous section can be generalized to larger query sets when increasing the ratio k/n.

Theorem 4: If the size of the query sets is bounded by $l \geq 2$, $n-1$ queries are necessary if $\frac{l-1}{2l-1}n \leq k \leq n-1$.

First note that for $l = 2$ this yields the result of Section 4.

Proof Sketch: We use the notation of the previous section except the following definition $d_t := \frac{2l-1}{l-1}k_t - n_t$. Again, the adversary distinguishes six cases:

Case 1 $Q_t = \{f_i\}$, $d_{t-1} \leq 1$.

Answer 0, $f_i \to c_i$.

Case 2 $Q_t = \{f_i\}$, $d_{t-1} \geq \frac{l}{l-1}$.

Answer 1, $f_i \to c_i$.

Case 3 $Q_t = \{b_i^s\}$.

Answer 1, $b_i^s \to c_i$, set all partners of b_i^s free.

Case 4 $Q_t = \{f_i, f_j, \ldots\}$, $d_{t-1} \leq 1 - l$.

Answer 0, $f_i \to c_i$, $f_j \to c_j$, ...

Case 5 $Q_t = \{f_i, f_j, \ldots\}$, $d_{t-1} \geq \frac{l(2-l)}{l-1}$.

Answer 1, $f_i \to b_i^t$, $f_j \to b_j^t$, ...

Case 6 $Q_t = \{b_i^s, x, \ldots\}$, $x, \ldots$ free or bound.

Answer 1, $b_i^s \to c_i$, set all partners of b_i^s free, the status of $x, \ldots$ is not changed.

Table 3 describes how d_t changes its value as a reaction to Q_t and the answer of the adversary. The contents of the table follows from the descriptions of the adversary strategy.

The remaining proof is similar to the proof of Theorem 3. $\square$

Case	Answer	$d_t - d_{t-1}$
1	0	$+1$
2	1	$-\frac{l}{l-1}$
3	1	0
4	0	$+\lvert Q_t\rvert \leq +l$
5	1	$-\frac{l}{l-1}$
6	1	0

Table 3: Change of the value of d_t.

6. DISCUSSION

In the statistical group testing problem the learner knows p and with high probability the number of essential objects is close to $k := np$. In the combinatorial problem the learner knows the number k of essential objects. Hence, the task is easier. But we are interested in the worst case scenario while Ungar (1960) considers the average case scenario. This makes the two approaches incomparable.

We have proved that the cut-off point of the combinatorial group testing problem is 1/3 if the size of the queries is bounded by 2. If our conjecture that the cut-off point is 1/3 in the general case is wrong, then there is some choice of the parameters k and n where $k \geq n/3$ and we can exceed the strategy of asking queries of size 1. But this is only possible with a strategy using also queries with at least three objects. One would expect that for an example with the smallest value of k/n we would be able to work with queries of size 1 and 2.

Our adversary has difficulties if larger query sets are allowed. In some situations (Case 4) the adversary may answer a query of two free objects by 0 which classifies two objects. The answer 0 to queries of at least three objects reveals too much information and the adversary should answer these queries by 1. Thus it becomes crucial to reveal less information than in Case 6, if a query contains a bound object. This makes adversaries hard to analyze.

REFERENCES

[1] Ahlswede, R. and Wegener, I. (1987). Search Problems. Wiley.

[2] Angluin, D. (1988). Queries and concept learning. Machine Learning 2, 318 – 342.

[3] Blum, A., Hellerstein, L., and Littlestone, N. (1995). Learning in the presence of finitely or infinitely many irrelevant attributes. Journal of Computer and System Sciences 50, 32 – 40.

[4] Bshouty, N.H. and Hellerstein, L. (1996). Attribute-efficient learning in query and mistake-bound models. Proc. of 9th Conf. on Computational Learning Theory COLT'96, 235 – 243.

[5] Dorfman, R. (1943). The detection of defective members of large populations. Ann. Math. Stat. 14, 436 – 440.

[6] Hegedűs, T. and Indyk, P. (1997). On learning disjunctions of zero-one threshold functions with queries. Techn. Rep. Univ. of Helsinki.

[7] Hellerstein, L., Pillaipakkamnatt, K., Raghavan, V., and Wilkins, D. (1996). How many queries are needed to learn? Journal of the ACM 43, 840 – 862.

[8] Kumar, S. (1970). Group-testing to classify all units in a trinomial sample. Studia Sci. Math. Hungar. 5, 229 – 247.

[9] Picard, C. (1965). Théorie des Questionnaires. Gauthier-Villars, Paris.

[10] Sobel, M. (1960). Group testing to classify efficiently all defectives in a binomial sample. Information and Decision Processes (Ed. Machol). McGraw-Hill, 127 – 161.

[11] Sobel, M. (1968). Binomial and hypergeometric group testing. Studia Sci. Math. Hungar. 3, 19 – 42.

[12] Sobel, M. and Groll, P.A. (1959). Group testing to classify efficiently all defectives in a binomial sample. Bell Systems Techn. J. 38, 1179 – 1259.

[13] Sterrett, A. (1957). On the detection of defective members of large populations. Ann. Math. Stat. 28, 1033 – 1036.

[14] Uehara, R., Tsuchida, K., and Wegener, I. (1997). Optimal attribute-efficient learning of disjunctions, parity, and threshold functions. Proc. of 3rd European Conference on Computational Learning Theory, EuroColt, LNCS in Artificial Intelligence 1208, 171 – 184.

[15] Ungar, P. (1960). The cut-off point for group testing. Commun. Pure Appl. Math. 13, 49 – 54.