Roadmap for Supplement In Section A we provide some additional technical preliminaries, including background on Boolean circuits. In Section B we give the deferred proofs from Section 3 for our main result, Theorem 3.1. In Section C.1 we give background on circuit lower bounds for TC^0, and in Section C we prove Theorem C.2 which effectively says that if one could exhibit generators with even logarithmic stretch which can fool constant-depth, sufficiently superlinear-size ReLU networks, then this would imply breakthrough circuit lower bounds.

A Additional Technical Preliminaries

More Notation and Miscellaneous Tools Let $\mathbf{1}_n$ denote the all-ones vector in n dimensions; when n is clear from context, we denote this by $\mathbf{1}$.

Given a vector $v \in \mathbb{R}^d$, we let $\|v\|$ denote its Euclidean norm. Given $r > 0$, let $B(v, r) \subset \mathbb{R}^d$ denote the Euclidean ball of radius r with center v. Given a matrix $\mathbf{W}$, we let $\|\mathbf{W}\|$ denote its operator norm. Let $\sigma_{\min}(\mathbf{W})$ denote its minimum singular value.

Given a distribution p, let $p^{\otimes n}$ denote the product measure given by drawing n independent samples from p.

Define the function $\mathrm{sgn}(x) \triangleq \begin{cases} 1 & x \geq 0 \\ -1 & x < 0 \end{cases}$.

Let $\phi : \mathbb{R} \to \mathbb{R}$ denote the ReLU activation $\phi(z) \triangleq \max(0, z)$. Let $\psi_\lambda : \mathbb{R} \to \mathbb{R}$ denote the leaky ReLU activation $\psi_\lambda(z) = z/2 + (1/2 - \lambda)|z|$. Note that

$$\psi_\lambda(z) = (1 - \lambda)\phi(z) - \lambda\phi(-z).$$

We will need the following well-known result:

Theorem A.1 (Kirszbraun extension)**.** *Given an arbitrary subset $S \subset \mathbb{R}^d$ and $f : S \to \mathbb{R}$ which is L-Lipschitz, there exists an L-Lipschitz extension $\widetilde{f} : \mathbb{R}^d \to \mathbb{R}$ for which $\widetilde{f}(y) = f(y)$ for all $y \in S$.*

We will need the following basic fact about composing Lipschitz functions:

Fact A.2. *If $g_1, \ldots, g_r : \mathbb{R}^d \to \mathbb{R}$ are Λ-Lipschitz and $h : \mathbb{R}^r \to \mathbb{R}$ is Λ'-Lipschitz, then the function*

$$x \mapsto h(g_1(x), \ldots, g_r(x))$$

is $\Lambda\Lambda'\sqrt{r}$-Lipschitz.

Proof. For any x, x', we have $|g_i(x) - g_i(x')| \leq \Lambda\|x - x'\|$, so $\left(\sum_{i=1}^r (g_i(x) - g_i(x'))^2\right)^{1/2} \leq \Lambda\sqrt{r}\|x - x'\|$. This implies that $|h(g_1(x), \ldots, g_r(x)) - h(g_1(x'), \ldots, g_r(x'))| \leq \Lambda\Lambda'\sqrt{r}\|x - x'\|$ as desired. □

Fact A.3. *The volume of a d-dimensional Euclidean ball of radius 1 is at most $(18/d)^{d/2}$.*

Proof. It is a standard fact that the volume of the ball can be expressed as $2^d \cdot \frac{(\pi/2)^{\lfloor d/2 \rfloor}}{d!!}$. If d is even, then $d!! = 2^{d/2} \cdot (d/2)! \geq 2^{d/2} \cdot e\left(\frac{d}{2e}\right)^{d/2} \geq (d/e)^{d/2}$. If d is odd, then $d!! = \frac{d!}{\lfloor d/2 \rfloor!! \cdot 2^{\lfloor d/2 \rfloor}} \geq \frac{e(d/e)^d}{e(d/2e)^{d/2} \cdot 2^{d/2}} = (d/e)^{d/2}$. We conclude that the volume is at most $(2\pi e/d)^{d/2} \leq (18/d)^{d/2}$. □

A.1 Concentration of Measure

We will use the following consequence of McDiarmid's inequality:

Lemma A.4 (McDiarmid's Inequality)**.** *Suppose $F : \{\pm 1\}^n \to \{\pm 1\}$ is such that for any $x, x' \in \{\pm 1\}^n$ differing on exactly one coordinate, $|F(x) - F(x')| \leq c$. Then*

$$\mathop{\mathbb{P}}_{x \sim \{\pm 1\}^n}[|f(x) - \mathbb{E}[f(x)]| > s] \leq \exp\left(-\frac{2s^2}{nc^2}\right).$$

Corollary A.5. *Given $F : \{\pm 1\}^n \to \{\pm 1\}$ which is Λ-Lipschitz, define the random variable $X \triangleq F(U_n)$. Then $X - \mathbb{E}[X]$ is $\Lambda\sqrt{2n}$-sub-Gaussian.*

Proof. Because F is Lipschitz, it satisfies the hypothesis of Lemma A.4 with $c = \Lambda$, so the corollary follows by the definition of sub-Gaussianity. □

Theorem A.6 (Theorem 1.1, (Rudelson & Vershynin, 2009)). *For $n, d \in \mathbb{N}$ with $n \geq d$, let $\mathbf{W} \in \mathbb{R}^{n \times d}$ be a random matrix whose entries are independent draws from $\mathcal{N}(0, 1)$. Then for every $\epsilon > 0$,*

$$\mathbb{P}\Big[\sigma_{\min}(\mathbf{W}) \leq \epsilon(\sqrt{n} - \sqrt{d-1})\Big] \leq (C\epsilon)^{n-d+1} + e^{-cn}$$

for absolute constants $C, c > 0$.

A.2 Boolean Circuits

In the context of pseudorandom generators, the set of all polynomial-sized Boolean circuits is the canonical family of discriminator functions to consider when formalizing what it means for a generator to fool all polynomial-time algorithms.

Here we review some basics about Boolean circuits; for a more thorough introduction to these concepts, we refer the reader to any of the standard textbooks on complexity theory, e.g. (Arora & Barak, 2009; Sipser, 1996).

Definition 7 (Boolean circuits). *Fix a set G of logical gates, e.g. $\wedge, \vee, \neg$. A Boolean circuit C is a Boolean function $\{\pm 1\}^n \to \{\pm 1\}$ given by a directed acyclic graph with n input nodes with in-degree zero and an output node with out-degree zero, where each node that isn't an input node is labeled by some logical gate in G. Unless otherwise specified, we will take G to be $\{\wedge, \vee, \neg\}$.*

The size *S of the circuit is the number of nodes in the graph, and the* depth *D is given by the length of the longest directed path in the graph. The value of C on input $x \in \{\pm 1\}^n$ is defined in an inductive fashion: the value at a node v in the graph is defined to be the evaluation of the gate at v on the in-neighbors of v (as the graph is acyclic, this is well-defined), and the value of C on x is then the value of the output node.*

We will occasionally also be interested in the number W of wires *in the circuit, i.e. the number of edges in the graph. Note that trivially*

$$S \leq W + 1. \tag{2}$$

Definition 8 (P/poly). *Given $T : \mathbb{N} \to \mathbb{N}$, let $\mathsf{SIZE}(T(n))$ denote the family of sequences of Boolean functions $\{f_n : \{\pm 1\}^n \to \{\pm 1\}\}$ for which there exist Boolean circuits $\{C_n\}$ with sizes $\{S_n\}$ that compute $\{f_n\}$ and such that $S_n \leq T(n)$.*

Let $\mathsf{P}/\mathsf{poly} \triangleq \bigcup_{c>1} \mathsf{SIZE}(n^c)$. We refer to (sequences of) functions in P/poly as functions computable by polynomial-sized circuits*.*

The following standard fact about bounded-depth Boolean circuits will make it convenient to translate between them and neural networks.

Lemma A.7 (See Theorem 1.1 in Section 12.1 of (Wegener, 1987)). *For any Boolean circuit C of size S and depth D with gate set G, there is another circuit C' of size $D \cdot S$ and depth D with gate set G which computes the same function as C but with the additional property that for any gate in C', all paths from an input to the gate are of the same length.*

The upshot of Lemma A.7 is that for any length ℓ, we can think of the gates of C' at distance ℓ from the inputs as comprising a "layer" in the circuit.

A less combinatorial way of formulating the complexity class captured by polynomial-sized circuits is in terms of Turing machines with *advice strings*.

Fact A.8 (See e.g. Theorem 6.11 in (Arora & Barak, 2009)). *A sequence of Boolean functions $\{f_n : \{\pm 1\}^n \to \{\pm 1\}\}$ is in P/poly if and only if there exists a sequence of* advice strings *$\{\alpha_n\}$, where $\alpha_n \in \{\pm 1\}^n$ for $a_n \leq \mathrm{poly}(n)$, and a Turing machine M which runs for at most $\mathrm{poly}(n)$ steps and, for any $n \in \mathbb{N}$, takes as input any $x \in \{\pm 1\}^n$ and the advice string α_n and outputs $M(x, \alpha_n) = f_n(x)$.*

This fact will be useful for translating discriminators computed by neural networks into discriminators given by polynomial-sized Boolean circuits.

A.3 More on GANs and PRGs

In this section we fill in some additional details regarding the contents of Section 2.1. We begin with a simple remark about Definition 1.

Remark A.9. In Definition 1, if $L = 1$, then $S = 0$ and the definition specializes to linear functions. That is, $\mathcal{C}^{\tau,\Lambda}_{1,0,d}$ is simply the class of affine linear functions $F(x) = \langle w, x\rangle + b$ for $w \in \mathbb{R}^d_\tau$ and $b \in \mathbb{R}_\tau$ satisfying $\|w\| \le \Lambda$.

Next, we fill in the proof of Lemma 2.1, which we restate below for the reader's convenience.

Lemma A.10. *Let $J : \mathbb{R}^s \to \mathbb{R}^r$ be a function each of whose output coordinates is computed by some network in $\mathcal{C}^{\tau_1,\Lambda_1}_{L_1,S_1,s}$, and let $f \in \mathcal{C}^{\tau_2,\Lambda_2}_{L_2,S_2,r}$. Then $f \circ J \in \mathcal{C}^{\tau,\Lambda}_{L,S,s}$ for $\tau = \max(\tau_1, \tau_2)$, $\Lambda = \Lambda_1\Lambda_2\sqrt{r}$, $L = L_1 + L_2$, and $S = (S_1 + 1)r + S_2$. Furthermore, for the network in $\mathcal{C}^{\tau,\Lambda}_{L,S,s}$ realizing $f \circ J$, the bias and weight vector entries in the output layer lie in $\mathbb{R}_{\tau_2}$.*

Proof. Suppose that the i-th output coordinate of J is computed by a neural network with weight matrices $\mathbf{W}^{(i)}_1 \in \mathbb{R}^{k^{(i)}_1 \times s}, \dots, \mathbf{W}^{(i)}_{L_1} \in \mathbb{R}^{1 \times k^{(i)}_{L_1-1}}$ and biases $b^{(i)}_1 \in \mathbb{R}^{k^{(i)}_1}, \dots, b^{(i)}_{L_1} \in \mathbb{R}$.

Define the $(\sum_{i=1}^r k^{(i)}_1) \times s$ weight matrix $\mathbf{W}_1$ by vertically concatenating the weight matrices $\mathbf{W}^{(1)}_1, \dots, \mathbf{W}^{(r)}_1$. For every $1 < j < L_1$ define the $(\sum_{i=1}^r k^{(i)}_j) \times (\sum_{i=1}^r k^{(i)}_{j-1})$ weight matrix $\mathbf{W}_j$ by diagonally concatenating the weight matrices $\mathbf{W}^{(1)}_j, \dots, \mathbf{W}^{(r)}_j$. Similarly, define the $r \times (\sum_{i=1}^r k^{(i)}_{L_1})$ matrix $\mathbf{W}_{L_1}$ by diagonally concatening the column vectors $\mathbf{W}^{(1)}_{L_1}, \dots, \mathbf{W}^{(r)}_{L_1}$. For the bias vectors in these layers, for every $1 \le j \le L_1$ define b_j to be the vector given by concatenating $b^{(1)}_j, \dots, b^{(r)}_j$.

Now suppose that f is computed by a neural network with weight matrices $\mathbf{W}_{L_1+1} \in \mathbb{R}^{k_{L_1+1} \times r}$, $\dots$, $\mathbf{W}_{L_1+L_2} \in \mathbb{R}^{1 \times k_{L_1+L_2-1}}$ and biases $b_{L_1+1} \in \mathbb{R}^{k_{L_1+1}}, \dots, b_{L_1+L_2} \in \mathbb{R}$. Then by design, for any $y \in \mathbb{R}^s$ we have

$$f(J(y)) = \mathbf{W}_{L_1+L_2}\phi(\mathbf{W}_{L_1+L_2-1}\phi(\cdots\phi(\mathbf{W}_1 y + b_1)\cdots) + b_{L_1+L_2-1}) + b_{L_1+L_2}.$$

This network has depth $L_1 + L_2$ and size

$$\left(\sum_{j=1}^{L_1-1}\sum_{i=1}^{r} k^{(i)}_j\right) + r + \sum_{j=L_1+1}^{L_1+L_2} k_j = r \cdot S_1 + r + S_2 = S.$$

The bit complexity of the entries of the weight matrices and biases are obviously bounded by $\max(\tau_1, \tau_2)$, and the Lipschitzness of the network is bounded by $\Lambda_1\Lambda_2\sqrt{r}$ by Fact A.2. □

Finally, we will also use the following standard tensorization property of Wasserstein distance later:

Fact A.11 (See e.g. Lemma 3 in (Mariucci & Reiß, 2018))**.** *If p, q satisfy $W_1(p, q) \le \epsilon$, then $W_1(p^{\otimes n}, q^{\otimes n}) \le \epsilon/\sqrt{n}$.*

A.4 Diverse Distributions: Missing Proofs

Here we fill in some missing details from Section 2.3. We first show that diverse distributions cannot be approximated by pushforwards of U_m if m is insufficiently large. This follows immediately from the definition of diversity:

Lemma A.12. *For any $0 < \beta < 1$, if $\mathcal{D}^*$ is a $(2^m, \beta)$-diverse distribution over $\mathbb{R}^d$, then for any function $G : \{\pm 1\}^m \to \mathbb{R}^d$, $W_1(G(U_m), \mathcal{D}^*) \ge \beta$.*

Proof. $G(U_m)$ is a uniform distribution on 2^m points, with multiplicity if there are multiple points in $\{\pm 1\}^m$ that map to the same point in $\mathbb{R}^d$ under G, so the claim follows by definition of diversity. □

Below we give some simple examples of diverse distributions.

Lemma A.13 (Discrete, well-separated distributions)**.** *For any $\alpha > 0$ and any $N, N' \in \mathbb{N}$ satisfying $N \le N'$. Let $\Omega \subseteq \mathbb{R}^d$ be a set of points such that for any $z, z' \in \Omega$, $\|z - z'\| \ge \alpha$. Then the uniform distribution μ on any N' points from Ω is (N, β)-diverse for $\beta = \alpha(1 - N/N')$.*

Proof. Take any discrete distribution ν supported on at most N points $y_1, \dots, y_N$ in $\mathbb{R}^d$. Consider the function $f : \mathbb{R}^d \to \mathbb{R}$: for any y in the support of ν, let $f(y) = 0$, and for any y not in the support of ν, let $f(y) = 1$. As a function from Ω to $\mathbb{R}$, where Ω inherits the Euclidean metric, f is clearly $1/\alpha$-Lipschitz over Ω. By Theorem A.1, there exists a $1/\alpha$-Lipschitz extension $\widetilde{f} : \mathbb{R}^d \to \mathbb{R}$ of f, and we have

$$|\mathbb{E}[f(\mu)] - \mathbb{E}[f(\nu)]| = |\mathbb{E}[f(\mu)]| \ge 1 - N/N',$$

so $W_1(\mu, \nu) \ge 1 - N/N'$ as desired. □

We now turn to examples of continuous distributions which are diverse. We first observe that a distribution is (N, β)-diverse if it satisfies certain small-ball probability bounds.

Definition 9. *For a distribution $\mathcal{D}$ over $\mathbb{R}^d$, define the Lévy concentration function $Q_{\mathcal{D}}(r) \triangleq \sup_{x' \in \mathbb{R}^d} \mathbb{P}_{x \sim \mathcal{D}}[\|x - x'\| \le r]$.*

Lemma A.14. *If a distribution $\mathcal{D}$ over $\mathbb{R}^d$ satisfies $Q_{\mathcal{D}}(r) \le \alpha$, then $\mathcal{D}$ is $(N, r(1 - N\alpha))$-diverse.*

Proof. Take any N points $z_1, \dots, z_N \in \mathbb{R}^d$. By the bound on $Q_{\mathcal{D}}(r)$, the union S of the balls of radius r around these points has Lebesgue measure at most $N\alpha$. Define the function $f : \{z_1, \dots, z_N\} \cup (\mathbb{R}^d \backslash S) \to \{0, 1\}$ to be zero on $\{z_1, \dots, z_N\}$ and one on $\mathbb{R}^d \backslash S$. This function is $1/r$-Lipschitz on its domain, so by Theorem A.1 there is an extension $f' : \mathbb{R}^d \to \mathbb{R}$ of f which remains $1/r$-Lipschitz on its domain. Define the function $f^*(x) \triangleq |f(x)|$. Note that for μ the uniform distribution on $\{z_1, \dots, z_N\}$,

$$|\mathbb{E}[f(\mu)] - \mathbb{E}[f(\mathcal{D})]| = |\mathbb{E}[f(\mathcal{D})]| \ge 1 - N\alpha,$$

so we conclude that $W_1(\mu, \mathcal{D}) \ge r(1 - N\alpha)$. □

Lemma A.15 (Uniform distribution on box)**.** *I_d is $(N, 1/2)$-diverse for $N \le \frac{1}{2}(d/18)^{d/2}$.*

Proof. By Fact A.3, $Q_{I_d}(r) \le (18r^2/d)^{d/2}$. Taking $r = 1$ and applying Lemma A.14 allows us to conclude that $W_1(\mu, I_d) \ge 1 - N(18/d)^{d/2}$, from which the lemma follows. □

As we stated in Lemma 2.3, a large family of pushforwards of the uniform distribution over $[0, 1]^d$ are similarly diverse. Below we restate this lemma and provide a complete proof.

Lemma A.16 (Random expansive leaky ReLU networks)**.** *Let $\gamma > 0$. For $k_0, \dots, k_L \in \mathbb{N}$ satisfying $k_i \ge (1 + \gamma)k_{i-1}$ for all $i \in [L]$, let $\mathbf{W}_1 \in \mathbb{R}^{k_1 \times k_0}, \mathbf{W}_2 \in \mathbb{R}^{k_2 \times k_1}, \dots, \mathbf{W}_L \in \mathbb{R}^{k_L \times k_{L-1}}$ be random weight matrices, where every entry of $\mathbf{W}_i$ is an independent draw from $\mathcal{N}(0, 1/k_i)$. For the function $F : \mathbb{R}^{k_0} \to \mathbb{R}^{k_L}$ given by*

$$F(x) \triangleq \mathbf{W}_L \psi_\lambda \left(\mathbf{W}_{L-1} \psi_\lambda \left(\cdots \psi_\lambda(\mathbf{W}_1 x) \cdots \right) \right),$$

we have that $F(I_{k_0})$ is $(2^m, \beta)$-diverse for

$$m = (k_0/2)\log(k_0/2) - k_L/\gamma - 1$$

$$\beta = \frac{1}{6}\left(\frac{\lambda}{2e^{1/\gamma}}\right)^L.$$

So for example, if γ, λ, L are constants, then $F(I_{k_0})$ is $(2^{\Omega(k_0 \log k_0)}, \Omega(1))$-diverse for k_0 sufficiently large.

We will prove this inductively by first arguing that pushing anticoncentrated distributions through leaky ReLU (Lemma A.17) or through mildly "expansive" random linear functions (Lemma A.18) preserves anticoncentration to some extent:

Lemma A.17. *Let $0 < \lambda \leq 1/2$. If a distribution $\mathcal{D}$ over $\mathbb{R}^d$ satisfies $Q_{\mathcal{D}}(r) \leq \alpha$, then the pushforward $\mathcal{D}' \triangleq \psi_\lambda(\mathcal{D})$ also satisfies $Q_{\mathcal{D}'}(\lambda r) \leq 2^d\alpha$, where here $\psi_\lambda(\cdots)$ denotes entrywise application of the leaky ReLU activation.*

Proof. Consider any ball $B(\nu, \lambda r)$ in $\mathbb{R}^d$. Take any orthant K_S of $\mathbb{R}^d$, given by points whose i-th coordinates are nonnegative for $i \in S$ and negative for $i \notin S$. Let B_S be the intersection of B with this orthant. Then $\psi_\lambda^{-1}(B_S)$ consists of points $z \in K_S$ for which

$$\sum_{i \in S}(\lambda z_i - \nu_i)^2 + \sum_{i \notin S}((1-\lambda)z_i - \nu_i)^2 \leq \lambda^2 r^2. \tag{3}$$

We can rewrite the left-hand side of (3) as

$$\lambda^2 \sum_{i \in S}(z_i - \nu_i\lambda)^2 + (1-\lambda)^2 \sum_{i \notin S}(z_i - \nu_i/(1-\lambda))^2 \geq \lambda^2 \|z - \nu(S)\|^2,$$

where in the last step we used $\lambda \leq 1/2$ and define the vector $\nu^S \in \mathbb{R}^d$ by

$$\nu_i^S = \begin{cases} \nu_i^S/\lambda & i \in S \\ \nu_i^S/(1-\lambda) & i \notin S \end{cases}.$$

In other words, $\psi_\lambda^{-1}(B_S)$ is contained in $K_S \cap B(\nu(S), r)$. In particular,

$$\psi_\lambda^{-1}(B) \subset \bigcup_S K_S \cap B(\nu(S), r),$$

so $\mathbb{P}_{x \sim \mathcal{D}'}[x \in B] \leq 2^d \cdot \alpha$ by a union bound. □

Lemma A.18. *Suppose $n, d \in \mathbb{N}$ satisfy $n \geq (1+\gamma)d$ for some $\gamma > 0$. Let $\mathbf{W} \in \mathbb{R}^{n \times d}$ be a matrix whose entries are independent draws from $\mathcal{N}(0, 1/n)$. If a distribution $\mathcal{D}$ over $\mathbb{R}^d$ satisfies $Q_{\mathcal{D}}(r) \leq \alpha$, then for the linear map $f : x \mapsto \mathbf{W}x$, the pushforward $\mathcal{D}' \triangleq f(\mathcal{D})$ satisfies $Q_{\mathcal{D}'}\left(\frac{\gamma r}{2(1+\gamma)}\right) \leq \alpha$ with probability at least $1 - \exp(-\Omega(\gamma d))$.*

Proof. By Theorem A.6, for any $\epsilon > 0$ we have that $\sigma_{\min}(\mathbf{W}) \geq \epsilon \cdot \left(1 - \sqrt{\frac{d-1}{n}}\right)$ with probability at least $1 - (C\epsilon)^{n-d+1} - e^{-cn}$. Taking $\epsilon = 1/2C$ and noting that $1 - \sqrt{\frac{d-1}{n}} \geq \frac{\gamma}{1+\gamma}$, we conclude that

$$\mathbb{P}\left[\sigma_{\min}(\mathbf{W}) \geq \frac{\gamma}{2(1+\gamma)}\right] \geq 1 - \exp(-\Omega(\gamma d)).$$

Condition on this event. Now for any $\nu \in \mathbb{R}^n$, if we write ν as $\mathbf{W}\mu + \mu^\perp$ where $\mu^\perp$ is orthogonal to the column span of $\mathbf{W}$, then $\|\mathbf{W}x - \nu\|^2 = \|\mathbf{W}(x-\mu)\|^2 + \|\mu^\perp\|^2$. So $\|\mathbf{W}x - \nu\| \leq \frac{\gamma r}{2(1+\gamma)}$ implies that $\|\mathbf{W}(x-\mu)\| \leq \frac{\gamma r}{2(1+\gamma)}$. But because $\sigma_{\min}(\mathbf{W}) \geq \frac{\gamma}{2(1+\gamma)}$, we conclude that $\|x - \mu\| \leq r$, from which the lemma follows. □

We are now ready to prove Lemma 2.3:

Proof of Lemma 2.3. By Lemma A.14 it suffices to bound the Lévy concentration function. We will induct on the layers of F. For $i \in [L]$, let $F^{(i)}$ denote the sub-network

$$\mathbf{W}_i\psi_\lambda\left(\mathbf{W}_{L-1}\psi_\lambda\left(\cdots\psi_\lambda(\mathbf{W}_1 x)\cdots\right)\right),$$

and let $\mathcal{D}_i$ denote the pushforward $F^{(i)}(I_{k_0})$, which is a distribution over $\mathbb{R}^{k_i}$. We would like to apply Lemma A.18 to each of the weight matrices $\mathbf{W}_1, \ldots, \mathbf{W}_L$, so condition on the event that the lemma holds for these matrices, which happens with probability at least $1 - L\exp(-\Omega(\gamma d))$.

Recalling from Fact A.3 that $Q_{I_{k_0}}(r) \leq (18r^2/k_0)^{k_0/2}$ for any $r > 0$, we get from Lemma A.18 applied to $\mathbf{W}_1$ that $Q_{\mathcal{D}_1}\left(\frac{\gamma r}{2(1+\gamma)}\right) \leq (18r^2/k_0)^{k_0/2}$.

Suppose inductively that we have shown that $Q_{\mathcal{D}_i}(r_i) \leq \alpha_i$ for some $r_i, \alpha > 0$. Then by Lemma A.17 and Lemma A.18 applied to weight matrix $\mathbf{W}_{i+1}$, we conclude that

$$Q_{\mathcal{D}_{i+1}}(r_{i+1}) \leq \alpha_{i+1} \quad \text{for} \quad r_{i+1} = \frac{\lambda\gamma r_i}{2(1+\gamma)}, \alpha_{i+1} = 2^{k_i}\alpha_i. \tag{4}$$

Unrolling the recursion (4), we conclude that $Q_{\mathcal{D}_L}(r_L) \leq \alpha_L$ for

$$r_L = r\lambda^{L-1}\left(\frac{\gamma}{2(1+\gamma)}\right)^L \geq r \cdot \left(\frac{\lambda\gamma}{2(1+\gamma)}\right)^L \geq r \cdot \left(\frac{\lambda}{2 \cdot e^{1/\gamma}}\right)^L$$

$$\alpha_L = 2^{k_1+\cdots+k_{L-1}}(18r^2/k_0)^{k_0/2} \leq 2^{k_L/\gamma}(18r^2/k_0)^{k_0/2}, \tag{5}$$

where the inequality in (5) follows from the fact that $k_1 + \cdots + k_{L-1} \leq k_{L-1}(1+1/\gamma) \leq k_L/\gamma$. By Lemma A.14, $F(I_{k_0}) = \mathcal{D}_L$ is $(N, r_L(1 - N\alpha_L))$-diverse. The lemma follows by taking $r = 1/3$ and $2^m = N = 1/2\alpha_L$. □

B Deferred Proofs from Section 3

B.1 Proof of Corollary 3.3

Corollary 3.3 is an immediate consequence of the following which appeared in (Chen et al., 2020b):

Lemma B.1. *For any function $P : \{\pm 1\}^k \to \{\pm 1\}$, there is a collection of k weight matrices $\mathbf{W}_1, \ldots, \mathbf{W}_k$ with entries in $\mathbb{R}_{O(k)}$ for which*

$$P(x) = \mathbf{W}_k\phi(\cdots\phi(\mathbf{W}_1 x)\cdots) \tag{6}$$

for all $x \in \{\pm 1\}^k$, and for which $\|\mathbf{W}_i\| \leq O(1)$. Furthermore, the size of the network on the right-hand side of (6) is at most $O(2^k \cdot k)$.

We give a proof for completeness to make explicit the dependence of the parameters on k.

Proof. Consider the Fourier expansion $F(x) = \sum_{S\subseteq[k]} \widehat{F}[S] \prod_{i\in S} x_i$. We show how to represent each Fourier basis function $\prod_{i\in S} x_i$ as a ReLU network with at most k layers. Observe that for any $x_1, x_2 \in \{\pm 1\}$,

$$x_1 \cdot x_2 = \phi(x_1 + x_2) + \phi(-x_1 - x_2) - \phi(x_2) - \phi(-x_2), \tag{7}$$

which is a two-layer neural network of size 4 whose two weight matrices have operator norm at most 3. Suppose inductively that for some $1 \leq m < n$, there exist weight matrices $\mathbf{W}'_1, \ldots, \mathbf{W}'_m$ for which $\prod_{i=1}^m x_i = \mathbf{W}'_m\phi(\cdots\phi(\mathbf{W}'_1 x)\cdots)$ for all $x \in \{\pm 1\}^k$, that this network has size $4m$, and that $\prod_{i=1}^m \|\mathbf{W}'_i\| \leq 6^m$.

We now show how to compute $\prod_{i=1}^{m+1} x_i$. Define $\mathbf{W}''_1$ by adding the m-th standard basis vector as a new row at the bottom of $\mathbf{W}'_1$. For every $1 < i \leq m$, define $\mathbf{W}''_i$ to be the matrix given by appending a column of zeros to the right of $\mathbf{W}''_i$ and then a new row at the bottom consisting of zeros except in the rightmost entry. Note that $\|\mathbf{W}''\|_1 = \max(1, \|\mathbf{W}'_i\|)$. Define the network $F_m : \mathbb{R}^k \to \mathbb{R}^2$ by $F_m(x) = \mathbf{W}''_m\phi(\cdots\phi(\mathbf{W}''_1 x)\cdots)$.

Letting $v, e \in \mathbb{R}^2$ be the vectors $(1,1)$ and $(0,1)$, we can use (7) to conclude that

$$\begin{aligned}\prod_{i=1}^{m+1} x_i &= \phi\left(\prod_{i=1}^m x_i + x_{m+1}\right) + \phi\left(-\prod_{i=1}^m x_i - x_{m+1}\right) - \phi(x_{m+1}) - \phi(-x_{m+1}) \\ &= \phi(v^\top F_m(x)) + \phi(-v^\top F_m(x)) - \phi(e^\top F_m(x)) - \phi(-e^\top F_m(x)).\end{aligned}$$

We can thus write $\prod_{i=1}^{m+1} x_i$ as the ReLU network

$$\prod_{i=1}^{m+1} x_i = \mathbf{W}'''_{m+1}\phi(\cdots\phi(\mathbf{W}'''_1 x)\cdots) \tag{8}$$

where

$$\mathbf{W}'''_{m+1} = (1, 1, -1, -1), \mathbf{W}'''_m = \begin{pmatrix} v^\top \mathbf{W}''_m \\ -v^\top \mathbf{W}''_m \\ e^\top \mathbf{W}''_m \\ -e^\top \mathbf{W}''_m \end{pmatrix}, \mathbf{W}'''_i = \mathbf{W}''_i \quad \forall 1 \le i < m.$$

Note that the entries of any $\mathbf{W}'''_i$ are in $\{0, \pm 1\}$ and thus have bit complexity at most 2. Additionally, $\|\mathbf{W}'''_{m+1}\| \le 2$, $\mathbf{W}'''_m \le 3\|\mathbf{W}''_m\| = 3 \cdot \max(1, \|\mathbf{W}'_m\|)$, and $\|\mathbf{W}''_i\| = \max(1, \|\mathbf{W}'_i\|)$ for all $1 \le i < m$, so $\prod_{i=1}^{m+1} \|\mathbf{W}'''_i\| \le 6^{m+1}$. Furthermore, the size of the network in (8) is $4m + 4$. This completes the inductive step and we conclude that any Fourier basis function $\prod_{i \in S} x_i$ can be implemented by an $|S|$-layer ReLU network with size $4|S|$ and the product of whose weight matrices' operator norms is at most $6^{|S|}$.

In particular, as the biases in the network are zero, we can rescale the weight matrices so they have equal operator norm, in which case they each have operator norm at most $O(1)$ and entries in $\mathbb{R}_{O(k)}$

Finally note that because the Fourier coefficients are given by $\mathbb{E}[F(x) \prod_{i \in S} x_i]$, they are all multiples of $1/2^k$ and thus have bit complexity $O(k)$. The proof follows from applying Lemma B.2 to these Fourier basis functions and λ given by the Fourier coefficients of P, as $\|\lambda\| = \|P\| = 1$. □

The above proof required the following basic fact:

Lemma B.2. *Let $\tau, \tau' \in \mathbb{N}$, and let $\lambda \in \mathbb{R}^r_\tau$. Given neural networks $F_1, \dots, F_r : \mathbb{R}^d \to \mathbb{R}$ each with L layers and whose weight matrices $\{\mathbf{W}_i^{(1)}\}, \dots, \{\mathbf{W}_i^{(r)}\}$ have operator norm bounded by some $R > 0$ and entries in $\mathbb{R}_{\tau'}$, their linear combination $\sum_i \lambda_i F_i$ is a neural network with L layers, size given by the sum of the sizes of $F_1, \dots, F_r$, and weight matrices $\mathbf{W}_1, \dots, \mathbf{W}_L$ with entries in $\mathbb{R}_{O(\tau+\tau')}$ and satisfying $\|\mathbf{W}_1\| \le R\sqrt{r}$, $\|\mathbf{W}_L\| \le R\|\lambda\|$, and $\|\mathbf{W}_i\| \le R$ for all $1 < i < L$. Here $\lambda \in \mathbb{R}^r$ is the vector with entries λ_i.*

Proof. Denote the i-th weight matrix of F_j by $\mathbf{W}_i^{(j)}$. Define $\mathbf{W}_1$ to be the vertical concatenation of $\mathbf{W}_1^{(1)}, \dots, \mathbf{W}_1^{(r)}$, and for every $1 < i < L$, define $\mathbf{W}_i$ to be the block diagonal concatenation of $\mathbf{W}_i^{(1)}, \dots, \mathbf{W}_i^{(r)}$. Finally, define $\mathbf{W}_L$ to be the row vector given by the product

$$\lambda^\top \begin{pmatrix} \mathbf{W}_L^{(1)} & 0 & \cdots & 0 \\ 0 & \mathbf{W}_L^{(2)} & \cdots & 0 \\ \vdots & 0 & \ddots & 0 \\ 0 & 0 & \cdots & \mathbf{W}_L^{(r)} \end{pmatrix}$$

For all $1 < i < L$, $\|\mathbf{W}_i\| \le \max_{j \in [r]} \|\mathbf{W}^{(i)}\|$, and additionally $\|\mathbf{W}_1\|^2 \le \sum_{j=1}^r \|\mathbf{W}_1^{(j)}\|^2$ and $\|\mathbf{W}_L\| \le \|\lambda\| \max_{j \in [r]} \|\mathbf{W}_L^{(j)}\|$. □

B.2 Proof of Lemma 3.4

We will need the following helper lemma about means of truncations of sub-Gaussian random variables:

Lemma B.3. *If Z is σ^2-sub-Gaussian and mean zero, then for any interval $I = [a, b]$ with $a \le 0 \le b$, we have $|\mathbb{E}[Z \cdot \mathbb{1}[Z \notin I]]| \le O(b - a + \sigma) \cdot \exp(-\min(-a, b)^2 / 2\sigma^2)$.*

Proof. Define the random variable $Z' = Z \cdot \mathbb{1}[Z \notin I]$. Then by integration by parts,

$$\begin{aligned}
\mathbb{E}[Z'] &\le \mathbb{E}[Z \cdot \mathbb{1}[Z > b]] \\
&= \int_0^\infty \mathbb{P}[Z' > t] dt \\
&= b\,\mathbb{P}[Z > b] + \int_b^\infty \mathbb{P}[Z > t] dt \\
&\le b \exp(-b^2/2\sigma^2) + O(\sigma \cdot \exp(-b^2/2\sigma^2)) \\
&\le O(b + \sigma) \cdot \exp(-b^2/2\sigma^2).
\end{aligned}$$

and similarly, $\mathbb{E}[Z'] \ge \mathbb{E}[Z \cdot \mathbb{1}[Z < -a]] \ge O(a - \sigma) \cdot \exp(-b^2/2\sigma^2)$, completing the proof. □

We now complete the proof of Lemma 3.4.

Proof of Lemma 3.4. Without loss of generality we can assume that $\mathbb{E}[X] = 0$ and $\mathbb{E}[Y] = \alpha$. If $\alpha \ge c\sigma$ for some sufficiently large absolute constant, then we can simply take $t = \alpha/2$ and get that $|\,\mathbb{P}[X > t] - \mathbb{P}[Y > t]| \ge 1/2$. Now suppose $\alpha < c\sigma$, and let $I = [-r, r+\alpha]$ for $r = \sigma\sqrt{\log(C\sigma/\alpha)}$ for some large constant $C > 0$. Note that by this choice of r,

$$r \exp(-r^2/2\sigma^2) \le O(\alpha),$$

where the constant factor can be made arbitrarily small by picking C sufficiently lage. Define the random variables $X' \triangleq X \cdot \mathbb{1}[X \in I]$ and $Y' \triangleq Y \cdot \mathbb{1}[Y \in I]$. Then

$$\alpha = \mathbb{E}[Y] - \mathbb{E}[X] = \mathbb{E}[Y'] - \mathbb{E}[X'] + \mathbb{E}[Y \cdot \mathbb{1}[Y \notin I]] - \mathbb{E}[X \cdot \mathbb{1}[X \notin I]]. \tag{9}$$

By Lemma B.3,

$$\mathbb{E}[X \cdot \mathbb{1}[X \notin I]] \le O(2r + \alpha + \sigma) \cdot \exp(-(r+\alpha)^2/2\sigma^2) \le O(r) \cdot \exp(-r^2/2\sigma^2) \le O(\alpha) \tag{10}$$

and similarly

$$\begin{aligned}
\mathbb{E}[Y \cdot \mathbb{1}[Y \notin I]] &\le \mathbb{E}[Y] \cdot \mathbb{P}[Y \notin I] + O(2r + \alpha + \sigma) \cdot \exp(-(r+\alpha)^2/2\sigma^2). \\
&\le 2\alpha \exp(-r^2/2\sigma^2) + O(r) \cdot \exp(-(r+\alpha)^2/2\sigma^2) \le O(\alpha).
\end{aligned} \tag{11}$$

Additionally, we have

$$\mathbb{E}[X'] - \mathbb{E}[Y'] = \int_0^{\alpha+r} (\Phi_{Y'}(z) - \Phi_{X'}(z)) dz - \int_{-\alpha}^0 (\Phi_{X'}(z) - \Phi_{Y'}(z)) dz \tag{12}$$

where $\Phi_Z(z)$ denotes the cdf at z of random variable Z. Putting (9), (10), (11), (12) together, we conclude that

$$\min\left(\int_0^{\alpha+r} (\Phi_{Y'}(z) - \Phi_{X'}(z)) dz, \int_{-\alpha}^0 (\Phi_{X'}(z) - \Phi_{Y'}(z)) dz\right) \ge \Omega(\alpha),$$

where the constant factor can be made arbitrarily close to 1/2 by making C sufficiently small. By averaging, we conclude that there exists $t \in [-\alpha, \alpha + r]$ for which

$$|\,\mathbb{P}[X' > t] - \mathbb{P}[Y' > t]| \ge \Omega(\alpha/r).$$

But $\mathbb{P}[X \notin I], \mathbb{P}[Y \notin I] \le O(\exp(-r^2/2\sigma^2)) \le O(\alpha/r)$, where the absolute constant can be made arbitrarily small by making C sufficiently small. The claim follows by a union bound, recalling the definition of X', Y'. □

B.3 Thresholds of Networks as Circuits

In the proof of Theorem 3.2, we also need the following basic fact that signs of ReLU networks can be computed in $\mathsf{P/poly}$.

Lemma B.4. *For any $f \in \mathcal{F}^*$, there is a Turing machine that, given any input y, outputs $\operatorname{sgn}(f(y))$ after $\operatorname{poly}(d)$ steps.*

Proof. Recall that the weight matrices $\mathbf{W}_1, \dots, \mathbf{W}_L$ of f have entries in $\mathbb{R}_\tau$ for $\tau = \mathrm{poly}(d)$. So for any $1 \le \ell \le L$, diagonal matrices $\mathbf{D}_1 \in \{0,1\}^{k_1 \times k_1}, \dots, \mathbf{D}_{\ell-1} \in \{0,1\}^{k_{\ell-1} \times k_{\ell-1}}$, and vector $y \in \{\pm 1\}^d$, every entry of the vector

$$\mathbf{W}_\ell \mathbf{D}_{\ell-1}(\mathbf{W}_{\ell-1}\mathbf{D}_{\ell-2}(\cdots(\mathbf{W}_1 y + b_1)\cdots) + b_{\ell-1}) + b_\ell$$

has bit complexity bounded by

$$\log_2\left(\ell \cdot 2^{O(\ell\tau)} \prod_{i=1}^{\ell-1} k_i\right) = O(\ell\tau + S) = \mathrm{poly}(d),$$

where in the second step we used that $\log(k_i) \le k_i$ for all $i \in [\ell - 1]$. So for any input to f, every intermediate activation has $\mathrm{poly}(d)$ bit complexity.

The Turing machine we exhibit for computing $\mathrm{sgn}(f(y))$ will compute the activations in the network layer by layer. The entries of $\mathbf{W}_1 y + b_1$ can readily be computed in $\mathrm{poly}(d)$ time. Now given the vector of activations

$$v = \mathbf{W}_\ell \phi(\cdots \phi(\mathbf{W}_1 y + b_1)\cdots) + b_\ell$$

for some $\ell \ge 1$ (where v is represented on a tape of the Turing machine as a bitstring of length $\mathrm{poly}(d)$), we need to compute $\mathbf{W}_{\ell+1}\phi(v) + b_{\ell+1}$. The ReLU activation can be readily computed in $\mathrm{poly}(d)$ time, so in $\mathrm{poly}(d)$ additional steps we can form this new vector of activations at the $(\ell+1)$-layer. So within $S \cdot \mathrm{poly}(d) = \mathrm{poly}(d)$ steps the Turing machine will have written down $f(y)$ (represented as a bitstring of length $\mathrm{poly}(d)$) on one of its tapes, after which it will return the sign of this quantity. □

B.4 Proof of Theorem 3.2

We now give a complete proof of Theorem 3.2:

Proof. The parameter m will be clear from context in the following discussion, so for convenience we will refer to $d(m)$ and G_m as d and G. Let k, P, G be such that the outcome of Assumption 1 holds, and $\mathsf{negl}(\cdot)$ denote the function indicating the extent to which G fools poly-sized circuits. By Corollary 3.3, every output coordinate of G is computable by a network in $\mathcal{C}^{\tau,\Lambda}_{L,S,m}$ for $\tau = O(k), \Lambda = \exp(O(k)), L = k, S = O(2^k k)$.

We first check that $W_1(G(U_m), U_d) > 1/3$. Note that $G(U_m)$ has support of size 2^m. In Lemma A.13 we can take $\mu = U_d$ and conclude that μ is $(2^m, 2(1 - 2^{m-d}))$-diverse, so $W_1(G(U_m), U_d) \ge 2(1 - 2^{m-d}) = 2(1 - 2^{m-m^c}) \ge 1$.

It remains to check that G fools $\mathcal{F}^*$ relative to U_d. Suppose to the contrary that there exists some $f \in \mathcal{F}^*$ and absolute constant $a > 0$ for which $|\mathbb{E}[f(G(U_m))] - \mathbb{E}[f(U_d)]| > 1/d^a$. We will argue that this implies there is a poly-sized circuit $C : \{\pm 1\}^d \to \{\pm 1\}$ distinguishing $G(U_m)$ from U_d.

First note that for any threshold $t \in \mathbb{R}_\tau$, by Lemma B.4 there is a Turing machine $\mathcal{M}_\tau : \{\pm 1\}^d \to \{\pm 1\}$ that computes $y \mapsto \mathrm{sgn}(f(y) - t)$ with τ bits of advice. So if there existed a threshold $t \in \mathbb{R}_\tau$ for which

$$|\mathbb{E}[\mathcal{M}_\tau(G(U_m))] - \mathbb{E}[\mathcal{M}_\tau(U_d)]| > 1/d^{a'}, \tag{13}$$

for some constant $a' > 0$, then by Fact A.8, there would exist a Boolean circuit C distinguishing $G(U_m)$ from U_d with non-negligible advantage, contradicting Assumption 1 and concluding the proof.

We will apply Lemma 3.4 to show the existence of such a threshold t. Specifically, define random variables $X = f(G(U_m))$ and $Y = f(U_d)$. By Corollary A.5 applied to the $\mathrm{poly}(d)$-Lipschitz function $f : \{\pm 1\}^d \to \{\pm 1\}$, $Y - \mathbb{E}[Y]$ is $\mathrm{poly}(d)$-sub-Gaussian. And recalling that $G \in \mathcal{C}^{\tau,\Lambda}_{L,S,m}$ for $\Lambda = \exp(O(k))$, we can apply Corollary A.5 to the $\mathrm{poly}(d) \cdot O_k(1)$-Lipschitz function $f \circ G : \{\pm 1\}^m \to \{\pm 1\}$ to conclude that $X - \mathbb{E}[X]$ is σ^2-sub-Gaussian for $\sigma \triangleq \mathrm{poly}(m) \cdot \exp(O(k)) = \mathrm{poly}(d)$. By Lemma 3.4, there exists a threshold t for which the left-hand side of (13) exceeds $\min(1/2, \widetilde{\Omega}(n^{-a}/\sigma))$, which is not negligible.

It remains to verify that t has bit complexity at most $\mathrm{poly}(d)$. As the entries in the weight matrices and biases in f all have bit complexity $\mathrm{poly}(d)$ and f has size and depth $\mathrm{poly}(d)$, $f(y)$ has bit

complexity $\mathrm{poly}(d)$ for any $y \in \{\pm 1\}^d$. Similarly, the entries in the weight matrices and biases in G all have bit complexity $O(k) = O(1)$, so $f(G(x))$ has bit complexity $\mathrm{poly}(d)$ for any $x \in \{\pm 1\}^m$. By the bound on t in Lemma 3.4 and our bound on σ above, t therefore also has $\mathrm{poly}(d)$ bit complexity. □

B.5 Proof of Lemma 3.6

Proof. Suppose to the contrary that there existed some function $f \in \mathcal{F}'$ for which

$$|\mathbb{E}[f(J(p))] - \mathbb{E}[f(J(q))]| > 2\epsilon \cdot \Lambda'.$$

By Lemma 2.1 and our choice of S'', the composition $f \circ J : \mathbb{R}^s \to \mathbb{R}$ can be computed by a network in $\mathcal{C}^{\tau, \Lambda\Lambda'\sqrt{r}}_{L,S,s}$ whose bias and weight vector entries in the output layer lie in $\mathbb{R}_{\tau'}$.

We first show why this would lead to a contradiction. Consider the function $h \triangleq \frac{1}{C} \cdot f \circ J$ for

$$C = 2^{\lceil \log_2 \Lambda' \sqrt{r} \rceil} \in [\Lambda', 2\Lambda'),$$

which can be computed by taking the network computing $f \circ J$ and scaling the bias and weight vector in the output layer by C. Note that this scaling results in bias and weight vector entries in the output layer for h with bit complexity $\tau' + \lceil \log_2 \Lambda' \sqrt{r} \rceil = \tau$. Furthermore, h is $\Lambda\Lambda'\sqrt{r}/C \leq \Lambda$-Lipschitz, so $h \in \mathcal{C}^{\tau,\Lambda}_{L,S,s}$. On the other hand, we would have

$$|\mathbb{E}[h(p)] - \mathbb{E}[h(q)]| > 2\epsilon\Lambda'\sqrt{r}/C \geq \epsilon,$$

yielding the desired contradiction of the assumption that $W_{\mathcal{F}}(p, q) \leq \epsilon$. □

B.6 Proof of Fact 3.7

Proof. Note that $h(U_n)$ is the uniform distribution over multiples of $1/2^n$ in the interval $[0, 1)$. Given any such multiple z, let p_z denote the uniform distribution over $[z, z + 1/2^n)$. One way of sampling from I_1 is thus to sample z from $h(U_n)$ and then sample from p_z.

Now consider any 1-Lipschitz function $f : \mathbb{R} \to \mathbb{R}$. Note that for any z' in the support of p_z, $|f(z) - f(z')| \leq 1/2^n \leq \epsilon$. We have

$$|\mathbb{E}[f(h(U_n))] - \mathbb{E}[f(I_1)]| = \left| \mathop{\mathbb{E}}_{z \sim h(U_n)} \left[\mathop{\mathbb{E}}_{z' \sim p_z} [f(z) - f(z')] \right] \right| \leq \epsilon$$

as desired. □

B.7 Proof of Lemma 3.8

Proof. Let $n \triangleq \lceil \log(1/\epsilon) \rceil$. For every $i \in [r]$, define $S_i \triangleq \{(i-1) \cdot n + 1, \ldots, i \cdot n\}$. Take J to be the linear function where for every $i \in [r]$, the i-th output coordinate of J is the linear function which maps $y \in \mathbb{R}^s$ to $\langle w_i, y + \mathbf{1} \rangle$ where w_i is zero outside of S_i and, over coordinates indexed by S_i, equal to the vector $(1/4, \ldots, 1/2^{n+1})$. Note that each output coordinate of J is computed by a function in $\mathcal{C}^{n+1, O(1)}_{1,0,s}$.

By Fact 3.7 and Fact A.11,

$$W_{\mathcal{F}^*}(J(U_s), I_r) \leq \mathrm{poly}(r) \cdot W_1(J(U_s), I_r) \leq \mathrm{poly}(r) \cdot \epsilon.$$

On the other hand, by Lemma 3.6 and the fact that the union of $\mathcal{C}^{\tau - \lceil \log_2 \Lambda\sqrt{r} \rceil, \Lambda}_{L-1, S-r, r}$ over $\tau, \Lambda, L, S = \mathrm{poly}(s)$ is still $\mathcal{F}^*$, we conclude that

$$W_{\mathcal{F}^*}(J(\widetilde{\mathcal{D}}), J(U_s)) \leq O(\epsilon\sqrt{r}),$$

from which the lemma follows by triangle inequality. □

B.8 Proof of Lemma 3.9

Proof. Let $J : \mathbb{R}^s \to \mathbb{R}^r$ be given by Lemma 3.8. We know that $W_{\mathcal{F}^*}(J(\widetilde{\mathcal{D}}), I_r) \le \epsilon \cdot \mathrm{poly}(r)$. By Lemma 3.6 applied to these two distributions and the generator function H, together with the fact that the union of $\mathcal{C}^{\tau'-\lceil \log_2 \Lambda\sqrt{d}\rceil,\Lambda}_{L-L',S-d(S'+1),d}$ over $\Lambda, L, S = \mathrm{poly}(s)$ is still $\mathcal{F}^*$, we thus have that $W_{\mathcal{F}^*}(H(J(\widetilde{\mathcal{D}})), H(I_r)) \le O(\epsilon\Lambda'\sqrt{d}) \cdot \mathrm{poly}(r) = \epsilon\Lambda' \cdot \mathrm{poly}(s)$.

We will thus take J' in the lemma to be $H \circ J$. By Lemma 2.1, each output coordinate of J' is computed by a function in $\mathcal{C}^{\max(O(\log(1/\epsilon)),\tau'),O(\Lambda'\sqrt{d})}_{L'+1,r+S',s}$ as claimed. □

B.9 Proof of Theorem 3.5

Proof of Theorem 3.5. The parameter m will be clear from context in the following discussion, so for convenience we will refer to $r(m), d(m), \epsilon(m), H_m, G_m$ as r, d, ϵ, H, G, and similarly for the network parameters τ', Λ', L', S'.

It is easy to verify condition 3 before we even define G: because $G(U_m)$ is a uniform distribution on 2^m points (with multiplicity) and $H(I_r)$ is $(2^m, \Omega(1))$-diverse, $W_1(G(U_m), I_d) \ge \Omega(1)$ as claimed.

Let $s = r \cdot \lceil \log(1/\epsilon) \rceil$. As we are assuming $\epsilon \ge \exp(-O(m))$, $s \le r \cdot m = m^c$ for some constant $c > 1$. If $s \le m$, then define $G' : \mathbb{R}^m \to \mathbb{R}^s$ to be the map given by projecting to the first s coordinates so that $G'(U_m)$ and U_s are identical as distributions. Otherwise, take G' to be the generator $G : \mathbb{R}^m \to \mathbb{R}^s$ constructed in Theorem 3.2, recalling that $W_{\mathcal{F}^*}(G'(U_m), U_s) \le \mathsf{negl}(m) \le \epsilon$.

Next, by applying Lemma 3.9 to $\widetilde{\mathcal{D}} = G'(U_m)$, we get a function $J' : \mathbb{R}^s \to \mathbb{R}^d$ each of whose output coordinates is computed by a function in $\mathcal{C}^{\max(O(\log(1/\epsilon)),\tau'),O(\Lambda'\sqrt{d})}_{L'+1,r+S',s}$ such that

$$W_{F_d}(J'(G'(U_m)), H(I_r)) \le \epsilon\Lambda' \cdot \mathrm{poly}(m) \le \epsilon \cdot \mathrm{poly}(m), \tag{14}$$

where the second step follows by our assumption on Λ'.

We will take $G \triangleq J' \circ G'$. (14) establishes condition 2 of the theorem. Finally, by Lemma 2.1, every output coordinate of G can be realized by a network in $\mathcal{C}^{\tau,\Lambda}_{L,S,m}$ for $\tau = \max(O(\log(1/\epsilon)), \tau', O(1))$, $\Lambda = O(\Lambda'\sqrt{ds}) = O(\Lambda') \cdot \mathrm{poly}(m)$, $L = L' + O(1)$, and $S = O(s) + r + S' = O(s) + S'$ (where we used the fact that $r = s/\lceil \log(1/\epsilon) \rceil < s$). This establishes condition 1 of the theorem. □

B.10 Proof of Lemma 3.10

Proof. Note that

$$h_\xi(x) = \phi(x/\xi + 1) - \phi(x/\xi - 1) - 1, \tag{15}$$

so we can take weight matrices

$$\mathbf{W}_1 = \begin{pmatrix} 1/\xi \\ 1/\xi \end{pmatrix} \qquad \mathbf{W}_2 = \begin{pmatrix} 1 & -1 \end{pmatrix}$$

and biases $b_1 = (1, -1)$ and $b_2 = -1$. Note that h_ξ is $1/\xi$-Lipschitz. We conclude that $h_\xi \in \mathcal{C}^{\tau,1/\xi}_{2,2,1}$. □

B.11 Proof of Lemma 3.11

Proof. We first verify that $W_1(G_0(U_m), G(\gamma_m)) \le \epsilon$. Take any 1-Lipschitz function f. Note that we can sample from U_m by sampling a vector g from γ_m, applying h_ξ entrywise to g, and replacing each resulting entry of $h_\xi(g)$ by its sign; importantly, the last step only affects entries $i \in [m]$ for which $|g_i| < \xi$.

We will define $\mathcal{E}$ to be the event that $|g_i| \ge \xi$ for all $i \in [m]$, noting that

$$\mathbb{P}[\mathcal{E}] \ge 1 - m \cdot \mathop{\mathbb{P}}_{g \sim \mathcal{N}(0,1)}[|g| < \xi] \ge 1 - m\xi\sqrt{2/\pi}.$$

We can thus write

$$\begin{aligned}
&|\mathbb{E}[f(G_0(U_m))] - \mathbb{E}[f(G(\gamma_m))]| \\
&\quad = \left| \mathop{\mathbb{E}}_{g\sim\gamma_m} [(f(G_0(h_\xi(g))) - f(G(g))) \cdot \mathbb{1}[\mathcal{E}] + (f(G_0(\mathrm{sgn}(h_\xi(g)))) - f(G(g))) \cdot \mathbb{1}[\mathcal{E}^c]] \right| \\
&\quad = \left| \mathop{\mathbb{E}}_{g\sim\gamma_m} [(f(G_0(\mathrm{sgn}(h_\xi(g)))) - f(G_0(h_\xi(g)))) \cdot \mathbb{1}[\mathcal{E}^c]] \right|. \qquad (16)
\end{aligned}$$

By Fact A.2, $f \circ G_0$ is $\Lambda''\sqrt{d}$-Lipschitz. Furthermore, because $h_\xi(g) \in [-1,1]^m$, $\|\mathrm{sgn}(h_\xi(g)) - h_\xi(g)\| \le \sqrt{m}$. We can thus upper bound (16) by

$$\le \Lambda''\sqrt{md} \cdot \mathbb{P}[\mathcal{E}^c] \le \Lambda''\xi\sqrt{(2/\pi)m^3 d} \le \epsilon$$

so $W_1(G_0(U_m), G(\gamma_m)) \le \epsilon$ as desired.

It remains to bound the complexity of G. For any $i \in [d]$, we can apply Lemma 2.1 with f given by the i-th output coordinate of G_0 and J given by the map which applies h_ξ to every entry of the input. We thus conclude that $G \in \mathcal{C}^{\tau,\Lambda}_{L,S,m}$ for $\tau = \max(\tau'', \log_2(1/\xi)) = \max(\tau'', O(\log(\Lambda'' md/\epsilon)))$, $\Lambda = \Lambda''\sqrt{m}/\xi = O(\Lambda''^2 m^2\sqrt{d}/\epsilon)$, $L = L'' + 2$, $S = 3m + S''$ as claimed. □

C Fooling ReLU Networks Would Imply New Circuit Lower Bounds

In this section we show that even exhibiting generators with logarithmic stretch that can fool all ReLU network discriminators of constant depth and slightly superlinear size would yield breakthrough circuit lower bounds.

First, in Section C.1 we review basics about average-case hardness and recall the state-of-the-art for lower bounds against TC^0. Then in Section C.2 we present and prove the main result of this section, Theorem C.2.

C.1 Average-Case Hardness and TC^0

One of the most common notions of hardness for a class of functions $\mathcal{F}$ is *worst-case hardness*, that is, the existence of functions which cannot be computed by functions in $\mathcal{F}$.

Definition 10 (Worst-case hardness)**.** *Given a class of Boolean functions $\mathcal{F}$, a sequence of functions $f_n : \{\pm1\}^n \to \{\pm1\}$ is* worst-case-hard *for $\mathcal{F}$ if for every $f : \{\pm1\}^n \to \{\pm1\}$ in $\mathcal{F}$, there is some input $x \in \{\pm1\}^n$ for which $f(x) \ne f_n(x)$.*

A more robust notion of hardness is that of *average-case hardness*, which implies worst-case hardness. For any $f_n \in \mathcal{F}$, rather than simply require that there is *some* input on which f and f_n disagree, we would like that over some fixed distribution over possible inputs, the probability that f and f_n output the same value is small. Typically, this fixed distribution is the uniform distribution over $\{\pm1\}^n$, but in many situations even showing average-case hardness with respect to less natural distributions is open.

Definition 11 (Average-case hardness)**.** *Given a class of Boolean functions $\mathcal{F}$, a function $\epsilon : \mathbb{N} \to [0, 1/2)$, and a sequence of distributions $\{\mathcal{D}_n\}_n$ over $\{\pm1\}^n$, a sequence of functions $f_n : \{\pm1\}^n \to \{\pm1\}$ is $(1/2+\epsilon(n))$*-average-case-hard for $\mathcal{F}$ with respect to $\{\mathcal{D}_n\}$ *if for every $f : \{\pm1\}^n \to \{\pm1\}$ in $\mathcal{F}$,*

$$\mathop{\mathbb{P}}_{x\sim\mathcal{D}_n}[f(x) = f_n(x)] \le \frac{1}{2} + \epsilon(n).$$

By a counting argument, for any reasonably constrained class $\mathcal{F}$ there must *exist* functions which are worst/average-case hard for $\mathcal{F}$. A central challenge in complexity theory has been to exhibit *explicit* hard functions for natural complexity classes. In the context of this work, by explicit we simply mean that there is a polynomial-time algorithm for evaluating the function.

The complexity class we will focus on in this section is TC^0, the class of constant-depth linear threshold circuits of polynomial size:

Definition 12 (Linear threshold circuits)**.** *A linear threshold circuit of size S and depth D is any Boolean circuit of size S and depth D whose gates come from the set G of all linear threshold functions mapping $x \in \{\pm 1\}^n$ to $\mathrm{sgn}(\langle w, x\rangle - b)$ for some arity $n \in \mathbb{N}$, vector $w \in \mathbb{R}^n$, and bias $b \in \mathbb{R}$. TC^0 is the set of all linear threshold circuits of size $\mathrm{poly}(n)$ and depth $O(1)$.*[6]

The best-known worst-case hardness result for TC^0 is that of (Impagliazzo et al., 1997) who showed:

Theorem C.1 ((Impagliazzo et al., 1997))**.** *Let $f_n : \{\pm 1\}^n \to \{\pm 1\}$ be the parity function on n bits. For any depth $D \geq 1$, any linear threshold circuit of depth D must have at least $n^{1+c\theta^{-D}}$ wires, where $c > 0$ and $\theta > 1$ are absolute constants.*

In (Chen et al., 2016) this worst-case hardness result was upgraded to an average-case hardness result with respect to the uniform distribution over the hypercube. Remarkably, this slightly superlinear lower bound from (Impagliazzo et al., 1997) has not been improved upon in over two decades!

We remark that the discussion about lower bounds for threshold circuits is a very limited snapshot of a rich line of work over many decades. We refer to the introduction in (Chen & Tell, 2019) for a more detailed overview of this literature.

C.2 Hardness Versus Randomness for GANs

For convenience, given sequences of parameters $D(m), S(m) \in \mathbb{N}$ (these will eventually correspond to the depth and size of the linear threshold circuits against which we wish to show lower bounds) let

$$\mathcal{C}_d(D, S) \triangleq \mathcal{C}^{\mathrm{poly}(d),\mathrm{poly}(d)}_{\Theta(D),3d+\Theta(S),d}.$$

This will comprise the family of ReLU network discriminators that we will focus on. We now show that if one could exhibit generators that can provably fool discriminators in $\mathcal{C}_d(D, S)$, then this would translate to average-case hardness against linear threshold circuits of depth D and size D. Formally, we show the following:

Theorem C.2. *There is an absolute constant $c > 0$ for which the following holds. Fix sequences of parameters $D(m), S(m) \in \mathbb{N}$. Suppose there is an explicit*[7] *sequence of generators $G_m : \mathbb{R}^m \to \mathbb{R}^{d(m)}$ for $d(m) \geq cm \log m$ such that $W_{\mathcal{C}_{d(m)}(D(m),S(m))}(G_m(I_m), I_{d(m)}) \leq \epsilon(m)$ for some $\epsilon(m) \geq 1/\mathrm{poly}(m)$ and such that each output coordinate of G_m is computable by a network in $\mathcal{F}^*$, then there exists a sequence of functions $h_{d(m)} : \{\pm 1\}^{d(m)} \to \{\pm 1\}$ in NP which are $(1/2+\epsilon(m)/2+m^{-\Omega(m)})$-average-case-hard with respect to some sequence of explicit distributions $\{\mathcal{D}_m\}$ for linear threshold circuits of depth $D(m)$ and size $S(m)$.*

Remark C.3. In particular, this shows that if we could exhibit explicit generators fooling all discriminators given by neural networks of polynomial Lipschitzness/bit complexity of depth $D(m)$ and size $O(d(m)^{1+\exp(-D(m)\cdot 99)})$, then by (2) we would get new average-case circuit lower bounds for TC^0. In fact it was shown by (Chen & Tell, 2019) that such a result would imply $\mathsf{TC}^0 \neq \mathsf{NC}^1$, which would be a major breakthrough in complexity. This can be interpreted in one of two ways: 1) it would be extraordinarily difficult to show that a particular generative model truly fools all constant-depth, barely-superlinear-size ReLU network discriminators, or 2) gives a learning-theoretic motivation for trying to prove circuit lower bounds.

Regarding the proof of Theorem C.2, note that the statement is closely related to existing well-studied connections between hardness and randomness in the study of pseudorandom generators. In fact, readers familiar with this literature will observe that Theorem C.2 is the GAN analogue of the "easy" direction of the equivalence between hardness and randomness: an explicit pseudorandom generator that fools some class of functions implies average-case-hardness for that class.

In order to leverage this connection however, we need to formalize the link between GANs (over continuous domains) and pseudorandom generators (over discrete domains) in the next lemma. It

[6] Sometimes TC^0 is defined with the gate set taken to consist of $\{\wedge, \vee, \neg\}$ and majority gates, though these two classes are equivalent up to polynomial overheads (Goldmann et al., 1992; Goldmann & Karpinski, 1998). Moreover, because a circuit of size S and depth D using the latter gate set is clearly implementable by a circuit of size S and depth D using the former gate set, so our lower bounds against the former gate set immediately translate to ones against the latter.

[7] By *explicit*, we mean that we are provided a way to evaluate these functions in polynomial time.

turns out that in the preceding sections we already developed most of the ingredients for establishing this connection.

Lemma C.4. *Suppose there is an explicit sequence of generators $G_m : \mathbb{R}^m \to \mathbb{R}^{d(m)}$ such that $W_{\mathcal{C}_{d(m)}(D(m),S(m))}(G_m(I_m), I_{d(m)}) \leq \epsilon(m)$ for some $\epsilon(m) = 1/\mathrm{poly}(m)$ and such that each output coordinate of G_m is computable by a network in $\mathcal{F}^*$. Then there is an explicit sequence of pseudorandom generators $G'_m : \{\pm 1\}^{n(m)} \to \{\pm 1\}^{d(m)}$ for $n(m) = \Theta(m \log m)$ that $2\epsilon(m)$-fool linear threshold circuits of depth $D(m)$ and size $S(m)$.*

Proof. As in the proofs of the theorems from Section 3, the parameter m will be clear from context, so we will drop m from subscripts and parenthetical references.

Recall the function h_ξ from Lemma 3.10; we will take $\xi = \epsilon/\mathrm{poly}(m)$. Also define $n \triangleq \Theta(\log(m/\epsilon))$ and recall from the proof of Lemma 3.8 the definition of the linear function $J : \mathbb{R}^{mn} \to \mathbb{R}^m$: for every $i \in [m]$, the i-th output coordinate of J is the linear function which maps $x \in \mathbb{R}^{mn}$ to $\langle w_i, x + \mathbf{1}\rangle$, where w_i is zero outside of indices $\{(i-1)\cdot n + 1, \ldots, i \cdot n\}$ and equal to the vector $(1/4, 1/8, \ldots, 1/2^{n+1})$ on those indices.

Given generator G fooling $\mathcal{C}_d$, we will show that the Boolean function $G' : \{\pm 1\}^{mn} \to \{\pm 1\}^d$ given by

$$G' = h_\xi \circ G \circ J$$

is a pseudorandom generator that fools TC^0 circuits. To that end, suppose there was a TC^0 circuit $f : \{\pm 1\}^d \to \{\pm 1\}$ for which $|\mathbb{E}[f(G'(U_{mn}))] - \mathbb{E}[f(U_d)]| > 2\epsilon$. We will show that this implies the existence of a ReLU network $f' \in \mathcal{C}_d(D, S)$ for which $|\mathbb{E}[f'(G(I_m))] - \mathbb{E}[f'(I_d)]| > \epsilon$.

Our proof proceeds in three steps: argue that

1. $f \circ h_\xi \in \mathcal{C}_d(D, S)$
2. $\mathbb{E}[f(U_d)] \approx \mathbb{E}[f(h_\xi(I_d))]$
3. $\mathbb{E}[G'(U_{mn})] \approx \mathbb{E}[f(h_\xi(G(I_m)))]$

Note that 2 and 3, together with the fact that f is a discriminator for G', imply that $f' \triangleq f \circ h_\xi$ is a discriminator for G. 1 then ensures that this discriminator is a ReLU network with the right complexity bounds, yielding the desired contradiction.

To show step 1, we will show that f can be computed by a network in $\mathcal{C}^{\mathrm{poly}(d),\mathrm{poly}(d)}_{O(1),\mathrm{poly}(d),d}$ and then apply Lemma 2.1 and Lemma 3.10. Suppose the threshold circuit computing f has depth D, where D is some constant. Recall from Lemma A.7 that we may assume, up to an additional blowup in size by D, that the constant-depth threshold circuit C computing f is comprised of layers $S_1, \ldots, S_D$ such that S_i consists of all gates in C for which any path from the inputs to the gate is of length i.

Let k_i denote the number of gates in S_i (where $k_D = 1$), and for each $j \in [k_i]$, suppose the linear threshold function computed by the j-th gate in S_i is given by $\mathrm{sgn}(\langle w_{i,j}, \cdot\rangle - b_{i,j})$ for $w_{i,j} \in \mathbb{R}^{k_{i-1}}$. As each linear threshold takes at most $\mathrm{poly}(d)$ bits as input, we can assume without loss of generality that $b_{i,j}$ and the entries of $w_{i,j}$ lie in $\mathbb{R}_\tau$ for $\tau = \mathrm{poly}(d)$. For this τ, note that for any $w \in \mathbb{R}^k_\tau, b \in \mathbb{R}_\tau, x \in \{\pm 1\}^k$,

$$\mathrm{sgn}(\langle w, x\rangle - b) = h_{\xi'}\left(\langle w, x\rangle - b\right),$$

for some $\xi' = 1/\mathrm{poly}(d)$, where $h_{1/\mathrm{poly}(d)}(\cdot)$ is the function defined in Lemma 3.10, and recall from the proof of Lemma 3.10 that it can be represented as a two-layer ReLU network via (15). For every $i \in [D]$, we can thus define two weight matrices $\mathbf{W}^{(1)}_i \in \mathbb{R}^{2k_i \times k_{i-1}}$ and $\mathbf{W}^{(2)}_i \in \mathbb{R}^{k_i \times 2k_i}$ by

$$\mathbf{W}^{(1)}_i = \frac{1}{\xi'} \cdot \begin{pmatrix} — & w_{i,1} & — \\ — & w_{i,1} & — \\ \vdots & \vdots & \vdots \\ — & w_{i,k_i} & — \\ — & w_{i,k_i} & — \end{pmatrix} \qquad \mathbf{W}^{(2)}_i = \begin{pmatrix} 1 & -1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -1 \end{pmatrix}$$

and biases $b_i^{(1)} \in \mathbb{R}^{2k_i}$ and $b_i^{(2)} \in \mathbb{R}^{k_i}$ by

$$b_i^{(1)} = (1, -1, 1, -1, \ldots, 1, -1) \qquad b_i^{(2)} = (-1, \ldots, -1)$$

so that for all $x \in \{\pm 1\}^d$,

$$f(x) = \mathbf{W}_D^{(2)} \phi \left(\mathbf{W}_D^{(1)} \phi \left(\cdots \phi \left(\mathbf{W}_1^{(2)} \phi \left(\mathbf{W}_1^{(1)} x + b_1^{(1)} \right) + b_1^{(2)} \right) \cdots \right) + b_D^{(1)} \right) + b_D^{(2)} \tag{17}$$

The entries of the weight matrices and bias vectors are clearly in $\mathbb{R}_{\mathrm{poly}(d)}$, and because each $h_{\xi'}$ is $\mathrm{poly}(d)$-Lipschitz and there are $D = O(1)$ layers in the circuit, the function in (17) is $\mathrm{poly}(d)$-Lipschitz as a function over $\mathbb{R}^d$. The size and depth of the network are within a constant factor of the size S and depth D of the circuit. Lemma 2.1 and Lemma 3.10 then imply that $f \circ h_\xi$ has depth $\Theta(D)$ and size $3d + \Theta(S)$, as well as Lipshitzness and bit complexity polynomial in m because $\epsilon \geq 1/\mathrm{poly}(m)$ so that $\xi \geq 1/\mathrm{poly}(m)$. Therefore, $f \circ h_\xi \in \mathcal{C}_d(D, S)$.

To show step 2, recall from Lemma 3.11 and Remark 3.12 that $W_1(U_m, h_\xi(I_m)) \leq \epsilon/\mathrm{poly}(m)$. Recalling that f is $\mathrm{poly}(d) = \mathrm{poly}(m)$-Lipschitz, we obtain the desired inequality $|\mathbb{E}[f(U_d)] - \mathbb{E}[f(h_\xi(I_d))]| \leq \epsilon/2$. Here the factor of $1/2$ is an arbitrary small constant coming from taking the $\mathrm{poly}(m)$ in the definition of ξ sufficiently large.

Finally, to show step 3, recall by Fact 3.7 that $W_1(J(U_{mn}), I_m) \leq \epsilon^2/\mathrm{poly}(m)$ by our choice of $n = \Theta(\log(m/\epsilon))$ (the ϵ^2 comes from taking the constant factor in the definition of n sufficiently large). By applying Fact A.2 to $f \circ h_\xi$ and G, we know that the composition $f \circ h_\xi \circ G$ is $\mathrm{poly}(m)/\epsilon$-Lipschitz. It follows that $|\mathbb{E}[G'(U_{mn})] - \mathbb{E}[f(h_\xi(G(I_m)))]| \leq \epsilon/2$. The factor of $1/2$ is an arbitrary small constant coming from taking the constant factor in the definition of n sufficiently large.

Putting everything together, we conclude by triangle inequality that

$$|\mathbb{E}[f(G(I_m))] - \mathbb{E}[f(I_d)]| > \epsilon,$$

a contradiction. □

The following lemma gives the standard transformation from pseudorandom generators to average-case hardness. We include a proof for completeness.

Lemma C.5 (Prop. 5 of (Viola, 2009)). *Suppose the sequence of functions $G_m : \{\pm 1\}^m \to \{\pm 1\}^{d(m)}$ $\epsilon(m)$-fools a class of Boolean functions $\mathcal{F}$. Define the function $h_{d(m)} : \{\pm 1\}^{d(m)} \to \{\pm 1\}$ by*

$$h_{d(m)}(x) = \begin{cases} 1 & \text{exists } y \in \{\pm 1\}^m \text{ such that } G(y) = x \\ -1 & \text{otherwise} \end{cases}.$$

Let $\mathcal{D}_{d(m)}$ be the distribution over $\{\pm 1\}^{d(m)}$ given by the uniform mixture between $U_{d(m)}$ and $G(U_m)$.

Then the sequence of functions $\{h_{d(m)}\}$ is $(1/2 + \epsilon'(m))$-average-case-hard for $\mathcal{F}$ with respect to $\{\mathcal{D}_{d(m)}\}$ for $\epsilon'(m) = \epsilon(m)/4 + 2^{m-d(m)-1}$.

Proof. As usual, we will omit most subscripts/parentheses referring to the parameter m. Let $f : \{\pm 1\}^d \to \{\pm 1\}$ be any function in $\mathcal{F}$. Then

$$\begin{aligned}
\mathbb{P}[f(\mathcal{D}) = h_d(\mathcal{D})] &= \frac{1}{2}\mathbb{P}[f(U_d) = h_d(U_d)] + \frac{1}{2}\mathbb{P}[f(G(U_m)) = h_d(G(U_m))] \\
&\leq \frac{1}{2}\left(\mathbb{P}[f(U_d) = 0] + \mathbb{P}[h_d(U_d) = 1]\right) + \frac{1}{2}\mathbb{P}[f(G(U_m)) = 1] \\
&\leq \frac{1}{2}\left(\mathbb{P}[f(U_d) = 0] + 2^{m-d}\right) + \frac{1}{2}\mathbb{P}[f(G(U_m)) = 1] \\
&\leq \frac{1}{2}\left(\mathbb{P}[f(U_d) = 0] + 2^{m-d}\right) + \frac{1}{2}\left(\mathbb{P}[f(U_d) = 1] + \epsilon/2\right) \\
&= \frac{1}{2} + \frac{\epsilon}{4} + 2^{m-d-1},
\end{aligned}$$

where in the second step we used a union bound and the fact that $h(G(U_m))$ is deterministically 1 by construction, in the third step we used the fact that $\mathbb{P}[h_d(U_d)] \leq 2^{m-d}$ because there are at most 2^m elements in the range of G, and in the fourth step we used the fact that G ϵ-fools functions in $\mathcal{F}$. □

We are now ready to prove Theorem C.2.

Proof of Theorem C.2. By Lemma C.4, we can construct out of the generators G_m an explicit sequence of pseudorandom generators that stretch $\Theta(m \log m)$ bits to $d(m) \geq c \cdot m \log m$ bits and $2\epsilon(m)$-fool linear threshold circuits of size $S(m)$ and depth $D(m)$. The theorem follows upon substituting this into Lemma C.5, which implies $(1/2 + \epsilon'(m))$-average-case-hardness for such circuits with respect to the explicit distributions $\mathcal{D}_{d(m)}$ defined in Lemma C.5, where $\epsilon'(m) = \epsilon(m)/2 + 2^{\Theta(m \log m) - d(m) - 1} = \epsilon(m)/2 + m^{-\Omega(m)}$, provided the absolute constant c is sufficiently large.

Finally, note that the average-case-hard functions $h_{d(m)}$ we get from Lemma C.5 are in NP because given an input x and a certificate y, one can easily verify whether $G(y) = x$. □