Algorithmic Dense Model Theorems and Weak Regularity

Russell Impagliazzo, CSE Department, UCSD and School of Mathematics, IAS*

November 5, 2009

Abstract

Green and Tao ([GT04]) used the existence of a dense subset indistinguishable from the primes under certain tests from a certain class to prove the existence of arbitrarily long prime arithmetic progressions. Tao and Ziegler ([TZ06]) showed some general conditions under which such a model exists. In [RTTV08], a quantitatively improved characterization was obtained using an argument based on Nisan's proof of the Impagliazzo hard-core set theorem ([I95]) from computational complexity. Gowers ([Gow08]) independently obtained a similar improvement.

We show that the existence of dense models can be reduced directly to the improved hardcore distribution results of Holenstein ([H05]). Using Holenstein's uniform proof of an optimal density hard-core set theorem, we show that the dense models that one derives have a canonical form, with models being (sampleable from) functions defined in terms of tests from the original class.

We give several applications, including generalizations of weak regularity lemmas ([FK99, K97, COCF]). For example, we show that any graph G with Δn^2 edges has a γ-cut-approximator of rank $2^{poly(1/\gamma,1/\log(1/\Delta)}$, whereas direct application of [FK99] gives rank $2^{O(1/\gamma^2\Delta^2)}$.

*Work supported by the Simonyi Fund, The Bell Company Fellowship and Fund for Math, and NSF grants DMS-083573, CNS-0716790 and CCF-0832797. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

1 Introduction

The notion of indistinguishability of two distributions under a class of tests ([Y82, GMR89]) has long been a fundamental concept in computational complexity, and in particular, for the foundations of cryptography. Recently, it is also becoming important in other areas of mathematics, in particular for additive combinatorics and number theory. Green and Tao ([GT04]) used the existence of a dense subset indistinguishable from the primes under certain tests from a certain class to prove the existence of arbitrarily long prime arithmetic progressions. Tao and Ziegler ([TZ06]) showed some general conditions under which such a model exists.

In [RTTV08, TTV09], a number of results translating between the two communities are given. For example, a quantitatively improved characterization was obtained using an argument similar to the non-constructive linear programming duality proof by Nisan of the hard-core set theorem ([I95]) from computational complexity. Gowers ([Gow08]) independently obtained a similar improvement, also using LP duality. [RTTV08, TTV09] also show that many of the standard "decomposition" theorems, such as the weak regularity theorem in graph theory, can be viewed as special cases of the dense model theorem or its generalizations.

Here, we give a precise connection between the hard-core set theorem and dense model theorem. We give a *direct reduction* from a strong form of dense model theorem to the hard-core set theorem ([I95, H05]). This allows us to convert any proof of the hard-core set theorem into a proof of the dense model theorem. Combining this reduction with *algorithmic* proofs of the hardcore set theorem using the *boosting* technique from computational learning theory ([I95, KS03, H05, BHK09]), we obtain an *algorithmic* version of the dense model theorem which also gives a *constructive characterization* of the models. ([TTV09] also obtained an algorithmic version of weak model theorem, but under a stronger assumption about the sets to be modeled.)

We show how to apply this general result to obtain algorithmic combinatorial "decomposition" theorems. The most important class of such results are versions of the weak regularity lemma ([FK99]) that apply to sparse graphs. We obtain the main result from [COCF] as a corollary, in a completely modular way. Because it is general and modular, our new proof of this result can be easily generalized, to think of our graph as a subgraph of any multigraph. An interesting case of this is to condition on degrees as in [ACOHKRS07]. It also allows us to apply our decompostion technique recursively, to get a new weak regularity theorem under much weaker assumptions about the graph.

Our work connects several disparate lines of research:

Computational entropy Consider a distribution on strings, such as descriptions of stock transactions or sunspot activity. While the distribution might be obviously not completely random, there may be computational limits to prediction or classification that go beyond the information-theoretic entropy of the distribution. So the amount of "randomness" as far as efficient algorithms (e.g., small Boolean circuits) are concerned may be greater than the amount of true randomness inherrent in the distribution. This issue arises in computational complexity theory and the theory of cryptography, as being related to pseudo-randomness, the power of randomness in computation, and randomness extraction.

Yao ([Y82]) first discusses the notion of computational entropy and gives a definition based on the amount of compression possible by feasible algorithms. Later, [HILL99] introduces a different formalization of computational entropy as the maximum Shannon entropy of a distribution indistinguishable from the given one by efficient algorithms (or small circuits). They use computational entropy as a tool in constructing cryptographic pseudo-random generators from one-way functions. Computational entropy has become a useful general conceptual tool for cryptography and the complexity theory of randomness. For example, [STV01] show how to use computational entropy for hardness amplification and constructions of pseudo-random generators for derandomization.

[BSW03] characterize distibutions with high pseudo-entropy in terms of a property we call *pseudo-density*. Essentially, their result says that any distribution which has no small circuit efficiently witnessing that it is small has high computational entropy, i.e., there is a high-entropy distribution indistinguishable from it to small circuits. This characterization turns out to be basically equivalent to the Dense Model Theorem. Their proof used a linear programming duality argument similar to the Nisan proof of the Harcore Set Theorem. Our constructive version shows that the high entropy distribution indistinguishable from a high pseudo-density set itself has small circuit complexity.

Additive combinatorics As noted before, the dense model theorem is a key step in the proof of the existence of arbitrarily large arithmetic progressions in the primes, and generalizations of this result ([GT04, TZ06]). This work has since been improved both quantitatively and qualitatively ([Gow08, RTTV08, TTV09]). In particular, the original work required the set in question (the primes) to be a dense subset of a pseudo-random set (the almost primes). For other sets of interest, such a pseudo-random set might not be obvious or might not exist. In contrast, dense model theorems in terms of pseudo-density ([RTTV08, BSW03]) work directly from the set to be modelled, by showing that for a certain class of tests, the expected value of the test on the set is not hugely greater than it is on the uniform distribution.

Second, in [TTV09], the model for the set has a small description in terms of the same set of tests. However, they required the set to be dense inside a pseudo-random set, not just pseudo-dense.

We give the first dense model theorem with both of these advantages simultaneously. We assume only pseudo-density, and give a construction of the model in terms of the same underlying family of tests. In many cases, we can do so efficiently, with an algorithm that either returns the description of the model or a test that violates pseduo-density. We call such an algorithm a modeller.

Hardness amplification Hardness amplification is the study of constructions that make a somewhat hard problem reliably hard on average. The hard-core set theorem is a central tool in hardness amplification, because it formalizes the intuitive notion of an instance to a somewhat hard problem being "hard" with some probability and "easy" otherwise. See for example [I95, IW97, STV01, O04, H05]. Here, we show the hardcore set theorem has applications beyond hardness amplification.

Boosting As observed by Klivans and Servedio ([KS03]), the constructive proofs of the hardcore set theorem apply the learning theory technique of boosting, and conversely, any boosting algorithm can be used to prove a version of hardcore set theorem. Boosting has already proved to be an amazingly useful algorithmic tool, with applications for learning ([S90, FS96a]), repeated playing of zero-sum games ([FS96b]) , and finding efficient approximations to convex optimization problems ([AHK05]). Here, we add yet another application: converting an approximation algorithm into a modeller. For example, we use the cut norm approximation algorithm of [AN04] to obtain the algorithmic weak regularity partitioning algorithm of [COCF]. Only the overall running time depends on the approximation factor, not the quality of the model, so even very poor approximation algorithms can be used in this general construction.

Regularity and Decomposition Theorems Many results in combinatorics show that modulo a high-level structure, the object is close to random in some respect. A classic example is the Szemeredi Regularity Lemma ([S78]), showing that any graph can be partitioned into a small number of pieces so that for most pairs of pieces, the density of any subgraphs is very close to the density of the bipartite graph between the pieces. The Weak Regularity Lemma of Frieze and Kannan ([FK99]) gives a similar result, with a partition into a smaller number of components, but where the error term is global, rather than for most pairs of components. However, the additive error term in these theorems swamps the number of edges in a sparse graph.

Kohayakawa [K97] gives a criterion under which even sparse graphs have regular partitions as in the Szemeredi regularity lemma [S78]. Coja-Oghlan, Cooper and Frieze [COCF] give a weak regularity lemma for sparse graphs and an efficient algorithm for finding a corresponding partition. Alon, Coja-Oghlan, Han, Kang, Rodl, and Schacht (ACHKRS07) modify the strong regularity lemma for sparse graphs so that nodes are weighted by degree, and also give an efficient version. For all of these results, the "high-level" structure is the partition, and the density between each pair, and the theorem is saying that, with a small error term, density of sub-graphs is close to that of random graphs with these densities giving edge probabilities between the parts.

We show how the efficient weak regularity lemma for sparse graphs of [COCF] follows directly from the dense model theorem (and an approximation algorithm of Alon and Naor for the cut norm [AN04], which is also used in the efficient results above.) We also give a generalization allowing the underlying probability model to have arbitrary edge weights rather than being uniform or proportional to degree.

We give a general decomposition theorem, showing how to break up any set into a "structured part" and a "random model". This is similar in spirit to the General Regularity Lemma of [TTV09], but we do not see any obvious implications in either direction. We also give a recursive version, that "telescopically" models increasing portions of the structured part of the set. Applying this to graph cuts, we get the following quantitative improvement to the weak regularity theorem for graphs:

Theorem 1. *For any Δ, Γ, there is a $T = 2^{poly(1/\Gamma, \log 1/\Delta)}$ so that: Let G be any undirected graph with Δn^2 edges. Then G has an Γ cut-approximator H which is a partition matrix of complexity T. (Here, a partition matrix of complexity T is one where we can partition the nodes of the graph into T sets, so that the value of $H_{u,v}$ just depends on which sets u and v are in.) Furthermore, we can compute H in polynomial time.*

We actually prove a more general version of the above as Corollary 2. Direct application of [FK99] would give $T = 2^{1/(\Gamma\Delta)^2}$ so our dependence on Δ is almost exponentially improved.

1.1 Notation and Definitions

Fix a basic finite universe U and a probability distibution σ on U. (We usually think of U as either the set of all n bit strings or all positive integers at most N, and σ as the uniform distribution. However, some of our proofs will require us to apply certain lemmas to non-uniform distributions σ, so we will be more general here.) A *test* is a Boolean function on U. (We can also consider more general tests, that map U to the real interval $[0, 1]$, but the case of real tests follows from that for Boolean tests by standard randomized rounding techniques.) Let T be a test and σ a distribution on U. We use $T[\sigma]$ as an abbreviation for $Prob_{x \in_\sigma U}[T(x) = 1]$. A *class* of tests $\mathcal{T}$ is a set of tests that contains the constant tests (always 0 and always 1) and is closed under complementation.

Two distributions ρ_1 and ρ_2 are *ϵ-indistinguishable* for $\mathcal{T}$ if $\forall T \in \mathcal{T}, |T[\rho_1] - T[\rho_2]| < \epsilon$. ρ is *ϵ-pseudorandom* for $\mathcal{T}$ if it is ϵ-indistinguishable for $\mathcal{T}$ from the uniform distribution on U.

A *measure* μ is a map from U to the real interval $[0, 1]$, with *density* $d(\mu) = \sum_{x \in U} \mu(x)\sigma(x)$. A measure μ of positive density induces the distribution $D_\mu(x) = \mu(x)\sigma(x)/d(\mu)$. We identify a set $S \subset U$ with the induced measure given by its characteristic function, and hence with the conditional distribution on x given $x \in S$. Thus, the density of a set S is $d(S) = \sum_{x \in S} \sigma(x) = Prob_{x \in_\sigma U}[x \in S]$. For test T and measure μ, we define $T[\mu] = T[D_\mu]$. In particular, $T[S] = Prob[T(x) = 1 | x \in S]$, since we identify S with the conditional distribution given $x \in S$.

If S is a set and μ a measure, we say that μ is an *ϵ-model* for S if the induced distribution on S and D_μ are ϵ-indistinguishable for $\mathcal{T}$. We are particularly interested in the case when S is of negligible size, and μ is dense, i.e., when $d(S) << d(\mu)$.

Note that if a set S has density $d(S)$, then for any test T, $T[U] = Prob[T(x) = 1] \geq Prob[x \in S]Prob[T(x) = 1 | x \in S] = d(S)T[S]$. Contrapositively, a test T with $T[U] < \delta T(S)$ is a "proof" that $d(S) < \delta$. However, since both $T[U]$ and $T[S]$ might be very small, random samples might not be enough to verify this "proof". To make the certificates easy to verify, we put in an additive error term. We call a test T an *(ϵ, δ)-distinctive* test if $T[U] < \delta T[S] - \epsilon$. (Hence, the inequality $T[U] < \delta T[S]$ can be verified using poly$(1/\epsilon)$ random samples.) A set S has *ϵ-pseudo-density* at least δ for $\mathcal{T}$ if there are no (ϵ, δ) distinctive tests in $\mathcal{T}$: $\forall T \in \mathcal{T}$, $T[U] \geq \delta T[S] - \epsilon$.

The *threshold function* $Th_{k,t}(b_1, ...b_t)$ is the Boolean function that is 1 if and only if at least k out of its t inputs are one. For $\mathcal{T}$ a class of tests, let $\mathcal{T}_t$ be the class of tests of the form $Th_{k,t'}(T_1, ..T_{t'})$ where $0 \leq k \leq t' \leq t$ and each $T_i \in \mathcal{T}$.

The *truncation* function $trunc(x)$ is a function from R to $[0, 1]$ defined by $trunc(x) = 0$ if $x < 0$, x if $0 \leq x \leq 1$, and 1 if $x > 1$. $tl_t(\mathcal{T})$ (truncated linear functions over $\mathcal{T}$ with size t) is the class of measures μ on

U of the following form: $\mu(x) = trunc((\sum_{i=1}^{i=t'}(T_i(x)))$ where $1 \leq t' \leq t$, and each $T_i \in \mathcal{T}$. If f is a Boolean function, the class of tests $\mathcal{T} \oplus f$ is the class of tests of the form $T'(x) = T(x) \oplus f(x)$, for some $T \in \mathcal{T}$. The class $\vee_k(\mathcal{T})$ is the class of tests that can be written as the logical or of at most k tests in $\mathcal{T}$, and $\wedge_k(\mathcal{T})$ the same for the logical and.

Let f be a Boolean function on U, and let ρ be a distribution on U. f is δ-hard for $\mathcal{T}$ on ρ if, $\forall T \in \mathcal{T}, Prob_{x \in_\rho U}[f(x) = T(x)] \leq 1 - \delta$. We say that f is ϵ-*hardcore* for $\mathcal{T}$ on ρ if and only if $\forall T \in \mathcal{T}, Prob_{x \in_\rho U}[f(x) = T(x)] \leq 1/2 + \epsilon$, i.e., if and only if f is $1/2 - \epsilon$ hard.

2 Statements of dense model theorems

The Green-Tao Dense Model Theorem can be paraphrased as follows:

Theorem 2. *There is a function $t = poly(1/\epsilon, 1/\delta)$ so that the following holds: Let $\mathcal{T}$ be a set of tests. Let $S \subset R \subset U$ be such that S is δ-dense in R (with base distribution uniform) and D_R is ϵ-pseudo-random for $\mathcal{T}_t$. Then there is a δ-dense distribution μ that is an $O(\epsilon)$-model for S with respect to tests in $\mathcal{T}$.*

[RTTV08] prove a somewhat simplified version of this theorem:

Theorem 3. *There is a function $t = poly(1/\epsilon, 1/\delta)$ so that the following holds: Let $\mathcal{T}$ be a set of tests. Let S be a subset of U that has ϵ-pseudo-density at least δ with respect to $\mathcal{T}_t$. Then there is a $\delta - O(\epsilon)$-dense measure μ that is an $O(\epsilon/\delta)$-model for S with respect to tests in $\mathcal{T}$.*

[1]

It is trivial to see that the second theorem implies a version of the first. All that is necessary is to show that if S has density δ within R and D_R is ϵ-pseudo-random, then S has ϵ-pseudo-density at least δ. But under these conditions, for any test T, $Prob_{x \in U}[T(x)] \geq Prob_{x \in R}[T(x)] - \epsilon \geq \delta Prob_{x \in S}[T(x)] - \epsilon$, which is precisely the requirement for ϵ-pseudo-density at least δ.

[RTTV08] used a proof modeled after the original hard-core set theorem [I95]. However, they used a final step very similar to that of a proof of an improved hard-core set theorem from [H05]. Our proof directly reduces the problem to Holenstein's hard-core set theorem. There were two proofs in [I95], both improved in [H05], a non-constructive game theoretic argument and a more constructive incremental argument later shown to be essentially the same as the concept of *boosting* in learning theory ([S90, FS96a, KS03]). [RTTV08] gave only a game theoretic proof, but by giving a direct reduction, we can use any proof of the Holenstein version of hard-core set theorem. Applying his second, boosting-style, proof, we obtain a more constructive characterization of the dense model for S:

Theorem 4. *There is a function $t = poly(1/\epsilon, 1/\delta)$ so that the following holds: Let $\mathcal{T}$ be a set of tests. Let S be a subset of U that has ϵ-pseudo-density at least δ with respect to $\mathcal{T}_t$. Then there is a $\delta - O(\epsilon)$-dense distribution μ that is an $O(\epsilon/\delta)$-model for S with respect to tests in $\mathcal{T}$. Moreover, $\mu \in tl_t(\mathcal{T})$ (i.e., has the following form: $\mu(x) = trunc(\sum_{i=1}^{i=t'}(g_i(x))$ where $t' < t$, and $g_i \in \mathcal{T}$ for $1 \leq i \leq t'$.)*

We also get an algorithmic version of the theorem:

Theorem 5. *Let $.1 > \epsilon > \epsilon' > 0, 1 > \delta > 4\beta > 0$ be given real numbers. There is a function $t = poly(1/\epsilon', 1/\delta)$ so that the following holds: Let $\mathcal{T}$ be a set of tests. Let $S \subseteq U$.*

Then there is an algorithm G running in time $poly(1/\epsilon, 1/\delta)$ and making at most that many oracle calls to four oracles A, A', B, and C as follows:

[1] The $O(\epsilon)$ slack term in the density can be moved into the error term by averaging the measure μ with the uniform distribution, for example. However, the $O(\epsilon/\delta)$ error term is tight, up to constant factors. Imagine a negligible size set S and a subset T of size $\epsilon/\delta|S|$, with $\mathcal{T}$ being the constants and the characteristic function of T. $\mathcal{T}_t = \mathcal{T}$, because every function in $\mathcal{T}$, and hence in $\mathcal{T}_t$, depends only on membership in T. The only non-trivial test to apply is membership in T, which is almost 0 on U and ϵ/δ on S (and $0 \geq \delta(\epsilon/\delta) - \epsilon$. So S is δ-pseudo-dense. However, on any measure of density δ, T has negligible probability, whereas $T(S)$ has probability ϵ/δ by construction, so T is a test that ϵ/δ distinguishes S from any large measure.

Sampler *A produces uniformly distributed samples $x \in S$, and A' produces uniformly distributed samples $x \in U$.*

Test evaluator *B, given x and a description of test $T \in \mathcal{T}$, computes $T(x)$.*

Approximate best distinguisher *C takes as input measures $\mu_1, \mu_0 \in tl_t(\mathcal{T} \oplus f)$ (described by up to t functions in $\mathcal{T}$ each) so that*

1. *μ_0 has density at least β on U*
2. *μ_1 has density at least β on S, and*
3. *there is an element $T \in \mathcal{T}$ distinguishing between D_{μ_0} on U and D_{μ_1} on S with advantage at least ϵ,*

and outputs a test $T' \in \mathcal{T}$ distinguishing the two distributions with advantage at least ϵ'.

Then, with high probability, G either produces a function $T' \in \mathcal{T}_t$ that is an (ϵ, δ)-distinctive test for S or a measure $\mu \in tl_t(\mathcal{T})$ of density at least $\delta - O(\epsilon)$ so that S is $O(\epsilon/\delta)$-indistinguishable from D_μ.

3 Hard-core set theorems

In [I95], the concept of a hard-core measure was introduced, and it was shown that every δ-hard function had a hard-core measure of density δ. Intuitively, a hard-core set theorem splits the instances into the "easy ones" and the hard-core, where the algorithm can do no better than random guessing. If the algorithm can compute f on all but a set H that is the "hard-core", it will also get the answer correct on H half the time, since more significant rate of failure means that the complement is an algorithm with a more significant rate of success on H. Thus, the failure overall should be half the density of H, or the hard-core should be of size 2δ, rather than the δ of [I95]. This gap was closed in [H05].

Theorem 6. *[H05] There is a function $t = poly(1/\epsilon, 1/\delta)$ so that the following holds: Let $\mathcal{T}$ be a set of tests. Let f be a Boolean function on U, and σ a distribution on U. If f is δ-hard for $\mathcal{T}_t$ on σ, then there is a measure μ of density 2δ on σ so that f is ϵ-hard-core on D_μ.*

By using a boosting-style argument, Holenstein also gives a more constructive algorithmic version of this theorem. Following a similar argument in [I95], but with some clever new twists, Holenstein describes a process where a set of tests in $\mathcal{T}$ evolves. Corresponding to the set of tests, there is a measure. If this measure has small density, he shows how to compute the function using the set of tests with probability more than $1 - \delta$, contradicting the hardness assumption. If not, and the measure is also not yet a hard-core, there is a prediction function in $\mathcal{T}$ that predicts f more than $1/2 + \epsilon$ of the time on the measure. This prediction function is added to the set. He shows that this process converges within a polynomial number of steps. While Holenstein stresses the algorithmic nature of this process, (assuming some method for obtaining the prediction function from the measure), it also has the advantage of having a very simple definition of the measures in question:

Theorem 7. *[H05] There is a function $t = poly(1/\epsilon, 1/\delta)$ so that the following holds: Let $\mathcal{T}$ be a set of tests. Let f be a Boolean function on U, and σ a distribution on U. If f is δ-hard for $\mathcal{T}_t$ on σ, then there is a measure $\mu \in tl_t(\mathcal{T} \oplus f)$ of density 2δ on σ so that f is ϵ-hard-core on D_μ.*

In fact, Holenstein gives an algorithmic form of this theorem.

Theorem 8. *[H05] Let $.1 > \epsilon > \epsilon' > 0, 1 > \delta > 2\beta > 0$ be given real numbers. There is a function $t = poly(1/\epsilon', 1/\delta)$ so that the following holds: Let $\mathcal{T}$ be a set of tests. Let f be a Boolean function on U, and σ a distribution on U.*

Then there is an algorithm G running in time $poly(1/\epsilon, 1/\delta$ and making at most that many oracle calls to three oracles A, B, and C as follows:

Labelled sampler *A produces samples $(x, f(x))$ so that x is distributed according to σ.*

Test evaluator *B, given x and a description of test $T \in \mathcal{T}$, computes $T(x)$.*

Approximate best predictor *C takes as input a measure $\mu \in tl_t(\mathcal{T} \oplus f)$ (described by up to t functions in $\mathcal{T}$) so that*

1. *μ has density at least β, and*
2. *there is an element $T \in \mathcal{T}$ predicting f on μ with advantage at least ϵ,*

and outputs a test $T' \in \mathcal{T}$ predicting f on μ with advantage at least ϵ'.

Then, with high probability, G either produces a function $T' \in \mathcal{T}_t$ predicting f with probability $1 - \delta$ or a measure $\mu \in tl_t(\mathcal{T} \oplus f)$ of density at least 2δ so that f is ϵ-hard-core on μ.

Combining our reduction with the above, we get the algorithmic dense model theorem (Theorem 5).

In the applications we give, the algorithms A, A' and B are usually trivial. Algorithm C can either be trivial (e.g., exhaustive search over $\mathcal{T}$) or very non-trivial (approximating the cut norm of a matrix ([AN04]).

4 The reduction

Here, we show how to reduce the model-existence theorems to the corresponding hard-core measure theorems. First, we give a high-level outline. Let S be a set which is (ϵ, δ)-pseudo-dense for $\mathcal{T}_t$.

We wish to use the hard-core set theorem to obtain a model for S. To do this, we need a somewhat hard Boolean function f. The obvious choice is to let f be the characteristic function of S, χ_S. However, the interesting case is when S is very small, so χ_S is approximated by 0 with very high probability. To avoid this, we need to magnify S, by warping the distribution so that we sample from S with some constant probability, say δ'. We will argue, for the right choice of δ', a test computes χ_S with probability $1 - \delta' + \epsilon$ if and only if it is (ϵ, δ) distinctive for the original distribution. Thus, by assumption, the function χ_S is δ' hard for this warped distribution. Thus, it has a hard core measure of density $2\delta'$. Since the constant functions are in our class of tests, such a measure must be evenly divided between elements of S and non-elements, a δ' fraction each. But since all of S has measure δ' in our new distribution, the part in S must be basically all of S. The part of the hard-core measure outside S must have density δ', but out of the $1 - \delta'$ fraction of the warped distribution outside S, so density $\delta'/(1 - \delta')$ in the original. So we need $\delta'/(1 - \delta') = \delta$. Somewhat miraculously, solving this equation for δ' also is the exact value required for the first step, transforming pseudo-density into hardness.

More precisely, let t be as in the Holenstein hard-core set theorems, for $\delta' = \delta/(1 + \delta)$ and for $\epsilon' = \epsilon/4$. Let S be a set which is (ϵ, δ)-pseudo-dense for $\mathcal{T}_t$.

Let $U' = \{(1, s) | s \in S\} \cup \{(0, x) | x \in U\}$. We think of $\mathcal{T}$ as a set of tests on U' by ignoring the first bit of the input. Let $\delta' = \delta/(1 + \delta)$, so that $\delta = \delta'/(1 - \delta')$. Consider the distribution σ that with probability δ' uniformly selects from $1 \times S$ and with probability $1 - \delta'$ uniformly selects from $0 \times U$. Let $f((b, x)) = b$ be the bit describing which case we sampled from. We claim that f is $(\delta' - \epsilon(1 - \delta))$-hard on σ for $\mathcal{T}_t$.

Otherwise, let $g \in \mathcal{T}_t$ compute f with probability $1 - \delta' + \epsilon(1 - \delta') = (1 - \delta')(1 + \epsilon)$. g could compute f correctly in two ways: x could be sampled from S, and $g(x) = 1$; or x could be sampled from U, and $g(x) = 0$. Since tests in $\mathcal{T}_t$ ignore the first input bit, this success probability is $\delta' g[S] + (1 - \delta')(1 - g[U])) =\geq (1 - \delta')(1 + \epsilon)$ Dividing through by $1 - \delta'$, and using $\delta = \delta'/(1 - \delta')$, we obtain: $\delta g[S] + (1 - g[U]) \geq 1 + \epsilon$ or equivalently, $g[u] \leq \delta g[S] - \epsilon$. Thus, g is (ϵ, δ)- distinctive, contradicting the pseudo-density assumption for S.

Therefore, applying the Holenstein hard-core measure theorem for f, $\mathcal{T}$, and σ, we obtain a measure $\mu((b, x))$ of density $2\delta' - 2(1 - \delta')\epsilon$ on which f is $\epsilon\delta'/4$ hard-core for $\mathcal{T}$. Let $\mu_1(x) = \mu(1, x)$ for $x \in S$ and similarly $\mu_0(x) = \mu(0, x)$ for $x \in U$. Then $d(\mu) = \delta' d(\mu_1) + (1 - \delta') d(\mu_0)$ (where $d(\mu_1)$ is taken with respect to the uniform distribution on S, whereas $d(\mu_0)$ is with respect to the uniform distribution on U).

Also, since constants are in $\mathcal{T}$, and since $f((b, x)) = b$, $|Prob_{(b,x) \in_{D_\mu} U'}[b = 1] - 1/2| \leq \epsilon'$. Then $2\epsilon' \geq |Prob_{(b,x) \in_{D_\mu} U'}[b = 1] - Prob_{(b,x) \in_{D_\mu} U'}[b = 0]| = |(\delta' d(\mu_1) - (1 - \delta') d(\mu_0))/d(\mu)|$. so $|\delta' d(\mu_1) - (1 -$

$\delta')d(\mu_0)| \leq 2\epsilon' d(\mu)$. Adding the equation above to this inequality , we get $2\delta' d(\mu_1) \geq d(\mu)(1-2\epsilon') \geq (2\delta' - 2\epsilon(1-\delta'))(1-2\epsilon')$, so $d(\mu_1) \geq 1 - \epsilon(1-\delta')/\delta' = 1 - \epsilon/\delta$. Similarly, $2(1-\delta')d(\mu_0) \geq d(\mu)(1-2\epsilon')$. Thus, $d(\mu_0) \geq ((\delta')/(1-\delta') - \epsilon)(1-2\epsilon') = (\delta - \epsilon)(1 - O(\epsilon)) = \delta - O(\epsilon)$. Therefore, $d(\mu_1) = 1 - O(\epsilon/\delta)$ and $d(\mu_0) = \delta - O(\epsilon)$.

We choose μ_0 as our hard-core measure. Since $d(\mu_1) = 1 - O(\epsilon/\delta)$, the statistical distance between D_{μ_1} and the uniform distribution on S is at most $O(\epsilon/delta)$. (To see this, we can write D_{μ_1} as a convex combination of uniform distributions on sets A of size $d(\mu_1)|S|$. Since each such distribution has statistical distance $2(1-|A|/|S|) = 2(1-d(\mu_1)) = O(\epsilon/\delta)$, the same is true for D_{μ_1}). Let (b,x) be chosen according to D_μ. For any $T \in \mathcal{T}$, $1/2 + \epsilon' \geq Prob[T(b,x) = b] = Prob[b=1]Prob[T((b,x)) = 1|b=1] + Prob[b=0]Prob[T((b,x)) = 0|b=0] \geq (1/2-\epsilon')T[D_{\mu_1}] + (1/2-\epsilon')(1-T[D_{\mu_0}]) \geq 1/2 - 2\epsilon' + 1/2(T[D_{\mu_1}] - T[[D_{\mu_0}])$. Thus, $(T[D_{\mu_1}] - T[[D_{\mu_0}]) \leq 4\epsilon' = \epsilon$. Since $T[S] - T[D_{\mu_1}] \leq O(\epsilon/\delta)$, we have $T[S] - T[D_{\mu_0} \leq O(\epsilon/\delta + \epsilon) = O(\epsilon/\delta)$. Thus, μ_1 is an $O(\epsilon/\delta$ model for S as claimed.

If we apply the boosting proof of Holenstein's theorem, μ has the form $trunc(\sum_i g_i(b,x) \oplus f(b,x))$. Since for μ_0 , $b = f(b,x) = 0$, and each g_i is independent of b, μ_0 has the form $trunc(\sum_i g_i(x))$ for at most t g_i's in $\mathcal{T}$.

4.1 Algorithmic version of the reduction

To make the above reduction algorithmic, i.e, to use Theorem 8 to prove Theorem 5, we also need to specify how to translate one group of parameters and algorithms to the other. In Theorem 5, we are given $.1 > \epsilon > \epsilon' > 0, 1 > \delta > 4\beta > 0$ and algorithms A, A', B, and C so that:

Sampler A is a probabilistic algorithm that produces uniformly distributed samples $x \in S$, and A' produces uniformly distributed samples $x \in U$.

Test evaluator B is an algorithm that given x and a description of test $T \in \mathcal{T}$ computes $T(x)$.

Approximate best distinguisher C is an algorithm that takes as input measures $\mu_1, \mu_2 \in tl_t(\mathcal{T} \oplus f)$ (described by up to t functions in $\mathcal{T}$ each) so that if μ_1 has density at least β on S and μ_2 has density at least β on U, and there is an element $T \in \mathcal{T}$ distinguishing between D_{μ_1} on U and D_{μ_2} on S with advantage at least ϵ, then C outputs a test $T' \in \mathcal{T}$ distinguishing the two distributions with advantage at least ϵ'.

Let $\epsilon_2 = 2\epsilon, \delta_2 = \delta + 2\epsilon$, $\epsilon_2' = \epsilon'/4$, and $\beta_2 = 2\beta + \epsilon$.

For U', σ and f as in the reduction, we'll define the following algorithms in order to apply Theorem /refUniformHardcore:

Labelled sampler $\overline{A}$, a probabilistic algorithm that produces samples $(x, f(x))$ so that x is distributed according to σ.

Test evaluator $\overline{B}$ an algorithm that given x and a description of test $T \in \mathcal{T}$ computes $T(x)$.

Approximate best predictor $\overline{C}$, an algorithm that takes as input a measure $\mu \in tl_t(\mathcal{T} \oplus f)$ (described by up to t functions in $\mathcal{T}$) so that if μ has density at least β_2, and there is an element $T \in \mathcal{T}$ predicting f on μ with advantage at least ϵ_2, then C outputs a test $T' \in \mathcal{T}$ predicting f on μ with advantage at least ϵ_2'

Define $\overline{A}$ as follows: Pick $b = 1$ with probability δ', and 0 otherwise. If $b = 1$, use A to produce $x \in_U S$, and return $((x,1),1)$, otherwise use A' to produce $x \in_U U$ and return $((x,0),0)$. (The bit b is both the last bit of the element of U' and the value of the function f).

We can let $\overline{B} = B$, since the requirement is identical.

Finally, define $\overline{C}$ as follows: $\overline{C}$ is given as input a measure μ of density at least β_2, so that there is some $T \in \mathcal{T}$ so that $Prob_{x \in_\mu U'}[T(x) = f(x)] \geq 1/2 + \epsilon_2$.

We first check to see if μ is very biased, and if it is, output a constant function as our predictor. Let $\gamma = |Prob_{x \in_{D_\mu} U'}[f(x) = 1] - 1/2|$ be the bias. We can estimate γ closely by generating random samples using $\overline{A}$. If $\gamma \geq \epsilon'_2$, we produce the appropriate constant function as our output.

Otherwise, the distribution is approximately balanced between 0's and 1's. Let $\mu_0, \mu_1 \in tl_t(\mathcal{T})$ be the results of setting the value of f in μ to 0 and 1 respectively.

Then $d(\mu) = \delta'(d(\mu_1)) + (1 - \delta')d(\mu_0) \geq \beta_2$. and the difference in probabilities of 1 and 0 is $2|\gamma| = |\delta'(d(\mu_1)) - (1 - \delta')d(\mu_0)|/d(\mu) \leq 2\epsilon'_2 = \epsilon$. Thus, $2\delta' d(\mu_0) \geq (1 - \epsilon)d(\mu) \geq \beta_2(1 - \epsilon) \geq 2\beta$, so $d(\mu_0) \geq \beta$. Similarly, $2(1 - \delta')d(\mu_1) \geq 2\beta$, so $d(\mu_1) \geq \beta$.

We claim the same test T that is a good predictor for f is also a good distinguisher for μ_0 and μ_1. For $x \in_\mu U'$, $1/2 + 2\epsilon = 1/2 + \epsilon_2 \leq Prob[T(x) = f(x)] = Prob[f(x) = 1]Prob[T(x) = 1|f(x) = 1] + Prob[f(x) = 0]Prob[T(x) = 0|f(x) = 0] = (1/2 + \gamma)T(\mu_1) + (1/2 - \gamma)(1 - T(\mu_0)) = 1/2(1 + (T(\mu_1) - T(\mu_0)) + \gamma(T(\mu_1) + T(\mu_0) - 1) \leq 1/2 + 1/2(T(\mu_1) - T(\mu_0)) + \epsilon/2$. Thus, $T(\mu_1) - T(\mu_0) \geq 3\epsilon > \epsilon$.

So μ_1 and μ_0 are both measure at least β and have a distinguisher of advantage ϵ. Thus, C on input μ_1, μ_0 produces a distinguisher $T' \in \mathcal{T}$ with advantage at least ϵ'. We can assume without loss of generality that $T'(\mu_1) \geq T'(\mu_0) + \epsilon'$. We claim T' is also a good predictor, working backwards through the above calculations: $Prob[T'(x) = f(x)] = Prob[f(x) = 1]Prob[T'(x) = 1|f(x) = 1] + Prob[f(x) = 0]Prob[T'(x) = 0|f(x) = 0] = (1/2 + \gamma)T'(\mu_1) + (1/2 - \gamma)(1 - T'(\mu_0)) = 1/2(1 + (T'(\mu_1) - T'(\mu_0)) + \gamma(T(\mu_1) + T(\mu_0) - 1) \geq 1/2(1 + \epsilon') - \gamma \geq 1/2 + \epsilon'/2 - \epsilon'/4 = 1/2 + \epsilon'/4 = 1/2 + \epsilon'_2$. So in this case we return T'.

Lastly, we need to show how to convert the output of the algorithmic hard-core set theorem into the output for the algorithmic dense model theorem. The algorithmic hard-core set either produces a function $g \in \mathcal{T}_t$ that predicts f with probability $1 - \delta'$ or a measure $\mu \in tl_t(\mathcal{T} \oplus f)$ that is measure at least $2\delta'$ and is $O(\epsilon/\delta)$ $\mathcal{T}$-hard-core for f. In the first case, we showed that g itself is an (ϵ, δ) distinctive test. In the second case, we showed that μ_0 , μ conditioned on $f = 0$, is a density $\delta - O(\epsilon)$ measure that is $O(\epsilon/\delta)$ indistinguishable from D_S. Note that μ_0 can be obtained from μ by substituting 0 for f in each occurence, and so each function is of the form $g_i \oplus 0 = g_i \in \mathcal{T}$. Thus, $\mu_1 \in tl_t(\mathcal{T})$ as claimed.

5 Weak regularity lemmas

In this section, we apply the constructive Dense Model Theorem to give an alternate proof of [COCF]. Following [RTTV08, TTV09], we consider the case of the set S being the edges of a graph and $\mathcal{T}$ cuts in this graph. For undirected graph G and two subsets A and B of vertices, let $E_G(S,T)$ be the multiset of edges with endpoints in both A and B, counting edges between nodes in $A \cup B$ twice (or equivalently, view each directed edge $\{u, v\}$ as a pair of directed edges (u, v) and (v, u), and let $E(A, B)$ be the set of directed edges (u, v) with $u \in A$ and $v \in B$.)

For H be a symmetric $n \times n$ matrix of non-negative real numbers, let $e(H) = \sum_i \sum_j H_{i,j}$ be the total value of entries in H. For $A, B \subseteq \{1, ..n\}$, let $e_H(A, B) = \sum_{i \in A} \sum_{j \in B} H_{i,j}$. We say that H is an *ϵ-cut approximator* for G if for every A, B $||E_G(A, B)|/|E(G)| - e_H(A, B)/e(H)| \leq \epsilon$.

Let $A_1, ...A_t$ be a partition of the vertices of G, and let $0 \leq \gamma_{i,j} \leq 1$ for each $1 \leq i \leq j \leq t$. The *partition matrix* for $A_1, ..A_t, \vec{\gamma}_{i,j}$ is the matrix H where $H_{u,v} = \gamma_{i,j}$ for every $u \in A_i, v \in A_j$. Note that H has at most t distinct rows, and hence is rank at most t.

We can reprove the following theorem of [COCF], which is an algorithmic version of weak regularity for sparse graphs.

Theorem 9. *[COCF] For every δ, ϵ, there are t, ϵ' so that there is a probabilistic polynomial time algorithm that, given an undirected graph G, with high probability, produces either:*

- *A cut (A, B) in G with $E(A, B) \geq ((1/\delta|A||B|/\binom{n}{2})) + \epsilon')E(G)$; or*
- *A partition $A_1, ...A_t$ and values $\gamma_{i,j} \in [0, 1]$, so that the corresponding partition matrix H is an $O(\epsilon/\delta)$-cut approximator for G and $e(H) \geq \delta n^2$.*

In our work, like the previous proofs, ϵ' is exponentially small and t is exponentially large in $poly(1/\delta, 1/\epsilon)$. However, both cases have an implicit representation that is only polynomially long in these parameters, and can be computed in polynomial time in $1/\epsilon, 1/\delta$. In [COCF], the implicit representation is a linear combination of rank 1 matrices; in ours, it is the truncation of the same. The proof is in the Appendix, Section 8.

We can also generalize this theorem easily, to give similar results for sub-graphs of a given graph or where edges are not weighted uniformly, e.g., where the “a priori” probability of an edge is proportional to the degrees of its vertices. Strong “degree-weighted” regularity lemmas and algorithmic versions thereof were proved by Alon, Coja-Oghlan, Han, Kang, Rodl, and Schacht (ACHKRS07). However, we believe that this is the first time the general “weighted regularity” problem was considered, and the first weak regularity lemma even for the “degree-weighted” special case.

The weighted version is more complicated to state than to prove. Let W be an $n \times n$ symetric matrix of non-negative integers. We think of W as represnting a Bayesian prior of how likely edges are to be in a graph with m edges, so think of picking a random multigraph with m edges by sampling m times from the edges of a multigraph where each edge $\{u, v\}$ appears $B_{u,v}$ times. In the unweighted case, W is identically 1, whereas we obtain the degree weighted case by setting $W_{u,v} = d_G(u)d_G(v)$, so that we are comparing G to a random graph where the expected degrees of nodes are proportional to their degrees in G. For matrices W, H, let $W \odot H$ be the pointwise product matrix, $W \cdot H_{i,j} = W_{j,j}H_{i,j}$. We can also let B be the adjacency matrix of any graph containing G as a subgraph.

Theorem 10. *For every δ, ϵ, there are t, ϵ' so that there is a probabilistic polynomial time algorithm that, given an undirected graph G, with high probability, produces either:*

- *A cut (A, B) in G with $E(S,T) \geq ((1/\delta e_W(S,T)/e(W)) + \epsilon')E(G)$; or*
- *A partition $S_1, ...S_t$ and values $\gamma_{i,j} \in [0, 1]$, so that for the corresponding partition matrix H, $B \cdot H$ is an ϵ-cut approximator for G and $e(W \cdot H) \geq \delta e(W)$.*

The proof is identical to the previous theorem, except we let U have $W[i, j]$ copies of the edge $\{i, j\}$. Note that all copies are identical for tests in $\mathcal{T}$, so μ and T will not distinguish between these copies.

6 Decomposition theorems

A decomposition theorem says that every object of a certain type can be broken up into a ‘strucured part” and a ‘random-looking” part. In this section, we use the constructive Dense Model Theorem recursively to provide some generic decompositions. We divide an arbitrary set S into a part that is contained within a very small defineable subset of the universe $S_1 \subset U_1$, and a part S_0 that is indistinguishable from a simple defineable distribution D_0 over the rest of the universe $U_0 = U - U_1$. Thus, we can view S_1, D_0 as giving the “structure” of S, and then view S_0 as close to a random set chosen according to D_0. We give two decomposition theorems. The first applies the Dense Model theorem interatively to carve the set into pieces that “look small” and “look large”; the second applies the first recursively.

Let T be a test. Let $U_0 = \{x \in U | T(x) = 0\}$, $U_1 = \{x \in U | T(x) = 1\}$, $S_0 = \{x \in S | T(x) = 0\}$, and $S_1 = \{x \in S | T(x) = 1\}$. Let $\alpha_0 = 1 - T(S)$ be the probability that T is 0. For D_0 a distribution on U_0, let $D(T, \alpha_0, D_0)$ be the following distribution on U: With probability α_0, pick $x \in U_0$ according to D_0. Otherwise, pick $x \in S_1$ uniformly.

Theorem 11. *Let $S \subset U$ and let $\mathcal{T}$ be any class of tests on U. For any ϵ, δ, there is a $t = poly(1/\epsilon, 1/\delta)$, a test $T \in \vee_{O(\epsilon/\delta)}\mathcal{T}_t$ and a measure $\mu_0 \in tl_t(\mathcal{T})$ on U_0 so that: $T(U) < \delta T(S)$, and $D(T, \alpha_0, D_{\mu_0})$ is $O(\epsilon/\delta)$ indistinguishable from U_S.*

Moreover, assuming the algorithms in Theorem /refUniformDenseModel can be extended to subsets defined by tests in $\mathcal{T}$, T and μ_0 can be found efficiently.

The Proof is in the Appendix, Section 9.

Let $D_t(\mathcal{T}, U)$ be the class of distributions on U that can be computed from at most t tests in $\mathcal{T}$, i.e., $D(x)$ is some function of $T_1(x), ...T_t(x)$.

Theorem 12. *Let $S \subset U$ and let $\mathcal{T}$ be any class of tests on U. For any ϵ, δ and integer l, there is a $t = poly(1/\epsilon, 1/\delta)$, a test $T \in \wedge_l(\vee_{O(\epsilon/\delta)} \mathcal{T}_t)$ and a distribution $D_0 \in D_{O(tl)}(\mathcal{T}, U)$ on U_0 so that: $T(U) \leq \delta^l T(S)$, and $D(T, \alpha_0, D_0)$ is $O(l\epsilon/\delta)$ indistinguishable from U_S.*

Moreover, assuming the algorithms in Theorem /refUniformDenseModel can be extended to subsets defined by tests in $\mathcal{T}$, T and μ_0 can be found efficiently.

The proof is in the Appendix, Section 9.

The following shows that we can have a simple model, if not a dense model, of even sets with very small pseudo-densities:

Corollary 1. *There exists a value of $t = poly(1/\gamma, 1/(\log \Delta))$ so that if S has pseudo-density $(\gamma\Delta/2, \Delta)$ to $\wedge_{\log \Delta + 1}(\vee_t(\mathcal{T}_t)))$, then there is a distribution D in $D_t(\mathcal{T}, U)$ so that D and U_S are indistinguishable within $O(\gamma)$ to $\mathcal{T}$.*

Proof: Let $\delta = 1/2, l = \log(1/\Delta) + 1, and \epsilon = \gamma/l$. Let T_l and D_l be as in the previous theorem. Then by pseudo-density, $\Delta T_l[S] - \gamma\Delta/2 \leq T_l[U] \leq 2^{-l} T_l[S] = \Delta/2 T_l[S]$. Therefore, $\alpha_l = T_l[S] \leq \gamma$. Thus, $D(T_l, \alpha, D_l)$ and D_l are statistically within γ of each other, and $D(T_l, \alpha, D_l)$ is indistinguishable within $O(l\epsilon) = \gamma$ from U_S.

We get as a special case the following smooth tradeoff between the quality and complexity of cut approximations via partitions for moderately pseudo-dense graphs. We believe this is new even for moderately dense graphs.

Corollary 2. *For any Δ, Γ, there are $T = 2^{poly(1/\Gamma, \log 1/\Delta)}$ and $\gamma = \Gamma\Delta/2T$ so that: Let G be any graph with pseudo-density (γ, Δ) with respect to cuts. Then G has an $O(\Gamma)$ cut-approximator H which is a partition matrix of complexity T. Furthermore, we can compute H in polynomial time.*

Proof. This is the special case of the above corollary. As before, because we can decompose any Boolean predicate of t cuts into $T = 8^t$ complete bipartite graphs, being (γ, Δ) pseudo-dense with respect to cuts means G is $(\gamma T = \Gamma\Delta/2, \Delta)$ pseudo-dense against Boolean combinations of t cuts. Then the previous corollary gives us a distribution D defined in terms of t cuts which is $O(\Gamma)$ indistinguishable to $U_{E(G)}$ by cuts. Then D can be written as a partition matrix of complexity T, by expanding the basic cuts into a partition. □

7 Conclusions

The main question this work raises is whether there are any applications of the framework in the field where dense model theorems originated, additive number theory. Are there sets where it is easy to bound the pseudo-density, but hard to define a pseudo-random set that it is a large subset of?

References

[AN04] N. Alon, A. Naor. Approximating the cut-norm via Grothendieck's inequality. Proc. 36'th STOC, 2004, pp. 72-80.

[ACOHKRS07] N. Alon, A. Coja-Oghlan, H. Han, M. Kang, V. Rodl, M. Schact: Quasi-randomness and algorithmic regularity for graphs with general degree distributions. Proc. 34th ICALP, 2007, pp. 789-800.

[AHK05] Sanjeev Arora, Elad Hazan, Satyen Kale: Fast Algorithms for Approximate Semidefinite Programming using the Multiplicative Weights Update Method. FOCS 2005, pp.339-348.

[BHK09] Boaz Barak, Moritz Hardt, Satyen Kale: The uniform hardcore lemma via approximate Bregman projections. SODA 2009, pp. 1193-1200.

[BSW03] Boaz Barak, Ronen Shaltiel, Avi Wigderson: Computational Analogues of Entropy. RANDOM-APPROX 2003, pp. 200-215.

[COCF] Amin Coja-Oghlan, Colin Cooper, Alan Frieze: An efficient regularity concept for sparse graphs and matrices. Proc. 20th SODA, 207-216.

[FS96a] Yoav Freund, Robert E. Schapire: Experiments with a New Boosting Algorithm. ICML 1996, pp.148-156.

[FS96b] Yoav Freund, Robert E. Schapire: Game Theory, On-Line Prediction and Boosting. COLT 1996, pp. 325-332.

[FK99] A. Frieze and R. Kannan, Quick approximations to matrices and applications. Combinatorica, volume 19, 1999, pp. 175-220.

[GMR89] Shafi Goldwasser, Silvio Micali, Charles Rackoff: The Knowledge Complexity of Interactive Proof Systems. SIAM J. Comput. (SIAMCOMP) 18(1), 1989. pp 186-208.

[Gow08] Timothy Gowers. Decompositions, approximate structure, transference, and the Hahn- Banach theorem. Preprint, 2008.

[GT04] Ben Green and Terence Tao, The primes contain arbitrarily long arithmetic progressions. To appear: Annals of mathematics, manuscript: 2004.

[HILL99] Johan Hstad, Russell Impagliazzo, Leonid A. Levin, Michael Luby: A Pseudorandom Generator from any One-way Function. SIAM J. Comput. 28(4): 1364-1396 (1999)

[H05] Thomas Holenstein: Key agreement from weak bit agreement. STOC 2005, pp. 664-673.

[I95] R. Impagliazzo, Hard-Core Distributions for Somewhat Hard Problems. FOCS 1995, pp. 538-545.

[IW97] Russell Impagliazzo, Avi Wigderson: $P = BPP$ if E Requires Exponential Circuits: Derandomizing the XOR Lemma. STOC, 1997, pp. 220-229.

[KS03] Adam R. Klivans, Rocco A. Servedio: Boosting and Hard-Core Set Construction. Machine Learning (ML) 51(3):217-238 (2003).

[K97] Y. Kohayakawa: Szemeredi's regularity lemma for sparse graphs. In F. Cucker, M. Shub (editors): Foundations of computational mathematics, 1997, pp. 216-230.

[O04] Ryan O'Donnell: Hardness amplification within NP. J. Comput. Syst. Sci. 69(1), 2004, pp. 68-94.

[RTTV08] Omer Reingold, Luca Trevisan, Madhur Tulsiani and Salil Vadhan: Dense Subsets of Pseudorandom Sets, FOCS 2008, pp. 76-85.

[S90] Robert E. Schapire: The Strength of Weak Learnability. Machine Learning 5, 1990, pp. 197-227.

[STV01] Madhu Sudan, Luca Trevisan, Salil P. Vadhan: Pseudorandom Generators without the XOR Lemma. J. Comput. Syst. Sci. (JCSS) 62(2), 2001, pp. 236-266.

[S78] E. Szemeredi: Regular partitions of graphs, Problemes Combinatoires et Theorie des Graphes Colloques Internationaux, CNRS vol. 260, 1978, pp. 399-401.

[TZ06] Terence Tao and Tamar Ziegler. The primes contain arbitrarily long polynomial progressions. arXiv:math/060050, 2006.

[TTV09] Luca Trevisan, Madhur Tulsiani and Salil Vadhan: Regularity, Boosting, and Efficiently Simulating Every High-Entropy Distribution, IEEE Computational Complexity Conference, 2009, pp.126-136.

[Y82] Andrew Chi-Chih Yao: Theory and Applications of Trapdoor Functions. FOCS 1982, pp 80-91.

8 Appendix: Proof of weak regularity lemma

Here we prove Theorem 9.

Proof. To prove this result, let U be the set of all undirected edges between pairs of distinct vertices in V, and let S be the set of edges in the graph G. For $A, B \subset V$, with $A \cap B = \emptyset$, let $T_{A,B}(\{u,v\}) = 1$ if $\{u,v\} \in E(A,B)$ and 0 otherwise. Let $\mathcal{T}$ be the set of such tests $T_{A,B}$.

To get an algorithmic hard core therorem, we need Sampling, Evaluation, and Approximate Distinguishing algorithms. The first two are trivial, produce a random edge of G, produce a random edge in U, and tell whether a given edge e is in $E(A,B)$. The Approximate Distinguishing algorithm is basically given to us by [AN04], who give a constant factor approximation algorithm for the cut norm of a matrix, where for $n \times n$ matrix M, $CutNorm(M) = max_{A \subset [n], B \subset [n]} |\sum_{i \in A, j \in B} M_{i,j}|$ Their algorithm returns a pair A, B with sum at least a constant fraction of $CutNorm(M)$, in polynomial time in M. Given measures μ_1 on S, and μ_0 on U, we can explicitly construct the distributions D_{μ_1} and D_{μ_0} and define symetric real-valued matrix M by $M_{u,v} = D_{\mu_1}(\{u,v\}) - D_{\mu_0}(\{u,v\})$. For cut (A,B), the distinguishing probability for $T_{A,B}$ is exactly $\sum_{u \in A, v \in B} M_{u,v}$. So if there is a test with distinguishing probability ϵ, the Alon Naor algorithm returns a pair (A_1, B_1) with sum $\Omega(\epsilon)$.

The one complication is that A_1 and B_1 may not be disjoint. If not, we can break the sum into a constant number of pieces, each between two of $A_1 - B_1$, $A_1 \cap B_1$ or $B_1 - A_1$. One of these pieces has sum $\Omega(\epsilon)$. All of these involve disjoint sets, except between $A_1 \cap B_1$ and itself. If this last sum is the largest, we can greedily partition this intersection so that we get at least half the sum crossing the partition. So in the end, we get disjoint sets A_2, B_2 whose cut has at least an $\Omega(\epsilon)$ distinguishing probability.

Using these algorithms in Theorem 5, for $t = poly(1/\epsilon, 1/\delta)$, we compute either a test $T \in Tau_t$ with $T[U] \leq \delta T[S] - \epsilon$, or a measure $\mu \in tl_t(\mathcal{T})$ of density at least δ and where U_S and D_μ are $O(\epsilon/\delta)$ indistinguishable for $\mathcal{T}$.

In the first case, we have a distinctive test in $\mathcal{T}_t$, but we want one in $\mathcal{T}$ itself. However, $T(e)$ only depends on whether e is in t cuts $E(A_i, B_i)$. Partition the nodes up into 4^t sets $C_1, \ldots C_{4^t}$, according to which of the A_i and B_i they are members of. T is constant on the set of edges between two groups in this partition, so there is a symetric relation R on $1, , ..4^t$ so that $T(\{u,v\}) = R(i,j)$ if $u \in C_i, v \in C_j$. Then we can write both $T[U]$ and $T[S]$ as the sum over i, j with $R(i,j) = 1$ of the probability of an edge falling in the corresponding cut. Since $T[U] \leq \delta T[S] - \epsilon$, there must be such a cut $E[C_i, C_j)]$ with $Prob_{e \in U}[e \in E(C_i, C_j)] \leq \delta Prob_{e \in S}[e \in E(C_i, C_j)] - \epsilon'$, where ϵ' is ϵ divided by the number of such pairs, and in particular, $\epsilon' \geq \epsilon/8^t$. Since the probability that a random edge in U is in $E(C_i, C_j)$ is $|C_i||C_j|/\binom{n}{2}$ and that for S is $E(G) \cap E(C_i, C_j)/|E|$, we can output C_i, C_j and satisfy the first clause of the statement for the algorithm.

In the second case, μ also depends on t cuts. We can similarly partition the vertices into 4^t subsets, where μ is constant on the edges between each pair of subsets. Thus, the matrix H can be just the value of μ itself. The condition that μ is $O(\epsilon/\delta)$ indistinguishable for $\mathcal{T}$ is the same as H being an $O(\epsilon/\delta)$ cut approximator for G. □

9 Proofs of decomposition theorems

Here, we prove Theorems 13 and 14. We'll also restate the theorems.

Theorem 13. *Let $S \subset U$ and let $\mathcal{T}$ be any class of tests on U. For any ϵ, δ, there is a $t = poly(1/\epsilon, 1/\delta)$, a test $T \in \vee_{O(\epsilon/\delta)} \mathcal{T}_t$ and a measure $\mu_0 \in tl_t(\mathcal{T})$ on U_0 so that: $T(U) < \delta T(S)$, and $D(T, \alpha_0, D_{\mu_0})$ is $O(\epsilon/\delta)$ indistinguishable from U_S.*

Moreover, assuming the algorithms in Theorem /refUniformDenseModel can be extended to subsets defined by tests in $\mathcal{T}$, T and μ_0 can be found efficiently.

Proof. Initially, let $T = 0$, so $S_0 = S, U_0 = U$. Repeatedly apply the dense model theorem to S_0, U_0 with $\delta' = \delta\alpha_0$. If $\alpha_0 < 10\epsilon/\delta$, we halt and output T and any measure, say identically 1. Otherwise, we apply the dense model theorem to either get a distinctive test for S_0 inside U_0 or a model for S_0. If in the i'th iteration, we get a test $T_i \in Tau_t$ with $T_i[U_0] \leq \delta/\alpha_0 T_i[S_0] - \epsilon$, we repace T by $T \vee T_i$. If we instead get a measure $\mu_0 \in Tau_t$ on U_0 that is $O(\epsilon/\delta\alpha_0)$ indistinguishable from U_{S_0}, we stop and output T, μ_0.

We maintain the invariant that $T(U) \leq \delta T(S)$. It is true initially, since both sides are 0. In the i'th iteration, T_i is $(\epsilon, \delta\alpha_0)$ distinctive within S_0, so $(T \vee T_i)[U] = T[U] + (T_i \wedge \neg T)[U] = T[U] + (T_i[U_0](1 - T[U])) \leq T[U] + T_i[U_0] \leq \delta T[S] + \alpha_0 \delta T_i[S_0] = \delta(T[S] + (1 - T[S])T_i[S_0]) = \delta(T[S] + (T_i \wedge \neg T)[S]) = \delta(T \vee T_i)[S]$.

Also, since $0 \leq T_i[U_0] \leq \alpha_0 \delta T_i[S_0] - \epsilon$, $T_i[S_0] \geq \epsilon/(\alpha_0\delta)$, and $(T_i \wedge \neg T)[S] \geq \epsilon/\delta$. So each iteration α_0 decreases by at least ϵ/δ, so there are at most δ/ϵ iterations.

If we terminate because α_0 is $O(\epsilon/\delta)$, both U_S and $D_{T,\alpha_0,\mu}$ are $O(\epsilon/\delta)$ close to U_{S_1}, since with probability $1 - alpha_0$, each samples uniformly from S_1. So they are $O(\epsilon/\delta)$ close to each other, statistically.

Otherwise, we find a measure μ_0 so that D_{μ_0} is $O(\epsilon/(\delta\alpha_0))$-indistinguishable from U_{S_0} by $\mathcal{T}$. Since both $D(T, \alpha_0, D_{\mu_0})$ and U_S sample uniformly from S_1 with probability $1 - \alpha_0$, any distinguishing probability for a test comes from the α_0 probability side conditioned on which the two distributions sample from U_{S_0} and D_{μ_0} respectively. Thus, D_S and D_{T,α_0,μ_0} are $O(\epsilon/\delta)$ indistinguishable.

□

Let $D_t(\mathcal{T}, U)$ be the class of distributions on U that can be computed from at most t tests in $\mathcal{T}$, i.e., $D(x)$ is some function of $T_1(x), ... T_t(x)$.

Theorem 14. *Let $S \subset U$ and let $\mathcal{T}$ be any class of tests on U. For any ϵ, δ and integer l, there is a $t = poly(1/\epsilon, 1/\delta)$, a test $T \in \wedge_l(\vee_{O(\epsilon/\delta)}\mathcal{T}_t)$ and a distribution $D_0 \in D_{O(tl)}(\mathcal{T}, U)$ on U_0 so that: $T(U) \leq \delta^l T(S)$, and $D(T, \alpha_0, D_0)$ is $O(l\epsilon/\delta)$ indistinguishable from U_S.*

Moreover, assuming the algorithms in Theorem /refUniformDenseModel can be extended to subsets defined by tests in $\mathcal{T}$, T and μ_0 can be found efficiently.

Proof. By induction on l. The case $l = 1$ follows directly from Theorem 13. For $l > 1$, by induction, we have:

1. a test $T_{l-1} \in \wedge_{l-1}(\vee_{O(\epsilon/\delta)}\mathcal{T}_t)$, where $T_{l-1}(U) \leq \delta^{l-1}T_{l-1}(S)$.(T_{l-1} splits S and U into S_0, S_1, U_0, U_1 by the value of $T_{l-1}(x)$)

2. $\alpha_{l-1} = 1 - T_{l-1}(S)$.

3. and a distribution $D_{l-1} \in D_{O(t(l-1))}(\mathcal{T}, U_0)$

so that $D(T_{l-1}, \alpha_{l-1}, D_{l-1})$ is $O((l-1)\epsilon/\delta)$ indistinguishable from U_S by tests in $\mathcal{T}$.

Apply Theorem 13 to S_1 with universe U_1 to get $T' \in \vee_{O(\epsilon/\delta}(\mathcal{T}_t)$, and $\mu' \in tl_t(\mathcal{T})$ so that $T'[U_1] \leq \delta T'[S_1]$ and for $D' = D_{\mu'}$ and $\alpha' = 1 - T'[S_1]$, U_{S_1} is $O(\epsilon/\delta)$ indistinguishable from $D(T', \alpha', D')$ (as a distribution on U_1). Let $T_l = T' \wedge T_{l-1}$. and from D' otherwise. Finally, let $\alpha_l = 1 - (T' \wedge T_{l-1})[S] = (\neg T' \vee \neg T_{l-1})[S] = (1 - T_{l-1}[S]) + (T_{l-1}[S])(1 - T'[S_1]) = \alpha_{l-1} + (1 - \alpha_{l-1})\alpha'$. Let D_l sample from D_{l-1} with probability α_{l-1}/α_l and from D' otherwise.

Then $D(T_l, \alpha_l, D_l)$ and $D(T_{l-1}, \alpha_{l-1}, D_{l-1})$ both sample from D_{l-1} with probability α_{l-1}. Otherwise the first samples from $D(T', \alpha', D')$ and the other from U_{S_1}. Since these last two distributions are $O(\epsilon/\delta)$ indistinguishable by $\mathcal{T}$, the preceding two distributions are also indistinguishable by the same amount. Since $D(T_{l-1}, \alpha_{l-1}, D_l)$ is $O(\epsilon/\delta(l-1))$ indistinguishable from U_S, $D(T_l, \alpha_l, D_l)$ is $O(l\epsilon/\delta)$ indistinguishable from U_S.

$T_l[U] = T'[U_1]T_{l-1}[U] \leq \delta T'[S_1]\delta^{l-1}T_{l-1}[S] = \delta^l T_l[S]$.

T_l has one more conjunct than T_{l-1}, and D_l is defined in terms of T', D_{l-1} and μ', so involves only $poly(1/\epsilon, 1/\delta)$ more tests in $\mathcal{T}$ than D_{l-1}. □

10 Appendix : Other examples of the general framework

In this appendix, we give several simple, direct applications of the Dense Model Theorem to both combinatorial and complexity-theoretic domains. The first two examples are basically "toy" problems, artificial demonstrations of the general technique, but ones that seem interesting and potentially useful to us. The third makes the characterization of computational entropy in [BSW03] more constructive.

10.1 Juntas

A d-junta is a Boolean function that depends on at most d of its inputs. Let $J_{d,n}$ be the class of d-juntas on n Boolean variables. Note there are at most $2^{2^d} n^d$ such juntas, and we can describe an element explicitly as a 2^d bit truth table and the set of variables that the junta depends on. Assume $S \subseteq \{0,1\}^n$ is given as a sampling procedure generating uniform elements of S. Let $\mathcal{T} = J_{d,n}$ and $U = \{0,1\}^n$. We have a sampling algorithm for U, and one for S is assumed. Given a description of a junta of the above form, to evaluate it is simple, finding the appropriate bits and looking up the value on the table. Finally, if d is constant, we can find an approximate best distinguisher between measures μ_0, μ_1 of measure at least β in $O(n^d poly(1/\epsilon, 1/\beta)$ time through exhaustive search. For each of $O(n^d)$ juntas, approximate the distinguishing probability to within say $\epsilon/4$ by generating $poly(1/\epsilon)$ samples from D_{μ_1} and D_{μ_0}. Each sample requires expected time $O(1/\beta)$, because uniform samples from S or U are accepted with probability $d(\mu_1), d(\mu_0)$ respectively. (We can save samples by using the same $O(\log |J_{n,d}|/\epsilon^2) = O((2^d + d \log n)/\epsilon^2)$ samples to estimate all probabilities.) Return the junta with the largest estimated distinguishing probability.

Therefore, from Theorem 5, for $t = poly(1/\epsilon, 1/\delta)$, we have an $O(n^d poly(1/\epsilon, 1/\delta))$ time algorithm that, given such a set S, either returns a (δ, ϵ)-distinctive $T \in \mathcal{T}_t$ for S or a $\delta - O(\epsilon)$ dense measure $\mu \in tl_t(\mathcal{T})$ that is an $O(\epsilon/\delta)$-model for S.

Note that $\mathcal{T}_t \subset J_{dt,n}$, so in the first case we find a constant size junta that is distinctive for S. Also functions in $tl_t(\mathcal{T})$ only depend on td variables. For a set of variables B, let U_B represent the uniform distribution on $\{0,1\}^B$. In the second case, there is a set of variables A of size td and a probability distribution on assignments to A, D_A, so that $\mu = D_A \times U_{n-A}$. In particular, we get:

Theorem 15. *For every fixed positive integer d and every fixed $0 < \epsilon < \delta < .1$, there is an integer k so that: For every $S \subseteq \{0,1\}^n$ either there is an (δ, ϵ)-distinctive k-junta T' for S or there is a set of variables A of size at most k so that for every d-junta T depending on a set of variables B, $B \cap A = \emptyset$, $|T(S) - T(U_n)| \leq O(\epsilon/\delta)$. Moreover, we can find either T' or A in $O(n^d)$ time given $O(\log n)$ uniform samples from S.*

Note that for this theorem, the form of the model, being determined by a small number of tests from $\mathcal{T}$, was more important than the density. Perhaps there is a direct argument not using dense model theorems.

10.2 Polynomials

Let $U = \{0,1\}^n$, and let $S \subseteq U$ be given by a sampling oracle. Let $\mathcal{T}$ be the closure under negations of the set of products of at most d variables over the real numbers. Let $\delta = 1/n^d$ and $\epsilon = 1/Wn^{3d}$.

As before, $\mathcal{T}$ has $O(n^d)$ elements, so we can approximate the best distinguisher by exhaustive search in $O(n^d)$ time. Applying Theorem 5, for $t = poly(1/\epsilon, 1/\delta) = O(Wn^{O(d)})$, we have an $O(n^O(d))$ time algorithm that, given such a set S, either returns a (δ, ϵ)-distinctive $T \in \mathcal{T}_t$ for S or a $\delta - O(\epsilon)$ dense measure $\mu \in tl_t(\mathcal{T})$ that is an $O(\epsilon/\delta)$-model for S.

Note that tests in $\mathcal{T}_t$ are signs of polynomials with degree d and integer-valued co-efficients at most t, and measures in tl_t are truncations of such polynomials, normalized by dividing by the sum of the co-efficients. Thus, we either compute a polynomial of degree d whose sign is $O(\delta, \epsilon)$ distinctive for S or a degree d polynomial $q(x_1, ...x_n)$ so that for every monomial of degree at most d, m, (letting $q[R]$ represent the expectation of function q on subset R), $|m[S] - trunc(q) * m[U]/trunc(q)[U]| \leq O(\epsilon/\delta) = O(1/Wn^{2d})$. Let p be any polynomial of degree d with co-efficients bounded by W in absolute value. Adding up all the error bounds for each co-efficient, we have $|p(S) - trunc(q) * p[U]/\alpha| \leq O(1/n^d)$. Thus, we can use q to approximate the expected value of any degree d polynomial with small co-efficeints on S.

We summarize this as:

Theorem 16. *For every fixed positive integer d, and every $W > 0$, for every $S \subseteq \{0,1\}^n$, either there is a degree d polynomial p so that $sign(p)$ is $(1/n^d, 1/Wn^{3d})$-distinctive for S or there is a degree d polynomial q and constant $\alpha = trunc(q)[U] > 1/n^d$ so that $|p(S) - trunc(q) * p[U]/\alpha| \leq O(1/n^d)$ for every degree d polynomial p with coefficients of absolute value at most W. Moreover, we can compute one of these two polynomials in time $n^{O(d)}$ given a sampling procedure for S.*

10.3 Characterization of computational entropy

For this subsection, let the universe U be $\{0,1\}^n$,and the family of tests $\mathcal{T}$ as Boolean circuits of size s on n inputs. [BSW03] proved the following:

Theorem 17. *[BSW03] Let $S \subset U$ be (ϵ, δ) pseudo-dense for circuits of size $O(spoly(1/\epsilon), 1/\delta))$. Then there is a measure μ on U of density δ so that U_S and D_μ are $O(\epsilon/\delta)$ -indistinguishable to circuits of size s.*

This can be rephrased as saying that the sets with $n - O(1)$ computational min-entropy are exactly those with constant pseudo-density.

We get the following refinements:

Theorem 18. *Let $S \subset U$ be (ϵ, δ) pseudo-dense for circuits of size $O(spoly(1/\epsilon), 1/\delta))$. Then there is a set H of cardinality $O(\delta 2^n)$ so that U_S and U_H are $O(\epsilon/\delta)$ -indistinguishable to circuits of size s. Moreover, H is recognizable by a circuit of size $O(s \log snpoly(1/\epsilon, 1/\delta))$.*

The proof follows directly from the constructive (but not algorithmic) Dense Model Theorem (Theorem 4). For $\mathcal{T}$ the set of functions computable by size s circuits, and t the polynomial in the statement of the theorem, functions in $\mathcal{T}_t and tl_t(\mathcal{T})$ are both computable by size $O(st)$ circuits. Therefore, S is pseudo-dense for $\mathcal{T}_t$, so we construct the desired measure $\mu \in tl_t(\mathcal{T})$. To convert μ to a set H recognized by a small circuit, let h be chosen from an $O(s \log s/\epsilon^2)$-wise independent family of functions from $\{0,1\}^n$ to $0, 1/q, 2/q, ...1$ for a large value of q. Let $H = \{x | h(x) \leq \mu(x)\}$. Standard probability estimates show that H is indistinguishable from D_μ with high probability, and that $d(H)$ is close to δ with high probability. Such hash functions can be described and computed in size $O(s \log sn/\epsilon^2)$.