On Learning Random DNF Formulas under the Uniform Distribution

Jeffrey C. Jackson[1]* and Rocco A. Servedio[2]

[1] Department of Mathematics and Computer Science, Duquesne University
Pittsburgh, PA 15282
jackson@mathcs.duq.edu
[2] Department of Computer Science, Columbia University
New York, NY 10027, USA
rocco@cs.columbia.edu

Abstract. We study the average-case learnability of DNF formulas in the model of learning from uniformly distributed random examples. We define a natural model of random monotone DNF formulas and give an efficient algorithm which with high probability can learn, for any fixed constant $\gamma > 0$, a random t-term monotone DNF for any $t = O(n^{2-\gamma})$. We also define an analogous model of random nonmonotone DNF and give an efficient algorithm which with high probability can learn a random t-term DNF for any $t = O(n^{3/2-\gamma})$. Our results have implications for the construction of cryptographic primitives from hard learning problems as proposed by Blum *et al.* [3].

1 Introduction

A *disjunctive normal form* formula, or DNF, is an AND of ORs of Boolean literals. A longstanding open question in computational learning theory is whether efficient algorithms exist for learning polynomial size DNF formulas. Many authors have studied this question in different learning models such as the model of exact learning from membership and equivalence queries [2, 8], the distribution-free PAC learning model [5, 11], the uniform distribution model [17, 16] and the model of uniform distribution learning with membership queries [13, 9].

Our focus is on learning DNF formulas under the uniform distribution (without membership queries). In 1990 Verbeurgt [17] gave an algorithm which can learn any poly(n)-size DNF in this model in time $n^{O(\log n)}$. No faster algorithms are known, and the question of whether poly(n)-time algorithms exist is now widely viewed as an important open problem. Blum *et al.* [3] showed that no algorithm which can be recast in the Statistical Query model can learn arbitrary polynomial-size DNF under the uniform distribution in $n^{o(\log n)}$ time.

The problem of learning *monotone* DNF formulas under uniform has also been much studied over the past decade [6, 7, 12, 15, 16, 18]). An algorithm is

* This material is based upon work supported by the National Science Foundation under Grant No. CCR-0209064.

known which learns any $2^{\sqrt{\log n}}$-term monotone DNF in poly(n) time [16], but no algorithm faster than that of [17] is known for arbitrary poly(n)-size monotone DNF. (The negative results of Blum *et al.* do not apply for monotone DNF.)

Learning Random DNF formulas: Motivation and Background. In this paper we study DNF learning from an average-case perspective, i.e. we consider the problem of learning *random* DNF formulas. One natural motivation for pursuing such a study is that since the problem is interesting and important but the worst-case version seems quite hard, it is natural to consider the average case. Additional motivation comes from related work in learning theory:

- Aizenstein & Pitt [1] posed the question of whether random DNF formulas are efficiently learnable. They proposed a model of random DNF in which each of the DNF's t terms is selected independently at random from all possible terms, and gave a membership and equivalence query algorithm which with high probability learns a random DNF generated in this way. As noted in [1], a limitation of this model is that with very high probability all terms will have length $\Omega(n)$. They also proposed another model which is parameterized by the (expected) length k of each term as well as the number of terms t, and asked whether random DNF can be efficiently learned in such a model. Our work considers a very similar model and gives an efficient uniform distribution algorithm for many interesting values of k and t.
- Blum *et al.* [3] considered the possibility of constructing cryptographic primitives such as pseudorandom generators and one-way functions based on the presumed intractability of certain learning problems. They defined an average-case model of learning in which examples are drawn from the uniform distribution and the target concept is drawn from some probability distribution over the concept class. They prove (Theorem 3.3 of [3]) that the existence of concept classes which are hard to learn in this model implies the existence of corresponding one-way functions whose circuit complexity is closely related to the circuit complexity of the concepts in the class. The motivation of Blum *et al.* was that since there are learning problems which seem hard yet have very low circuit complexity, it is possible to thus obtain cryptographic primitives with low circuit complexity.
  The learning model which we consider corresponds exactly to this framework of Blum *et al.*: we consider uniform distribution learning of DNFs which are selected according to a natural probability distribution over DNF formulas. Our positive learning results thus indicate that certain natural approaches to constructing cryptographic primitives along the lines suggested by Blum *et al.* are in fact not secure.
- Finally, the current work is similar in spirit to recent work by the authors on learning random decision trees of logarithmic depth under the uniform distribution [10]. We note that the algorithm in [10] works by constructing a decision tree hypothesis, and its proof of correctness depends heavily on Fourier properties specific to log-depth DTs. Thus we must use a very different algorithm and analysis in the current paper.

Our Results. We consider the following natural model for a random t-term k-DNF: each of the t terms is selected independently and uniformly from the set of all terms of length exactly k. We also consider a monotone version of the model in which each term is required to be monotone.

Our main results are polynomial time algorithms which with high probability (over the choice of target function as well as the choice of examples) will successfully learn random DNF generated from these models for a fairly wide range of values of k and t. (As we discuss later, for a given value of t there is only a small range of values of k which are interesting for uniform distribution learning, so in the rest of this section we only discuss t.) In more detail, for the monotone model our algorithm can learn t-term monotone DNF for any $t = O(n^{2-\gamma})$ where $\gamma > 0$; this algorithm can achieve any error rate $\epsilon > 0$ in $\mathrm{poly}(n, 1/\epsilon)$ time with high probability. For the general (nonmonotone) model, our algorithm can learn t-term DNF for any $t = O(n^{\frac{3}{2}-\gamma})$; this algorithm cannot achieve arbitrarily small error but can achieve error $\epsilon = o(1)$ for any $t = \omega(1)$. Detailed statements of our results are given in Theorems 3 and 6.

Our algorithms work in two stages: we first identify pairs of variables which cooccur in some term of the target DNF, and then use these pairs to reconstruct terms. For monotone DNF we can with high probability exactly identify those pairs of variables which cooccur in some term. For nonmonotone DNF with high probability we can identify most pairs of variables which cooccur in some term; this enables us to learn to fairly (but not arbitrarily) high accuracy.

We give preliminaries in Section 2. Section 3 and 4 contain our results for monotone and nonmonotone DNF respectively. Section 5 concludes.

2 Preliminaries

We first describe our models of random monotone and nonmonotone DNF. Let $\mathcal{M}_n^{t,k}$ be the probability distribution over monotone t-term DNF induced by the following random process: each term is independently and uniformly chosen at random from all $\binom{n}{k}$ monotone ANDs of size exactly k over variables $v_1, \ldots, v_n$. For nonmonotone DNF, we write $\mathcal{D}_n^{t,k}$ to denote the following probability distribution over t-term DNF: first a monotone DNF is selected from $\mathcal{M}_n^{t,k}$, and then each occurrence of each variable in each term is independently negated with probability $1/2$. (Equivalently, a draw from $\mathcal{D}_n^{t,k}$ is obtained by independently selecting t terms from the set of all terms of length exactly k).

Given a Boolean function $\phi : \{0,1\}^n \to \{0,1\}$, we write $\Pr[\phi]$ to denote $\Pr_{x \sim U_n}[\phi(x) = 1]$, where U_n denotes the uniform distribution over $\{0,1\}^n$. We write log to denote $\log_2$ and ln to denote natural log.

We use the following Chernoff bound [Theorem A.12, Alon & Spenser]: Let $B(t,p)$ denote the binomial distribution with parameter p, i.e. a draw from $B(t,p)$ is a sum of t independent p-biased 0/1 Bernoulli trials. Then for $\beta > 1$,

$$\Pr_{S \sim B(t,p)}[S \geq \beta pt] \leq \left(e^{\beta-1}\beta^{-\beta}\right)^{pt} < (e/\beta)^{\beta pt}.$$

The following bound will also be useful:

McDiarmid bound [14]: *Let $X_1, \ldots X_m$ be independent random variables taking values in a set Ω. Let $F\colon \Omega^m \to \mathbf{R}$ be such that for all $i \in [m]$ we have*

$$|F(x_1, \ldots, x_m) - F(x_1, \ldots, x_{i-1}, x_i', x_{i+1}, \ldots, x_m)| \leq c_i$$

for all $x_1, \ldots, x_m$ and x_i' in Ω. Let $\mu = \mathbf{E}[F(X_1, \ldots, X_m)]$. Then for all $\tau > 0$,

$$\Pr\left[|F(X_1, \ldots, X_m) - \mu| > \tau\right] < \exp(-\tau^2/(c_1^2 + \cdots + c_m^2)).$$

Finally, we remind the reader that in the uniform distribution learning model the learner is given a source of labeled examples $(x, f(x))$ where each x is uniformly drawn from $\{0,1\}^n$ and f is the unknown function to be learned. The goal of the learner is to efficiently construct a hypothesis h which with high probability has low error relative to f under the uniform distribution, i.e. $\Pr_{x \sim U_n}[h(x) \neq f(x)] \leq \epsilon$ with probability $1 - \delta$.

3 Learning Random Monotone DNF

3.1 Interesting Parameter Settings

Consider a random draw of f from $\mathcal{M}_n^{t,k}$. It is intuitively clear that if t is too large relative to k then a random $f \in \mathcal{M}_n^{t,k}$ will likely have $\Pr[f] \approx 1$; similarly if t is too small relative to k then a random $f \in \mathcal{M}_n^{t,k}$ will likely have $\Pr[f] \approx 0$. Such cases are not very interesting from a learning perspective since a trivial algorithm can learn to high accuracy. We are thus led to the following definition:

Definition 1. *For $0 < \alpha \leq 1/2$, a pair of values (k,t) is said to be* monotone α-interesting *if $\alpha \leq \mathbf{E}_{f \in \mathcal{M}_n^{t,k}}[\Pr[f]] \leq 1 - \alpha$.*

Throughout the paper we will assume that $0 < \alpha \leq 1/2$ is a fixed constant independent of n and that $t \leq p(n)$, where $p(\cdot)$ is a fixed polynomial (and we will also make assumptions about the degree of p). The following lemma proved in Appendix A gives necessary conditions for (k,t) to be monotone α-interesting: (As Lemma 1 indicates, we may always think of k as being roughly $\log t$.)

Lemma 1. *If (k,t) is monotone α-interesting then $\alpha 2^k \leq t \leq 2^{k+1} \ln \frac{2}{\alpha}$.*

3.2 Properties of $\mathcal{M}_n^{t,k}$

Throughout the rest of Section 3 we assume that $\alpha > 0$ is fixed and (k,t) is a monotone α-interesting pair where $t = O(n^{2-\gamma})$ for some $\gamma > 0$. In this section we develop some useful probabilistic lemmas regarding $\mathcal{M}_n^{t,k}$. All of the proofs, which are relatively straightforward, are given in Appendix B.

Our first lemma does not require that f be drawn from $\mathcal{M}_n^{t,k}$.

Lemma 2. *Any monotone DNF f with $t \geq 2$ terms each of size k has* $\Pr[\overline{f}] \geq \alpha^3$.

Let f^i denote the projected function obtained from f by first removing term T_i from the monotone DNF for f and then restricting all of the variables which were present in term T_i to 1. For $\ell \neq i$ we write T_ℓ^i to denote the term obtained by setting all variables in T_i to 1 in T_ℓ, i.e. T_ℓ^i is the term in f^i corresponding to T_ℓ. Note that if $T_\ell^i \not\equiv T_\ell$ then T_ℓ^i is smaller than T_ℓ.

The next two lemmas show that each variable appears in a limited number of terms and that therefore not too many terms T_ℓ^i in f^i are smaller than their corresponding terms T_ℓ in f. In these and later lemmas, "n sufficiently large" means that n is larger than a constant which depends on α but not on k or t.

Lemma 3. *For n sufficiently large, with probability at least $1 - \delta_{\text{many}} = 1 - n(\frac{ekt^{3/2}\log t}{n2^{k-1}\alpha^2})^{2^{k-1}\alpha^2/\sqrt{t}\log t}$ over the random draw of f from $\mathcal{M}_n^{t,k}$ we have that every variable v_j, $1 \leq j \leq n$, appears in at most $2^{k-1}\alpha^2/\sqrt{t}\log t$ terms of f.*

Note that since (k,t) is a monotone α-interesting pair and $t = O(n^{2-\gamma})$ for some fixed $\gamma > 0$, for sufficiently large n this probability bound is non-trivial.

Lemma 4. *For n sufficiently large, with probability at least $1 - \delta_{\text{small}} = 1 - tk(\frac{ek(t-1)\log t}{n2^k})^{2^k/(\log t)}$ over the random draw of f from $\mathcal{M}_n^{t,k}$ we have that for all $1 \leq i \leq n$ at most $2^k/\log t$ terms T_ℓ^i with $\ell \neq i$ in the projection f^i are smaller than the corresponding terms T_ℓ in f.*

There is probably little overlap between any pair of terms in f:

Lemma 5. *With probability at least $1 - t^2(\frac{k^2}{n})^{\log\log t} = 1 - \delta_{\text{shared}}$ over the random draw of f from $\mathcal{M}_n^{t,k}$, for all $1 \leq i, j \leq t$ no set of $\log\log t$ or more variables belongs to two distinct terms T_i and T_j in f.*

Putting the preceding lemmas together, we can show that for f drawn from $\mathcal{M}_n^{t,k}$, with high probability each term is "uniquely satisfied" by a noticeable fraction of assignments. More precisely, we have:

Lemma 6. *For n sufficiently large, with probability at least $1 - \delta_{\text{many}} - \delta_{\text{small}} = 1 - \delta_{\text{usat}}$ over the random draw of f from $\mathcal{M}_n^{t,k}$, f is such that for all $i = 1, \ldots, t$ we have $\Pr_x[T_i$ is satisfied by x but no other T_j is satisfied by $x] \geq \frac{\alpha^3}{2^{k+2}}$.*

On the other hand, we can upper bound the probability that two terms of a random DNF f will be satisfied simultaneously:

Lemma 7. *With probability at least $1 - \delta_{\text{shared}}$ over the random draw of f from $\mathcal{M}_n^{t,k}$, for all $1 \leq i < j \leq n$, $\Pr[T_i \wedge T_j] \leq \frac{\log t}{2^{2k}}$.*

3.3 Identifying cooccurring variables

In this section we show how to identify pairs of variables (v_i, v_j) which cooccur in some term of f.

First, some notation. Given a monotone DNF f over variables $v_1, \ldots, v_n$, define DNF formulas g_{**}, g_{1*}, g_{*1} and g_{11} over variables $v_3, \ldots, v_n$ as follows:

- g_{**} is the disjunction of the terms in f that contain neither v_1 nor v_2;
- g_{1*} is the disjunction of the terms in f that contain v_1 but not v_2 (but with v_1 removed from each of these terms);
- g_{*1} is defined similarly as the disjunction of the terms in f that contain v_2 but not v_1 (but with v_2 removed from each of these terms);
- g_{11} is the disjunction of the terms in f that contain both v_1 and v_2 (with both variables removed from each term).

We thus have $f = g_{**} \vee (v_1 g_{1*}) \vee (v_2 g_{*1}) \vee (v_1 v_2 g_{11})$. Note that any of g_{**}, g_{1*}, g_{*1}, g_{11} may be an empty disjunction which is identically false.

We can empirically estimate each of the following using uniform random examples $(x, f(x))$:

$$p_{00} := \Pr_x[g_{**}] = \Pr_{x \in U_n}[f(x) = 1 \mid x_1 = x_2 = 0]$$
$$p_{01} := \Pr_x[g_{**} \vee g_{*1}] = \Pr_{x \in U_n}[f(x) = 1 \mid x_1 = 0, x_2 = 1]$$
$$p_{10} := \Pr_x[g_{**} \vee g_{1*}] = \Pr_{x \in U_n}[f(x) = 1 \mid x_1 = 1, x_2 = 0]$$
$$p_{11} := \Pr_x[g_{**} \vee g_{*1} \vee g_{1*} \vee g_{11}] = \Pr_{x \in U_n}[f(x) = 1 \mid x_1 = 1, x_2 = 1].$$

It is clear that g_{11} is nonempty if and only if v_1 and v_2 cooccur in some term of f; thus we would ideally like to obtain $\Pr_{x \in U_n}[g_{11}]$. While we cannot obtain this probability from p_{00}, p_{01}, p_{10} and p_{11}, the following lemma, proved in Appendix C, shows that we can estimate a related quantity:

Lemma 8. *Let P denote the sum $p_{11} - p_{10} - p_{01} + p_{00}$. Then $P = \Pr[g_{11} \wedge \overline{g}_{1*} \wedge \overline{g}_{*1} \wedge \overline{g}_{**}] - \Pr[g_{1*} \wedge g_{*1} \wedge \overline{g}_{**}]$.*

More generally, let P_{ij} be defined as P but with v_i, x_i, v_j, and x_j substituted for v_1, x_1, v_2, and x_2, respectively, throughout the definitions of the g's and p's above. The following lemma shows that, for most random choices of f, for all $1 \leq i, j \leq n$, the value of P_{ij} is a good indicator of whether or not v_i and v_j cooccur in some term of f:

Lemma 9. *For n sufficiently large and $t \geq 4$, with probability at least $1 - \delta_{\text{small}} - \delta_{\text{shared}} - \delta_{\text{usat}}$ over the random draw of f from $\mathcal{M}_n^{t,k}$, we have that for all $1 \leq i, j \leq n$ (i) if v_i and v_j do not cooccur in some term of f then $P_{ij} \leq 0$; (ii) if v_i and v_j do cooccur in some term of f then $P_{ij} \geq \frac{\alpha^4}{8t}$.*

Proof. Part (i) holds for any monotone DNF by Lemma 8. For (ii), we first note that with probability at least $1 - \delta_{\text{many}} - \delta_{\text{small}} - \delta_{\text{usat}}$, a randomly chosen f will have all of the following properties:

1. Each term in f is uniquely satisfied with probability at least $\alpha^3/2^{k+2}$ (by Lemma 6);
2. Each pair of terms T_i and T_j in f are both satisfied with probability at most $\log t/2^{2k}$ (by Lemma 7); and
3. Each variable in f appears in at most $2^{k-1}\alpha^2/\sqrt{t}\log t$ terms (by Lemma 3).

We call such an f *well-behaved*. For the sequel, assume that f is well-behaved and also assume without loss of generality that $i = 1$ and $j = 2$. We consider separately the two probabilities $\rho_1 = \Pr[g_{11} \wedge \overline{g}_{1*} \wedge \overline{g}_{*1} \wedge \overline{g}_{**}]$ and $\rho_2 = \Pr[g_{1*} \wedge g_{*1} \wedge \overline{g}_{**}]$ whose difference defines $P_{12} = P$. By property (1) above, $\rho_1 \geq \alpha^3/2^{k+2}$, since each instance x that uniquely satisfies a term T_j in f containing both v_1 and v_2 also satisfies g_{11} while falsifying all of g_{1*}, g_{*1}, and g_{**}. Since (k,t) is monotone α-interesting, this implies that $\rho_1 \geq \alpha^4/4t$. On the other hand, clearly $\rho_2 \leq \Pr[g_{1*} \wedge g_{*1}]$. By property (2) above, for any pair of terms consisting of one term from g_{1*} and the other from g_{*1}, the probability that both terms are satisfied is at most $\log t/2^{2k}$. Since each of g_{1*} and g_{*1} contains at most $2^{k-1}\alpha^2/\sqrt{t}\log t$ terms by property (3), by a union bound we have $\rho_2 \leq \alpha^4/(4t\log t)$, and the lemma follows given the assumption that $t \geq 4$. □

Thus, our algorithm for finding all of the cooccurring pairs of a randomly chosen monotone DNF consists of estimating P_{ij} for each of the $n(n-1)/2$ pairs (i,j) so that all of our estimates are—with probability at least $1-\delta$—within an additive factor of $\alpha^4/16t$ of their true values. The reader familiar with discrete multivariate Fourier analysis will readily recognize that P_{12} is just $\hat{f}(110_{n-2})$ and that in general all of the P_{ij} are simply second-order Fourier coefficients. Therefore, by the standard Hoeffding bound, a uniform random sample of size $512t^2\ln(n^2/\delta)/\alpha^8$ is sufficient to estimate all of the P_{ij}'s to the specified tolerance with overall probability at least $1-\delta$. This gives us the following:

Theorem 1. *For n sufficiently large and any $\delta > 0$, with probability at least $1 - \delta_{\text{many}} - \delta_{\text{small}} - \delta_{\text{usat}} - \delta$ over the choice of f from $\mathcal{M}_n^{t,k}$ and the choice of random examples, the above algorithm runs in $O(n^2t^2\log(n/\delta))$ time and identifies exactly those pairs (v_i, v_j) which cooccur in some term of f.*

3.4 Forming a hypothesis from pairs of cooccurring variables

In this section we show how to construct an accurate DNF hypothesis for a random f drawn from $\mathcal{M}_n^{t,k}$.

Identifying all k-cliques. By Theorem 1, with high probability we have complete information about which pairs of variables (v_i, v_j) cooccur in some term of f. We thus may consider the graph G with vertices $v_1, \ldots, v_n$ and edges for precisely those pairs of variables (v_i, v_j) which cooccur in some term of f. This graph is a union of t randomly chosen k-cliques from $\{v_1, \ldots, v_n\}$ which correspond to the t terms in f. We will show how to efficiently identify (with high probability over the choice of f and random examples of f) all of the k-cliques

in G. Once these k-cliques have been identified, as we show later it is easy to construct an accurate DNF hypothesis for f.

The following lemma shows that with high probability over the choice of f, each pair (v_i, v_j) cooccurs in at most a constant number of terms:

Lemma 10. *Fix $1 \leq i < j \leq n$. For any $C \geq 0$ and all sufficiently large n, we have $\Pr_{f \in \mathcal{M}_n^{t,k}}[$some pair of variables (v_i, v_j) cooccur in more than C terms of $f] \leq (\frac{tk^2}{n^2})^C = \delta_C$.*

Proof. For any fixed $r \in \{1, \ldots, t\}$ we have that $\Pr[v_i$ and v_j cooccur in term $T_r] = \frac{k(k-1)}{n(n-1)} \leq \frac{k^2}{n^2}$. Since these events are independent for all r, the probability that there is any collection of C terms such that v_i and v_j cooccur in all C of these terms is at most $\binom{t}{C} \cdot (\frac{k^2}{n^2})^C \leq (\frac{tk^2}{n^2})^C$. □

By Lemma 10 we know that, for any given pair (v_i, v_j) of variables, with probability at least $1 - \delta_C$ there are at most Ck other variables v_ℓ such that (v_i, v_j, v_ℓ) all cooccur in some term of f. Suppose that we can efficiently (with high probability) identify the set S_{ij} of all such variables v_ℓ. Then we can perform an exhaustive search over all $(k-2)$-element subsets S' of S_{ij} in at most $\binom{Ck}{k} \leq (eC)^k = n^{O(\log C)}$ time, and can identify exactly those sets S' such that $S' \cup \{v_i, v_j\}$ is a clique of size k in G. Repeating this over all pairs of variables (v_i, v_j), we can with high probability identify all k-cliques in G.

Thus, to identify all k-cliques in G it remains only to show that for every pair of variables v_i and v_j, we can determine the set S_{ij} of those variables v_ℓ that cooccur in at least one term with both v_i and v_j. Assume that f is such that all pairs of variables cooccur in at most C terms, and let T be a set of variables of cardinality at most C having the following properties:

- In the projection $f_{T \leftarrow 0}$ of f in which all of the variables of T are fixed to 0, v_i and v_j do not cooccur in any term; and
- For every set $T' \subset T$ such that $|T'| = |T| - 1$, v_i and v_j do cooccur in $f_{T' \leftarrow 0}$.

Then T is clearly a subset of S_{ij}. Furthermore, if we can identify all such sets T, then their union will be S_{ij}. There are only $O(n^C)$ possible sets to consider, so our problem now reduces to the following: given a set T of at most C variables, determine whether or not v_i and v_j cooccur in $f_{T \leftarrow 0}$.

The proof of Lemma 9 shows that f is well-behaved with probability at least $1 - \delta_{\text{many}} - \delta_{\text{small}} - \delta_{\text{usat}}$ over the choice of f. Furthermore, if f is well-behaved then it is easy to see that for every $|T| \leq C$, $f_{T \leftarrow 0}$ is also well-behaved, since $f_{T \leftarrow 0}$ is just f with $O(\sqrt{t})$ terms removed (by Lemma 3). That is, removing terms from f can only make it more likely that the remaining terms are uniquely satisfied, does not change the bound on the probability of a pair of remaining terms being satisfied, and can only decrease the bound on the number of remaining terms in which a remaining variable can appear. Furthermore, Lemma 8 holds for any monotone DNF f. Therefore, if f is well-behaved then the proof of Lemma 9 also shows that for every $|T| \leq C$, the P_{ij}'s of $f_{T \leftarrow 0}$ can be used to identify the

cooccurring pairs of variables within $f_{T\leftarrow 0}$. What remains is to show that we can efficiently simulate a uniform example oracle for $f_{T\leftarrow 0}$ so that these P_{ij}'s can be accurately estimated.

In fact, for a given set T, we can simulate a uniform example oracle for $f_{T\leftarrow 0}$ by filtering the examples from the uniform oracle for f so that only examples setting the variables in T to 0 are accepted. Since $|T| \leq C$, the filter accepts with constant probability at least $1/2^C$. A Chernoff argument shows that if all P_{ij}'s are estimated using a single sample of size $2^{C+10}t^2 \ln(2(C+2)n^C/\delta)/\alpha^8$ (filtered appropriately when needed) then all of the estimates will have the desired accuracy with probability at least $1-\delta$.

This gives us the following:

Theorem 2. *For n sufficiently large, any $\delta > 0$, and any fixed $C \geq 2$, with probability at least $1 - \delta_{\text{small}} - \delta_{\text{shared}} - \delta_{\text{usat}} - \delta_C - \delta$ over the random draw of f from $\mathcal{M}_n^{t,k}$ and the choice of random examples, all of the k-cliques of the graph G can be identified in time $O(n^C t^3 k^2 \log(n/\delta))$.*

The main learning result for monotone DNF. We now have a list $T'_1, \ldots, T'_N$ (with $N = O(n^C)$) of length-k monotone terms which contains all t true terms $T_1, \ldots, T_t$ of f. Now observe that the target function f is simply an OR of some subset of these N "variables" $T_1, \ldots, T_N$, so the standard elimination algorithm can be used to PAC learn the target function.

Call the above described entire learning algorithm A. In summary, we have proved the following:

Theorem 3. *Fix γ, $\alpha > 0$ and $C \geq 2$. Let (k,t) be a monotone α-interesting pair. For any $\epsilon > 0$, $\delta > 0$, and $t = O(n^{2-\gamma})$, algorithm A will with probability at least $1 - \delta_{\text{many}} - \delta_{\text{small}} - \delta_{\text{usat}} - \delta_C - \delta$ (over a random choice of DNF from $\mathcal{M}_n^{t,k}$ and the randomness of the example oracle) produce a hypothesis h that ϵ-approximates the target with respect to the uniform distribution. Algorithm A runs in time polynomial in n, $\log(1/\delta)$, and $1/\epsilon$.*

4 Nonmonotone DNF

4.1 Interesting Parameter Settings

As with $\mathcal{M}_n^{t,k}$, we are interested in pairs (k,t) for which $\mathbf{E}_{f\in\mathcal{D}_n^{t,k}}[\Pr[f]]$ is between α and $1-\alpha$:

Definition 2. *For $\alpha > 0$, the pair (k,t) is said to be* α-interesting *if $\alpha \leq \mathbf{E}_{f\in\mathcal{D}_n^{t,k}}[\Pr[f]] \leq 1-\alpha$.*

It is easy to give an explicit formula for $\mathbf{E}_{f\in\mathcal{D}_n^{t,k}}[\Pr[f]]$ which will be useful later. For any fixed $x \in \{0,1\}^n$ we have $\Pr_{f\in\mathcal{D}_n^{t,k}}[f(x)=0] = (1-\frac{1}{2^k})^t$, and thus by linearity of expectation we have $\mathbf{E}_{f\in\mathcal{D}_n^{t,k}}[\Pr[f]] = 1-(1-\frac{1}{2^k})^t$.

Throughout the rest of Section 4 we assume that $\alpha > 0$ is fixed and (k,t) is an α-interesting pair where $t = O(n^{3/2-\gamma})$ for some $\gamma > 0$.

4.2 Properties of $\mathcal{D}_n^{t,k}$

In this section we develop analogues of Lemmas 6 and 7 for $\mathcal{D}_n^{t,k}$. The $\mathcal{D}_n^{t,k}$ analogue of Lemma 7 follows directly from the proof of Lemma 7, and we have:

Lemma 11. *With probability at least* $1 - \delta_{\text{shared}}$ *over the random draw of* f *from* $\mathcal{D}_n^{t,k}$, *for all* $1 \leq i < j \leq n$, $\Pr[T_i \wedge T_j] \leq \frac{\log t}{2^{2k}}$.

In Appendix D we use McDiarmid's bound to prove a $\mathcal{D}_n^{t,k}$ version of Lemma 6:

Lemma 12. *With probability at least* $1 - t\left((t-1)(\frac{k^2}{n})^{\log\log t} + \exp\left(\frac{-\alpha^2 t}{2\ln^2(1/\alpha)\log^2 t}\right)\right)$ $= 1 - \delta'_{\text{usat}}$, *a random* f *drawn from* $\mathcal{D}_n^{t,k}$ *is such that for each* $i = 1, \ldots, t$, *we have* $P_i \equiv \Pr_x[T_i$ *is satisfied by* x *but no other* T_j *is satisfied by* $x] \geq \frac{\alpha}{2^{k+1}}$.

4.3 Identifying (most pairs of) cooccurring variables

Recall that in Section 3.3 we partitioned the terms of our monotone DNF into four disjoint groups $f = g_{**} \vee (v_1 g_{1*}) \vee (v_2 g_{*1}) \vee (v_1 v_2 g_{11})$. depending on what subset of $\{v_1, v_2\}$ was present in each term. Now, in the nonmonotone case, we will partition the terms of our general DNF f into nine disjoint groups depending on whether each of v_1, v_2 is unnegated, negated, or absent:

$$f = g_{**} \vee (v_1 g_{1*}) \vee (\overline{v_1} g_{0*}) \vee (v_2 g_{*1}) \vee (v_1 v_2 g_{11}) \vee (\overline{v_1} v_2 g_{01}) \vee (\overline{v_2} g_{*0}) \vee (v_1 \overline{v_2} g_{10}) \vee (\overline{v_1}\,\overline{v_2} g_{00})$$

Thus g_{**} contains those terms of f which contain neither v_1 nor v_2 in any form; g_{0*} contains the terms of f which contain $\overline{v_1}$ but not v_2 in any form (with $\overline{v_1}$ removed from each term); g_{*1} contains the terms of f which contain v_2 but not v_1 in any form (with v_2 removed from each term); and so on. Each $g_{\cdot,\cdot}$ is thus a DNF (possibly empty) over literals formed from $v_3, \ldots, v_n$.

We can empirically estimate each of

$$\begin{aligned}
p_{00} &:= \Pr_x[g_{**} \vee g_{0*} \vee g_{*0} \vee g_{00}] = \Pr_{x \in U_n}[f(x) = 1 \mid x_1 = 0, x_2 = 0] \\
p_{01} &:= \Pr_x[g_{**} \vee g_{0*} \vee g_{*1} \vee g_{01}] = \Pr_{x \in U_n}[f(x) = 1 \mid x_1 = 0, x_2 = 1] \\
p_{10} &:= \Pr_x[g_{**} \vee g_{1*} \vee g_{*0} \vee g_{10}] = \Pr_{x \in U_n}[f(x) = 1 \mid x_1 = 1, x_2 = 0] \\
p_{11} &:= \Pr_x[g_{**} \vee g_{1*} \vee g_{*1} \vee g_{11}] = \Pr_{x \in U_n}[f(x) = 1 \mid x_1 = 1, x_2 = 1].
\end{aligned}$$

It is easy to see that $\Pr[g_{11}]$ is either 0 or else at least $\frac{4}{2^k}$ depending on whether g_{11} is empty or not. Thus, ideally we would like to be able to accurately estimate each of $\Pr[g_{00}], \Pr[g_{01}], \Pr[g_{10}]$ and $\Pr[g_{11}]$; if we could do this then we would have complete information about which pairs of literals involving variables v_1 and v_2 cooccur in terms of f. Unfortunately, the probabilities $\Pr[g_{00}], \Pr[g_{01}], \Pr[g_{10}]$ and $\Pr[g_{11}]$ cannot in general be obtained from p_{00}, p_{01}, p_{10} and p_{11}. However, we will show that we can efficiently obtain some partial information which enables us to learn to fairly high accuracy.

As before, our approach is to accurately estimate the quantity $P = p_{11} - p_{10} - p_{01} + p_{00}$. We have the following lemmas proved in Appendix E:

Lemma 13. *If all four of g_{00}, g_{01}, g_{10} and g_{11} are empty, then P equals*

$$\begin{aligned}&\Pr[g_{1*} \wedge g_{*0} \wedge (\text{ no other } g_{\cdot,\cdot})] + \Pr[g_{0*} \wedge g_{*1} \wedge (\text{ no other } g_{\cdot,\cdot})] \\ &\quad - \Pr[g_{1*} \wedge g_{*1} \wedge (\text{ no other } g_{\cdot,\cdot})] - \Pr[g_{0*} \wedge g_{*0} \wedge (\text{ no other } g_{\cdot,\cdot})]. \quad (1)\end{aligned}$$

Lemma 14. *If exactly one of g_{00}, g_{01}, g_{10} and g_{11} is nonempty (say g_{11}), then P equals (1) plus*

$$\begin{aligned}&\Pr[g_{11} \wedge g_{1*} \wedge g_{*0} \wedge (\text{ no other } g_{\cdot,\cdot})] + \Pr[g_{11} \wedge g_{0*} \wedge g_{*1} \wedge (\text{ no other } g_{\cdot,\cdot})] \\ &- \Pr[g_{11} \wedge g_{1*} \wedge g_{*1} \wedge (\text{ no other } g_{\cdot,\cdot})] - \Pr[g_{11} \wedge g_{0*} \wedge g_{*0} \wedge (\text{ no other } g_{\cdot,\cdot})] \\ &+ \Pr[g_{11} \wedge g_{0*} \wedge (\text{ no other } g_{\cdot,\cdot})] + \Pr[g_{11} \wedge g_{*0} \wedge (\text{ no other } g_{\cdot,\cdot})] + \Pr[g_{11} \wedge (\text{ no other } g_{\cdot,\cdot})].\end{aligned}$$

Using the above two lemmas we can show that the value of P is a good indicator for distinguishing between all four of $g_{00}, g_{01}, g_{10}, g_{11}$ being empty versus exactly one of them being nonempty:

Lemma 15. *For n sufficiently large and $t \geq 4$, with probability at least $1 - \delta'_{\text{usat}} - \delta_{\text{shared}} - \delta_{\text{many}}$ over a random draw of f from $\mathcal{D}_n^{t,k}$, we have that: (i) if v_1 and v_2 do not cooccur in any term of f then $P \leq \frac{\alpha^2}{8t}$.; (ii) if v_1 and v_2 do cooccur in some term of f and exactly one of g_{00}, g_{01}, g_{10} and g_{11} is nonempty, then $P \geq \frac{3\alpha^2}{16t}$.*

Proof. With probability at least $1 - \delta'_{\text{usat}} - \delta_{\text{shared}} - \delta_{\text{many}}$ a randomly chosen f from $\mathcal{D}_n^{t,k}$ will have all of the following properties:

1. Each term in f is uniquely satisfied with probability at least $\alpha/2^{k+1}$ (by Lemma 12);
2. Each variable in f appears in at most $2^{k-1}\alpha^2/\sqrt{t}\log t$ terms (by Lemma 3); and
3. Each pair of terms T_i and T_j in f are both satisfied with probability at most $\log t/2^{2k}$ (by Lemma 11).

For the sequel assume that we have such an f. We first prove (i) by showing that P—as represented by (1) of Lemma 13—is at most $\frac{\alpha^4}{t\log t}$. By property 3 above, for any pair of terms consisting of one term from g_{1*} and the other from g_{*0}, the probability that both terms are satisfied is at most $\log t/2^{2k}$. Since each of g_{1*} and g_{*0} contains at most $2^{k-1}\alpha^2/\sqrt{t}\log t$ terms by property 2, a union bound gives $\Pr[g_{1*} \wedge g_{*0} \wedge (\text{ no other } g_{\cdot,\cdot})] \leq \Pr[g_{1*} \wedge g_{*0}] \leq \frac{\alpha^4}{4t\log t}$. A similar argument holds for the three other summands in (1), so P is at most $\frac{\alpha^4}{t\log t} \leq \frac{\alpha^2}{8t}$ since $\alpha \leq 1/2$ and $t \geq 4$.

We now prove (ii). By an argument similar to the above we have that the first six summands in the expression of Lemma 14 are each at most $\frac{\alpha^4}{4t\log t}$ in magnitude. Now observe that each instance x that uniquely satisfies a term T_j in f containing both v_1 unnegated and v_2 unnegated must satisfy g_{11} and no other $g_{\cdot,\cdot}$. Thus under the conditions of (ii) the last summand in Lemma 14 is at least $\frac{\alpha}{2^{k+1}}$ by property 1 above, so we have that (ii) is at least $\frac{\alpha}{2^{k+1}} - \frac{5}{2}\frac{\alpha^4}{t\log t}$. Since

(k,t) is α-interesting we have $\frac{t}{2^k} \geq \alpha$, and from this and the constant bounds on α and t it is easily shown that $\frac{\alpha}{2^{k+1}} \geq \frac{\alpha^2}{2t}$ and $\frac{5}{2}\frac{\alpha^4}{t \log t} \leq \frac{5\alpha^2}{16t}$, from which the lemma follows after simplifying the difference of these quantities. □

It is clear that an analogue of Lemma 15 holds for any pair of variables v_i, v_j in place of v_1, v_2. Thus, for each pair of variables v_i, v_j, if we decide whether v_i and v_j cooccur (negated or otherwise) in any term on the basis of whether P_{ij} is large or small, we will err only if two or more of $g_{00}, g_{01}, g_{10}, g_{11}$ are nonempty.

We now show that for $f \in \mathcal{D}_n^{t,k}$, with very high probability there are not too many pairs of variables (v_i, v_j) which cooccur (with any sign pattern) in at least two terms of f. (Note that this immediately bounds the number of pairs (v_i, v_j) which have two or more of the corresponding $g_{00}, g_{01}, g_{10}, g_{11}$ nonempty.)

Lemma 16. *Let $d > 0$ and $f \in \mathcal{D}_n^{t,k}$. The probability that more than $(d+1)t^2k^4/n^2$ pairs of variables (v_i, v_j) each cooccur in two or more terms of f is at most $\exp(-d^2t^3k^4/n^4)$.*

Proof. We use McDiarmid's inequality, where the random variables are the terms $T_1, \ldots, T_t$ chosen independently from the set of all possible terms of length k and $F(T_1, \ldots, T_t)$ denotes the number of pairs of variables (v_i, v_j) that cooccur in at least two terms. For each $\ell = 1, \ldots, t$ we have $\Pr[T_\ell$ contains both v_1 and $v_2] \leq \frac{k^2}{n^2}$, so by a union bound we have $\Pr[f$ contains at least two terms which contain both v_1 and v_2 in any form$] \leq \frac{t^2k^4}{n^4}$. By linearity of expectation we have $\mu = \mathbf{E}[F] \leq \frac{t^2k^4}{n^2}$. Since each term involves at most k^2 pairs of cooccurring variables, we have $|F(T_1, \ldots, T_t) - F(T_1, \ldots, T_{i-1}, T_i', T_{i+1}, \ldots, T_t)| \leq c_i = k^2$. We thus have by McDiarmid's inequality that $\Pr[F \geq t^2k^4/n^2 + \tau] \leq \exp(-\tau^2/(tk^4))$. Taking $\tau = dt^2k^4/n^2$, we have $\Pr[F \geq (d+1)t^2k^4/n^2] \leq \exp(-d^2t^3k^4/n^4)$. □

Taking $d = n^2/(t^{5/4}k^4)$ in the above lemma (note that $d > 1$ for n sufficiently large since $t^{5/4} = O(n^{15/8})$), we have $(d+1)t^2k^4/n^2 \leq 2t^{3/4}$ and the failure probability is at most $\exp(-\sqrt{t}/k^4) = \delta_{\text{cooccur}}$. The results of this section (together with a standard analysis of error in estimating each P_{ij}) thus yield:

Theorem 4. *For n sufficiently large and for any $\delta > 0$, with probability at least $1 - \delta_{\text{cooccur}} - \delta'_{\text{usat}} - \delta_{\text{shared}} - \delta_{\text{many}} - \delta$ over the random draw of f from $\mathcal{D}_n^{t,k}$ and the choice of random examples, the above algorithm runs in $O(n^2t^2 \log(n/\delta))$ time and outputs a list of pairs of variables (v_i, v_j) such that: (i) if (v_i, v_j) is in the list then v_i and v_j cooccur in some term of f; and (ii) at most $N_0 = 2t^{3/4}$ pairs of variables (v_i, v_j) which do cooccur in f are not on the list.*

4.4 Reconstructing an accurate DNF hypothesis

Now we show how to construct a good hypothesis for the target DNF from a list of pairwise cooccurrence relationships as provided by Theorem 4. As in the monotone case, we consider the graph G with vertices $v_1, \ldots, v_n$ and edges for precisely those pairs of variables (v_i, v_j) which cooccur (with any sign pattern)

in some term of f. As before this graph is a union of t randomly chosen k-cliques $S_1, \ldots, S_t$ which correspond to the t terms in f, and as before we would like to find all k-cliques in G. However, there are two differences now: the first is that instead of having the true graph G, we instead have access only to a graph G' which is formed from G by deleting some set of at most $N_0 = 2t^{3/4}$ edges. The second difference is that the final hypothesis must take the signs of literals in each term into account. To handle these two differences, we use a somewhat different reconstruction procedure than we used for monotone DNF in Section 3.4; this reconstruction procedure only works for $t = O(n^{3/2-\gamma})$ where $\gamma > 0$.

We first show how to identify (with high probability over the choice of f) We then show how to form a DNF hypothesis from the set of all k-cliques in G'.

We now describe an algorithm which, for $t = O(n^{3/2-\gamma})$ with $\gamma > 0$, with high probability runs in polynomial time and identifies all the k-cliques in G' which contain edge (v_1, v_2). Running the algorithm at most tk^2 times on all edges in G' will give us with high probability all the k-cliques in G'. The algorithm is:

- Let Δ be the set of vertices v_j such that v_1, v_2, v_j form a triangle in G'. Run a brute-force algorithm to find all $(k-2)$-cliques in the graph induced by Δ.

It is clear that the algorithm finds every k-clique which contains edge (v_1, v_2). To bound the algorithm's running time, it suffices to give a high probability bound on the size of Δ in the graph G (clearly Δ only shrinks in passing from G to G'). The following lemma (proved in Appendix F) gives such a bound:

Lemma 17. *Let G be a random graph as described above and let $0 < \gamma < \frac{1}{4}$. For any $t = O(n^{3/2-\gamma})$ and any $C > 0$ we have that with probability $1 - O\left(\frac{\log^{6C} n}{n^{2\gamma C}}\right)$ the size of Δ in G is at most Ck.*

By Lemma 17, doing a brute-force search which finds all k-cliques in the graph induced by Δ takes at most $\binom{Ck}{k} \leq \left(\frac{eCk}{k}\right)^k = (eC)^{O(\log n)} = n^{O(\log C)}$ time steps. Thus we can efficiently with high probability identify all the k-cliques in G'. How many of the "true" cliques $S_1, \ldots, S_t$ in G are not present as k-cliques in G'? By Lemma 10, with probability at least $1 - t^2(\frac{tk^2}{n^2})^C$ each edge (v_i, v_j) participates in at most C cliques from $S_1, \ldots, S_t$. Since G' is missing at most N_0 edges from G', with probability at least $1 - t^2(\frac{tk^2}{n^2})^C$ the set of all k-cliques in G' is missing at most CN_0 "true" cliques from $S_1, \ldots, S_t$.

Summarizing the results of this section so far, we have:

Theorem 5. *Fix $C \geq 2$. Given a DNF formula f drawn from $\mathcal{D}_n^{t,k}$ and a list of pairs of cooccurring variables as described in Theorem 4, with probability at least $1 - 1/n^{\Omega(C)}$ the above procedure runs in $n^{O(\log C)}$ time and constructs a a list $Z_1, \ldots, Z_{N'}$ (where $N' = n^{O(\log C)}$) of k-cliques which contains all but at most CN_0 of the cliques $S_1, \ldots, S_t$.*

We construct a hypothesis DNF from the list $Z_1, \ldots, Z_{N'}$ of candidate k-cliques as follows: for each Z_i we form all 2^k possible terms which could have given rise to Z_i (corresponding to all 2^k sign patterns on the k variables in

Z_i). We then test each of these $2^k N'$ potential terms against a sample of M randomly drawn negative examples and discard any terms which output 1 on any negative example; the final hypothesis h is the OR of all surviving terms. Any candidate term T' which has $\Pr_{x \in U_n}[T'(x) = 1 \ \& \ f(x) = 0] \geq \frac{\epsilon}{2^{k+1}N'}$ will survive this test with probability at most $\exp(-\epsilon M / 2^{k+1} N')$. Taking $\epsilon = 1/2^k$ and $M = (1/\epsilon) 2^{k+1} N' \log^2 n$ we have that with probability $1 - 1/n^{\omega(1)}$ each term in the final hypothesis contributes at most $\epsilon / 2^{k+1} N'$ toward the false positive rate of h, so with high probability the false positive rate of h is at most $\epsilon = 1/2^k$.

The false negative rate of h is at most $\frac{1}{2^k}$ times the number of terms in f which are missing in h. Since the above algorithm clearly will not discard any term in f (since such a term will never cause a false negative mistake), we need only bound the number of terms in f which are not among our $2^k N'$ candidates. With probability at least $1 - t/\binom{n}{k} = 1 - \delta_{\text{clique}}$ each true clique $S_1, \ldots, S_t$ in G gives rise to exactly one term of f (the only way this does not happen is if two terms consist of literals over the exact same set of k variables), so Theorem 5 implies that h is missing at most CN_0 terms of f. Thus the false negative rate is at most $CN_0/2^k \leq 2Ct^{3/4}/2^k = 1/\Omega(t^{1/4})$.

All in all the following is our main learning result for nonmonotone DNF:

Theorem 6. *Fix γ, $\alpha > 0$ and $C \geq 2$. Let (k, t) be a monotone α-interesting pair. For f randomly chosen from $\mathcal{D}_n^{t,k}$, with probability at least $1 - \delta_{\text{cooccur}} - \delta'_{\text{usat}} - \delta_{\text{shared}} - \delta_{\text{many}} - \delta_{\text{clique}} - 1/n^{\Omega(C)}$ the above algorithm runs in $\tilde{O}(n^2 t^2 + n^{O(\log C)})$ time and outputs a hypothesis h whose error rate relative to f under the uniform distribution is at most $1/\Omega(t^{1/4})$.*

One can straightforwardly verify from the definitions of the various δ's that for any $t = \omega(1)$ as a function of n, the failure probability of the algorithm is $o(1)$ and the algorithm learns to accuracy $1 - o(1)$.

5 Discussion and Conclusions

We have shown that several natural models of random DNF formulas can be efficiently learned to high accuracy under the uniform distribution.

Several directions for future work present themselves. We can currently only learn random DNFs with $o(n^{3/2})$ terms ($o(n^2)$ terms for monotone DNF); can stronger results be obtained which hold for all polynomial-size DNF? Also, our current results for $t = \omega(1)$-term DNF let us learn to some $1 - o(1)$ accuracy but we cannot yet achieve an arbitrary inverse polynomial error rate for nonmonotone DNF. Finally, another interesting direction is to explore other natural models of random DNF formulas, perhaps by allowing some variation among term sizes or dependencies between terms.

Acknowledgement. Avrim Blum suggested to one of us (JCJ) the basic strategy that learning monotone DNF with respect to uniform might be reducible to finding the cooccurring pairs of variables in the target function.

References

[1] H. Aizenstein and L. Pitt. On the learnability of disjunctive normal form formulas. *Machine Learning*, 19:183–208, 1995.

[2] D. Angluin. Negative results for equivalence queries. *Machine Learning*, 5:121–150, 1990.

[3] A. Blum, M. Furst, M. Kearns, and R. Lipton. Cryptographic Primitives Based on Hard Learning Problems. In *Advances in Cryptology – CRYPTO '93*, pages 278–291, 1993.

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

[5] N. Bshouty. A subexponential exact learning algorithm for DNF using equivalence queries. *Information Processing Letters*, 59:37–39, 1996.

[6] N. Bshouty and C. Tamon. On the fourier spectrum of monotone functions. *Journal of the ACM*, 43(4):747–770, 1996.

[7] T. Hancock and Y. Mansour. Learning monotone k-μ DNF formulas on product distributions. In *Proceedings of the Fourth Annual Conference on Computational Learning Theory*, pages 179–193, 1991.

[8] L. Hellerstein and V. Raghavan. Exact learning of DNF formulas using DNF hypotheses. In *Proceedings of the 34th Annual Symposium on Theory of Computing*, pages 465–473, 2002.

[9] J. Jackson. An efficient membership-query algorithm for learning DNF with respect to the uniform distribution. *Journal of Computer and System Sciences*, 55:414–440, 1997.

[10] J. Jackson and R. Servedio. Learning random log-depth decision trees under the uniform distribution. In *Proceedings of the 16th Annual Conf. on Computational Learning Theory and 7th Kernel Workshop*, pages 610–624, 2003.

[11] A. Klivans and R. Servedio. Learning DNF in time $2^{\tilde{O}(n^{1/3})}$. In *Proceedings of the Thirty-Third Annual Symposium on Theory of Computing*, pages 258–265, 2001.

[12] L. Kucera, A. Marchetti-Spaccamela, and M. Protassi. On learning monotone DNF formulae under uniform distributions. *Information and Computation*, 110:84–95, 1994.

[13] Y. Mansour. An $O(n^{\log \log n})$ learning algorithm for DNF under the uniform distribution. *Journal of Computer and System Sciences*, 50:543–550, 1995.

[14] C. McDiarmid. On the method of bounded differences. In *Surveys in Combinatoric 1989*, pages 148–188. London Mathematical Society Lecture Notes, 1989.

[15] Y. Sakai and A. Maruoka. Learning monotone log-term DNF formulas under the uniform distribution. *Theory of Computing Systems*, 33:17–33, 2000.

[16] R. Servedio. On learning monotone DNF under product distributions. In *Proceedings of the Fourteenth Annual Conference on Computational Learning Theory*, pages 473–489, 2001.

[17] K. Verbeurgt. Learning DNF under the uniform distribution in quasi-polynomial time. In *Proceedings of the Third Annual Workshop on Computational Learning Theory*, pages 314–326, 1990.

[18] K. Verbeurgt. Learning sub-classes of monotone DNF on the uniform distribution. In *Proceedings of the Ninth Conference on Algorithmic Learning Theory*, pages 385–399, 1998.

A Proof of Lemma 1

Proof of Lemma 1: One side is easy: if $t < \alpha 2^k$ then each of the t terms of f is satisfied by a uniform random example with probability at most α/t, and consequently $\Pr[f(x) = 1] \leq \alpha$. Note that by our assumptions on t and α we thus have that $k = O(\log n)$ for any monotone α-interesting pair (k, t).

We now show that if $t > 2^{k+1} \log \frac{2}{\alpha}$, then $\mathbf{E}_{f \in \mathcal{M}_n^{t,k}}[\Pr[f]] > 1 - \alpha$. Let us write $|x|$ to denote $x_1 + \cdots + x_n$ for $x \in \{0,1\}^n$. It is easy to see that $\Pr[f(x) = 1]$, viewed as a random variable over the choice of $f \in \mathcal{M}_n^{t,k}$, depends only on the value of $|x|$. We have

$$\mathbf{E}_{f \in \mathcal{M}_n^{t,k}}[\Pr[f]] = \sum_{r=0}^{n} \mathbf{E}_{f \in \mathcal{M}_n^{t,k}}[\Pr[f(x) = 1 \mid |x| = r] \cdot \Pr[|x| = r].$$

A standard tail bound on the binomial distribution implies that

$$\Pr_{x \in U_n}\left[|x| \leq n/2 - \sqrt{n \log(2/\alpha)}\right] < \alpha/2.$$

Thus it suffices to show that for any x with $|x| \geq n/2 - \sqrt{n \log(2/\alpha)}$, we have $\Pr_{f \in \mathcal{M}_n^{t,k}}[f(x) = 1] \geq 1 - \alpha/2$.

Fix an $x \in \{0,1\}^n$ with $|x| = w \geq n/2 - \sqrt{n \log(2/\alpha)}$. Let T_1 be a random monotone term of length k. We have

$$\Pr_{T_1}[T_1(x) = 1] = \frac{w(w-1)\cdots(w-k+1)}{n(n-1)\cdots(n-k+1)} \geq \frac{1}{2^{k+1}}$$

where the inequality is implied by our conditions on k and α. Since the terms of f are chosen independently, this implies that

$$\Pr_f[f(x) = 0] \leq \left(1 - \frac{1}{2^{k+1}}\right)^t \leq \exp\left(\frac{-t}{2^{k+1}}\right).$$

If $t/2^{k+1} > \ln \frac{2}{\alpha}$ then this bound is at most $\alpha/2$. □

B Proof of Lemmas 2, 3, 4, 5, 6, and 7

Proof of Lemma 2:

We write $T_1, T_2, \ldots, T_t$ to denote the terms of f. We have

$$\begin{aligned}\Pr[\overline{f}] &= \Pr[\overline{T_1} \wedge \overline{T_2} \wedge \cdots \wedge \overline{T_t}] = \Pr[\overline{T_1} \mid \overline{T_2} \wedge \cdots \overline{T_t}] \Pr[\overline{T_2} \mid \overline{T_3} \wedge \cdots \overline{T_t}] \cdots \Pr[\overline{T_{t-1}} \mid \overline{T_t}] \Pr[\overline{T_t}] \\ &\overset{(*)}{\geq} \prod_{i=1}^{t} \Pr[\overline{T_i}] = \left(1 - \frac{1}{2^k}\right)^t \geq \left(1 - \frac{1}{2^k}\right)^{2^{k+1} \ln(2/\alpha)} \geq \left(\frac{1}{4}\right)^{2 \ln \frac{2}{\alpha}} > \alpha^3.\end{aligned}$$

The first inequality holds since $\Pr[f(x) = 1 | g(x) = 1] \geq \Pr[f(x) = 1]$ for any monotone Boolean functions f, g on $\{0,1\}^n$ (see e.g. Corollary 7, p. 149 of [4]).

The second inequality holds by Lemma 1, and the third holds since $(1-1/x)^x \geq 1/4$ for all $x \geq 2$. □

Proof of Lemma 3:

Fix any variable v_j. For each term T_ℓ we have that v_j occurs in T_ℓ with probability k/n. Since the terms are chosen independently, the number of occurrences of v_j is binomially distributed according to $B(t,p)$ with $p = k/n$. Taking $\beta = n2^{k-1}\alpha^2/kt^{3/2}\log t$ in the Chernoff bound (which is greater than 1 for sufficiently large n), the probability that v_j appears in $\beta pt = 2^{k-1}\alpha^2/\sqrt{t}\log t$ or more terms is at most $\left(\frac{ekt^{3/2}\log t}{n2^{k-1}\alpha^2}\right)^{2^{k-1}\alpha^2/\sqrt{t}\log t}$. The lemma follows by the union bound over the n variables v_j. □

Proof of Lemma 4:

We will modify the proof of the previous lemma slightly. We first fix a value $1 \leq i \leq t$ which will act as the index of a distinguished term T_i, and we also fix a value $1 \leq j \leq k$ which will be the index of a distinguished variable within T_i. By taking $\beta = \frac{n2^k}{k(t-1)\log t}$ in the Chernoff bound we have that the probability over the choice of the $t-1$ terms other than T_i that v_j also appears in $\beta p(t-1) = \frac{2^k}{\log t}$ or more terms is at most $\left(\frac{ek(t-1)\log t}{n2^k}\right)^{2^k/(\log t)}$. We then again apply the union bound, this time over tk different choices of i and j. □

Proof of Lemma 5:

We are interested in upper bounding the probability p_i that $\log\log t$ or more of the variables in a fixed term T_i belonging to f also appear in some other term T_ℓ of f, for any $\ell \neq i$. First, a simple counting argument shows that the probability that a fixed set of $\log\log t$ variables appears in a set of k variables randomly chosen from among n variables is at most $(k/n)^{\log\log t}$. Since there are $\binom{k}{\log\log t}$ ways to choose a fixed set of $\log\log t$ variables from term T_i, we have $p_i \leq \binom{k}{\log\log t}\left(\frac{k}{n}\right)^{\log\log t}(t-1)$. The lemma follows by the union bound over the t probabilities p_i. □

Proof of Lemma 6:

Given an f drawn according to $\mathcal{M}_n^{t,k}$ and given any term T_i in f, we are interested in the probability over uniformly drawn instances that T_i is satisfied and T_ℓ is not satisfied for all $\ell \neq i$. Let $\overline{T_{\ell \neq i}}$ represent the formula that is satisfied by an assignment x if and only if all of the T_ℓ with $\ell \neq i$ are not satisfied by x. We want a lower bound on

$$\Pr[T_i \wedge \overline{T_{\ell \neq i}}] = \Pr[\overline{T_{\ell \neq i}} \mid T_i] \cdot \Pr[T_i].$$

Since $\Pr[T_i] = 1/2^k$, what remains is to show that with very high probability over random draw of f, $\Pr[\overline{T_{\ell \neq i}} \mid T_i]$ is bounded below by $\alpha^3/4$ for all T_i. That is, we need to show that $\Pr[\overline{f^i}] \geq \alpha^3/4$ with very high probability.

We have that all of the following statements hold with probability at least $1 - \delta_{\text{usat}}$ for every $1 \leq i \leq n$ for a random f from $\mathcal{M}_n^{t,k}$:

1. $\Pr[\overline{f^i}] \geq \prod_{\ell:\ell\neq i} \Pr[\overline{T_\ell^i}]$: this follows from Equation $(*)$ in the proof of Lemma 2.
2. $\prod_{\ell:T_\ell^i \equiv T_\ell} \Pr[\overline{T_\ell^i}] > \alpha^3$. This holds because the terms in this product are a subset of the terms in Equation $(*)$ (in the proof of Lemma 2).
3. At most $2^k/\log t$ terms T_ℓ with $\ell \neq i$ are smaller in f^i than they are in f (by Lemma 4).
4. No term in f^i has fewer than $k - \log\log t$ variables (by Lemma 5).

These conditions together imply that $\Pr[\overline{f^i}] \geq \alpha^3 \left(1 - \frac{\log t}{2^k}\right)^{2^k/\log t} \geq \alpha^3/4$ using the fact that $\left(1 - \frac{1}{x}\right)^x \geq 1/4$ for all $x \geq 2$. □

Proof of Lemma 7:

By Lemma 5, with probability at least $1 - \delta_{\text{shared}}$ f is such that, for all $1 \leq i < j \leq n$, terms T_i and T_j share at most $\log\log t$ variables. Thus for each pair of terms a specific set of at least $2k - \log\log t$ variables must be simultaneously set to 1 in an instance in order for both terms to be satisfied. □

C Proof of Lemma 8:

P gets a net contribution of 0 from those x which belong to $g_{*,*}$ (since each such x is added twice and subtracted twice in P). We proceed to analyze the contributions to P from the remaining 8 subsets of the events g_{11}, g_{1*} and g_{*1}:

- P gets a net contribution of 0 from those x which are in $g_{1*} \wedge \overline{g}_{*1} \wedge \overline{g}_{**}$ since each such x is counted in p_{11} and p_{10} but not in p_{01} or p_{00}. Similarly P gets a net contribution of 0 from those x which are in $g_{*1} \wedge \overline{g}_{1*} \wedge \overline{g}_{**}$.
- P gets a net contribution of $\Pr[g_{11} \wedge \overline{g}_{1*} \wedge \overline{g}_{*1} \wedge \overline{g}_{**}]$ since each such x is counted in p_{11}.
- P gets a net contribution of $-\Pr[g_{1*} \wedge g_{*1} \wedge \overline{g}_{**}]$ since each such x is counted in p_{01}, p_{10} and p_{11}. □

D Proof of Lemma 12

We show that $P_1 \geq \frac{\alpha}{2^{k+1}}$ with probability at least $1 - \delta'_{\text{usat}}/t$; the lemma follows by a union bound. We first show that $\mathbf{E}_{f \in \mathcal{D}_n^{t,k}}[P_1] \geq \frac{\alpha}{2^k}$. For any fixed $x \in T_1$, we have $\Pr[\overline{T}_2(x) \wedge \cdots\cdots \wedge \overline{T}_t(x)] = (1 - 2^{-k})^{t-1} > (1 - 2^{-k})^t \geq \alpha$ where the last inequality holds since (k, t) is α-interesting. Since a 2^{-k} fraction of all $x \in \{0,1\}^n$ belong to T_1, by linearity of expectation we have $\mathbf{E}_{f \in \mathcal{D}_n^{t,k}}[P_1] \geq \frac{\alpha}{2^k}$.

Now we show that with high probability the deviation of P_1 from its expected value is low. Given any fixed length-k term T_1, let Ω denote the set of all length-k terms T which satisfy $\Pr[T_1 \wedge T] \leq \frac{\log t}{2^{2k}}$. By reasoning as in the proof of Lemma 11, with probability at least $1 - (t-1)(\frac{k^2}{n})^{\log\log t}$ each of $T_2, \ldots, T_t$ belongs to Ω, so we henceforth assume that this is in fact the case, i.e. we condition on the

event $\{T_2, \ldots, T_t\} \subset \Omega$. Note that under this conditioning we have that each of $T_2, \ldots, T_t$ is selected uniformly and independently from Ω.

We now use McDiarmid's inequality where the random variables are the randomly selected terms $T_2, \ldots, T_t$ from Ω and $F(T_2, \ldots, T_t)$ denotes P_1, i.e.

$$F(T_2, \ldots, T_t) = \Pr_x[T_1 \text{ is satisfied by } x \text{ but no } T_j \text{ with } j \geq 2 \text{ is satisfied by } x].$$

Since each T_j belongs to Ω, we have $|F(T_2, \ldots, T_t) - F(T_2, \ldots, T_{j-1}, T'_j, T_{j+1}, \ldots, T_t)| \leq c_i = \frac{\log t}{2^{2k}}$ for all $j = 2, \ldots, t$. Taking $\tau = \frac{\alpha}{2^{k+1}}$, McDiarmid's inequality implies that $\Pr\left[P_1 \geq \frac{\alpha}{2^{k+1}}\right]$ is at most

$$\exp\left(\frac{-\alpha^2/(4 \cdot 2^{2k})}{(t-1)(\frac{\log t}{2^{2k}})^2}\right) = \exp\left(\frac{-\alpha^2 2^{2k}}{4(t-1)\log^2 t}\right) < \exp\left(\frac{-\alpha^2 2^{2k}}{2t \log^2 t}\right) \leq \exp\left(\frac{-\alpha^2 t}{2\ln^2(1/\alpha)\log^2 t}\right)$$

where the last inequality holds since (k, t) is α-interesting. Combining all the failure probabilities, the lemma is proved. □

E Proof of Lemmas 13 and 14

Proof of Lemma 13: Since all four of g_{00}, g_{01}, g_{10} and g_{11} are empty we need only consider the five events $g_{**}, g_{*0}, g_{0*}, g_{*1}$ and g_{1*}. We now analyze the contribution to P from each possible subset of these 5 events:

- P gets a net contribution of 0 from those x which belong to $g_{*,*}$ (and to any other subset of the remaining four events) since each such x is counted in each of p_{00}, p_{01}, p_{10} and p_{11}. It remains to consider all 16 subsets of the four events g_{*0}, g_{0*}, g_{*1} and g_{1*}.
- P gets a net contribution of 0 from those x which are in at least 3 of the four events g_{*0}, g_{0*}, g_{*1} and g_{1*} since each such x is counted in each of p_{00}, p_{01}, p_{10} and p_{11}. P also gets a net contribution of 0 from those x which are in exactly one of the four events g_{*0}, g_{0*}, g_{*1} and g_{1*}. It remains to consider those x which are in exactly two of the four events g_{1*}, g_{0*}, g_{*1} and g_{*0}.
- P gets a net contribution of 0 from those x which are in g_{1*} and g_{0*} and no other events, since each such x is counted in each of p_{00}, p_{01}, p_{10} and p_{11}. The same is true for those x which are in g_{*1} and g_{*0} and no other events.
- P gets a net contribution of $-\Pr[g_{1*} \wedge g_{*1} \wedge (\text{ no other } g_{.,.} \text{ occurs})]$ from those x which are in g_{1*} and g_{*1} and no other event. Similarly, P gets a net contribution of $-\Pr[g_{0*} \wedge g_{*0} \wedge (\text{ no other } g_{.,.} \text{ occurs})]$ from those x which are in g_{0*} and g_{*0} and no other event. P gets a net contribution of $\Pr[g_{1*} \wedge g_{*0} \wedge (\text{ no other } g_{.,.} \text{ occurs})]$ from those x which are in g_{1*} and g_{*0} and no other event, and gets a net contribution of $\Pr[g_{0*} \wedge g_{*1} \wedge (\text{ no other } g_{.,.} \text{ occurs})]$ from those x which are in g_{0*} and g_{*1} and no other event. □

Proof of Lemma 14: We suppose that g_{11} is nonempty. We wish to analyze the contribution to P from all 64 subsets of the six events $g_{**}, g_{1*}, g_{0*}, g_{*1}, g_{*0}$ and g_{11}. From Lemma 13 we know this contribution for the 32 subsets which do not include g_{11} is (1) so only a few cases remain:

- P gets a net contribution of 0 from those x which are in g_{11} and in g_{**} and in any other subset of events (each such x is counted in each of p_{11}, p_{01}, p_{10} and p_{00}). Similarly, P gets a contribution of 0 from those x which are in g_{11} and in at least three of $g_{1*}, g_{0*}, g_{*1}, g_{*0}$. So it remains only to analyze the contribution from subsets which contain g_{11}, contain at most two of $g_{1*}, g_{0*}, g_{*1}, g_{*0}$, and contain nothing else.
- An analysis similar to that of Lemma 13 shows that P gets a net contribution of $\Pr[g_{11} \wedge g_{1*} \wedge g_{*0} \wedge (\text{ no other } g_{\cdot,\cdot})] + \Pr[g_{11} \wedge g_{0*} \wedge g_{*1} \wedge (\text{ no other } g_{\cdot,\cdot})] - \Pr[g_{11} \wedge g_{1*} \wedge g_{*1} \wedge (\text{ no other } g_{\cdot,\cdot})] - \Pr[g_{11} \wedge g_{0*} \wedge g_{*0} \wedge (\text{ no other } g_{\cdot,\cdot})]$ from those x which are in g_{11}, in exactly two of $\{g_{1*}, g_{0*}, g_{*1}, g_{*0}\}$, and in no other events. So it remains only to consider subsets which contain g_{11} and at most one of $g_{1*}, g_{0*}, g_{*1}, g_{*0}$ and nothing else.
- P gets a contribution of 0 from x which are in g_{11} and g_{1*} and in nothing else; likewise from x which are in g_{11} and g_{*1} and in nothing else. P gets a contribution of $\Pr[g_{11} \wedge g_{0*} \wedge (\text{ no other } g_{\cdot,\cdot})]$ from x which are in g_{11} and g_{0*} and in nothing else, and a contribution of $\Pr[g_{11} \wedge g_{*0} \wedge (\text{ no other } g_{\cdot,\cdot})]$ from x which are in g_{11} and g_{*0} and in nothing else.
- P gets a net contribution of $\Pr[g_{11} \wedge (\text{ no other } g_{\cdot,\cdot})]$ from those x which are in g_{11} and in no other event. □

F Proof of Lemma 17

In order for v_1, v_2, v_j to form a triangle in G, it must be the case that either (i) some clique S_i contains $\{1, 2, j\}$; or (ii) there is some pair of cliques S_a, S_b with $2 \notin S_a$ and $\{1, j\} \subset S_a$ and $1 \notin S_b$ and $\{2, j\} \subset S_b$.

For (i), we have from Lemma 10 that v_1 and v_2 cooccur in more than C terms with probability at most $(\frac{tk^2}{n^2})^C$. Since each term in which v_1 and v_2 cooccur contributes at most $k-2$ vertices v_j to condition (i), the probability that more than $C(k-2)$ vertices v_j satisfy condition (i) is at most $\left(\frac{tk^2}{n^2}\right)^C = O(1/n^{C/2})$.

For (ii), let A be the set of those indices $a \in \{1, \ldots, t\}$ such that $2 \notin S_a$ and $1 \in S_a$, and let S_A be $\cup_{a \in A} S_a$. Similarly let B be the set of indices b such that $1 \notin S_b$ and $2 \in S_b$, and let S_B be $\cup_{b \in B} S_b$. It is clear that A and B are disjoint. For each $\ell = 1, \ldots, t$ we have that $\ell \in A$ independently with probability at most $p = \frac{k}{n}$, so $E[|A|] \leq tk/n$. We now consider two cases:

Case 1: $t \leq n/\log n$. In this case we may take $\beta = \frac{n \log n}{tk}$ in the Chernoff bound, and we have that $\Pr[|A| \geq \beta pt]$ equals

$$\Pr[|A| \geq \log n] \leq \left(\frac{e}{\beta}\right)^{\beta pt} \leq \left(\frac{ek}{\log^2 n}\right)^{\log n} = \left(\frac{e}{\Omega(\log n)}\right)^{\log n} = \frac{1}{n^{\omega(1)}}.$$

The same bound clearly holds for B. Note that in Case 1 we thus have $|S_A|, |S_B| \leq k \log n$ with probability $1 - 1/n^{\omega(1)}$.

Case 2: $t > n/\log n$. In this case we may take $\beta = \log n$ in the Chernoff bound and we obtain

$$\Pr[|A| \geq \beta pt] = \Pr[|A| \geq \frac{tk\log n}{n}] \leq \left(\frac{e}{\log n}\right)^{kt(\log n)/n} < \left(\frac{e}{\log n}\right)^{k} = \frac{1}{n^{\omega(1)}}$$

where the last inequality holds since $k = \Omega(\log n)$ (since $t > n/\log n$ and (k,t) is α-interesting). In Case 2 we thus have $|S_A|, |S_B| \leq \frac{tk^2 \log n}{n}$ with probability $1 - 1/n^{\omega(1)}$.

Let S'_A denote $S_A - \{1\}$ and S'_B denote $S_B - \{2\}$. Since A and B are disjoint, it is easily seen that conditioned on S'_A being of some particular size s'_A, all $\binom{n-2}{s'_A}$ s'_A-element subsets of $\{3,\ldots,n\}$ are equally likely for S'_A. Likewise, conditioned on S'_B being of size s'_B, all $\binom{n-2}{s'_B}$ s'_B-element subsets of $\{3,\ldots,n\}$ are equally likely for S'_B. Thus, the probability that $|S'_A \cap S'_B| \geq C$ is at most

$$\binom{s'_B}{C}\left(\frac{s'_A}{n-2}\right)^C \leq \left(\frac{s'_A s'_B}{n-2}\right)^C. \tag{2}$$

(since the expression on the left is an upper bound on the probability that any collection of C elements in S'_B all coincide with elements of S'_A).

In Case 1 ($t \leq n/\log n$) we may assume that s'_A, s'_B are each at most $k\log n$, and thus (2) is at most $(\frac{2k^2\log^2 n}{n})^C$. In Case 2 ($t > n/\log n$) we may assume that $s'_A, s'_B \leq \frac{tk^2\log n}{n}$, and thus (2) is at most $(\frac{2t^2k^4\log^2 n}{n^3})^C = O(\frac{\log^{6C} n}{n^{2\gamma C}})$.

Thus all in all, we have that except with probability $O(1/n^{C/2})$ event (i) contributes at most $C(k-2)$ vertices v_j such that $\{1,2,j\}$ forms a triangle, and except with probability $O\left(\frac{\log^{6C} n}{n^{2\gamma C}}\right)$ event (ii) contributes at most C vertices v_j such that $\{1,2,j\}$ forms a triangle. This proves the lemma. □