Electronic Colloquium on Computational Complexity, Report No. 61 (1998)

On Frequent Sets of Boolean Matrices

Robert H. Sloan*
Email: sloan@eecs.uic.edu.
Dept. of EE & Computer Science
University of Illinois at Chicago
851 S. Morgan St. Rm 1120
Chicago, IL 60607-7053

Ken Takata†
Email: ktakat1@uic.edu
Dept. of Math., Stat., & Comp. Sci.
University of Illinois at Chicago

György Turán‡
Email: gyt@uicvm.uic.edu.
Dept. of Math., Stat., & Comp. Sci.
University of Illinois at Chicago,
Research Group on Artificial Intelligence
Hungarian Academy of Sciences

Abstract

Given a Boolean matrix and a threshold t, a subset of the columns is *frequent* if there are at least t rows having a 1 entry in each corresponding position. This concept is used in the algorithmic, combinatorial approach to knowledge discovery and data mining. We consider the complexity aspects of frequent sets. An explicit family of subsets is given that requires exponentially many rows to be represented as the family of frequent sets of a matrix, with any threshold. Examples are given of families that can be represented by a small matrix with

*Corresponding author. Partially supported by NSF grant CCR-9800070. Member, National Center for Data Mining at University of Illinois at Chicago.

†Partially supported by NSF grant CCR-9800070.

‡Partially supported by NSF grant CCR-9800070, ESPRIT grant 20237, and OTKA grant T-016349. Member, National Center for Data Mining at University of Illinois at Chicago.

ISSN 1433-8092

threshold t, but that require a significantly larger matrix if the threshold is less than t. We also discuss the connections of these problems to circuit complexity and the existence of efficient listing algorithms.

1 Introduction

Let A be a 0-1 matrix with m rows and n columns, and let t, $0 \leq t \leq m+1$ be an integer called the frequency threshold, or *threshold*. A subset $I \subseteq \{1, \ldots, n\}$ of the columns is *frequent* if there are at least t rows that have a 1 entry in each column belonging to I, and it is *infrequent* otherwise. Clearly, a subset of a frequent set is also frequent, and a superset of an infrequent set is also infrequent. On the other hand, *every* family of subsets that is closed under taking subsets can be represented as the system of frequent sets of a matrix, even when the threshold is restricted to 1.

Frequent sets are used in the algorithmic, combinatorial approach to knowledge discovery and data mining [1, 18, 19]. (Some papers use the term "large itemset" for frequent sets.) Here a matrix, for example, can represent the transaction database of a department store. Columns can correspond to products, rows can correspond to transactions, and a set of products is frequent if many transactions involve each of these products.

In this paper we study the complexity aspects of frequent sets. Given a family of subsets that is closed under taking subsets, what is the smallest matrix that has this family as its frequent sets? The size of a matrix is measured by the number of its rows. The existence of families requiring exponentially many rows follows from a standard counting argument in Boolean complexity theory. We give an *explicit* family of subsets that requires exponentially many rows. The role of the frequency threshold is also considered. We show that there are families that can be represented by a small matrix with some threshold t, but that require a significantly larger matrix if the threshold is required to be anything less than t. Related results, for a more general model and a different range of parameters, are obtained by Jukna [15].

The proofs use techniques from circuit complexity theory and combinatorics [4, 8, 13, 16, 23]. The problems studied here have a reformulation in terms of threshold circuits, and thus the results can be interpreted as separation and trade-off results for monotone perceptrons.

The paper is organized as follows. In Section 2 we give preliminaries on frequent sets. Section 3 contains the exponential lower bound. Section 4 gives the size-threshold trade-offs, with the combinatorial lemmas proven

separately in Section 4.1 . In Section 5 we reformulate the results in terms of circuit complexity. Sections 6 and 7 contain a brief discussion of the application of frequent sets in knowledge discovery and data mining, and several open problems, mostly concerning the existence of efficient listing algorithms related to frequent sets.

2 Preliminaries

Instead of a subset $I \subseteq [n] = \{1, \ldots, n\}$, we usually refer to the characteristic vector $v_I \in \{0,1\}^n$ of I, and so we talk about *frequent* and *infrequent vectors.* The 0-set of a vector in $\{0,1\}^n$ is the subset of $[n]$ that corresponds to its 0 coordinates; similarly for 1-set.

Given a matrix A and a threshold t, we consider the set of *infrequent* vectors, corresponding to the Boolean function $f_{A,t} : \{0,1\}^n \to \{0,1\}$. Thus, $f_{A,t}(x) = 1$ iff x is the characteristic vector of an infrequent set. We say that $f_{A,t}$ is *represented* by A and t. (We consider infrequent sets rather than frequent sets for convenience of notation.) Clearly, $f_{A,t}$ is always a *monotone* function. (A Boolean function is monotone if $f(x) = 1$ and $y \geq x$ implies $f(y) = 1$, where $y \geq x$ means that every component of y is at least as large as the corresponding component of x. A *maximal 0-vector* x of a monotone Boolean function f is a vector x such that $f(x) = 0$, but if y is obtained from x by switching any of its 0 components to 1, then $f(y) = 1$.) On the other hand, every monotone Boolean function f is represented by some matrix A and threshold t. This follows from the fact that infrequent sets of matrices with threshold 1 are equivalent to monotone conjunctive normal form (CNF) expressions.

Proposition 1 *A monotone Boolean function of n variables f is represented by some $m \times n$ matrix A with threshold $t = 1$ if and only if f has a monotone CNF with m clauses.*

Proof. Let f be a monotone Boolean function in the variables $x_1, \ldots, x_n$, and $C_1 \wedge \cdots \wedge C_m$ be a monotone CNF for f. Then the $m \times n$ matrix A defined by $a_{ij} = 0$ iff x_j is contained in C_i, represents f with threshold 1. Conversely, let A be an m-row matrix representing f with threshold 1. Then $C_1 \wedge \cdots \wedge C_m$ is a monotone CNF for f, where C_i contains a variable x_j iff $a_{ij} = 0$. □

Let $T_k^n(x_1, \ldots, x_n)$ be the "at-least-k-out-of-n" threshold function. Thus, $T_k^n(x_1, \ldots, x_n) = 1$ iff at least k of the variables $x_1, \ldots, x_n$ are set to 1. The

unique minimal monotone CNF of $T_k^n(x_1, \ldots, x_n)$ is

$$\bigwedge_{i_1 < \cdots < i_{n-k+1}} (x_{i_1} \vee \cdots \vee x_{i_{n-k+1}}),$$

having $\binom{n}{n-k+1} = \binom{n}{k-1}$ clauses.

By J_n we denote the $n \times n$ matrix that has 0's in its diagonals, and 1's everywhere else. The following proposition follows directly from the definitions.

Proposition 2 *The matrix J_n with threshold t ($0 \leq t \leq n+1$) represents the function T_{n-t+1}^n.*

Now we note that representing monotone Boolean functions as infrequent sets of matrices with arbitrary thresholds can actually be more powerful than representing them as CNFs, which correspond to threshold 1 by Proposition 1.

Proposition 3 *$T_{n/2}^n$ is represented by J_n with threshold $\lfloor n/2 \rfloor + 1$. Every matrix representing $T_{n/2}^n$ with threshold 1 must have at least $\binom{n}{\lceil n/2 \rceil - 1}$ rows.*

Proof. We only have to prove the second claim, and this follows from Proposition 1 by noting that the unique minimal monotone CNF of $T_{n/2}^n$ has $\binom{n}{\lceil n/2 \rceil - 1}$ clauses. □

3 A lower bound for the number of rows

In this section we discuss lower bounds for the number of rows needed to represent a monotone Boolean function f. First we note that the standard counting argument from Boolean complexity theory (see, e.g., [25]) shows that most monotone Boolean functions require large matrices.

Proposition 4 *Almost all n-variable monotone Boolean functions require $\Omega(\frac{2^n}{n^{3/2}})$ rows to be represented by a matrix, with any threshold.*

Proof. There are at least $2^{c\frac{2^n}{\sqrt{n}}}$ n-variable monotone Boolean functions for some constant c (see, e.g. [25]), and at most $m^2 2^{mn}$ matrices of $\leq m$ rows with a threshold. The bound follows by comparing these quantities. □

Now we give a family of explicit Boolean functions which require exponentially many rows. The proof of the lower bound uses the "discriminator method" of [13] for threshold circuits.

Theorem 5 *The minimal number of rows required to represent the function* $f_n(x_1, \ldots, x_n, y_1, \ldots, y_n) = \bigvee_{i=1}^{n}(x_i \wedge y_i)$ *is* 2^n.

Proof. The upper bound follows from using the CNF as in Proposition 1. For the lower bound, let A be an $m \times 2n$ matrix with threshold t, representing f_n. Let a_i be the i'th row of A, and $\bar{a}_i$ be its complement. Let $V = \{v_1, \ldots, v_n\}$ be the set of minimal 1-vectors of f_n. Thus, for v_i one has $x_i = y_i = 1$ and all the other components are 0. Also, let W be the set of maximal 0-vectors of f_n. Thus, $|W| = 2^n$, and for every vector in $|W|$, exactly one of x_i and y_i is 1, for every i. Let D_1 be the uniform probability distribution on V and let D_2 be the uniform probability distribution on W.

If x is any vector in $\{0,1\}^{2n}$ then let $\#(x)$ be the number of rows i such that $x \wedge \bar{a}_i \neq 0$ (where $\wedge$ denotes componentwise "and" and 0 is the all 0 vector). Thus $\#(x)$ is the number of rows for which the 0-set of the row intersects the 1-set of x. Put $r = m + 1 - t$. Then, by definition, if $v \in V$ then $\#(v) \geq r$, and if $w \in W$ then $\#(w) \leq r - 1$. Taking expectations with respect to D_1 and D_2, it follows that $E_{D_1}(\#(\mathbf{x})) \geq r$ and $E_{D_2}(\#(\mathbf{x})) \leq r-1$.

But

$$E_{D_1}(\#(\mathbf{x})) = \sum_{i=1}^{m} \mathrm{Prob}_{D_1}(\mathbf{x} \wedge \bar{a}_i \neq 0) \geq r$$

and

$$E_{D_2}(\#(\mathbf{x})) = \sum_{i=1}^{m} \mathrm{Prob}_{D_2}(\mathbf{x} \wedge \bar{a}_i \neq 0) \leq r - 1.$$

Subtracting these two inequalities we get

$$\sum_{i=1}^{m} \mathrm{Prob}_{D_1}(\mathbf{x} \wedge \bar{a}_i \neq 0) - \mathrm{Prob}_{D_2}(\mathbf{x} \wedge \bar{a}_i \neq 0) \geq 1.$$

Thus it must be the case that for some i

$$\mathrm{Prob}_{D_1}(\mathbf{x} \wedge \bar{a}_i \neq 0) - \mathrm{Prob}_{D_2}(\mathbf{x} \wedge \bar{a}_i \neq 0) \geq 1/m.$$

Hence the lower bound follows if we show that for every $b \in \{0,1\}^{2n}$ it holds that

$$\mathrm{Prob}_{D_1}(\mathbf{x} \wedge b \neq 0) - \mathrm{Prob}_{D_2}(\mathbf{x} \wedge b \neq 0) \leq 1/2^n.$$

This is certainly true if b is such that $x_i = y_i = 1$ for some i, as in this case $\mathrm{Prob}_{D_2}(\mathbf{x} \wedge b \neq 0) = 1$. Otherwise let us assume that b is such that exactly one of x_i and y_i is equal to 1 for s values of i, and $x_i = y_i = 0$ for

the remaining $n-s$ values of i ($0 \leq s \leq n$). Counting the number of vectors in V and W intersecting b we get

$$\mathrm{Prob}_{D_1}(\mathbf{x} \wedge b \neq 0) = s/n$$

and

$$\mathrm{Prob}_{D_2}(\mathbf{x} \wedge b \neq 0) = 1 - \frac{1}{2^s}.$$

Now if $s = n$ then $s/n - (1 - 1/2^s) = 1/2^n$. Otherwise, we claim that $s/n \leq 1 - 1/2^s$. Indeed, if $1 \leq s \leq n/2$ then $s/n \leq 1/2 \leq 1 - 1/2^s$; if $n/2 < s < n$ then $s/n \leq 1 - 1/n < 1 - 1/2^{n/2} < 1/2^s$ whenever $n \geq 5$. If $n \leq 5$ or $s = 0$ the statement is obvious. □

In fact, the proof shows that the matrix corresponding to the CNF is the unique representation of the function $f_n(x_1, \ldots, x_n, y_1, \ldots, y_n) = \bigvee_{i=1}^{n}(x_i \wedge y_i)$ with 2^n rows.

4 The role of the frequency threshold

We now turn to "trade-offs" between the size of the threshold and the number of rows. Examples are given that show that even decreasing the threshold by 1 may result in a significant increase in the number of rows. We show that such a trade-off phenomenon holds for the threshold functions T^n_{n-t+1} represented by the matrix J_n with a threshold t, for every fixed constant t. For small values of t ($2 \leq t \leq 4$) we get sharper bounds than in the general case, therefore these cases are discussed separately.

Proposition 6 *T^n_{n-1} is represented by J_n with threshold 2. The minimal number of rows needed to represent T^n_{n-1} with threshold < 2 is $\binom{n}{2}$.*

Proof. The unique minimal monotone CNF of T^n_{n-1} is $\bigwedge_{i \neq j}(x_i \vee x_j)$ with $\binom{n}{2}$ clauses. The proposition then follows from Proposition 1. □

The combinatorial lemmas used in the proofs of the remaining theorems of this section are given separately in Section 4.1.

Theorem 7 *T^n_{n-2} is represented by J_n with threshold 3. The minimal number of rows needed to represent T^n_{n-2} with threshold < 3 is $\binom{n-1}{2} + 1$.*

Proof. Let A be a matrix representing T^n_{n-2} with threshold less than 3. If the threshold is 1, then the number of rows is at least $\binom{n}{3}$ by Proposition

1. So let us assume that the threshold is 2. In this case let us consider the collection $\mathcal{S}$ of subsets of $[n]$ consisting of the 0-sets of each row of A.[1] Then, by considering the 0-sets of frequent and infrequent vectors, it follows that $\mathcal{S}$ has the following property: every ≤ 2 element subset of $[n]$ contains ≤ 1 set from $\mathcal{S}$ and every ≥ 3 element subset of $[n]$ contains ≥ 2 subsets from $\mathcal{S}$. Conversely, every collection $\mathcal{S}$ of subsets of $[n]$ with this property can be used to construct a matrix that represents T^n_{n-2} with threshold 2. Hence, what we need is a lower bound for the size of collections of subsets having this property. Such a lower bound is provided by Lemma 10 in Section 4.1. □

The proofs of the next two theorems are analogous, using Lemmas 11 and 12, and Lemma 13, respectively.

Theorem 8 *T^n_{n-3} is represented by J_n with threshold 4. The minimal number of rows needed to represent T^n_{n-3} with threshold < 4 is between $(\frac{1}{87} + o(1))n^3$ and $(\frac{1}{24} + o(1))n^3$.*

Proof. The argument is analogous to the proof of the previous theorem. In this case the threshold may be 1, 2 or 3. If the threshold is 1, a lower bound of $\binom{n}{4}$ follows from Proposition 1. If the threshold is 2 or 3, then the required lower bound follows from Lemma 11, respectively Lemma 12 in Section 4.1. □

Finally, we present the bound for the case when t is an arbitrary fixed constant.

Theorem 9 *Let $t \geq 5$ be a fixed constant. Then T^n_{n-t+1} is represented by J_n with threshold t. The minimal number of rows needed to represent T^n_{n-t+1} with threshold $< t$ is $\Theta(n^{t-1})$.*

Proof. As in the previous theorems, let A be a matrix representing T^n_{n-t+1} with threshold $1 \leq l < t$, and consider the collection $\mathcal{S}$ of the 0-sets of the rows of A. Then every set of size $\leq t-1$ contains $< l \leq t-1$ sets from $\mathcal{S}$, and every set of size $\geq t$ contains $\geq l$ sets from $\mathcal{S}$. In Lemma 13 we show the following: if $\mathcal{S}$ is a set of subsets of $[n]$ such that for every $A \subseteq [n]$ it holds that if $|A| \leq k$ then A contains fewer than k sets from $\mathcal{S}$, and if $|A| > k$ then A contains *at least one* set from $\mathcal{S}$, then $|\mathcal{S}| = \Omega(n^k)$. Note that in this version we do not have to talk about multiple occurrences of a set, as deleting multiple occurrences will still preserve the conditions. The theorem thus follows from Lemma 13 by putting $k = t-1$. □

[1] A *collection* of subsets may contain a given subset several times.

4.1 Combinatorial lemmas

In this section we prove the lemmas used in the preceding trade-off results. Before formulating them, let us give a brief description of the general combinatorial problem, which is partially solved here.

The collection $\mathcal{S}$ consisting of the singletons $\{1\}, \ldots, \{n\}$ in $[n]$ has the following property: if a subset $A \subseteq [n]$ has size $\leq k$ than it contains $\leq k$ sets from $\mathcal{S}$, and if it has size $> k$ then it contains $> k$ sets from $\mathcal{S}$. Thus "large" (i.e., size $> k$) and "small" (i.e., size $\leq k$) sets can be distinguished by counting how many sets they contain from $\mathcal{S}$ and using k as the cut-off point. Now the general question is the following: how many sets do we need in our collection to distinguish large (i.e., size $> k$) from small (i.e., size $\leq k$) sets if the cut-off point must be a number l that is *less than* k? The set of all k-element sets gives a solution with $\binom{n}{k}$ sets, using any number l, $2 \leq l < k$, as the cut-off point. The results below show that if k is a fixed constant then this solution is optimal up to order of magnitude. Note that $\mathcal{S}$ may contain repeated sets, and sets of *different sizes*. As the examples in Lemmas 10 and 11 below indicate, using sets of different sizes *does* help to reduce the size of $\mathcal{S}$.

The first lemma is used in the proof of Theorem 7.

Lemma 10 *Let $\mathcal{S}$ be a collection of subsets of $[n]$ ($n \geq 3$) such that for every $A \subseteq [n]$ it holds that*

1. *if $|A| \leq 2$ then A contains at most 1 set from $\mathcal{S}$,*
2. *if $|A| \geq 3$ then A contains at least 2 sets from $\mathcal{S}$.*

Then $|\mathcal{S}| \geq \binom{n-1}{2} + 1$. Furthermore, there is an $\mathcal{S}$ of this size with the required properties.

Proof. A set of subsets $\mathcal{S}$ of size $\binom{n-1}{2} + 1$ with the required properties is given set by the singleton $\{1\}$ and all pairs $\{i, j\}$ with $2 \leq i, j \leq n$.

In order to prove the lower bound, we note that it may be assumed that $\mathcal{S}$ contains no sets of size ≥ 4, as Condition 2 is satisfied for every A if it is satisfied for every A of size 3, and in this case sets of size ≥ 4 are "of no help". Furthermore, Condition 1 implies that $\mathcal{S}$ contains at most one singleton.

It may also be assumed that $\mathcal{S}$ contains no triples. To see this, assume that the triple $\{i, j, k\}$ occurs in $\mathcal{S}$, one or more times. If $\{i, j, k\}$ contains at least two sets of size ≤ 2 from $\mathcal{S}$ then all its copies can be deleted from $\mathcal{S}$ without violating Conditions 1 and 2. If it contains exactly one set of size

≤ 2 from $\mathcal{S}$, then we distinguish two cases. If this set is a singleton, say $\{i\}$, then we can delete all copies of $\{i,j,k\}$ and add the pair $\{j,k\}$ to $\mathcal{S}$. If this set is a pair, say $\{i,j\}$, then we can delete all copies of $\{i,j,k\}$ and add the pair $\{i,k\}$ to $\mathcal{S}$. In both cases, Conditions 1 and 2 continue to hold and the size of $\mathcal{S}$ is not increased. If $\{i,j,k\}$ does not contain any set of size ≤ 2 form $\mathcal{S}$, then we can delete all its copies and add the pairs $\{i,j\}$ and $\{i,k\}$ to $\mathcal{S}$. Again, Conditions 1 and 2 are not violated, and the size of $\mathcal{S}$ is not increased as it must have contained at least 2 copies of $\{i,j,k\}$. Repeating this process, we can eliminate all the triples from $\mathcal{S}$.

Now, if $\mathcal{S}$ contains exactly one singleton $\{i\}$, then Condition 1 implies that it cannot contain any pair $\{i,j\}$. By Condition 2, $\mathcal{S}$ must contain ≥ 2 sets from every triple $\{i,j,k\}$. Hence $\mathcal{S}$ must contain *all* the pairs $\{j,k\}$ where j and k are different from i, and so $|\mathcal{S}| \geq \binom{n-1}{2} + 1$.

Otherwise $\mathcal{S}$ contains only pairs. By Condition 2, there cannot be any two pairs $\{i,j\}$ and $\{j,k\}$ missing from $\mathcal{S}$. Thus the pairs missing from $\mathcal{S}$ must be pairwise disjoint, and so their number is at most $\lfloor n/2 \rfloor$. Hence in this case

$$|\mathcal{S}| \geq \binom{n}{2} - \left\lfloor \frac{n}{2} \right\rfloor \geq \binom{n-1}{2} + 1$$

if $n \geq 3$. □

The next two lemmas are used in the proof of Theorem 8.

Lemma 11 *Let $\mathcal{S}$ be a collection of subsets of $[n]$ such that for every $A \subseteq [n]$ it holds that*

1. *if $|A| \leq 3$ then A contains at most 2 sets from $\mathcal{S}$,*
2. *if $|A| \geq 4$ then A contains at least 3 sets from $\mathcal{S}$.*

Then $|\mathcal{S}| \geq (\frac{1}{87} + o(1))n^3$. Furthermore, there is an $\mathcal{S}$ of size $(\frac{1}{24} + o(1))n^3$ with the required properties.

Proof. A set of subsets $\mathcal{S}$ of size $(\frac{1}{24} + o(1))n^3$ with the required properties is given by the set of triples $\{i,j,k\}$ with either $1 \leq i,j,k \leq \lfloor n/2 \rfloor$ or $\lfloor n/2 \rfloor + 1 \leq i,j,k \leq n$ and the set of pairs $\{i,j\}$ with $1 \leq i \leq \lfloor n/2 \rfloor$, $\lfloor n/2 \rfloor + 1 \leq j \leq n$. In other words, $\mathcal{S}$ consists of a complete bipartite graph with half the vertices on each side, and all possible triples contained in each side.

For the lower bound note that Condition 1 implies that $\mathcal{S}$ can contain at most two singletons. Deleting these, the remaining subsets on $\geq n-2$

elements still satisfy Conditions 1 and 2, and it is sufficient to prove the bound for these subsets. Thus it may be assumed that the collection $\mathcal{S}$ of subsets of $[n]$ contains no singletons.

Let the number of pairs, triples and quadruples in $\mathcal{S}$ be e_2, e_3 and e_4, respectively. (Each pair, triple or quadruple is counted with its multiplicity.) We now count how many times each pair, triple and quadruple in $\mathcal{S}$ can be included in some 4-subset of $[n]$. As any 4-subset of $[n]$ must contain at least 3 subsets from $\mathcal{S}$ by Condition 2, we get

$$e_2\binom{n-2}{2} + e_3(n-3) + e_4 \geq 3\binom{n}{4} = \left(\frac{1}{8} + o(1)\right) n^4.$$

Now if $e_4 \geq n^{7/2}$ then we are done. Otherwise it holds that

$$e_2\binom{n-2}{2} + e_3(n-3) \geq \left(\frac{1}{8} + o(1)\right) n^4.$$

We distinguish two cases.

Case 1 $e_2 < 0.226n^2$.
Then

$$\begin{aligned} e_3 &\geq \left(\left(\frac{1}{8} + o(1)\right) n^4 - 0.226n^2\binom{n-2}{2}\right)/(n-3) \\ &= \left(\left(\frac{1}{8} + o(1)\right) n^4 - (0.113 + o(1))n^4\right)/(n-3) \\ &= (0.012 + o(1))n^3 \\ &> \left(\frac{1}{87} + o(1)\right) n^3, \end{aligned}$$

and the lower bound follows.

Case 2 $e_2 \geq 0.226n^2$.

By Condition 1, every pair may occur at most twice in $\mathcal{S}$. If a pair $\{i,j\}$ occurs twice in $\mathcal{S}$, then, again by Condition 1, there cannot be any other pair $\{i,k\}$ in $\mathcal{S}$, where $j \neq k$. Hence the number of pairs that occur twice in $\mathcal{S}$ is at most $n/2$. After deleting these pairs, we are still left with $(0.226 + o(1))n^2$ pairs. Let us consider the graph formed by the remaining pairs. Let d be its maximal degree, let v be a vertex of degree d, and let $N(v)$ be the set of neighbors of v. As

$$\frac{dn}{2} \geq (0.226 + o(1))n^2,$$

it follows that $d \geq (0.452 + o(1))n$. Now, by Condition 1, no pair in $\mathcal{S}$ can join two neighbors of v. Hence, if f_3 and f_4 denote the number of triples and quadruples, respectively, of $\mathcal{S}$ contained in $N(v)$, then Condition 2 implies

$$f_3(d-3) + f_4 \geq 3\binom{d}{4} = \left(\frac{1}{8} + o(1)\right) d^4.$$

As $f_4 \leq e_4 < n^{7/2} = O(d^{7/2})$, it follows that

$$\begin{aligned} f_3 &\geq \left(\frac{1}{8} + o(1)\right) d^4/(d-3) \\ &\geq \left(\frac{1}{8} + o(1)\right) d^3 \\ &\geq \left(\frac{1}{8} + o(1)\right) 0.452^3 n^3 \\ &\geq (0.0115 + o(1))n^3 \\ &> \left(\frac{1}{87} + o(1)\right) n^3 \ , \end{aligned}$$

again implying the lower bound. □

Lemma 12 *Let $\mathcal{S}$ be a collection of subsets of $[n]$ such that for every $A \subseteq [n]$ it holds that*

1. *if $|A| \leq 3$ then A contains at most 1 set from $\mathcal{S}$,*
2. *if $|A| \geq 4$ then A contains at least 2 sets from $\mathcal{S}$.*

Then $|\mathcal{S}| \geq (\frac{1}{24} + o(1))n^3$.

Proof. Again, we may assume that $\mathcal{S}$ contains no singletons. Now Condition 1 implies that every pair or triple may occur at most once. Also by Condition 1, any two pairs must be disjoint, hence there can be at most $n/2$ pairs. Thus the number of 4-subsets containing ≥ 2 pairs is $\leq \binom{n/2}{2}$. The remaining 4-subsets must contain at least one triple or quadruple. If the number of triples and quadruples in $\mathcal{S}$ is e_3 and e_4, respectively, then

$$e_3(n-3) + e_4 \geq \binom{n}{4} - \binom{n/2}{2} = \left(\frac{1}{24} + o(1)\right) n^4.$$

So either it holds that $e_4 \geq n^{7/2}$ or it holds that $e_3 \geq (\frac{1}{24} + o(1))n^3$, proving the claim in both cases. □

The final lemma is used in the proof of Theorem 9.

Lemma 13 *Let $k \geq 3$ be fixed. Let $\mathcal{S}$ be a set of subsets of $[n]$ such that for every $A \subseteq [n]$ it holds that*

1. *if $|A| \leq k$ then A contains fewer than k sets from $\mathcal{S}$,*
2. *if $|A| > k$ then A contains at least 1 set from $\mathcal{S}$.*

Then $|\mathcal{S}| = \Omega(n^k)$. Furthermore, there is a collection of size $O(n^k)$ with the required properties.

Proof. The set of all k-subsets of $[n]$ gives the upper bound.

In order to prove the lower bound, we note again that by Condition 1, $\mathcal{S}$ contains at most $k-1$ singletons. Deleting these elements, we get a set of size $\geq n-k+1$, for which Conditions 1 and 2 still hold. It is sufficient to prove the bound for this set. Thus we may assume that $\mathcal{S}$ is a set of subsets of $[n]$ containing no singletons.

As $\binom{k}{r} \geq k$ for $k \geq 3$ and $2 \leq r \leq k-1$, Condition 1 implies the following

Fact For every r, $2 \leq r \leq k-1$, there is no subset A of size k such that all r-subsets of A are in $\mathcal{S}$.

Ramsey's theorem [8] for two colors states that for every $r \geq 2$, i and j there is a number $R^r(i,j)$ such that if $\mathcal{H}$ is any set of r-subsets of a set of size at least $R^r(i,j)$, then either there are i elements all of whose r-subsets are in $\mathcal{H}$, or there are j elements all of whose r-subsets are not in $\mathcal{H}$.

The outline of the argument is the following. By a repeated application of Ramsey's theorem and the Fact above it follows that every sufficiently large subset of $[n]$ contains a set of size $k+1$ that contains no set of size $\leq k-1$ from $\mathcal{S}$. By Condition 2, this set must contain a set of size k or $k+1$ from $\mathcal{S}$. Hence, the sets of size k and $k+1$ in $\mathcal{S}$ must be "dense" in some sense, which implies the lower bound.

More precisely, let us define the numbers $N_{k-2}, N_{k-3}, \ldots, N_2$ and N_1 by

$$\begin{aligned} N_{k-2} &= R^{k-1}(k, k+1), \\ N_i &= R^{i+1}(k, N_{i+1}) \text{ for } i = k-3, \ldots, 2, 1, \end{aligned}$$

and let $N = N_1$.

We show that every N-subset of $[n]$ contains a $(k+1)$-subset that contains no set of size $\leq k-1$ from $\mathcal{S}$. Consider an arbitrary N-subset B of $[n]$.

Claim For every j, $2 \leq j \leq k-2$, B has a subset B_j of size N_j that contains no subset of size $\leq j$ from $\mathcal{S}$.

The claim is proved by induction on j. For $j = 2$, we use $N = N_1 = R^2(k, N_2)$. From the Fact above, no k-element subset of B has all its pairs

in $\mathcal{S}$. Hence, by Ramsey's theorem, B must have a subset B_2 of size N_2 that contains no pair from $\mathcal{S}$. For the induction step, assume that the subset B_j of size N_j has no subset of size $\leq j$ in $\mathcal{S}$. The Fact implies again that no k-element subset of B_j has all its $(j+1)$-subsets in $\mathcal{S}$. Hence, as $N_j = R^{j+1}(k, N_{j+1})$, we get that B_j has a subset B_{j+1} of size N_{j+1} that does not contain any set of size $\leq j+1$ from $\mathcal{S}$. Thus B has a subset B_{k-2} of size N_{k-2} that does not contain any set of size $\leq k-2$ from $\mathcal{S}$, proving the Claim.

As $N_{k-2} = R^{k-1}(k, k+1)$, applying the Fact for $r = k-1$, it follows that B contains a $(k+1)$-subset B_{k-1} that contains no set of size $\leq k-1$ from $\mathcal{S}$. Now, by Condition 2, B_{k-1} has a subset from $\mathcal{S}$. This subset must have size k or $k+1$. Hence we showed that every N-subset of $[n]$ contains a set of size k or $k+1$ from $\mathcal{S}$.

Let $\mathcal{S}_k$, resp. $\mathcal{S}_{k+1}$, denote the number of k-element, respectively $(k+1)$-element subsets in $\mathcal{S}$. Then

$$\begin{aligned}
\binom{n}{N} &\leq |\mathcal{S}_k| \binom{n-k}{N-k} + |\mathcal{S}_{k+1}| \binom{n-k-1}{N-k-1} \\
&\leq (|\mathcal{S}_k| + |\mathcal{S}_{k+1}|) \binom{n-k}{N-k} \\
&\leq |\mathcal{S}| \binom{n-k}{N-k},
\end{aligned}$$

using

$$\binom{n-k-1}{N-k-1} \leq \binom{n-k}{N-k}.$$

Hence, noting that N is a constant depending only on k, we get

$$|\mathcal{S}| \geq \frac{\binom{n}{N}}{\binom{n-k}{N-k}} = \frac{n(n-1)\cdots(n-k+1)}{N(N-1)\cdots(N-k+1)} = \Omega\left(n^k\right).$$

□

We note that Lemma 11 follows from Lemma 13, but the constant would be 1/504 instead of 1/87.

5 Monotone perceptrons

It was noted in Proposition 1 that matrices with threshold 1 are equivalent to monotone CNFs. This correspondence can be generalized to the case

of an arbitrary threshold by replacing CNFs with a more general class of circuits.

A *threshold gate* is a gate computing a function T_t^m for some m and t. A *perceptron* is a depth-2 circuit consisting of $\wedge$'s of variables or their negations at the bottom, followed by a threshold gate. If the inputs to the $\wedge$ gates are all unnegated variables then the perceptron is *monotone*. Perceptrons form an important class of neural networks [2, 20].

Depth-2 circuits consisting of $\vee$'s of variables or their negations, followed by a threshold gate can be called *dual perceptrons*. If the inputs to the $\vee$ gates are all unnegated variables then the dual perceptron is *monotone*.

This terminology is justified by the following fact. The *dual* of a Boolean function $f(x_1, \ldots, x_n)$ is $\overline{f(\bar{x}_1, \ldots, \bar{x}_n)}$. For example, the dual of $T_k^m(x_1, \ldots, x_m)$ is $T_{m-k+1}^m(x_1, \ldots, x_m)$.

Proposition 14 *Let C be a monotone perceptron having m $\wedge$'s at the bottom and a threshold gate T_{m-t+1}^m as the final gate. Then the dual of the function computed by C is computed by the monotone dual perceptron obtained from C by replacing the $\wedge$-gates by $\vee$-gates and replacing the final gate by T_t^m. Conversely, let D be a monotone dual perceptron having m $\vee$'s at the bottom and a threshold gate T_{m-t+1}^m as the final gate. Then the dual of the function computed by D is computed by the monotone perceptron obtained from D by replacing the $\vee$-gates by $\wedge$-gates and replacing the final gate by T_t^m.*

The correspondence between matrices with threshold 1 and monotone CNFs generalizes to the general case as follows.

Proposition 15 *A monotone Boolean function f is represented by some m-row matrix with threshold t if and only if f can be computed by a monotone dual perceptron having m $\vee$'s at the bottom and a threshold gate T_{m-t+1}^m as the final gate.*

Proof. The two-way simulation is the same as for Proposition 1, with the $\vee$-gates corresponding to the 0-sets of the rows. □

With the translation given by Propositions 14 and 15, Theorem 5 gives an exponential lower bound for monotone perceptrons, showing, for example, that they cannot always simulate CNFs polynomially.

Theorem 16 *Every monotone perceptron computing the function $g_n(x_1, \ldots, x_n, y_1, \ldots, y_n) = \bigwedge_{i=1}^n (x_i \vee y_i)$ has at least 2^n $\wedge$-gates.*

We also get the following size-threshold trade-off as a direct consequence of Theorems 7, 8 and 9. It can also proved directly using Lemmas 10, 11, 12 and 13.

Theorem 17 *Let $t \geq 3$ be a fixed constant. Then T_t^n is computed by a monotone perceptron consisting of the single threshold gate T_t^n. The minimal number of gates required by a monotone perceptron that computes T_t^n with threshold $< t$ is $\Theta(n^{t-1})$.*

6 Listing frequent sets

A possible objective for knowledge discovery and data mining is to *produce a list of potentially "interesting" relationships that are "true" in the database* [1, 9, 12, 18, 19]. It is assumed that the user, who may be the manager of the department store, has some notion of "interestingness", based on which he will select some of these relationships as truly interesting. For example, noticing that beer and pretzels are often sold together, he can use this information to design sales, product location, etc. Frequent sets, besides providing an example of a notion of interestingness, can also be used as the computational basis for computing other kinds of relationships, such as association rules [1, 18]. As mentioned above, one of the algorithmic problems arising in this context is that of efficient listing of a class of objects.

In particular, there are many algorithms given for listing all maximal frequent sets of a matrix with a given threshold, making use of the fact that the family of frequent sets is closed under inclusion [1, 19]. Thus these algorithms are in fact general methods for listing the maximal 0-vectors of monotone Boolean functions based on function-evaluation queries. (A function-evaluation or membership query in this context corresponds to checking if a given set of columns is an infrequent set.) In this general framework one can prove lower bounds for the complexity of listing algorithms in terms of the size of the *boundary* of a monotone Boolean function (the boundary consists of the maximal 0-vectors and the minimal 1-vectors) [9, 12, 19]. On the other hand, there may exist efficient algorithms that make use of additional information provided by the matrix. For example, let us consider the $n \times 2n$ matrix $J_n J_n$ with threshold 1. This matrix has only n maximal frequent sets, but it has 2^n minimal infrequent sets. Thus, although it has only a few maximal frequent sets, it has a large boundary, and therefore the general algorithms would run for a long time to produce a small output. As the matrix itself has only a few rows, it may be possible that this extra information could be used to get a faster algorithm. As a first step in

this direction, it is of some interest to know the restrictions imposed on the family of frequent sets by the size of the matrix and the threshold, and this leads to the questions studied in this paper.

The question whether the maximal frequent sets of matrix with a given threshold can be listed with polynomial delay in terms of the size of the matrix, appears to be an important open problem that could also be of interest from the point of view of practical applications. (The definitions of different efficiency criteria for listing algorithms are given in, e.g., [5, 7, 14].) In the rest of this section we mention some related listing problems, which are known to be easy or hard in some sense.

Perhaps the central problem in the area is that of listing the maximal 0-vectors of a monotone disjunctive normal form, which is equivalent to finding the minimal hitting sets (or transversals) of a hypergraph and many other listing problems [3, 5, 21]. A recent important result of Fredman and Khachiyan [6] shows that this can be done with incremental quasipolynomial time. Listing the maximal 0-vectors of a monotone CNF can be done trivially in polynomial delay, as these are the complements of the minimal characteristic vectors of the clauses.

Disjunctive normal forms and CNFs are depth 2 formulas. On the other hand, listing the maximal 0-vectors of a depth-3 monotone Boolean formula is provably difficult. This was essentially proved by Lawler, Lenstra and Rinnooy Kan [17] and Gurvich and Khachiyan [10]. The first of those two articles formulates its result in terms of independence systems, without specifying the complexity of the formulas needed to implement these systems. The second article shows the hardness of a related, but somewhat different problem. Nevertheless, both constructions actually imply the hardness of this problem, and therefore we formulate both of them below, without proving the correctness claims. The negative result applies to the "most liberal" definition of an efficient listing algorithm, where one requires the algorithm only to be polynomial in the total size of the input and the final output, called polynomial total time [7, 14].

Theorem 18 *If $P \neq NP$, then there is no algorithm running in polynomial total time that lists all maximal 0-vectors of depth-3 monotone Boolean formulas.*

Proof. We show that such an algorithm $\mathcal{A}$ could be used to decide CNF-unsatisfiability in polynomial time. Let $\varphi = C_1 \wedge \cdots \wedge C_m$ be a CNF expression in the variables $x_1, \ldots, x_n$. WLOG, we assume that no clause is a subset of another.

Let us introduce new variables $y_1, \ldots, y_n$, and let

$$\varphi^* = C'_1 \wedge \cdots \wedge C'_m$$

be obtained from φ by replacing $\bar{x}_i$ by y_i for every $i = 1, \ldots, n$.

Also, let $\psi = D_1 \vee \ldots \vee D_m$ be the negation of φ in disjunctive normal form obtained by using the De Morgan laws, and let

$$\psi^* = D'_1 \vee \cdots \vee D'_m$$

be obtained from ψ by replacing $\bar{x}_i$ by y_i for every $i = 1, \ldots, n$.

Construction 1 (Lawler–Lenstra–Rinnooy Kan [17])

$$\Phi_1 = \left(\psi^* \vee \bigvee_{i=1}^{n} (x_i \wedge y_i) \right) \wedge \bigwedge_{i=1}^{n} (x_i \vee y_i).$$

Claim. Φ_1 has at least n maximal 0-vectors, and φ is unsatisfiable if and only if Φ_1 has exactly n maximal 0-vectors.

Construction 2 (Gurvich–Khachiyan [10])

$$\Phi_2 = \varphi^* \wedge \bigvee_{i=1}^{n} (x_i \wedge y_i).$$

Claim. Φ_2 has at least m maximal 0-vectors, and φ is unsatisfiable if and only if Φ_2 has exactly m maximal 0-vectors.

Based on either one of these claims, the satisfiability of φ can be decided by running $\mathcal{A}$ for a polynomial number of steps and checking its output. □

Now, using the circuit terminology of the previous section, the problem of listing all maximal frequent sets of a matrix with a given threshold, asks for listing the maximal 0-vectors of a monotone dual perceptron. This class of circuits is more general than CNFs, but it is *incomparable* in its computational power to depth-3 formulas. In one direction this follows from Theorem 5 which shows that even monotone disjunctive normal forms cannot be polynomially simulated by monotone dual perceptrons. In the other direction this follows from results in circuit complexity theory showing that the majority function cannot be computed by any $\{\wedge, \vee\}$-circuit of bounded depth and polynomial size [25].

7 Further remarks and open problems

The problem of listing all *minimal infrequent sets* is also interesting, although it is less important from the point of view of practical applications. An equivalent formulation of this problem is the following: given a collection $\mathcal{S}$ of m subsets of $[n]$ and a number k, list all minimal subsets that intersect at least k subsets from $\mathcal{S}$. Thus, this problem generalizes the minimal hitting set problem, where $k = m$, and it would be interesting to know if it can still be solved in incremental quasipolynomial time. It follows from a general result of Gurvich and Khachiyan [10] that the maximal frequent sets and the minimal infrequent sets can be listed *together* in incremental quasipolynomial time in the size of the matrix.

In the special case when $\mathcal{S}$ is a set and every subset in $\mathcal{S}$ has size 2, the problem specializes to listing all minimal vertex sets of a graph that cover at least k edges, or, turning to the complements of the vertex sets, to listing all maximal vertex sets that contain at most $m-k$ edges. This problem, which may be called the "maximal fairly independent set problem," generalizes the problem of listing all maximal independent sets in a graph, which can be solved with polynomial delay [14, 24]. Can maximal fairly independent sets also be listed with polynomial delay?

Besides frequent sets, there are several other interesting notions of interestingness, such as those corresponding to various statistical tests, studied in the *GUHA* project (see, e.g. Hájek and Havránek [12]). One could study the representational and listing complexity aspects of these notions as well. Some related complexity questions are discussed in Pudlák and Springsteel [22].

Concerning the combinatorial problem discussed in Section 4.1, it would be interesting to extend the results to cases when k may grow with n and to get sharper bounds for constant k. The proof of Lemma 13 provides very small constants because of the repeated application of Ramsey's theorem.

References

[1] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. Fast discovery of association rules. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, pages 307–328. AAAI Press, Menlo Park, California, 1996.

[2] R. Beigel, N. Reingold, and D. Spielman. The perceptron strikes back (preliminary report). In *Proceedings of the Sixth Annual Structure in Complexity Theory Conference*, pages 286–291, 1991.

[3] J. C. Bioch and T. Ibaraki. Complexity of identification and dualization of positive Boolean functions. *Information and Computation*, 123:50–63, 1995.

[4] B. Bollobás. *Combinatorics: Set Systems, Hypergraphs, Families of Vectors and Combinatorial Probability.* Cambridge University Press, 1986.

[5] T. Eiter and G. Gottlob. Identifying the minimal transversals of a hypergraph and related problems. *SIAM J. Comput.*, 24:1278–1304, 1995.

[6] M. L. Fredman and L. Khachiyan. On the complexity of dualization of monotone disjunctive normal forms. *Journal of Algorithms*, 21:618–628, 1996.

[7] L. A. Goldberg. *Efficient Algorithms for Listing Combinatorial Objects.* Distinguished Dissertations in Computer Science. Cambridge University Press, 1993.

[8] R. L. Graham, B. L. Rothschild, and J. H. Spencer. *Ramsey Theory.* Interscience Series in Discrete Mathematics. Wiley, 1980.

[9] D. Gunopulos, R. Khardon, H. Mannila, and H. Toivonen. Data mining, hypergraph transversals, and machine learning. In *Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 12–15, 1997.

[10] V. Gurvich and L. Khachiyan. On generating the irredundant conjunctive and disjunctive normal forms of monotone Boolean functions. RUTCOR Research Report RRR 35-95, Rutgers Center for Operations Research, 1997. Also available as LCSR-TR-251, Dept. Computer Science, Rutgers University, 1995. To appear in *Discrete Applied Math.*

[11] V. Gurvich and L. Khachiyan. On the frequency of the most frequently occurring variable in dual monotone DNFs. *Discrete Math.*, 169:245–248, 1997.

[12] P. Hájek and T. Havránek. *Mechanizing Hypothesis Formation: Mathematical Foundations for a General Theory.* Springer, 1978.

[13] A. Hajnal, W. Maass, P. Pudlák, M. Szegedy, and G. Turán. Threshold circuits of bounded depth. *Journal of Computer and System Sciences*, 46:129–154, 1993.

[14] D. S. Johnson, M. Yannakakis, and C. H. Papadimitriou. On generating all maximal independent sets. *Inf. Process. Lett.*, 27:119–123, 1988.

[15] S. Jukna. Computing threshold functions by depth-3 threshold circuits with smaller thresholds of their gates. *Inf. Process. Lett.*, 56:147–150, 1995.

[16] G. O. H. Katona, T. Nemetz, and M. Simonovits. On a problem of Turán in the theory of graphs. *Mat. Lapok*, 15(228–238), 1964. (In Hungarian).

[17] E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Generating all maximal independent sets: NP-hardness and polynomial-time algorithms. *SIAM J. Comput.*, 9(3):558–565, 1980.

[18] H. Mannila and H. Toivonen. Multiple uses of frequent sets and condensed representations. In *Second International Conference on Knowledge Discovery and Data Mining*, pages 189–194, 1996.

[19] H. Mannila and H. Toivonen. Levelwise search and borders of theories in knowledge discovery. Series of Publications C C-1997-8, University of Helsinki, Dept. Computer Science, 1997.

[20] M. Minsky and S. Papert. *Perceptrons*. MIT Press, Cambridge, MA, 1969.

[21] N. Mishra and L. Pitt. Generating all maximal independent sets of bounded-degree hypergraphs. In *Proc. 10th Annu. Conf. on Comput. Learning Theory*, pages 211–217. ACM Press, New York, NY, 1997.

[22] P. Pudlák and F. N. Springsteel. Complexity of mechanized hypothesis formation. *Theoret. Comput. Sci.*, 8:203–225, 1979.

[23] K.-Y. Siu, V. Roychowdhury, and T. Kailath. *Discrete Neural Computation: A Theoretical Foundation*. Prentice Hall, 1995.

[24] S. Tsukiyama, M. Ide, H. Ariyoshi, and I. Shirakawa. A new algorithm for generating all the maximal independent sets. *SIAM J. Comput.*, 6(3):505–517, 1977.

[25] I. Wegener. *The Complexity of Boolean Functions.* Wiley–Teubner, 1987.

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'