

THE UNIVERSITY OF CHICAGO

ON INTERACTIONS OF VAPNIK-CHERVONENKIS AND LITTLESTONE
DIMENSIONS WITH LOGIC AND COMBINATORICS

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE PHYSICAL SCIENCES
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

DEPARTMENT OF MATHEMATICS
AND
DEPARTMENT OF COMPUTER SCIENCE

BY
OLGA MEDRANO MARTÍN DEL CAMPO

CHICAGO, ILLINOIS

JUNE 2026

Para mis queridos padres, Olga y Adán;
para mi buen hermano, Adán;
para mi familia, biológica y escogida;
y para tanta gente que me apoyó en el camino.

Instructions on not Giving up

by Ada Limón ¹

More than the fuchsia funnels breaking out
of the crabapple tree, more than the neighbor's
almost obscene display of cherry limbs shoving
their cotton candy-colored blossoms to the slate
sky of Spring rains, it's the greening of the trees
that really gets to me. When all the shock of white
and taffy, the world's baubles and trinkets, leave
the pavement strewn with the confetti of aftermath,
the leaves come. Patient, plodding, a green skin
growing over whatever winter did to us, a return
to the strange idea of continuous living despite
the mess of us, the hurt, the empty. Fine then,
I'll take it, the tree seems to say, a new slick leaf
unfurling like a fist to an open palm, I'll take it all.

1. Ada Limón, *The Carrying: Poems*. First edition. Minneapolis, Minnesota: Milkweed Editions, 2018.

TABLE OF CONTENTS

LIST OF FIGURES	viii
LIST OF TABLES	ix
ACKNOWLEDGMENTS	x
ABSTRACT	xii
1 INTRODUCTION	1
2 OPTIMAL ALGORITHMS FOR LEARNING PARTITIONS WITH FAULTY OR- ACLES	13
2.1 Introduction	13
2.1.1 Background and related results	16
2.1.2 Discussion of the Model	17
2.2 Technical preliminaries	19
2.3 A hierarchy of problems	19
2.4 Algorithmic results	24
2.5 Lower bound techniques: correlation clustering and the Chip-Liar game . . .	25
2.6 Algorithm for weighted ell-PL with k-known	27
2.7 Additional relevant background	32
2.8 Equivalence between weighted problem and doubly constrained one	34
2.9 Deferred proofs from Section 2.6	35
2.10 The algorithm for k-unknown	42
2.11 Lower bound results	50
2.11.1 Lower bounds for the easy problem: ell-false negatives	50
2.11.2 Lower bounds for the easy problem: ell-false positives	52
2.12 One-sided error lower bounds strategies	54
2.12.1 A responder strategy for the RUCCfn game	54
2.12.2 A responder strategy for the RUCCfp game	56
2.13 Conclusions, Limitations, and Future Directions	60
3 EPSILON-SATURATION FOR STABLE GRAPHS AND LITTLESTONE CLASSES	62
3.1 Introduction	62
3.2 Notation	63
3.3 Preliminaries: good, excellent, stable, Littlestone	64
3.4 Definition of epsilon saturation for the hypothesis class	72
3.5 Preservation of dimension for small epsilon	75
3.6 Characterization of epsilon saturation for Littlestone classes	80
3.7 Intermediate epsilon	83
3.8 Conclusions for Littlestone classes	89
3.9 Definition of epsilon saturation for graphs	90
3.10 Characterization of epsilon saturation for stable graphs	94

3.11	Examples	99
3.11.1	Cliques	99
3.11.2	Matchings	100
3.11.3	Random graphs	100
3.11.4	Half-graphs	101
3.11.5	An example where saturation takes at least two steps	102
3.11.6	An example where k extension steps are needed before saturation	103
3.12	Examples for large epsilon	105
3.12.1	An example where VC-dimension goes to infinity	105
3.12.2	An example of a good set which is not an excellent set	106
3.12.3	Extensions	107
3.12.4	Realizability and the value of epsilon	108
3.13	Open questions	108
4	LOCAL COMBINATORIAL ANALOGUES FOR BOUNDED VC DIMENSION	110
4.1	Introduction	110
4.1.1	Prior and Related Work	111
4.2	Chapter Overview	111
4.3	Background	112
4.3.1	Notation	112
4.3.2	Littlestone and VC dimensions	112
4.3.3	Good sets	114
4.3.4	Antigood sets	114
4.4	Contributions of this chapter	115
4.4.1	Characterization of classes of bounded VC dimension	115
4.4.2	Existence of sets with small antigood sets	116
4.5	Proof of Theorem 4.4.4	117
4.6	Comparison with Littlestone characterization	123
4.7	Analysis of the half graph and some examples	124
4.7.1	Countable half graph	124
4.7.2	Good sampling	125
4.7.3	Interaction of B , epsilon, and the size of the epsilon-antigood set of B	127
4.7.4	Other examples of VC classes	129
4.8	Comparison with proof via Haussler Packing	131
4.9	Comparison with proof via VC Regularity Lemma	134
4.10	Concluding remarks	137
4.11	Appendix: referenced and used results	137
4.11.1	Regularity Lemma	137
4.11.2	Fundamental Theorem of Statistical Learning	139
4.11.3	Haussler's Packing Lemma	140
5	OPEN QUESTIONS AND EXAMPLES	142
5.1	Follow ups from Chapter 3	142
5.2	Comparison of results in Chapter 4	146

REFERENCES	149
----------------------	-----

LIST OF FIGURES

1.1	Diagram of an ϵ -hypothesis in the class given by a matching between two countable sets, X and $\mathcal{H}$	5
2.1	Diagram showing the lattice of problems considered in this chapter: ℓ -PL, weighted ℓ -PL, ℓ -PL ^{FP} , and ℓ -PL ^{FN} . Algorithmic results propagate from left to right while lower bound results propagate right to left.	22
2.2	Drawing that shows an interim partition $\mathcal{C}^*$ used for the $(k-1)$ responder strategy for the RUCC ^{FP} game.	57
3.1	Phase diagram showing the regimes of possible values of the parameter $\epsilon \in (0, 1/2)_{\mathbb{R}}$, and the corresponding statements of what we showed about $\text{Ldim}(\mathcal{H}_{\infty})$ and $\text{VC}(\mathcal{H}_{\infty})$, given that $\text{Ldim}(\mathcal{H}) = \ell$ and $\text{VC}(\mathcal{H}) = d$	90
3.2	A diagram showing a bigraph with $V = A \cup B$, $A = \{a_i\}_{i \in [6]}$, and $B = \{b_j\}_{j \in [6]}$, which are connected in a way so that while A and B are both ϵ -good, neither is ϵ -excellent, for $\epsilon = 21/60$	107
4.1	Drawing showing an example of a pair of vertices (A, B) that is connected so that it is ϵ -good in the asymmetric sense, but not in the symmetric sense, for every $\epsilon \in (0, 1/2)_{\mathbb{R}}$	115

LIST OF TABLES

2.1	Table showing the query complexity of the different variants of the ℓ -PL problem, for both the k -known and the k -unknown setting.	20
4.1	Table comparing the results and the dependencies on the sizes and properties of the near equipartitions on parameters for three different Regularity Lemmas. . .	139
5.1	A table summarizing the results on symmetric good pairs and large sets with small antigood sets characteristic of classes of VC dimension bounded by some fixed $d \in \mathbb{N}$	147
5.2	A table summarizing the results on symmetric good pairs and large sets with small antigood sets characteristic of classes of VC dimension bounded by some fixed $d \in \mathbb{N}$. There, the parameter $\epsilon \in (0, 1/2)_{\mathbb{R}}$ is scaled for.	148

ACKNOWLEDGMENTS

To those alluded to on this page: thank you. Your presence meant a lot and was, on occasions, what helped me keep going towards this goal. I strive to live on in gratitude for all that each of you did to help me.

I appreciate the support, advice, and wise insights of Maryanthe Malliaris: her patience and feedback helped nurture a belief in my creativity and capacity to work towards becoming a scientist, to appreciate the little wins and the lessons that come with doing math. Thanks for helping me to work to become a better mathematician and computer scientist, step by step. I appreciate the advice, feedback, and understanding of Alexander Razborov: a lot was learned from him, including the importance of being resilient and responsible in academic contexts. I appreciate the patience and insightful mathematics conversations with Leonardo Nagami Coregliano: a lot was learned from his mentoring and feedback, and many improvements that this thesis underwent owe a lot to his help.

Thanks to Shay Moran for the support and the insightful mathematical conversations. The time we collaborated along with Maryanthe was one that I enjoyed very much, and learned a lot from. Thanks to Denis Hirschfeldt, Phokion Kolaitis, Mariana Vicaria, Cindy Wyels, Zsuzsanna Szaniszlo, Enrique Treviño, Gordon Kindlmann, Grant Ho, Aaron Potechin, and Caroline Terry, and many more for the mentorship and insightful mathematical conversations, which motivated me to keep working on improving as a researcher. Thanks to Adela DePavia and Erasmo Tani, for the time that we collaborated together and all the insightful conversations. Thanks to the Logic and Combinatorics group and to the Theory group at UChicago. Thanks to each person who shared their career, pedagogical, or math advice. This list, like the ones that will follow, anonymizes the names of many amazing people. For what it's worth this sends gratitude to those who shared their help along the way.

Thanks to my family for everything. Mom and Dad, thank you so much for your immense love and support, and for believing in me, even when the going got tough. Adán, thank you for your straight-to-the-point and helpful advice, your beautiful music, and the reminders

to laugh, smile, problem-solve, exercise, and be playful once in a while. Kevin, thank you for listening to, believing in, and accepting me through and through so many times: much respect, love and gratitude to you and yours. Thanks to my friends from back home, from our neighborhood, from school, from math olympiads, and from undergrad, for the meaningful times we shared and for your cheerful presence in my life. Thanks to everyone who has been a part of my education.

Thanks to the amazing people in Math, CS, PSD, CAM, TTIC and the wider campus and off-campus community whose friendship, solidarity, allyship, or simply being helped me keep going many times; to Student Wellness, healthcare providers, and UChicago CARES, for supporting me to heal and take care; to Calvert House and Zen at Rockefeller, for the space and community; to the instructors and community at Ratner for being welcoming and turning every day into something to look forward to; and to UCMA, IHouse, CI+I, SACNAS, FSILG, SHPE, IGSAB, for giving me opportunities to make precious and esteemed friendships; to the HPKW Food Pantry, UCSC, PSI, GSU, and AWM for showing me the power of community and collective, purposeful actions. To everyone who took part in organizing, giving talks, and showing up for the Graduate Logic and Combinatorics Seminar. To everyone with whom I co-worked or co-organized something during this time. To Phoenix STEM, for all the support, motivation, and career advice; to the Math Office, CS Office, Math Teaching, ELI, UGrad, and MyChoice teams, for their service and provided resources, advice, opportunities, and training. Thanks to everyone who provided essential services that made a living possible for so many of us, for your showing up every day to help others.

To all who are mentioned here, individually or collectively: this was nothing short of a journey, and crossing paths with you was nothing short of a miracle. For the laughs, the tears, the space, the hugs, the chalk, the words, the conversations, the explanations, the talks, the walks, and a ton more: thank you.

ABSTRACT

This thesis has five chapters. A longer overview of each will be given in Chapter 1.

Chapter 2 consists of joint work with Adela DePavia and Erasmo Tani on *Optimal Algorithms for Learning Partitions with Faulty Oracles* [57]. Consider a learner seeking to fully recover a partition of a set with $n < \infty$ elements into k groups (k might be known or not to the learner), based on answers to oracle queries of the form “Are u and v in the same group?”, for u, v in the set. We define exact partition recovery problems with bounded tolerance for faulty oracle answers, false positives, false negatives, or a weighted combination of both. We designed optimal algorithms for this task and proved they achieve optimal query complexity. We discuss the asymmetry between false negative robustness, and false positive robustness.

Chapter 3 consists of joint work with Maryanthe Malliaris and Shay Moran, on *Epsilon-saturation for stable graphs and Littlestone classes* [104]. We introduce the concept of the ϵ -saturation of a (bi)graph, which is obtained by taking the union closure after inductively adding virtual elements, determined by pointwise weighted ϵ -majorities (cf. good and excellent sets from the Stable Regularity Lemma [107]). In the Littlestone class case, we characterize this closure as the smallest ϵ -saturated class containing the initial one. We show the following, for a class $\mathcal{H}$ of Littlestone dimension ℓ and Vapnik-Chervonenkis dimension d : if $\epsilon < 1/(\ell + 1)$, then $\text{Ldim}_\infty(\mathcal{H}) = \text{Ldim}(\mathcal{H}(\epsilon)) = \ell$; if $\epsilon < 1/(d + 1)$, then $\text{VC}(\mathcal{H}_\infty(\epsilon)) = \text{VC}(\mathcal{H}(\epsilon)) = d$; but if $\epsilon \geq 1/(d + 1)$, then $\text{VC}(\mathcal{H}_\infty(\epsilon))$, and thus $\text{Ldim}(\mathcal{H}_\infty(\epsilon))$ may go to infinity; if $\epsilon \in [1/(Cd), 1/(d + 1)]$ for a numerical constant $C \approx 27.7259$, then $\text{Ldim}(\mathcal{H}_\infty(\epsilon))$, though it may change. We show some analogous results on the case of ϵ -saturation of stable graphs.

Chapter 4 consists of *Local Combinatorial analogues for bounded VC dimension*. From the work of Malliaris and Shelah in [107], and Malliaris and Moran in [105], a class is Littlestone if and only if within any $A \subseteq X$, there exists $A' \subseteq A$ of size at least $c_1|A|$ that is ϵ -good. In analogy to this, we characterize the property of bounded VC by the existence of so-called *symmetric ϵ -good pairs*. We prove this in three ways: first, we use combinatorial techniques

including Pigeonhole Principle; second, we apply Haussler’s Packing Lemma (see [86]); last, we apply the VC Regularity Lemma, (see [101, 16, 68]).

Chapter 5 states and explores open questions and further research directions.

CHAPTER 1

INTRODUCTION

This thesis is composed of the work for three papers. Chapters 2, 3, and 4 will cover each of them individually. Lastly, a fifth chapter presents some open questions and possible further research directions.

Chapter 2 consists of a project done jointly with colleagues Adela DePavia and Erasmo Tani [57]. This work was accepted for the poster session of NeurIPS 2024. We study the active graph learning problem of recovering a partition of a finite set of n objects into k parts, or clusters via same-cluster queries to an oracle. The t -th query that the learner asks is of the form: *Are u_t and v_t in the same cluster?*, where u_t and v_t are elements of the set in question. The parameter n is known to the learner, and the parameter k may or may not be. The oracle answers $r_t \in \{+1, -1\}$, corresponding to *yes* and *no* answers. The learner may be adaptive, having the option to decide the next query based on previous answers. We study variants of this problem in which the oracle may return false answers nonconsistently up to some finite error tolerance. These are the variants we studied:

1. The ℓ -PL^{FP} problem of *partition learning with ℓ false positives*, where the oracle may provide at most $\ell \in \mathbb{N}$ many false positives: $r_t = +1$, and u and v do not belong to the same cluster, or $u_t \not\sim v_t$.
2. The ℓ -PL^{FN} problem of *partition learning with ℓ false negatives*, where the oracle may provide at most $\ell \in \mathbb{N}$ many false negatives: $r_t = -1$, and u_t and v_t belong to the same cluster, or $u_t \sim v_t$.
3. Weighted ℓ -PL problem of *partition learning with weighted false positive and false negative tolerance*, where given $\ell_{\text{yes}}, \ell_{\text{no}}$, the oracle is $(\ell_{\text{yes}}, \ell_{\text{no}})$ -faulty: its answers to finitely many queries asked by the learner satisfy the following if $\ell_{\text{yes}} \neq 0$ and $\ell_{\text{no}} \neq 0$:

$$\frac{|\{t : u_t \not\sim_{\mathcal{C}} v_t \wedge r_t = +1\}|}{\ell_{\text{yes}}} + \frac{|\{t : u_t \sim_{\mathcal{C}} v_t \wedge r_t = -1\}|}{\ell_{\text{no}}} \leq 1,$$

the following in the respective cases that $\ell_{\text{no}} = 0$ or $\ell_{\text{yes}} = 0$:

$$\frac{|\{t : u_t \not\sim_{\mathcal{C}} v_t \wedge r_t = +1\}|}{\ell_{\text{yes}}} \leq 1, \quad \frac{|\{t : u_t \sim_{\mathcal{C}} v_t \wedge r_t = -1\}|}{\ell_{\text{no}}} \leq 1,$$

and $|\{t : u_t \not\sim_{\mathcal{C}} v_t \wedge r_t = +1\}| = |\{t : u_t \sim_{\mathcal{C}} v_t \wedge r_t = -1\}| = 0$ in the case that $\ell_{\text{yes}} = \ell_{\text{no}} = 0$.

4. Unweighted ℓ -PL problem of *partition learning with ℓ faulty answers*, where the oracle is (ℓ, ℓ) -faulty for some $\ell \in \mathbb{N}$ (just denoted as ℓ -faulty), meaning that the number of false answers is no more than ℓ .

Both in the ℓ -PL^{FP} problem and in the weighted ℓ -PL problem, the number of false positives cannot exceed ℓ_{yes} . Analogously, the number of false negatives cannot exceed ℓ_{no} in neither of these problems. Note that each of these individual conditions does not imply that the weighted ℓ -PL problem's condition is satisfied. However, in Appendix B we argue that, within a factor of 2 of error tolerance, the weighted ℓ -PL problem is equivalent to the *asymmetric ℓ -PL problem*, which is defined as a variant in which the oracle may return up to ℓ_{yes} false positive answers and ℓ_{no} false negative ones.

The work of Reyzin and Srivastava [125] provides algorithms that, under the assumption of an always-truthful oracle, recover the exact partition in $\text{RS}^k(n, k) = n(k-1) - \binom{k}{2}$ queries when the number of clusters k is known to the learner, and $\text{RS}^u(n, k) = nk - \binom{k+1}{2}$ queries when the number of clusters k is unknown to the learner. Liu and Mukherjee [99] show tight lower bounds, implying that such algorithms are optimal. Several works that followed, including [22], [128], [53], [31], and [33], study the benefits of accessing a same-cluster oracle for improvements in related problems.

In our work, one of the main results is the construction of an algorithm that solves the weighted ℓ -PL problem, with optimal query complexity. Optimality is shown via tight lower bounds found using a Rényi–Ulam style 2-player game based on *correlation clustering*. More concretely, our main results (Theorems 4.1 and 4.3) say the following:

Theorem (DePavia, M., Tani). *There exist algorithms (Algorithm 1 and Algorithm 9, respectively), which with optimal query complexity:*

1. *solve the weighted ℓ -PL problem, fully recovering $\mathcal{C}$ in the k -known setting with query complexity bounded by $O(\text{RS}^k(n, k) + (n - k)\ell_{\text{yes}} + k^2\ell_{\text{no}})^1$;*
2. *solve the weighted ℓ -PL problem, fully recovering $\mathcal{C}$ in the k -unknown setting with query complexity bounded by $O(\text{RS}^u(n, k) + (n - k)\max\{\ell_{\text{yes}}, \ell_{\text{no}}\} + k^2\ell_{\text{no}})^2$.*

While algorithms for the other variants of the problem are also given, the ones alluded to above can specialize to each of the problems we study. That is, by letting $\ell_{\text{no}} < 1$, $\ell_{\text{yes}} < 1$, or $\ell_{\text{yes}} = \ell_{\text{no}}$, we obtain the ℓ -PL^{FP}, the ℓ -PL^{FN}, and the unweighted ℓ -PL problems, respectively.

A last note is that the asymmetry between the complexity cost for false positive and false negative robustness is discussed in more detail in the chapter. Briefly, it comes from the observation that certifying k -clusters would have different requirements in each case. In the case of false positives, this would require a spanning forest with $(n - k)$ edges corresponding to +1 answers. In the case of false negatives, this would require a clique of size k with each one of the $O(k^2)$ edges corresponding to -1 answers.

Chapter 3 consists of a project done jointly with Maryanthe Malliaris and Shay Moran [104]. The objects we study are pairs $(X, \mathcal{H})$, where every element of $\mathcal{H}$ is a function, or hypothesis, from the domain X to $\{0, 1\}$. The following example gives an initial view of the question studied. Consider such a pair $(X, \mathcal{H})$ where $\mathcal{H}$ consists of the singletons, $h_{x_0}: X \rightarrow \{0, 1\}$ for each $x_0 \in X$, such that $h_{x_0}(x) = 1$ if $x = x_0$ and 0 otherwise. While the all-zero function is not present in the collection $\mathcal{H}$, it can be presented as a weighted average for a finite set of functions. Concretely, given $\epsilon \in (0, 1/2)$, we define a *weighted ϵ -good set* as a set $H = \{(h_i, \gamma_i) : i < i_H\}$ of size $i_H < \infty$, where all $h_i \in \mathcal{H}$ are given

1. $\text{RS}^k(n, k) = n(k - 1) - \binom{k}{2}$.

2. $\text{RS}^u(n, k) = nk - \binom{k+1}{2}$.

a finitely supported probability distribution by the real weights $\gamma_i \in (0, 1]$, and with the property that for all $x \in X$, there exists a value $\mathbf{t} = \mathbf{t}(x, H) = \mathbf{t}_H(x) \in \{0, 1\}$ for which

$$\sum \{\gamma_i : h_i(x) = \mathbf{t}\} \geq 1 - \epsilon, \text{ and hence } \sum \{\gamma_i : h_i(x) = 1 - \mathbf{t}\} < \epsilon.$$

When the weights γ_i are equal, we may just say the set is unweighted, or just ϵ -good. Such set has the property that majority values are defined on the entire set X , and thus it determines a total function $\mathbf{t}_H : X \rightarrow \{0, 1\}$. Any function $\mathbf{t}_H$ obtained from an ϵ -weighted good set H is called an ϵ -hypothesis.

Note that ϵ -good sets (see above; also see Definition 4.3.9) were defined first in the context of the Stable Regularity Lemma in logic and finite combinatorics (see [107]), in analogy to indiscernible sets in model theory, as is explained in [108]. Although there is no direct correspondence to ϵ -good sets from learning theory, let us mention some of these connections which motivate this work.

A first connection is that online learnable classes are exactly those of finite Littlestone dimension (see [97, 26]), which are also exactly those with linear-sized ϵ -good sets (see [107]; also [105]). These sets are shown to exist by using Littlestone mistake trees. A second connection is that ϵ -good sets provide a $\{0, 1\}$ weighted majority: notions of (not necessarily $\{0, 1\}$ valued) weighted majorities is also relevant in learning. For example, multiplicative weights algorithms from the work of Littlestone and Warmuth [98] take weighted additive majorities; these are applied in the work of Malliaris and Moran [105] to find linear-sized ϵ -excellent sets in Littlestone classes. Another example from the work of Hannecke, Livni and Moran [85] uses mixed strategies in online learning; moreover, one of their main results, Littlestone Minimax Theorem, is applied in the setting of our work. A third connection is through the Standard Optimal Algorithm, from the work of Littlestone [97]. In this case, learning is done by taking Littlestone dimension majorities.

Figure 1.1 below shows an approximation of the all-zero function as the average of, say,

the first $\lceil 1/\epsilon \rceil$ elements.

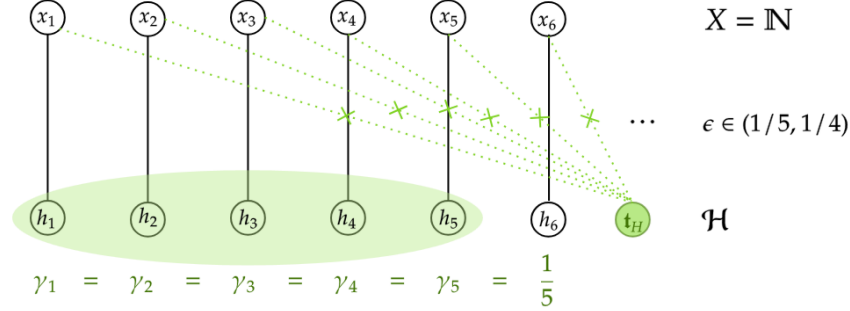


Figure 1.1: We fixed $\epsilon \in (1/5, 1/4)$. In this particular example, if we give each of the first 5 functions a weight of $\gamma_i = 1/5$, then the weighted majority $\mathbf{t}_H$ of these $5 = \lceil 1/\epsilon \rceil$ hypotheses is the function that returns $\mathbf{t}_H(x) = 0$ for all $x \in X$. Let us call the all-zero function $\mathbf{t}_0$.

There are some questions that follow from this example. For instance: *What happens if we add all the functions that can be ϵ -approximated by $\mathcal{H}$ in this way, to $\mathcal{H}$? Call the object we obtain from doing this $\mathcal{H}_1(\epsilon)$. Now, what happens if we do the same process on $\mathcal{H}_1(\epsilon)$? What happens as we iterate this n times to obtain $\mathcal{H}_n(\epsilon)$? How does $\mathcal{H}_n(\epsilon)$ change as n increases? Ultimately, what happens in the limit as $n \rightarrow \infty$?*

Sometimes, this process stops quickly. In the example of the singleton class mentioned above, $\mathcal{H}_1(\epsilon) = \mathcal{H} \cup \{\mathbf{t}_0\}$, where $\mathbf{t}_0$ is the all-zero function, since this was the only ϵ -hypothesis added to $\mathcal{H}$. Indeed, any (trivial or not) ϵ -hypothesis $\mathbf{t}_H$ coming from the ϵ -good set $\{(h_i, \gamma_i) : i < i_H\}$ sends no more than 1 element in X to 1. Otherwise, there would exist two distinct x_1, x_2 in X , with $\mathbf{t}_H(x_1) = \mathbf{t}_H(x_2) = 1$, thus $\sum_{i:h_i(x_1)=1} \gamma_i > 1 - \epsilon$ and $\sum_{i:h_i(x_2)=1} \gamma_i > 1 - \epsilon$. While the indices appearing in these two sums form disjoint sets, the weight sums add up to more than $2 - 2\epsilon > 1$, giving a contradiction. Similarly, we can show by induction on n that every $h \in \mathcal{H}_n(\epsilon)$ is such that $|h^{-1}(1)| \leq 1$. Thus, $\mathcal{H}_n(\epsilon) = \mathcal{H}_1(\epsilon)$ for all finite n .

It turns out that in the general case, more interesting things can possibly happen. We give some definitions below to have notation by which to address these questions.

More formally, let $\epsilon \in (0, 1/2)$ be a fixed parameter. Define (weighted and unweighted)

ϵ -good sets and ϵ -hypotheses as we did above. Then, define $\mathcal{H}_0(\epsilon) = \mathcal{H}$ and

$$\mathcal{H}_{n+1}(\epsilon) := \mathcal{H}_n(\epsilon) \cup \{\mathbf{t}_H : H \text{ is a weighted } \epsilon\text{-good set of } \mathcal{H}_n(\epsilon)\}.$$

Then, the ϵ -saturation $\mathcal{H}_\infty(\epsilon)$ for a class $(X, \mathcal{H})$ (see Definition 3.4.9), is given by the union $\mathcal{H}_\infty(\epsilon) = \bigcup_{n < \omega} \mathcal{H}_n(\epsilon)$. Note that here we have a hypothesis class, and only $\mathcal{H}$ is modified by adding ϵ -hypotheses, while X is unchanged throughout. Two natural variants come from working with (a) a bipartite graph both of whose sides are updated simultaneously, and (b) a graph $\mathcal{G} = (V, E)$ where V and E are updated symmetrically. Both of these are included in the chapter but are omitted now for simplicity of this introduction.

We focus on Littlestone classes. A main reason is that while any given class contains trivial ϵ -good sets – for each $h \in \mathcal{H}$, there is an ϵ -good set of size 1, $\{(h, 1)\}$ –, Littlestone classes have the characteristic property that every finite $H \subseteq \mathcal{H}$ contains a linear-sized ϵ -good set, or ϵ -excellent set (see the work of Malliaris and Shelah [107], and the work of Malliaris and Moran [105]). Thus, our initial class has Littlestone dimension $\ell := \text{Ldim}(\mathcal{H})$. Recall that its VC dimension $d := \text{VC}(\mathcal{H})$ is less than or equal than ℓ (see Definitions 3.3.13 and 3.3.15 below). Also, while ℓ being finite implies d being finite, it is possible to have ℓ being infinite and d being finite.

One of the main results of this paper is Theorem 3.8.1, which gives the following complexity trichotomy (see also Figure 3.1):

Theorem (Malliaris, M., Moran). *Let $(X, \mathcal{H})$ be a hypothesis class such that $\text{Ldim}(\mathcal{H}) = \ell$ and $\text{VC}(\mathcal{H}) = d$. The ϵ -saturation of these classes satisfy the following:*

- (a) When $\epsilon \leq 1/(\ell + 1)$, $\text{Ldim}(\mathcal{H}_\infty(\epsilon)) = \text{Ldim}(\mathcal{H}) = \ell$;
- (b) When $\epsilon \leq 1/(d + 1)$, $\text{VC}(\mathcal{H}_\infty(\epsilon)) = \text{VC}(\mathcal{H}) = d$;
- (c) For $\epsilon < 1/(Cd)$, for some constant $C \gtrsim 27.7259$, $\text{Ldim}(\mathcal{H}_\infty(\epsilon)) < \infty$ although it need not stay the same as $\text{Ldim}(\mathcal{H})$;

(d) For $\epsilon > 1/(d+1)$, there exists an $\mathcal{H}$ such that both $\text{Ldim}(\mathcal{H}_\infty(\epsilon))$ and $\text{VC}(\mathcal{H}_\infty(\epsilon))$ to be infinite.

Let us emphasize that in (c), we bound ϵ only in terms of VC dimension and find that Littlestone remains finite. Also in that case, the Littlestone dimension of $\mathcal{H}_\infty(\epsilon)$ is controlled by the VC dimension, which is usually considered a weaker notion as $\text{VC}(\mathcal{H}) \leq \text{Ldim}(\mathcal{H})$.

In working with hypothesis classes, we were essentially working with bipartite graphs. For graphs more generally, the process of obtaining the saturation $\mathcal{G}_\infty(\epsilon)$ is distinct (see Definition 3.9.10). The notion of ϵ -good sets used to determine ϵ -hypotheses has to be replaced with the stronger notion of ϵ -excellent sets, to determine ϵ -vertices (see Definition 3.9.4). We find the following complexity trichotomy in the case of graph ϵ -saturation (see Conclusion 3.10.9 in the paper).

Theorem (Malliaris, M., Moran). *Let $\mathcal{G}$ be a graph.*

(a) If $\text{Ldim}(\mathcal{G}) := \ell < \infty$, and $\epsilon < 1/2^{\ell+1}$, then $\text{Ldim}(\mathcal{G}) = \text{Ldim}(\mathcal{G}_\infty(\epsilon)) = \ell$.

(b) If in addition $\epsilon < 1/2^{\ell+2}$, then $\text{Thr}(\mathcal{G}) = \text{Thr}(\mathcal{G}_\infty(\epsilon))$.³

(c) If $\text{VC}(\mathcal{G}) := d < \infty$, and $\epsilon < 1/2^{d+1}$, then $\text{VC}(\mathcal{G}) = \text{VC}(\mathcal{G}_\infty(\epsilon))$.

This work allows to see some connections and interactions between the fields of learning theory and model theory. Although not emphasized in this introduction, saturation is defined for infinite structures in Model Theory, and an important part of our work is how the property of saturation is defined in this possibly finite context (cf. the Definition 3.6.1 for ϵ -saturated; Definition 3.4.8 for the class case, and Definition 3.10.1 for the graph case). In particular, we find that for the case of classes, if $\mathcal{H}_\infty(\epsilon)$ is Littlestone, then it is ϵ -saturated, and moreover, that every $\mathcal{H}'$ that is ϵ -saturated and contains $\mathcal{H}$, also contains $\mathcal{H}_\infty(\epsilon)$ (Theorem 3.6.2). For

3. Here, $\text{Thr}(\mathcal{G})$ is the *threshold dimension of the graph $\mathcal{G}$* , defined as the supremum of the values k such that $\mathcal{G}$ has a k -half graph: a set of vertices $\{x_1, \dots, x_k, h_1, \dots, h_k\}$ such that there exists an edge between x_i and h_j if and only if $i < j$.

the case of stable graphs, we find that $\mathcal{G}_\infty(\epsilon)$ is an ϵ -saturated object if $\mathcal{G}_\infty(\epsilon)$ is Littlestone (Theorem 3.10.3).

For the proofs in this work, we use theorems and methods from combinatorics, statistical learning, logic, Stable Regularity from the work of Malliaris and Shelah [107], and Littlestone Minimax from the work of Hannecke, Livni and Moran [85].

Chapter 4 consists of solo work. In it, we consider a class $(X, \mathcal{H})$ and study it as a bigraph. Suppose we assume that $\text{VC}(\mathcal{H}) \leq d < \infty$. This assumption means that the largest possible set that can be shattered is of finite size d . According to Sauer–Shelah–Perles Lemma [129, 133], this is saying that $\mathcal{H}$ eventually does not realize more than polynomially many different patterns. This bigraph, however, cannot a priori be guaranteed to have the same characteristics of a Littlestone class because we only know that its Littlestone dimension is lower bounded by its VC dimension, but $\text{Ldim}(\mathcal{H})$ need not be finite. Recall that for Littlestone classes, a characterization was given in the work of Malliaris and Shelah [107], as well as in the work of Malliaris and Moran [105]. Part of this characterization says, in words, that a given class $\mathcal{H}$ is of finite Littlestone dimension, if and only if there exists a constant $c = c(\epsilon) > 0$ such that the following happens: any large finite set $H \subseteq \mathcal{H}$ contains some $H' \subseteq H$, of size at least $|H'| \geq c|H|$, and that is (*unweighted*) ϵ -good, namely, for every $x \in X$, there exists $\mathbf{t} \in \{0, 1\}$ such that $|\{h \in H' : h(x) = \mathbf{t}\}| \geq (1 - \epsilon)|H'|$. In fact, this set H has to moreover contain a linear-sized (*unweighted*) ϵ -excellent set, a term that will not be defined in this introduction, for purposes of exposition. Note that, unlike in Chapter 3, in this chapter we will only consider the unweighted versions for ϵ -good sets.

In this chapter we set out to explore the natural analogues of unweighted ϵ -good sets in VC classes by allowing for some error. First, given $\epsilon > 0$ and $B \subseteq \mathcal{H}$, we define its ϵ -antigood set $\text{Anti}(B, \epsilon)$ as follows:

$$\text{Anti}(B, \epsilon) := \{x \in X : \nexists \mathbf{t} \in \{0, 1\} \text{ for which } |\{h \in B : h(x) = \mathbf{t}\}| \geq (1 - \epsilon) \cdot |B|\}.$$

Second, we define a *symmetric ϵ -good pair*, as a pair (X', H') of finite sets $X' \subseteq X$ and $H' \subseteq H$, such that the following holds:

$$\forall x \in X', \exists \mathbf{t} \in \{0, 1\} \text{ such that } |\{h \in H' : h(x) = \mathbf{t}\}| > (1 - \epsilon)|H'|,$$

$$\forall h \in H', \exists \mathbf{t} \in \{0, 1\} \text{ such that } |\{x \in X' : h(x) = \mathbf{t}\}| > (1 - \epsilon)|X'|.$$

In words, every $x \in X'$ has a majority vote on H' , and every $h \in H'$ has a majority vote on X' . Equivalently, this definition gives a pair of sets such that $X' \cap \text{Anti}(H', \epsilon) = \emptyset$ and $H' \cap \text{Anti}(X', \epsilon) = \emptyset$ (see Definition 4.4.2). Such pairs are the analogue to ϵ -good sets that we find in this chapter.

Consider the following motivating example where the class is of bounded VC dimension, but not Littlestone. Concretely, let us consider a countable half graph $(X, \mathcal{H}_\omega)$, where $X := \{x_i : i < \omega\}$ and $\mathcal{H}_\omega := \{h_j : j < \omega\}$ follow the pattern

$$h_j(x_i) = 1, \text{ or in other words } x_i \sim h_j, \text{ iff } i < j.$$

If we fix our parameter $\epsilon < 1/2$, then any finite set H of $\mathcal{H}_\omega$ (of size at least 2) is not ϵ -good, since there exists a vertex x_i in X such that the sets $\{h_j : j \leq i\}$ and $\{h_j : j > i\}$ are off in size by at most 1, and thus not every element of X has a well-defined majority vote $\mathbf{t}(x_i, H)$. In other words, $x_i \in \text{Anti}(H, \epsilon)$ and hence $\text{Anti}(H, \epsilon)$ is nonempty.

We elaborate on two interesting properties of this example. First, the size of $\text{Anti}(H, \epsilon)$ only depends on the difference $\max\{j : h_j \in H\} - \min\{j : h_j \in H\}$, which holds no relation with $|H|$ a priori. Second, for every $A \subseteq X$ and $B \subseteq \mathcal{H}$ finite, there exists an ϵ -symmetric good pair (A', B') of vertex subsets of A and B respectively, such that $|A'| \geq \epsilon|A|/4$ and $|B'| \geq |B|/4$. In words, there exists a symmetric ϵ -good pair of large size. This example is discussed in further detail in Section 4.7 of this chapter.

More generally, we say that a class $\mathcal{H}$ has (c_1, c_2, ϵ) -good sampling when the following

happens. Given $\epsilon \in (0, 1/2)$ there exist constants $c_1 = c_1(\epsilon)$ and $c_2 = c_2(\epsilon)$, such that for every $A \subseteq X$ and $B \subseteq \mathcal{H}$ finite, there exists an ϵ -symmetric good pair (A', B') of vertex subsets of A and B respectively, such that $|A'| \geq c_1|A|$ and $|B'| \geq c_2|B|$.

We show that such property characterizes classes of VC dimension at most d . For this result, we have three different proofs and statements. The first one uses the concept of VC mistake tree (see Definition 4.3.7), and the Pigeonhole Principle:

Theorem (Theorem 4.4.4). *Let $\mathcal{H}$ be a hypothesis class consisting of functions from X to $\{0, 1\}$. The following are equivalent:*

- (a) $\text{VC}(\mathcal{H}) = d < \infty$;
- (b) $\forall \epsilon \in (0, 1/2)$ there exist constants c_1, c_2 such that $(X, \mathcal{H})$ has (c_1, c_2, ϵ) -good sampling;
- (c) $\exists \epsilon_0 \in (0, 1/2)$ and constants c_1, c_2 so that the pair $(X, \mathcal{H})$ has (c_1, c_2, ϵ_0) -good sampling.

One remark is that there is a basic lower bound for the constants c_1 and c_2 , and they depend on d and on ϵ only:

$$c_1 \geq \frac{\epsilon}{2^{d+1}} \quad \text{and} \quad c_2 \geq \frac{1}{2^{d+1}}.$$

As a closing remark on the first proof, the example stated above (see Example 4.7.2) is a class with the property of $(\epsilon/4, 1/4, \epsilon)$ -good sampling.

The second proof that good sampling characterizes classes of VC dimension at most d uses techniques from Haussler's Packing Lemma [86] on the bigraph, which we also interpret as a set system:

Proposition (Theorem 4.8.3). *Let $\epsilon > 0$, and let $B = B_\epsilon(h_0)$ be any δ -ball centered at a hypothesis h_0 . Then the size of the set $\text{Anti}(B, \epsilon)$ is no larger than:*

$$\max \left\{ \frac{|B| - 1}{|B|} \cdot \frac{\delta}{\epsilon(1 - \epsilon)} \cdot |A|, |A| \right\}.$$

Note that $|A|$ is always larger than the size of any antigood set, regardless of the choice of $\delta > 0$. Certain particular choices of δ give a small bound for $|\text{Anti}(B, \epsilon)|$ in terms of $|A|$.

A remark is that the above bounds the size of the resulting antigood set of B , whose complement can be halved using Lemma 4.5.2 to obtain a symmetric good pair. In particular, for certain choices of $\delta > 0$, (and in particular, $\delta = \epsilon/2$) the upper bound on the size of the antigood set:

$$|\text{Anti}(B, \epsilon)| \leq \frac{|B| - 1}{|B|} \cdot \frac{1}{2(1 - \epsilon)} \cdot |A|,$$

is lower than the one in the main result via the combinatorial proof. Equivalently, the lower bound for the symmetric good pair size is higher. However, the lower bound on the resulting set $B = B_\delta(h_0)$ is much smaller, at least since the lower bound $|B| \cdot (\epsilon/60)^d$ that can be obtained using this approach is smaller. For the purposes of finding the symmetric good pair, we obtain a set $B' \subseteq B_\delta(h_0)$ such that $|B'| \geq (1/2) \cdot |B_\delta(h_0)| \geq (1/2) \cdot |B| \cdot (\epsilon/60)^d$, forming a symmetric 2ϵ -good pair with a set $A' \subseteq A \setminus \text{Anti}(B, \epsilon)$ of size at least

$$\frac{1}{2} \cdot \left(1 - \frac{|B| - 1}{2|B|(1 - \epsilon)}\right) |A|.$$

This indicates that there is a tradeoff between the pair of constants (c_1, c_2) obtained with the first and the second approach. The tradeoff depends on the parameter $\epsilon > 0$, and on the size of the set $B \subseteq \mathcal{H}$.

The third proof uses the equipartition that is obtained from directly applying the VC Regularity Lemma on the bigraph in question [68, 16, 101].

Proposition (Theorem 4.9.1). *Consider a class $(X, \mathcal{H})$ such that $\text{VC}(\mathcal{H}) = d$, finite subsets $A \subseteq X$, $B \subseteq \mathcal{H}$, and a parameter $\epsilon > 0$. It is possible to find a symmetric $(100^2\epsilon/99)$ -good pair (A', B') of subsets of A and B , respectively, such that*

$$|A'| \geq \frac{99 \cdot |A|}{100 \cdot m}, \quad |B'| \geq \frac{99 \cdot |B|}{100 \cdot m},$$

for some $m = O(1/\epsilon^{2d+1})$.

A similar tradeoff analysis for the resulting lower bounds on both sides of the pair can be done. The number of parts in the VC-equipartition is $m = O(1/\epsilon^{2d+1})$, implying that $|A'| \geq 99|A|/(100m) = \Omega(|A| \cdot \epsilon^{2d+1})$ (and $|B'| = \Omega(|B| \cdot \epsilon^{2d+1})$). The dependence on the parameter ϵ is of higher degree (recall that $c_1 \geq \epsilon^d/2^{d+1}$ and $c_2 \geq 1/2^{d+1}$ in Theorem 4.4.4). Therefore, we can expect the lower bound of the size of the symmetric pair in this result to be smaller than that in the first main theorem. By this argument, the lower bound obtained by Theorem 4.9.1 is lower. This however does not rule out other bounds coming from different proofs.

Chapter 5 briefly elaborates on examples, further questions, and research directions.

CHAPTER 2

OPTIMAL ALGORITHMS FOR LEARNING PARTITIONS WITH FAULTY ORACLES

JOINT WITH ADELA DEPAVIA AND ERASMO TANI

We consider a clustering problem where a learner seeks to partition a finite set by querying a faulty oracle. This models applications where learners crowdsource information from non-expert human workers or conduct noisy experiments to determine group structure. The learner aims to exactly recover a partition by submitting queries of the form “are u and v in the same group?” for any pair of elements u and v in the set. Moreover, because the learner only has access to faulty sources of information, they require an error-tolerant algorithm for this task: i.e., they must fully recover the correct partition, even if up to ℓ answers are incorrect, for some error-tolerance parameter ℓ . We study the question: for any given error-tolerance ℓ , what is the minimum number of queries needed to learn a finite set partition of n elements into k groups? We design algorithms for this task and prove that they achieve optimal query complexity. To analyze our algorithms, we first highlight a connection between this task and correlation clustering. We then use this connection to build a Rényi–Ulam style analytical framework for this problem, which yields matching lower bounds. Our analysis also reveals an inherent asymmetry between the query complexity necessary to be robust against false negative errors as opposed to false positive errors.

2.1 Introduction

Learning cluster structure from data is a fundamental task in machine learning. While the statistical setting is typically concerned with using batch data to approximately recover cluster structure with high probability, some applications allow for the learner to make explicit queries, and some require exact recovery guarantees.

We highlight two key settings. In several scientific domains, particularly bioinformatics, researchers conduct physical experiments to learn whether two objects are part of the same class [24, 30, 76, 136]. Another major application is learning cluster structure by collecting information from human workers via crowdsourcing services [40, 114, 151]. While some traditional methods focus on querying workers for class labels, alternative approaches use simpler *same-cluster* queries. For example, [151] point out that, given images of birds, it may be easier for non-experts to correctly answer the question “Do these two birds belong to the same species?” as opposed to “What species does this bird belong to?” In both of these application domains, the learner typically seeks to minimize the number of queries made, as queries require carrying out potentially expensive measurements or time-consuming experiments.

We consider a very general setting and develop algorithms that make no assumptions about the specific type of data involved, but rather rely solely on the same-cluster queries to infer structure. In particular, we model this task by assuming that the algorithm only interfaces with the data via a *same-cluster oracle*. In this setting, the learner obtains information about a hidden partition $\mathcal{C}$ of a finite set V by repeatedly choosing two elements $u, v \in V$ and asking questions of the form “Are u and v part of the same cluster?”, i.e. given $\mathcal{C}$ and V , a same-cluster oracle is an oracle $\alpha_{\mathcal{C}}$ that takes as input u and v in V , and returns 1 if u and v belong to the same cluster in $\mathcal{C}$, and -1 otherwise. The power and limitations of same-cluster oracles have been studied in a variety of settings, including the clustering problem described above [22, 99, 125, 128].

However, previous work on this model makes the arguably unrealistic assumption that all queries return the correct answer. In the motivating applications, queries are often at risk of failure that results from relying on non-expert workers or suffering from experimental noise. The learner’s goal is to recover the partition in spite of these errors. Such errors are also often not persistent: in the presence of noise, repeating an experiment multiple times may yield different answers, and querying different human workers may result in conflicting

responses. Nonetheless, practitioners may need to achieve exact recovery of the underlying partition. However, existing theory either focuses on the error-free regime [99, 125], assumes that errors are persistent [53, 123], or focuses on probabilistic guarantees rather than exact recovery [40].

To address this literature gap, we study algorithms for partition learning via same-cluster oracles that are robust to non-persistent errors. A typical way to incorporate uncertainty and noise is to assume that errors occur independently at random with some small probability on every query. However, it is not possible to guarantee exact recovery of the underlying clusters in this model. Instead, we focus on a model in which the learner sets an error-tolerance parameter ℓ , and they require guaranteed full recovery of the hidden partition as long as the error incurred is within this tolerance. In particular, we say a same-cluster oracle $\alpha_{\mathcal{C}}$ is ℓ -faulty if it may return an incorrect answer up to ℓ times. We do not assume that errors are persistent: if a single pair of elements is queried repeatedly, an ℓ -faulty oracle may return inconsistent responses. We define the *ℓ -bounded error partition learning* (ℓ -PL) problem, as the problem of exactly recovering a hidden partition via access to an ℓ -faulty same-cluster oracle.

We design algorithms for the ℓ -PL problem and related variants, and analyze their query complexity. We introduce a two-player game based on *correlation clustering*, and we show that the minimax value of this game is closely linked to the query complexity of the ℓ -PL problem. We then use this game as a framework to give tight lower bounds for the query complexity of this task, proving the proposed algorithms are optimal.

We find that the ℓ -PL problem occupies a unique position at the intersection of different research streams. In fact, the study of this problem complements work on clustering with same-cluster query advice, and the techniques used for its analysis draw from the theory of graph learning with oracle queries and the study of Rényi–Ulam liar games¹. We are hopeful that the impact of these connections will go beyond the results presented in this chapter.

1. We provide the reader with relevant background from these areas in Section 2.1.1 and Section 2.7.

2.1.1 Background and related results

Clustering with same-cluster oracles. In the error-free regime, [99] studied the query complexity of recovering partitions of a finite set in different oracle models. They prove tight lower bounds on the query complexity of learning partitions with error-free same-cluster oracles, and point out that an algorithm proposed by [125] exactly achieves this query complexity.

In many motivating applications of this model queries may return the wrong answer, such as when labels result from non-expert human input or noisy scientific experiments. Korlakai Vinayak and Hassibi [151] consider multiple query models for collecting information from human workers, pointing out that same-cluster queries can accomplish similar end goals as label queries while potentially being easier questions for non-experts to answer correctly. They also provide an algorithm that works for the setting in which triangle queries—which ask the worker to provide all pairwise relationships between three data points—are made. Mazumdar and Saha [114] initiate the formal study of clustering with same-cluster oracles in the presence of persistent i.i.d. errors. They point out that under these assumptions, learning a partition has a strong relationship to recovering community structure in the stochastic block model (SBM). This work inspired a productive line of research on the i.i.d. noise model, which also leverages connections to the SBM to study related problems. In fact, this problem has been studied in the setting when the number of clusters $k = 2$ in [78], when the underlying clusters are nearly-balanced [123], and in the semi-random noise model, in which errors occur with some i.i.d. probability, but when they do occur the (erroneous) answer may be chosen adversarially [53]. Models for non-persistent error have also been considered. Chen, Vinayak, and Hassibi [40] study same-cluster queries in the presence of i.i.d. error, allowing for repeated querying of pairs. They give an efficient algorithm with recovery guarantees that does not require a-priori knowledge of the probability of query failure. Similar results were also subsequently established in [81]. Our work complements this line of research by considering a setting in which exact recovery is possible in spite of errors and providing a

full characterization of the query complexity of the problem in this setting.

Rényi–Ulam games. In his autobiography, Stanisław Ulam, introduced the following two-player game [147]. One player (the responder) thinks of a number in $x \in [N]$ for some $N \in \mathbb{N}$, and the other player (the questioner), given N , tries to guess x by asking only yes/no questions. The main twist to this setup is that the responder is allowed to lie up to ℓ times. The term “Rényi–Ulam games” has since been used to identify a wide range of problems involving asking questions to an oracle who is allowed some limited amount of lying (see e.g. [121]). The question of finding the worst-case query complexity of the ℓ -PL problem can be naturally formulated as a Rényi–Ulam game, in which the learning algorithm takes the role of the questioner, and the oracle plays the part of the responder.

Correlation clustering. Bansal, Blum, and Chawla [25] introduced correlation clustering. In this problem, one is given a signed graph $G = (V, E, \sigma)$, where $\sigma: E \rightarrow \{\pm 1\}$ is a function representing one’s prior belief about which pairs of vertices belong to the same cluster. The goal of the problem is to return a partition of the vertices of G into clusters that minimizes the amount of disagreement with the edge signs given by σ . The problem is known to be NP-hard. Many variations of problem have been proposed, including versions with weighted edges [7, 56], a version where the number of clusters k is fixed [73], and an agreement maximization setting. Different assumptions have also been studied, such as instance stability and noisy partial information [102, 103, 111].

2.1.2 Discussion of the Model

In the previous sections, we introduced a problem in which a learner has to exactly recover a hidden partition by making same-cluster queries that could be subject to up to ℓ errors. Other plausible models for learning partitions with errors could be considered. In this section, we discuss some of the key design features of the model and provide justification and examples.

In general, even in the regime in which the number k of clusters is known to the learner, it is not possible to recover the partition exactly unless one is able to resolve all but at most 2 of the pairwise same-cluster relationship between the elements. This follows from a result of Reyzin and Srivastava (this is a key idea leveraged in the proof of Proposition 3 in [125]). In particular, simple extensions of these core arguments imply that in the setting in which the answers to the queries are persistent, it is impossible to solve the problem even for small constant values of ℓ [57, 125]. Furthermore, this implies that even in the non-persistent error-model, one cannot solve the problem for a value of ℓ that grows linearly with n . Below, we give two general motivating examples, in the technology and scientific domains respectively, illustrating the role of ℓ in partition learning tasks.

Example 1: Robustness to Misinformation. Consider a setting where a user is trying to cluster a dataset by crowdsourcing information in the form of same-cluster questions. However, the user suspects that an ill-intentioned competitor organization is attempting to corrupt the learning process by entering a number of bad actors in the crowd to strategically mislabel queries. If the user selects a new person every time they submit a query, then the number of adversarial answers they encounter is finite and does not grow with the number of queries submitted. In this scenario, ℓ plays the role of a security parameter, and the algorithm is guaranteed to be robust to up to ℓ -many poisoned responses. The user can set ℓ based on, e.g., their prior belief about the resources of the competitor organization. Our results can be interpreted as quantifying the cost (in queries / crowd size) of implementing a fixed security parameter ℓ .

Example 2: Trustworthy Science. Consider a setting in which a scientist is attempting to group items into classes by running experiments that reveal whether two items are in the same class, such as the example of clustering compatible molecules described in [81]. The scientist has limited resources (e.g. limited materials or time) and can only conduct a finite number of experiments. Our results allow the scientist to derive the maximum number of

errors to which their learning procedure can be tolerant, given their fixed query budget. They can use this maximum value as the setting for ℓ , and then use our algorithms to guide their choice of experiments. Our analysis would then allow them to measure the significance of their findings by quantifying the number of experiments that would need to have failed for the finding to be incorrect.

2.2 Technical preliminaries

Basic definitions Given a positive integer n we denote with $[n]$ the set $\{1, \dots, n\}$. Given any finite set V we denote with $\binom{V}{2}$ the set $\{\{i, j\} \subseteq V \mid i \neq j\}$. A graph $G = (V, E)$ is a pair containing a finite vertex set V and a subset $E \subseteq \binom{V}{2}$. Given a finite set V a k -partition of V is a collection of k pairwise disjoint non-empty subsets $\mathcal{C} = \{C_1, \dots, C_k\}$ of V such that $\bigcup_{a=1}^k C_a = V$. We will denote by n the cardinality $|V|$ of V and we may assume without loss of generality that $V = [n]$. Given two elements u and v of V , we write $u \sim_{\mathcal{C}} v$ (resp. $u \not\sim_{\mathcal{C}} v$) for the statement $\exists C \in \mathcal{C}, \{u, v\} \subseteq C$ (resp. $\nexists C \in \mathcal{C}, \{u, v\} \subseteq C$). We denote by $[u]_{\mathcal{C}}$ the equivalence class of u with respect to $\mathcal{C}$, i.e., $[u]_{\mathcal{C}}$ is the unique $C \in \mathcal{C}$ containing u . Given two partitions $\mathcal{C}_1$ and $\mathcal{C}_2$ we say $\mathcal{C}_2$ is a *refinement* of $\mathcal{C}_1$ if for every $C \in \mathcal{C}_2$ there is a $C' \in \mathcal{C}_1$ such that $C \subseteq C'$.

2.3 A hierarchy of problems

In the introduction we consider learning a partition $\mathcal{C}$ of a finite set V by asking questions of the form “Are u and v part of the same cluster in $\mathcal{C}$?”, where up to ℓ of the answers may be incorrect. We refer to this as the ℓ -PL problem. We characterize the hardness of the ℓ -PL problem by considering a family of related tasks. In particular, we introduce a hierarchy of problems with different degrees of difficulty (see Figure 2.1) and establish matching upper lower bounds in nearly all of these tasks. We summarize our results in Table 2.1.

At the bottom of the hierarchy, we have the problem first studied by Reyzin and Srivas-

	Query Complexity	
	k -known	k -unknown
Weighted ℓ -PL (Thm. 2.4.1, 2.4.3, and 2.5.1)	$\Theta(\text{RS}^k(n, k) + (n - k)\ell_{\text{yes}} + k^2\ell_{\text{no}})$	$O(\text{RS}^u(n, k) + (n - k)\max\{\ell_{\text{yes}}, \ell_{\text{no}}\} + k^2\ell_{\text{no}})$
ℓ -PL (Cor. 2.4.2 and 2.4.4, Thm. 2.5.1)	$\Theta(\text{RS}^k(n, k) + (n - k)\ell + k^2\ell)$	$\Theta(\text{RS}^u(n, k) + (n - k)\ell + k^2\ell)$
ℓ -PL ^{FN} (Thm. 2.4.1, 2.4.3, and 2.11.1)	$\Theta(\text{RS}^k(n, k) + k^2\ell)$	$\Theta(\text{RS}^u(n, k) + (n - k)\ell + k^2\ell)$
ℓ -PL ^{FP} (Thm. 2.4.1, 2.4.3, and 2.11.3)	$\Theta(\text{RS}^k(n, k) + (n - k)\ell)$	$\Theta(\text{RS}^u(n, k) + (n - k)\ell)$
Error-Free (Results from [99, 125])	$\text{RS}^k(n, k)$	$\text{RS}^u(n, k)$

Table 2.1: The query complexity of the different variants of the ℓ -PL problem, for both the k -known and the k -unknown setting. We obtain matching upper and lower bounds for all the variants with the exception of the weighted ℓ -PL problem in k -unknown setting, for which the upper bound does not match the lower bound (given for the k -unknown case) exactly. The results are given in terms of the complexity $\text{RS}^k(n, k)$ and $\text{RS}^u(n, k)$ (defined in Equations (2.3.1) and (2.3.2)) of solving the problem without errors.

tava [125] of learning partitions with same-cluster queries, where the answer to every query is guaranteed to be correct. They give an algorithm that can learn the underlying partition with:

$$\text{RS}^k(n, k) \stackrel{\text{def}}{=} n(k - 1) - \binom{k}{2} \quad (2.3.1)$$

many queries when the number k of clusters is known to learner and:

$$\text{RS}^u(n, k) \stackrel{\text{def}}{=} nk - \binom{k + 1}{2} \quad (2.3.2)$$

when k is unknown to the learner. Recently, Liu and Mukherjee [99] showed that no algorithm can guarantee full recovery of the underlying partition in fewer than $\text{RS}^k(n, k)$ (resp. $\text{RS}^u(n, k)$) queries when k is known (resp. unknown) to the learner, showing that the algorithms of Reyzin and Srivastava are optimal down to the exact constants. Since this is the “easiest” version of the problem we will be considering, the lower bounds of Liu and

Mukherjee immediately imply the same lower bounds for all the other problems in the family.

We then consider two variants of this problem, each only admitting one-sided error. In the first variant, which we refer to as the ℓ -bounded error partition learning with false positives (ℓ -PL^{FP}) problem, the oracle might return the answer $+1$ even when two elements are not part of the same cluster. We refer to this kind of fault as a *false positive* answer. Here, we restrict the oracle to return at most ℓ many false positive answers, for some fixed positive integer ℓ , and to always return -1 when the two elements being queried are not part of the same cluster.

In the second variant, which we refer to as the ℓ -PL^{FN} problem, the oracle behaves the opposite way, and it may return -1 even if the two elements being queried are part of the same cluster in the hidden partition (a *false negative answer*), but it cannot return a false positive answer. Similarly to the previous variant, we restrict the number of false negative answers to be at most some fixed ℓ . We give precise definitions of the ℓ -PL^{FP} and ℓ -PL^{FN} problems below.

The ℓ -PL problem is then harder than all of the three variants described above (false positives, false negatives and no error), and hence, it inherits hardness results from all three of these problems. Instead of directly designing an algorithm for the ℓ -PL problem, we consider a yet more general version of the problem, and use results from that generalized version to obtain algorithms for all of the other problems in the class. This final variant is a more fine-grained problem, which allows false positive answers and false negative answers to incur different penalties. In particular we consider a weighted version of ℓ -PL, which is formalized in the following definition:

Definition 2.3.1 ($(\ell_{\text{yes}}, \ell_{\text{no}})$ -Faulty Oracle and Weighted ℓ -PL Problem). *Let V be a finite set, and $\mathcal{C}$ be a partition of V . A $(\ell_{\text{yes}}, \ell_{\text{no}})$ -faulty oracle for $\mathcal{C}$ is an algorithm $\alpha_{\mathcal{C}}$ which, given as input a pair of elements uv , returns a value $r \in \{\pm 1\}$ so that for any sequence of queries $\{u_t v_t\}_{t \in [T]}$ the sequence of responses $\{r_t = \alpha_{\mathcal{C}}(u_t v_t)\}_{t \in [T]}$ satisfies the following if*

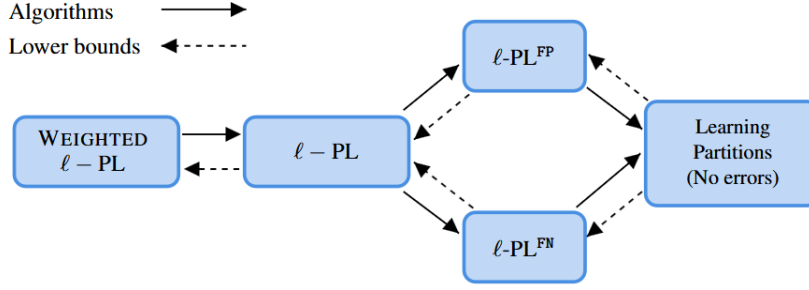


Figure 2.1: The lattice of problems considered in this chapter. Algorithmic results propagate from left to right while lower bound results propagate right to left.

$\ell_{\text{yes}} \neq 0$ and $\ell_{\text{no}} \neq 0$:

$$\frac{|\{t : u_t \not\sim_{\mathcal{C}} v_t \wedge r_t = +1\}|}{\ell_{\text{yes}}} + \frac{|\{t : u_t \sim_{\mathcal{C}} v_t \wedge r_t = -1\}|}{\ell_{\text{no}}} \leq 1,$$

the following in the respective cases that $\ell_{\text{no}} = 0$ or $\ell_{\text{yes}} = 0$:

$$\frac{|\{t : u_t \not\sim_{\mathcal{C}} v_t \wedge r_t = +1\}|}{\ell_{\text{yes}}} \leq 1, \quad \frac{|\{t : u_t \sim_{\mathcal{C}} v_t \wedge r_t = -1\}|}{\ell_{\text{no}}} \leq 1,$$

and $|\{t : u_t \not\sim_{\mathcal{C}} v_t \wedge r_t = +1\}| = |\{t : u_t \sim_{\mathcal{C}} v_t \wedge r_t = -1\}| = 0$ in the case that $\ell_{\text{yes}} = \ell_{\text{no}} = 0$.

Note that we assume the oracle can maintain an internal state, so its answers may depend on the query history. The weighted ℓ -PL problem asks one to recover an unknown k -partition of V given access to an $(\ell_{\text{yes}}, \ell_{\text{no}})$ -faulty oracle for $\mathcal{C}$.

Observe that by setting $\ell_{\text{yes}} < 1$ (resp. $\ell_{\text{no}} < 1$) one would require the oracle to not return any false positive (resp. false negative) answer. As a convention, we will write $\ell_{\text{yes}} = 0$ or $\ell_{\text{no}} = 0$ to indicate any setting that ensures the oracle may not return any false positive or false negative answers respectively. We note that, up to rescaling ℓ_{yes} and ℓ_{no} by a factor of 2, this problem is equivalent to a version of the problem in which the oracle has separate constraints for the number of false positive answers and false negative answers, a simple fact which we prove in Section 2.8.

Given this definition, the (unweighted) ℓ -PL can be cast as a homogeneous version of the

above, one in which $\ell_{\text{yes}} = \ell_{\text{no}}$, as follows:

Definition 2.3.2 (ℓ -Faulty Oracle and ℓ -PL Problem). *An ℓ -faulty oracle is an (ℓ, ℓ) -faulty oracle. The ℓ -PL problem is the problem of learning a partition with access to an ℓ -faulty oracle.*

The $\ell\text{-PL}^{\text{FP}}$ and $\ell\text{-PL}^{\text{FN}}$ problems discussed above can now be formally described as special instances of the weighted ℓ -PL problem.

Definition 2.3.3 ($\ell\text{-PL}^{\text{FP}}$ Problem and $\ell\text{-PL}^{\text{FN}}$ Problem). *The $\ell\text{-PL}^{\text{FP}}$ problem is the problem of learning a partition with access to an $(\ell, 0)$ -faulty oracle. Intuitively, this corresponds to the problem of learning partitions subject to at most ℓ false-positive responses and no false negatives. The $\ell\text{-PL}^{\text{FN}}$ problem is the problem of learning a partition with access to an $(0, \ell)$ -faulty oracle. This corresponds to learning partitions subject to at most ℓ false-negatives responses and no false positives.*

In the rest of the chapter, we will always assume that the set V , its cardinality n , and the parameters $(\ell_{\text{yes}}, \ell_{\text{no}})$ are known to the learner. We will consider both the k -known setting, in which the number k of clusters is known to learner, and the k -unknown setting, in which it is not. It is easy to see that the query complexity of every problem in the k -unknown setting is no smaller than that of the same problem in the k -known setting, since any learning algorithm could simply ignore the value of k . We design separate algorithms for the k -known and the k -unknown setting, while the lower bounds, which we only prove for the k -known setting, apply directly to the k -unknown setting.

In the next section we describe a number of algorithmic results to solve variants of the ℓ -PL problem. In particular, Theorem 2.4.1 describes the performance of an algorithm for the weighted ℓ -PL problem in the k -known setting. We briefly comment that it is actually not necessary for the number of false negatives ℓ_{no} to be bounded or known in advance in order to run this algorithm and achieve optimal query complexity. Rather the value of ℓ_{no} that appears in the query complexity analysis will be the true number of false-negative

responses that occur during the execution of the algorithm. In particular, this implies that in the version of the problem in which no false positives can occur, one does not need to set any upper bound on the error at all. In contrast in the k -unknown setting, in order to achieve the bound described in Theorem 2.4.3 our proposed algorithm does require a user-chosen value for ℓ_{no} . Here, a simple argument shows that it is impossible to guarantee one has found the correct answer unless they can upper bound both ℓ_{yes} and ℓ_{no} .

2.4 Algorithmic results

Our first result is an algorithm for the weighted ℓ -PL problem in the k -known setting.

Theorem 2.4.1. *There exists an algorithm for the weighted ℓ -PL problem which recovers the full partition $\mathcal{C}$ in the k -known setting with query complexity bounded by*

$$O(\text{RS}^k(n, k) + (n - k)\ell_{\text{yes}} + k^2\ell_{\text{no}}).$$

We observe an asymmetry in the dependence on ℓ_{yes} and ℓ_{no} respectively, indicating that false-negative and false-positive errors cause the algorithm to incur different costs. In Section 2.5 we prove lower bounds showing that this is not a mere artifact of the algorithm or its analysis, but rather a fundamental aspect of the problem.

We describe the algorithm and prove Theorem 2.4.1 in Section 2.6. Throughout the chapter, when considering the k -known setting, we shall assume that $k \notin \{1, n\}$, for otherwise the problem is trivially solved without the need to query the oracle. Because the weighted ℓ -PL problem generalizes the ℓ -PL problem, this yields an algorithm for the ℓ -PL problem as well:

Corollary 2.4.2. *There exists an algorithm for the ℓ -PL problem which learns the correct partition in the k -known setting with query complexity bounded by*

$$O(\text{RS}^k(n, k) + (n - k)\ell + k^2\ell).$$

As our final algorithmic result, in Section 2.10 we show that the algorithm can be adapted to obtain optimal asymptotic performance for the ℓ -PL problem in the k -unknown setting.

Theorem 2.4.3. *There exists an algorithm for the weighted ℓ -PL problem which recovers the full partition $\mathcal{C}$ in the k -unknown setting with query complexity bounded by*

$$O\left(\text{RS}^u(n, k) + (n - k) \max\{\ell_{\text{yes}}, \ell_{\text{no}}\} + k^2 \ell_{\text{no}}\right).$$

An immediate corollary bounds the query complexity of the ℓ -PL problem in the k -unknown setting.

Corollary 2.4.4. *There exists an algorithm for the ℓ -PL problem which learns the correct partition in the k -unknown setting with query complexity bounded by*

$$O\left(\text{RS}^u(n, k) + \ell(n - k) + k^2 \ell\right).$$

In the following section we show that all the bounds discussed in this section are asymptotically optimal. While the results are stated only asymptotically, we note that the upper bounds we obtain are only a small constant factor away from the lower bounds, and we give the exact upper bounds in Section 2.10, where we prove Theorem 2.4.3.

2.5 Lower bound techniques: correlation clustering and the Chip–Liar game

Underlying our lower bound analysis is a connection between partition learning problems, Rényi–Ulam and Chip–Liar games, and correlation clustering. The Rényi–Ulam game, as defined in the background is equivalent to the following Chip–Liar game (see e.g. Chapter 15, in the book of Alon and Spencer [18]). In the game, N chips, numbered 1 to N are placed on a game board that has $\ell + 1$ positions, labeled $0, \dots, \ell$. At the start of the game, all of the chips begin at position 0 on the board. The game takes place in rounds. On each

round the questioner player Q selects a subset S of the chips, and the responder player R decides on one of the following moves: they can either increase the position of every chip in S by 1, or increase the position of every chip in $\bar{S}$ by 1. If a chip at position ℓ is in a group whose position is advanced, the chip is said to *have fallen off the board*. After this point such chips will no longer advance. The rules of the game constrain R 's responses; R must ensure that on every round, at least one chip remains on the board at position $i \leq \ell$. The game terminates when there is a unique chip remaining on the board.

Note that one can think of the ℓ -PL problem as a constrained version of this Chip-Liar game. In the k -known setting², the chips are equivalent to k -partitions of the finite set V , so that the number N of chips is the Stirling Number of the second kind $N = \left\{ \begin{smallmatrix} n \\ k \end{smallmatrix} \right\}$. Moreover, unlike the general Chip-Liar game, in our setup the questioner may only select specific subsets S : for any pair of elements $u, v \in V$, the questioner can select S to be the set of all partitions in which u and v are part of the same cluster. When the questioner submits a query ("Are u and v in the same group?"), all chips whose partitions are inconsistent with the response advance by one position on the board.

This allows one to adopt the following perspective. One may think of the queries $\{u_t v_t\}_{t \in [T]}$ made by the questioner together with the signs given by the responses to the queries as making up an instance of the correlation clustering problem. The position of a chip on the board will then be equal the cost of the corresponding partition as a solution of the correlation clustering instance constructed. In Section 2.11.1 and Section 2.11.2, we formalize this intuition by defining *Rényi-Ulam Correlation Clustering* (RUCC) games, which are the key tools we use to lower bound query complexity.

Leveraging the above techniques, we lower bound the query complexity of the ℓ -PL^{FN} and ℓ -PL^{FP} problems. Combining these lower bounds with results in the error-free regime yield the following lower bounds on the ℓ -PL problem.

2. Note that an analogous game can be defined for the k -unknown setting, where chips are equivalent to any partition of V and N is the Bell number B_n .

Theorem 2.5.1. *Every algorithm for the ℓ -PL problem requires at least:*

$$\Omega(\max\{\text{RS}(n, k), \ell(n - k), \ell k^2\})$$

queries both in the k -known and in the k -unknown setting. Moreover every algorithm for the weighted ℓ -PL problem requires at least:

$$\Omega(\max\{\text{RS}(n, k), \ell_{\text{yes}}(n - k), \ell_{\text{no}} k^2\})$$

queries both in the k -known and in the k -unknown setting. Here $\text{RS}(n, k)$ represents $\text{RS}^k(n, k)$ in the k -known setting, and $\text{RS}^u(n, k)$ in the k -unknown setting.

This theorem is a corollary of Theorems 2.11.1 and 2.11.3, which we prove in Sections 2.12.1 and 2.12.2.

Intuitively, the difference in complexity we observe between the case of false negative errors and the case of false positive errors is due to the following. On one hand, certifying the existence of k -clusters using positive answers requires building a spanning forest, which has $(n - k)$ edges. On the other hand, accomplishing the same task using negative answers requires constructing a clique on k vertices which requires $O(k^2)$ edges. This intuition is made precise in Section 2.11 in the proofs of Theorem 2.11.1 and 2.11.3 respectively.

2.6 Algorithm for weighted ℓ -PL with k -known

In this section we give an algorithm that solves the weighted ℓ -PL problem in the k -known setting with asymptotically optimal query complexity. We later adapt this algorithm to the k -unknown setting in Section 2.10.

We begin by briefly providing intuition for the algorithm. The algorithm maintains $\mathcal{C}'$ a refinement of the correct hidden partition $\mathcal{C}$. At the start of the algorithm $\mathcal{C}'$ is the set of all singleton elements in V . The algorithm repeatedly makes sequences of queries that

guarantee progress towards either (a) learning $\mathcal{C}$, in the form of establishing that a pair of vertices u and v must be part of the same cluster, or (b) certifying that an error has occurred in the oracle's responses.

In particular, the algorithm repeatedly attempts to construct $(k + 1)$ -cliques of -1 responses. If at any point the oracle responses form such a clique, then a false-negative response must have occurred, as vertices in V belong to at most k distinct clusters (note that k is assumed to be known to the algorithm). In this case, the algorithm has certified a false-negative response. On the other hand, the only way that this process may be interrupted is if the algorithm receives a $(+1)$ response to some query uv . However, if such a positive response is received, repeatedly querying the pair uv will either quickly determine that u and v must be part of the same cluster or it will reveal that an error has occurred. In the first case, the algorithm makes progress towards learning $\mathcal{C}$, and in the second case it has certified the occurrence of an error.

We provide the pseudocode for Algorithm 1 and describe in prose the main functions of each subroutine. The algorithm is comprised of subroutines **Learn**, **Get_New**, **Insert** and **Compare**:

Learn This is the core component of the algorithm. It maintains a partition $\mathcal{C}'$ for V . Throughout the algorithm, $\mathcal{C}'$ is guaranteed to be a refinement of the true hidden partition $\mathcal{C}$. $\mathcal{C}'$ is initialized as the set of singletons for every element in V . The algorithm then repeatedly tries to construct a clique of negative answers, supported on some set $S \subseteq V$ until $|S| = k + 1$. It does so by inserting new vertices v into S by calling the subroutine **Insert**. **Insert** may then either make progress towards constructing the clique (by increasing the size of S) or towards learning the hidden partition by decreasing the size of $\mathcal{C}'$ (merging two clusters together).

Get_New Given $S \subseteq V$ and $\mathcal{C}'$ a partition of V , **Get_New** returns an element $v \in V \setminus S$ representing a cluster in $\mathcal{C}'$ that's not yet represented in S , i.e. $v \notin [u]_{\mathcal{C}'}$ for any $u \in S$.

Insert Given an element v and a subset $S \subseteq V$, **Insert** compares v against every element in S by passing u and v to the function **Compare**. **Compare** returns $\text{result} \in \{\pm 1\}$. If **Compare** returns a negative result for each $u \in S$, then **Insert** adds v to the subset S . On the other hand, if **Compare** returns a positive result for v and some element $u \in S$, then **Insert** updates the partition $\mathcal{C}'$ to reflect that u and v must be in the same cluster. The latter is done by merging $[u]_{\mathcal{C}'}$ with $[v]_{\mathcal{C}'}$.

Compare Given two vertices u and v , **Compare** repeatedly submits query uv to the oracle. **Compare** maintains **count** to record information about the number of positive and negative responses received for this query. If at any point **Compare** has received strictly more negative responses than positive responses, it returns $\text{result} = -1$.

Algorithm 1: $\text{Learn}(V, \alpha, k, \ell_{\text{yes}})$

Input: A finite set V , an $(\ell_{\text{yes}}, \ell_{\text{no}})$ -faulty oracle $\alpha_{\mathcal{C}}$, a target number of clusters k , and error parameters ℓ_{yes} .

Output: A partition $\mathcal{C}'$ of V .

```

1  $S \leftarrow \{\}, \mathcal{C}' \leftarrow \{\{v\} \mid v \in V\}$ 
2 while  $|\mathcal{C}'| > k$  do
3    $v \leftarrow \text{Get\_New}(S, \mathcal{C}')$ 
4    $\mathcal{C}', S \leftarrow \text{Insert}(V, \alpha, S, \mathcal{C}', \ell_{\text{yes}})$ 
5   if  $|S| = k + 1$  then
6      $S \leftarrow \{\}$ 
7   end
8 end
9 return  $\mathcal{C}'$ 
```

We briefly remark that for the k -unknown setting, Algorithm 9 proposed in the appendix, works in an analogous fashion. The key difference between the two settings is that rather than building cliques of size $k + 1$, Algorithm 9 builds the largest clique possible. We then give a charging argument that shows that this strategy does not lead to a significantly higher query complexity. The algorithm and analysis are presented in Section 2.10.

Algorithm 2: Get_New($S, \mathcal{C}'$)

Input: A partition $\mathcal{C}'$ of some finite set V , and a subset $S \subset V$ such that $|S| \leq |\mathcal{C}'|$

Output: An element $v \in V$ representing a set in $\mathcal{C}'$ which is currently not represented by any element of S .

```
1 for  $v \in V$  do
2   if  $[v]_{\mathcal{C}'} \cap S = \emptyset$  then
3     return  $v$ 
4   end
5 end
6 return  $\perp$ 
```

Algorithm 3: Insert($V, \alpha, \mathcal{C}', S, \ell_{\text{yes}}$)

Input: A finite set V , a $(\ell_{\text{yes}}, \ell_{\text{no}})$ -faulty oracle α for V , a partial clique S , a candidate partition $\mathcal{C}'$ of V , and error parameter ℓ_{yes} .

Output: A new candidate cluster $\mathcal{C}'$, partial clique S .

```
1 for  $u \in S$  do
2   // Returns either result = -1 or +1
3   result  $\leftarrow$  Compare( $u, v, \alpha, \ell_{\text{yes}}$ )
4   // If +1: it is guaranteed that  $u$  and  $v$  are part of the same cluster and  $\mathcal{C}'$  is
   // edited to reflect this
5   if result = +1 then
6     new_set  $\leftarrow [v]_{\mathcal{C}'} \cup [u]_{\mathcal{C}'}$ 
7      $\mathcal{C}' \leftarrow (\mathcal{C}' \setminus \{[v]_{\mathcal{C}'}, [u]_{\mathcal{C}'}\}) \cup \{\text{new\_set}\}$ 
8     return  $\mathcal{C}', S$ 
9   end
10 end
11 // If  $S$  is empty or if result = -1 for every  $u \in S$  then a new item is inserted into  $S$ 
12  $S \leftarrow S \cup \{v\}$ 
13 return  $\mathcal{C}', S$ 
```

Algorithm 4: $\text{Compare}(u, v, \alpha, \ell_{\text{yes}})$

Input: A pair of vertices u and v from some finite set V , a same-cluster oracle α for V , and error parameter ℓ_{yes} .

Output: $\text{result} \in \{\pm 1\}$.

```
1 // Initialize count to track "how many more times have we observed  $r = +1$ 
   compared to  $r = -1$ "
2 count  $\leftarrow 0$ 
3 while count  $\geq 0$  do
4    $r \leftarrow \alpha(u, v)$ 
5   if  $r = -1$  and count  $> 0$  then
6     count  $\leftarrow \text{count} - 1$ 
7   else if  $r = -1$  and count  $= 0$  then
8     /* If the number of  $-1$  responses exceeds the number of  $+1$  responses seen so
       far, we return  $-1$  */
9     return  $-1$ 
10  else
11    count  $\leftarrow \text{count} + 1$ 
12  end
13  // If count/ $\ell_{\text{yes}} > 1$  then we have verified that  $u \sim_{\mathcal{C}_*} v$  so return result  $= +1$ 
14  if count/ $\ell_{\text{yes}} > 1$  then
15    return 1
16  end
17 end
```

2.7 Additional relevant background

Comparison of random and adversarial error. Most of the prior work on clustering with faulty same-cluster queries focuses on the random-error model. Different models of random error have been considered. The most similar works to this chapter are [81] and [40]. These papers consider non-persistent random error. In this setting, the best known upper bounds on the query complexity of partition learning are

$$\tilde{O}\left(\frac{nk}{f(p)}\right),$$

where p is the probability of error and f is some function that tends to 0 as $p \rightarrow 1/2$ [40, 81]. It is difficult to compare the scaling of query complexity under random error as a function of probability of error versus the scaling of query complexity under adversarial error as a function of parameter ℓ . However one interesting qualitative distinction is that in this work the tight upper and lower bounds on query complexity under adversarial error never pay for a term as large as $n \cdot k \cdot \ell$. Instead in the setting of adversarial error, query complexity bounds scale with $(n + k^2) \cdot \ell$.

Comparison of persistent and non-persistent error. Prior literature has also studied the persistent random error version of this problem. The seminal work of [114] inspired a series of followup works in this setting [53, 78]. For this problem, upper and lower bounds which match up to log factors are known: [114] establish a lower bound stating that any algorithm achieving probability of recovery at least $3/4$ must make at least

$$\Omega\left(\frac{nk}{(1-2p)^2}\right)$$

queries. They further give a (time-inefficient) algorithm which recovers the maximum-likelihood clustering estimate with high probability when $p < 1/2$, using

$$O\left(\frac{nk \log(n)}{(1 - 2p)^2}\right)$$

queries [114]. In particular, while a priori the persistent-error versions of the problem can be no easier than the non-persistent-error ones, we are not aware of any result that shows a gap in query complexity between these two different settings.

We note that a model imposing persistent adversarial error does not allow for any non-trivial algorithmic results.

Active learning of graphs with oracles. The field of graph learning focuses on learning graph structure with oracle access to the graph in question.³ Numerous settings have been considered in the literature, including instances where the target graphs are matchings [9], stars or cliques [8], or Hamiltonian cycles [136]. Further research examines Las Vegas algorithms [2] and the role that adaptivity plays in the query complexity of these problems [36, 77]. In a related paper, [125] give an algorithm that learns the partitions of a graph with n vertices using $O(nk)$ shortest path queries, as well as an algorithm that learns such partitions using $O(n \log n)$ edge counting queries. Angluin and Chen [20] give an algorithm that can learn a graph with n vertices using $O(\log n)$ edge detection queries per edge. Following this line of inquiry, Liu and Mukherjee [99] find that the exact worst-case number of queries needed to learn a partition of $[n]$ into k clusters in the same-cluster query model is exactly $n(k - 1) - \binom{k}{2}$ if k is known by the learner a priori, and $nk - \binom{k+1}{2}$ otherwise.

Geometric clustering with oracle advice. Ashtiani, Kushagra and Ben-David [22] introduce the problem of clustering data when the learner can make limited same-cluster

3. In this chapter, we invoke the phrase “graph learning” in reference to problems which seek to exactly recover graph structure by actively making queries to some graph oracle, as opposed to the wide field of problems pertaining to learning graphs from data, which often focus on approximate or probabilistic results (i.e. such topics as surveyed by [153]).

queries, and they show that this leads to a polynomial-time algorithm for k-means clustering under certain geometric assumptions. In recent years, this model of Ashtiani, Kushagra and Ben-David has inspired a variety of other related works which aim to understand the benefits of accessing a same-cluster oracle (e.g., [31, 33, 53]). Ailon, Bhattacharya, and Jaiswal [6] show that making a small number of queries to a same-cluster oracle allows one to improve exponentially the dependency on the approximation parameter ε and on the number k of clusters for polynomial time approximation schemes for some versions of correlation clustering. Saha and Subramanian [128] also consider the problem of using same-cluster queries to aid the solution of a correlation clustering instance, and they give algorithms with performance guarantees that are functions of the optimal value of the input instance.

2.8 Equivalence between weighted problem and doubly constrained one

In this section, we argue that, up to changing the parameters by a factor of at most 2, the weighted ℓ -PL problem is equivalent to one in which the oracle may return up to ℓ_{yes} many false positive answers and ℓ_{no} many false negative ones. We refer to this problem as the asymmetric ℓ -PL problem.

Definition 2.8.1 (Asymmetric ℓ -PL Problem). *Given a finite set V , and a partition $\mathcal{C}$ of V , an $(\ell_{\text{yes}}, \ell_{\text{no}})$ -asymmetric oracle for $\mathcal{C}$ is an algorithm $\alpha_{\mathcal{C}}$ which, given as input a pair of elements uv , returns a value $r \in \{\pm 1\}$ so that for any sequence of queries $\{(u_t, v_t)\}_{t \in [T]}$, the sequence of responses $\{r_t = \alpha_{\mathcal{C}}(u_t v_t)\}_{t \in [T]}$ satisfies:*

$$|\{t \mid r_t = 1 \wedge u_t \not\sim_{\mathcal{C}} v_t\}| \leq \ell_{\text{yes}} \quad \text{and} \quad |\{t \mid r_t = -1 \wedge u_t \sim_{\mathcal{C}} v_t\}| \leq \ell_{\text{no}}.$$

The asymmetric ℓ -PL problem is the problem of recovering $\mathcal{C}$ by making queries to an $(\ell_{\text{yes}}, \ell_{\text{no}})$ -asymmetric oracle.

It is easy to see that any $(\ell_{\text{yes}}, \ell_{\text{no}})$ -faulty oracle is also an $(\ell_{\text{yes}}, \ell_{\text{no}})$ -asymmetric oracle and hence the query complexity of the asymmetric ℓ -PL problem is no larger than that of the weighted ℓ -PL problem for the same choice of parameters. Conversely, we have the following:

Proposition 2.8.2. *For any algorithm which solves the weighted ℓ -PL problem by making at most $q(n, k, \ell_{\text{yes}}, \ell_{\text{no}})$ queries, there exists an algorithm which solves the asymmetric ℓ -PL problem by making at most $q(n, k, 2\ell_{\text{yes}}, 2\ell_{\text{no}})$ queries.*

Proof. This simply follows from the fact that any $(\ell_{\text{yes}}, \ell_{\text{no}})$ -asymmetric oracle is also an $(2\ell_{\text{yes}}, 2\ell_{\text{no}})$ -faulty oracle. $\square$

2.9 Deferred proofs from Section 2.6

Query Accounting. For the sake of analysis, we will assume that the algorithm maintains a global history of all the queries made to the oracle and the responses received. We will also let the algorithm assign labels to query-response pairs. This is done in order to charge the queries made to some measure of progress towards either learning the partitions or certifying that errors have occurred. The lines of pseudocode assigning labels to the query history, written in **blue**, are not necessary for the algorithm to work correctly but rather are included only to facilitate the analysis. Note also that Algorithm 5 is the same as Algorithm 1 for **Learn**. Similarly, Algorithm 6 is the same as Algorithm 2 for **Get_New**.

Note that Algorithm 3 and Algorithm 7 differ only in line 7.

Note that Algorithm 4 and Algorithm 8 differ only in lines 9, 10, 17 and 18, i.e., where labels occur.

Lemma 2.9.1. *Throughout the execution of Algorithm 5, $\mathcal{C}'$ is always a refinement of the true partition.*

Proof of Lemma 2.9.1. $\mathcal{C}'$ is essentially a global variable. We begin by noting that $\mathcal{C}'$ is trivially a refinement of the correct partition at the beginning of the algorithm. The only

Algorithm 5: Learn($V, \alpha, k, \ell_{\text{yes}}$)

Input: A finite set V , an $(\ell_{\text{yes}}, \ell_{\text{no}})$ -faulty oracle α , a target number of clusters k , and error parameters ℓ_{yes} .

Output: A partition $\mathcal{C}'$ of V .

```
1  $S \leftarrow \{\}$ ,  $\mathcal{C}' \leftarrow \{\{v\} \mid v \in V\}$ 
2 while  $|\mathcal{C}'| > k$  do
3    $v \leftarrow \text{Get\_New}(S, \mathcal{C}')$ 
4    $\mathcal{C}', S \leftarrow \text{Insert}(V, \alpha, S, \mathcal{C}', \ell_{\text{yes}}, \ell_{\text{no}})$ 
5   if  $|S| = k + 1$  then
6      $S \leftarrow \{\}$ 
7   end
8 end
9 return  $\mathcal{C}'$ 
```

Algorithm 6: Get_New($S, \mathcal{C}'$)

Input: A partition $\mathcal{C}'$ of some finite set V , and a subset $S \subset V$ such that $|S| \leq |\mathcal{C}'|$

Output: An element $v \in V$ representing a set in $\mathcal{C}'$ which is currently not represented by any element of S .

```
1 for  $v \in V$  do
2   if  $[v]_{\mathcal{C}'} \cap S = \emptyset$  then
3     return  $v$ 
4   end
5 end
6 return  $\perp$ 
```

Algorithm 7: $\text{Insert}(V, \alpha, \mathcal{C}', S, \ell_{\text{yes}})$

Input: A finite set V , an $(\ell_{\text{yes}}, \ell_{\text{no}})$ -faulty oracle α for V , a partial clique S , a candidate partition $\mathcal{C}'$ of V , and error parameters ℓ_{yes} .

Output: A new candidate cluster $\mathcal{C}'$, partial clique S .

```
1 for  $u \in S$  do
2   // Returns either result = -1 or +1
3   result  $\leftarrow \text{Compare}(u, v, \alpha, \ell_{\text{yes}}, \ell_{\text{no}})$ 
4   // If +1: it is guaranteed that  $u$  and  $v$  are part of the same cluster and  $\mathcal{C}'$  is
   // edited to reflect this
5   if result = +1 then
6     Re-label all ‘clique’ queries made during this execution of Insert (and all
     // calls to Compare therein) as ‘merge-’
7     new_set  $\leftarrow [v]_{\mathcal{C}'} \cup [u]_{\mathcal{C}'}$ 
8      $\mathcal{C}' \leftarrow (\mathcal{C}' \setminus \{[v]_{\mathcal{C}'}, [u]_{\mathcal{C}'}\}) \cup \{\text{new\_set}\}$ 
9     return  $\mathcal{C}', S$ 
10  end
11 end
12 // If  $S$  is empty or if result = -1 for every  $u \in S$  then a new item is inserted into  $S$ 
13  $S \leftarrow S \cup \{v\}$ 
14 return  $\mathcal{C}', S$ 
```

point in the algorithm at which $\mathcal{C}'$ is modified is on Lines 7 and 8 of **Insert**. In that step, the algorithm modifies $\mathcal{C}'$ by merging the cluster containing v with the cluster containing u . These lines are executed only if the value of the variable **result**, output by **Compare**, is +1. This happens only if $\text{count}/\ell_{\text{yes}} > 1$ on Line 16 of **Compare**, in which case the latest call to **Compare** has obtained more than ℓ_{yes} many uv queries returning +1.

We now argue that if that happens, u and v must have been part of the same cluster in the ground-truth partition $\mathcal{C}$. If u and v were not part of the same of the same cluster in $\mathcal{C}$, we derive a contradiction, as this would imply:

$$\text{FP} \geq \text{count} > \ell_{\text{yes}}$$

and hence $\text{FP}/\ell_{\text{yes}} > 1$. Hence, if $\mathcal{C}'$ was a refinement of $\mathcal{C}$ before merging the clusters containing u and v , then it is still a refinement of $\mathcal{C}$ after merging the two clusters. This invariant is then maintained throughout the algorithm, and the lemma holds. $\square$

Algorithm 8: $\text{Compare}(u, v, \alpha, \ell_{\text{yes}})$

Input: A pair of vertices u and v from some finite set V , a same-cluster oracle α for V , and error parameters ℓ_{yes} .

Output: $\text{result} \in \{\pm 1\}$.

```
1 // Initialize count to track “how many more times have we observed  $r = +1$ 
   compared to  $r = -1$ ”
2 count  $\leftarrow 0$ 
3 while count  $\geq 0$  do
4    $r \leftarrow \alpha(u, v)$ 
5   if  $r = -1$  and count  $> 0$  then
6     count  $\leftarrow \text{count} - 1$ 
7   else if  $r = -1$  and count  $= 0$  then
8     /* If the number of  $-1$  responses exceeds the number of  $+1$  responses seen so
       far, we return  $-1$  */
9     Label the last query as ‘clique’
10    Label all other queries created on this execution of Compare as ‘spurious’
11    return  $-1$ 
12  else
13    count  $\leftarrow \text{count} + 1$ 
14  end
15 // If count/ $\ell_{\text{yes}} > 1$  then we have verified that  $u \sim_{\mathcal{C}_*} v$  so return result  $= +1$ 
16 if count/ $\ell_{\text{yes}} > 1$  then
17   Label the last count-many queries which received positive responses as
     ‘merge+’
18   Label all other queries made during this execution of Compare as ‘spurious’
19   return 1
20 end
21 end
```

Lemma 2.9.2. *At the end of Algorithm 5, the history of queries contains at most $(\ell_{\text{yes}} + 1)(n - k)$ queries labelled ‘merge+’ and at most $k(n - k)$ queries labelled ‘merge-’.*

Proof of Lemma 2.9.2. Consider a point in the algorithm in which **Compare** returns $+1$. When that happens, the **if** block starting on Line 5 of **Insert** is executed; we refer to this as a *merge event*.

We begin by arguing that at most $n - k$ merge events can occur during an execution of Algorithm 5. At the start of the algorithm we have $|\mathcal{C}'| = n$. At each merge event two clusters in $\mathcal{C}'$ are merged and the cardinality of $\mathcal{C}'$ decreases by 1. Since $\mathcal{C}'$ is always a refinement of the hidden k -partition $\mathcal{C}$ (by Lemma 2.9.1), this implies that at most $n - k$ merge events can occur during the execution of the algorithm.

Both ‘merge-’ and ‘merge+’ queries are only created in correspondence with a merge event. We now show that on any merge event, at most k ‘merge-’ queries and at most $\ell_{\text{yes}} + 1$ ‘merge+’ queries are created. By the above argument these results imply the statement of the lemma.

We start by showing that at most k ‘merge-’ queries are created during each merge event. ‘merge-’ queries are created when some ‘clique’ queries are re-labelled as ‘merge-’. Specifically, all the ‘clique’ queries created during every execution of **Compare** within the current execution of **Insert** are relabelled as ‘merge-’. On any execution of **Insert**, at most one ‘clique’ query is created for every element in the current set S . During any iteration of the **for** loop at **Insert** Line 1, the set S is of size at most k , implying that at most k ‘clique’ queries are created during one execution of **Insert**. As a result at most k ‘merge-’ queries are created at every merge event.

Similarly, ‘merge+’ queries are only created in correspondence with a merge event. Specifically, merge events occur when **Compare** returns **result** = $+1$. When this happens, ‘merge+’ queries are created at **Compare** Line 17. When Line 17 is executed, the number of queries labelled ‘merge+’ is equal to the current value of **count**, which is at most $\ell_{\text{yes}} + 1$. Therefore, at most $\ell_{\text{yes}} + 1$ ‘merge+’ queries are created in correspondence with each merge

event, concluding the proof. $\square$

Lemma 2.9.3. *At the end of the algorithm, the history of queries contains at most $2 \max\{\ell_{\text{yes}}, \ell_{\text{no}}\}$ queries labelled ‘spurious’.*

Proof of Lemma 2.9.3. Each query uv labelled ‘spurious’ can be paired uniquely with another query uv labelled **spurious** which returned the opposite answer. Hence, the number of erroneous answers returned by the oracle is at least the number of **spurious** queries divided by two. This number is upper bounded by $\max\{\ell_{\text{yes}}, \ell_{\text{no}}\}$ and the result follows. $\square$

Lemma 2.9.4. *At the end of the algorithm, the history of queries contains at most $(\ell_{\text{no}} + 1) \binom{k+1}{2}$ queries labelled ‘clique’.*

Proof of Lemma 2.9.4. The algorithm only labels queries as ‘clique’ in Line 9 of **Compare**. Thus every ‘clique’ query is made between two elements u and v such that both u and v were inserted into some construction of a clique, supported on the set S .

If S reaches cardinality $k + 1$, the **if** block on Line 5 of **Learn** is executed. We refer to this occurrence as a *clique event*. When a clique event occurs, queries have made up new a clique of -1 responses of size $k + 1$. In order for all of these responses to have been correct, the ground-truth partition would need to have been size at least $k + 1$, hence every time a clique event happens, a false negative error must have occurred. Since every clique event can be charged to a false negative error, at most ℓ_{no} clique events can occur.

We will charge every ‘clique’ query made before a clique event to the first clique event following the query’s creation. These queries represent a distinct edge uv of a clique built on S which eventually reaches size $k + 1$, and hence exactly $\binom{k+1}{2}$ many of these ‘clique’ queries will occur per clique event. Finally, all ‘clique’ queries occurring after the last clique event make up the edges of a clique supported on the set S , which never reaches size $k + 1$, and hence there are less than $\binom{k+1}{2}$ such queries.

The result then follows. $\square$

Lemma 2.9.5. *If Algorithm 5 makes a finite number of queries, then it must terminate.*

Proof of Lemma 2.9.5. We show that if Algorithm 5 makes a finite number of queries, then Line 4 of Algorithm 5 is executed a finite number of times. This implies termination of Algorithm 5.

When Line 4 of Algorithm 5 executes, Algorithm 5 calls Algorithm 7, **Insert**. If S is nonempty on the execution of Line 4, then **Insert** calls **Compare** on some pair of elements. On every call to **Compare**, Line 4 of **Compare** executes at least once, and hence the algorithm makes at least one query. If S is empty on the execution of Line 4, then **Insert** increases the size of S by 1. This implies that on the next execution of Line 4, S will be nonempty. Thus in order for Algorithm 5 to execute Line 4 infinitely many times, the algorithm must make infinitely many queries, yielding the result. $\square$

We now analyze the number of queries made by the algorithm. We begin by noting that when the algorithm terminates, every query made will hold exactly one of the following labels: ‘merge-’, ‘merge+’, ‘clique’, ‘spurious’. The analysis proceeds by showing how labels of each category contribute to either (a) correctly identifying the true partition, or (b) correctly certifying the occurrence of errors.

Equipped with these lemmata, we now prove Theorem 2.4.1 on the existence of an algorithm for the weighted ℓ -PL problem in the k -known setting.

Proof of Theorem 2.4.1. Let $|\text{‘merge-’}|, |\text{‘merge+’}|, |\text{‘spurious’}|, |\text{‘clique’}|$ be the number of ‘merge-’, ‘merge+’, ‘spurious’ and ‘clique’ queries made by the algorithm respectively. By Lemmata 2.9.2, 2.9.3, and 2.9.4, the algorithm makes at most

$$\begin{aligned} & |\text{‘merge-’}| + |\text{‘merge+’}| + |\text{‘spurious’}| + |\text{‘clique’}| \\ & \leq nk - k^2 + (\ell_{\text{yes}} + 1)(n - k) + 2 \max\{\ell_{\text{yes}}, \ell_{\text{no}}\} + (\ell_{\text{no}} + 1) \binom{k+1}{2} \\ & = O\left(\text{RS}^k(n, k) + (n - k)\ell_{\text{yes}} + k^2\ell_{\text{no}}\right) \end{aligned}$$

many queries. Note that we used that $\text{RS}^k(n, k) = n(k-1) - \binom{k}{2}$ from [125]. By Lemma 2.9.5

this implies that the algorithm must terminate.

When the algorithm terminates, the condition in the while loop of **Learn** is not met, and hence $|\mathcal{C}'| \leq k$. But since, by Lemma 2.9.1 $\mathcal{C}'$ is a refinement of $\mathcal{C}$, it must hold that upon termination $|\mathcal{C}'| = k$ and hence $\mathcal{C}' = \mathcal{C}$. The algorithm then returns the correct hidden partition $\mathcal{C}$. $\square$

2.10 The algorithm for k -unknown

In this section we describe an algorithm for the weighted ℓ -PL problem in the k -unknown setting, and further prove that it achieves optimal asymptotic performance for the unweighted version of the problem. A complete description of the algorithm is given in the pseudocode below. Note that the algorithm makes use of subroutines defined in Section 2.10. The main challenge in adapting Algorithm 5 in Section 2.6 is that the algorithm crucially relies on knowing the value of k to build a lower bound to the number of observed errors, by counting the number of $(k + 1)$ -cliques of -1 answers observed.

To overcome this problem, the new algorithm will instead simply aim to build the largest possible clique of -1 answers. We show that, remarkably, when false positive and false negative errors are equivalent ($\ell_{\text{yes}} = \ell_{\text{no}} = \ell$), this suffices to match the optimal guarantees of Algorithm 5, up to small constant factors.

Algorithm 9: Learn_k_unknown($V, \alpha, \ell_{\text{yes}}, \ell_{\text{no}}$)

Input: A finite set V , an $(\ell_{\text{yes}}, \ell_{\text{no}})$ -faulty same-cluster oracle α , error parameters

ℓ_{yes} and ℓ_{no} .

Output: A partition $\mathcal{C}'$ of V .

```
1  $\mathcal{C}' \leftarrow \{\{v\} \mid v \in V\}, \text{idx} \leftarrow 1$ 
2  $S_{\text{idx}} \leftarrow \{\}$ 
3 while  $\text{idx} \leq \ell_{\text{no}} + 1$  do
4    $v \leftarrow \text{Get\_New}(S, \mathcal{C}')$ 
5   if  $v = \perp$  then
6      $\text{idx} + = 1$ 
7      $S_{\text{idx}} \leftarrow \{\}$ 
8   end
9   else
10     $\mathcal{C}', S_{\text{idx}} \leftarrow \text{Insert}(V, \alpha, S_{\text{idx}}, \mathcal{C}', \ell_{\text{yes}})$ 
11  end
12 end
13 return  $\mathcal{C}'$ 
```

The intuition behind the success of Algorithm 9 is that, for the algorithm to construct large cliques (significantly larger than the cardinality k of the hidden partition), the oracle must return many incorrect answers. We formalize this intuition in Lemma 2.10.1.

Consider the execution of Algorithm 9. The algorithm will repeatedly construct cliques of -1 queries supported on vertex sets S_{idx} , adding elements to S_{idx} until all elements of V belong to an equivalence class $[u]_{\mathcal{C}'}$ for some $u \in S_{\text{idx}}$. Once this condition is met, the algorithm increments idx and initializes the next S_{idx} as the empty set. The following lemma shows that the maximum size of any such set S_{idx} constructed by the algorithm is bounded as a function of the true unknown partition size k and the error parameter ℓ_{no} .

Consider the set S_i constructed by Algorithm 9 while $\text{idx} = i$. We will denote by n_i the maximum cardinality attained by this set S_i . For convention we will also set $n_0 = n$. Moreover, we let ℓ_i denote the number of false negative responses returned while $\text{idx} = i$.

Lemma 2.10.1. *For any $i \geq 1$, we have:*

$$n_i \leq k + \sqrt{2k\ell_i}. \quad (2.10.1)$$

Proof. Note that, if the hidden partition has cardinality k , and the algorithm obtains a clique of -1 responses on n_i vertices, then the number of false negative errors contained in the clique responses is at least the minimum number of within-cluster edges of any k -partition of the complete graph K_{n_i} . I.e., a lower bound to the number of false negative errors that must have occurred is given by:

$$\begin{aligned} \ell_i &\geq \min \left\{ \sum_{j=1}^k \binom{c_j}{2} \mid c_1, \dots, c_k \in \mathbb{N}, \sum_{j=1}^k c_j = n_i \right\} \\ &\geq \min \left\{ \sum_{j=1}^k \frac{c_j(c_j - 1)}{2} \mid c_1, \dots, c_k \in \mathbb{R}_{>0}, \sum_{j=1}^k c_j = n_i \right\}. \end{aligned}$$

Note that the function $f(x) = x(x-1)/2$ is convex, and hence by Karamata's Inequality (Theorem 1 from [92]) the right-hand side above is lower bounded by:

$$\ell_i \geq \sum_{j=1}^k \frac{\frac{n_i}{k} \left(\frac{n_i}{k} - 1 \right)}{2} = \frac{1}{2} n_i \left(\frac{n_i}{k} - 1 \right).$$

Re-arranging, we find that $n_i^2 - kn_i \leq 2\ell_i k$. Completing the square we obtain:

$$\left(n_i - \frac{1}{2}k \right)^2 \leq 2\ell_i k + \frac{1}{4}k^2$$

giving:

$$n_i - \frac{1}{2}k \leq \sqrt{2\ell_i k + \frac{1}{4}k^2} \leq \sqrt{2\ell_i k} + \frac{1}{2}k,$$

and hence:

$$n_i \leq \sqrt{2\ell_i k} + k,$$

as needed. □

Crucially, we note that as with Algorithm 5, the partition $\mathcal{C}'$ maintained by Algorithm 9 is always a refinement of the ground-truth.

Lemma 2.10.2. *Throughout the execution of Algorithm 9, $\mathcal{C}'$ is always a refinement of the true partition.*

The proof of this lemma is identical to the proof of Lemma 2.9.1.

Lemma 2.10.3. *Algorithm 9 terminates.*

Proof. We show that Lines 6 and 10 in Algorithm 9 execute finitely many times. This implies termination of the algorithm.

Every time Line 6 executes, the value of `idx` is incremented by 1. Once the value of `idx` exceeds $\ell_{\text{no}} + 1$, the **while** loop in Algorithm 9 exits, and thus Line 6 can only execute finitely many times.

To show that Line 10 executes finitely many times, we show that every time Line 10 executes, either $\mathcal{C}'$ decreases in cardinality or S_{idx} increases in cardinality. This follows from considering Algorithm 7, **Insert**. On each call to **Insert**, either `result` = +1 for some $u \in S$, in which case the cardinality of $\mathcal{C}'$ is reduced in Line 8, or `result` = -1 for every $u \in S$. In the latter case, the size of S is increased in Line 13.

By Lemma 2.10.2, $\mathcal{C}'$ is always a refinement of $\mathcal{C}$ and thus its cardinality cannot decrease more than $(n - k)$ times. Similarly, for each value `idx` = i , $|S_i| \leq n$ so S_i can increase in size finitely many times for each value of `idx` $\in [1, \ell_{\text{no}} + 1]$. This implies that Line 10 can only execute finitely many times, completing the proof. □

Lemma 2.10.4. *When Algorithm 9 terminates $\mathcal{C}'$ is equal to the ground-truth partition $\mathcal{C}$.*

Proof. By Lemma 2.10.2 $\mathcal{C}'$ is always a refinement of $\mathcal{C}$. Therefore to show that the partition $\mathcal{C}'$ returned is equal to $\mathcal{C}$, it remains to show that the cardinality of $\mathcal{C}'$ upon termination is the true unknown value k .

Assume for the sake of contradiction that this does not hold. Once initialized, the only place where $\mathcal{C}'$ is modified during the execution of Algorithm 9 is in Lines 7 and 8 of **Insert**. There, the algorithm modifies $\mathcal{C}'$ by merging two previously disjoint equivalence classes. This reduces the number of disjoint equivalence classes in $\mathcal{C}'$, and so in particular the cardinality of $\mathcal{C}'$ monotonically decreases during the execution of Algorithm 9. Thus if upon termination the cardinality of $\mathcal{C}'$ is strictly greater than k , then this was true at all points during the algorithm's execution.

Recall that n_i is defined to be the maximum cardinality attained by set S_i . In particular, S_i attains maximum size when Line 6 of Algorithm 9 is executed. When this happens, S_i contains exactly one representative from every cluster in $\mathcal{C}'$ at the time of execution. If the size of $\mathcal{C}'$ is strictly greater than k at all points during the execution of Algorithm 9 then $n_i > k \ \forall i \in [0, \ell_{\text{no}} + 1] \cap \mathbb{N}$.

In particular for every value $i \in [1, \ell_{\text{no}} + 1] \cap \mathbb{N}$, for every distinct $u, v \in S_i$ Algorithm 9 receives at least one response -1 for query (u, v) . This response is consistent with the scenario in which u and v belong to disjoint sets in the ground truth partition $\mathcal{C}$. If $n_i > |\mathcal{C}|$ then at least one of these responses must have been a false negative error. As this holds for every value $i \in [1, \ell_{\text{no}} + 1] \cap \mathbb{N}$, this implies least $\ell_{\text{no}} + 1$ false negative responses occur during the execution of Algorithm 9. In particular this violates the assumption that α is an $(\ell_{\text{yes}}, \ell_{\text{no}})$ -faulty oracle, and we thus conclude that the size of $\mathcal{C}'$ cannot be greater than k upon termination, implying that the partition returned by the algorithm satisfies $\mathcal{C}' = \mathcal{C}$. $\square$

As with the analysis of Algorithm 5, bounding the query complexity of Algorithm 9 reduces to bounding the number of queries with labels ‘merge-’, ‘merge+’, ‘clique’, ‘spurious’ upon termination of the algorithm.

Lemma 2.10.5. *When Algorithm 9 terminates the history of queries contains at most $(1 + \sqrt{2})(\ell_{\text{no}} + 1)k^2$ queries labelled ‘clique’.*

Proof. All queries labeled ‘clique’ upon termination must have been created during the construction of some set S_i while $\text{idx} = i$. Given n_i the maximum cardinality attained by set S_i , the number of ‘clique’ queries present upon termination that were constructed while $\text{idx} = i$ is exactly

$$\binom{n_i}{2} = \frac{1}{2}(n_i^2 - n_i) \leq \frac{k^2 + 2k\sqrt{2k\ell_i} + 2k\ell_i - n_i}{2},$$

where the upper bound follows from Lemma 2.10.1 and ℓ_i denotes the number of false negative responses returned by the oracle while $\text{idx} = i$. Summing the total number of such queries over all iterates $\text{idx} = 1, \dots, \ell_{\text{no}} + 1$, we obtain:

$$\begin{aligned} \sum_{i=1}^{\ell_{\text{no}}+1} \binom{n_i}{2} &\leq \frac{1}{2} \sum_{i=1}^{\ell_{\text{no}}+1} \left(k^2 + 2k\sqrt{2k\ell_i} + 2k\ell_i - n_i \right) \\ &= \frac{1}{2}(\ell_{\text{no}} + 1)k^2 + \sqrt{2} \sum_{i=1}^{\ell_{\text{no}}+1} k\sqrt{k\ell_i} + k \sum_{i=1}^{\ell_{\text{no}}+1} \ell_i - \frac{1}{2} \sum_{i=1}^{\ell_{\text{no}}+1} n_i \\ &\leq \frac{1}{2}(\ell_{\text{no}} + 1)k^2 + \sqrt{2}k \sum_{i=1}^{\ell_{\text{no}}+1} \frac{\ell_i + k}{2} + k \sum_{i=1}^{\ell_{\text{no}}+1} \ell_i - \frac{1}{2} \sum_{i=1}^{\ell_{\text{no}}+1} n_i \\ &\leq \frac{1}{2}(\ell_{\text{no}} + 1)k^2 + \frac{k\ell_{\text{no}}}{\sqrt{2}} + \frac{(\ell_{\text{no}} + 1)k^2}{\sqrt{2}} + k\ell_{\text{no}} - \frac{1}{2}k(\ell_{\text{no}} + 1). \end{aligned}$$

The last line follows by observing that by definition of ℓ_i , $\sum_{i=1}^{\ell_{\text{no}}+1} \ell_i \leq \ell_{\text{no}}$.

We can simplify the above expression using the fact that for integer k it holds that $k \leq k^2$, to obtain the bound:

$$\begin{aligned} &\frac{1}{2}(\ell_{\text{no}} + 1)k^2 + \frac{k\ell_{\text{no}}}{\sqrt{2}} + \frac{(\ell_{\text{no}} + 1)k^2}{\sqrt{2}} + k\ell_{\text{no}} - \frac{1}{2}k(\ell_{\text{no}} + 1) \\ &= \left(\frac{1}{2} + \frac{1}{\sqrt{2}} \right) \left((\ell_{\text{no}} + 1)k^2 + k\ell_{\text{no}} \right) - \frac{k}{2} \\ &\leq (1 + \sqrt{2})(\ell_{\text{no}} + 1)k^2. \end{aligned}$$

□

Lemma 2.10.6. *Upon termination of Algorithm 9, the history of queries contains at most $2 \max\{\ell_{\text{yes}}, \ell_{\text{no}}\}$ queries labelled ‘spurious’.*

The proof follows analogously to that of Lemma 2.9.3.

Lemma 2.10.7. *Upon termination of Algorithm 9, the history of queries contains at most $(\ell_{\text{yes}} + 1)(n - k)$ queries labelled ‘merge+’, and at most*

$$(n - k) \max \left\{ (1 + \sqrt{2})k, k + \ell_{\text{no}}\sqrt{2} \right\}$$

queries labelled ‘merge-’.

Proof. Similarly to the analysis of Algorithm 5, we analyze the numbers of ‘merge+’ and ‘merge-’ queries, by considering *merge events*. A merge event is defined as the execution of Line 5 in **Insert**.

Recall that n_i is define to be the maximum cardinality attained by set S_i . In particular, S_i attains maximum size when Line 6 of Algorithm 9 is executed. When this happens, S_i contains exactly one representative from every cluster in $\mathcal{C}'$ at the time of execution. Thus $n_i - n_{i+1}$ is equal to the number of clusters merged while $\text{idx} = (i + 1)$, i.e. the number of merge events that occurred while $\text{idx} = (i + 1)$. Moreover the size of $\mathcal{C}'$ upon termination of Algorithm 9 is equal to $n_{\ell_{\text{no}}+1}$, and hence by Lemma 2.10.4 we have $n_{\ell_{\text{no}}+1} = k$. This implies

$$\sum_{i=0}^{\ell_{\text{no}}} (n_i - n_{i+1}) = n_0 - n_{\ell_{\text{no}}+1} = n - k.$$

As with the proof of Lemma 2.9.2, we note that each ‘merge+’ query can be charged to a unique merge event in such a way that exactly $\ell_{\text{yes}} + 1$ ‘merge+’ queries are charged to each merge event. As a result, the number $|\text{‘merge+’}|$ of ‘merge+’ queries satisfies:

$$|\text{‘merge+’}| = \sum_{i=0}^{\ell_{\text{no}}} (\ell_{\text{yes}} + 1)(n_i - n_{i+1}) = (\ell + 1)(n - k)$$

Similarly, ‘merge-’ queries are only created in correspondence with a merge event. Specifically, ‘merge-’ are created when ‘clique’ queries are re-labelled as ‘merge-’. When $\text{idx} = (i + 1)$ on any single execution of **Insert**, at most one ‘clique’ query is created for every element in the current set $S_{(i+1)}$. Hence at most n_{i+1} ‘merge-’ queries correspond to each merge event that occurs while $\text{idx} = (i + 1)$. Summing over all values attained by idx , this implies the total number of queries labeled ‘merge-’ is bounded by

$$\begin{aligned}
|\text{‘merge-’}| &\leq \sum_{i=0}^{\ell_{\text{no}}} n_{i+1}(n_i - n_{i+1}) \\
&\leq \sum_{i=0}^{\ell_{\text{no}}} (k + \sqrt{2k\ell_{i+1}})(n_i - n_{i+1}) \\
&= k \sum_{i=0}^{\ell_{\text{no}}} (n_i - n_{i+1}) + \sqrt{2k} \sum_{i=0}^{\ell_{\text{no}}} (n_i - n_{i+1}) \sqrt{\ell_{i+1}},
\end{aligned}$$

where the second inequality follows from the bound in Lemma 2.10.1. Invoking the fact that $\sum_{i=0}^{\ell_{\text{no}}} (n_i - n_{i+1}) = n - k$, we can bound the above and obtain:

$$\begin{aligned}
|\text{‘merge-’}| &\leq k \cdot (n - k) + \sqrt{2k} \cdot \max_i \sqrt{\ell_{i+1}} \cdot \sum_{i=0}^{\ell_{\text{no}}} (n_i - n_{i+1}) \\
&\leq \left(k + \sqrt{2k\ell_{\text{no}}} \right) (n - k) \\
&\leq (n - k) \cdot \max \left\{ (1 + \sqrt{2})k, k + \ell_{\text{no}}\sqrt{2} \right\},
\end{aligned}$$

yielding the result. □

Equipped with the above lemmata, we proceed to prove Theorem 2.4.3.

Proof of Theorem 2.4.3. By Lemmata 2.10.3 and 2.10.4, Algorithm 9 terminates and returns the correct hidden partition. It thus remains to bound the number of queries made during execution.

Let $|\text{‘merge-’}|, |\text{‘merge+’}|, |\text{‘spurious’}|, |\text{‘clique’}|$ be the number of ‘merge-’, ‘merge+’,

‘spurious’ and ‘clique’ queries made by the algorithm respectively. By Lemmata 2.10.5, 2.10.6, and 2.10.7, the algorithm makes at most

$$\begin{aligned}
& |\text{‘merge-’}| + |\text{‘merge+’}| + |\text{‘spurious’}| + |\text{‘clique’}| \\
& \leq (n - k) \max \left\{ (1 + \sqrt{2})k, k + \ell_{\text{no}}\sqrt{2} \right\} + (\ell_{\text{yes}} + 1)(n - k) \\
& \quad + 2 \max \{ \ell_{\text{yes}}, \ell_{\text{no}} \} + (1 + \sqrt{2})(\ell_{\text{no}} + 1)k^2 \\
& = O \left(\text{RS}^u(n, k) + (n - k) \max \{ \ell_{\text{yes}}, \ell_{\text{no}} \} + k^2 \ell_{\text{no}} \right)
\end{aligned}$$

many queries. Note that we used that $\text{RS}^u(n, k) = nk - \binom{k+1}{2}$ from [125]. $\square$

2.11 Lower bound results

2.11.1 Lower bounds for the easy problem: ℓ -false negatives

We prove the following lower bound on the query complexity of the $\ell\text{-PL}^{\text{FN}}$ problem.

Theorem 2.11.1. *Every algorithm for the $\ell\text{-PL}^{\text{FN}}$ problem which guarantees full recovery of the correct partitions requires $\text{RS}^k(n, k) + \Omega(k^2\ell)$ queries in the worst case.*

Because the $\ell\text{-PL}^{\text{FN}}$ problem is no harder than the $\ell\text{-PL}$ problem, this immediately implies the same lower bound on the query complexity of the $\ell\text{-PL}$ problem.

To establish this result, we introduce and analyze a game between a Questionner and a Responder, representing the algorithm and the oracle respectively.

Definition of the RUCC^{FN} game. We define the following game, which we call the RUCC^{FN} game. The game is played by two players, the questioner (Q) and the responder (R) and it is parametrized by a finite set V of cardinality n , a positive integer $k \in \{2, \dots, n - 1\}$, and a non-negative integer ℓ . The goal of the questioner is to infer a k -partition of V given input from the responder. At the start of the game, we are given a complete undirected

graph G_0 on vertex set V . $G_0 = (V, E, w_0^+, w_0^-)$ has two associated edge weight functions $w_0^+ : E \rightarrow \mathbb{R}$ and $w_0^- : E \rightarrow \mathbb{R}$, both of which are set to zero at the beginning of the game.

At each iteration t , the questioner Q submits a query uv for distinct u and v in V . The responder R then issues a response $r_t \in \{\pm 1\}$. If $r_t = +1$, then w^+ is updated by setting $w_t^+(uv) = w_{t-1}^+(uv) + 1$. Otherwise, if $r_t = -1$, then w^- is updated by setting $w_t^-(uv) = w_{t-1}^-(uv) + 1$. We then set $G_t = (V, E, w_t^+, w_t^-)$.

For any partition $\mathcal{C}$ of V , we define the cost:

$$\text{cost}_{G_t}^{\text{FN}}(\mathcal{C}) \stackrel{\text{def}}{=} \sum_{\substack{uv \in E \\ u \sim_{\mathcal{C}} v}} w_t^-(uv) + \mathbb{I} \left(\sum_{\substack{uv \in E \\ u \not\sim_{\mathcal{C}} v}} w_t^+(uv) = 0 \right),$$

where $\mathbb{I}$ is the convex indicator function, that gives the value 1 when the partition satisfies the condition within the parentheses, and ∞ otherwise. Intuitively, the value of $\text{cost}_{G_t}^{\text{FN}}(\mathcal{C})$ is the number of incorrect negative (-1) answers that the responder would have to have given if $\mathcal{C}$ was the correct partition, or infinity if the responder would have had to given any incorrect positive ($+1$) answer.

We require that the responder must, at each iteration t , reply in a way that guarantees that there exists some k -partition $\mathcal{C}$ of V such that:

$$\text{cost}_{G_t}^{\text{FN}}(\mathcal{C}) \leq \ell.$$

The game terminates when $\text{cost}_{G_t}^{\text{FN}}(\mathcal{C})$ is strictly more than ℓ for all but one k -partition $\mathcal{C}$ of V , indicating that the responder is left with a single feasible partition and hence they have learned the ground truth. If the game terminates at iteration T , then the payoff to the responder player is T , and the payoff to the questioner is $-T$.

Given a questioner strategy Q and a responder strategy R , we let $\text{Game}^{\text{FN}}(Q, R)$ be the number of iterations the game lasts when questioner Q plays against responder R , as a function of the parameters n , k and ℓ . The query complexity of the ℓ -PL^{FN} problem is the

value:

$$\min_{Q \in \mathcal{Q}} \max_{R \in \mathcal{R}} \text{Game}^{\text{FN}}(Q, R),$$

where $\mathcal{Q}$ is the set of all possible questioner strategies and $\mathcal{R}$ is the set of all possible responder strategies. A simple argument then shows:

Lemma 2.11.2. *Let f be a non-negative real valued function. If there exists a responder strategy R^* such that:*

$$\forall n, k, \ell \in \mathbb{N}_0 : \quad \min_{Q \in \mathcal{Q}} \text{Game}^{\text{FN}}(Q, R^*) \geq f(n, k, \ell),$$

then the query complexity of the ℓ -PL^{FN} problem is at least $f(n, k, \ell)$.

Proof. This follows from:

$$\min_{Q \in \mathcal{Q}} \max_{R \in \mathcal{R}} \text{Game}^{\text{FN}}(Q, R) \geq \min_{Q \in \mathcal{Q}} \text{Game}^{\text{FN}}(Q, R^*) \geq f(n, k, \ell). \quad \square$$

In Theorem 2.12.1 in Section 2.12.1 we show that there exists a responder strategy R^* satisfying

$$\min_{Q \in \mathcal{Q}} \text{Game}^{\text{FN}}(Q, R^*) \geq (\ell + 1) \left(\binom{k+1}{2} - 1 \right) = \Omega(\ell k^2)$$

queries. Lemma 2.11.2 and Theorem 2.12.1, together with the lower bounds of $(k-1)n - \binom{k}{2}$ (for the case when k is known) and $kn - \binom{k+1}{2}$ (for the case when k is unknown) in Liu and Mukherjee's work for the problem without errors [99] immediately imply Theorem 2.11.1.

2.11.2 Lower bounds for the easy problem: ℓ -false positives

We prove the following lower bound on the query complexity of the ℓ -PL^{FP} problem:

Theorem 2.11.3. *Every algorithm for the ℓ -PL^{FP} problem which guarantees exact recovery of the correct partition requires $\text{RS}^k(n, k) + \Omega(\ell(n-k))$ queries in the worst case.*

As in Section 2.11.1, this result immediately implies the same lower bound for the ℓ -PL problem. In order to analyze the query complexity of learning partitions with ℓ false positives, we consider a very similar game to that defined above in Section 2.11.1.

Definition of the RUCC^{FP} game. The RUCC^{FP} game is defined with the same setup as the RUCC^{FN} game. The questioner Q , responder R , finite set V , and gameplay graph $G_t = (V, E, w_t^+, w_t^-)$ are all defined as in Section 2.11.1. The core distinction between the RUCC^{FP} game and the RUCC^{FN} game is the notion of cost employed. For the RUCC^{FP} game, for any partition $\mathcal{C}$ of V we define the cost

$$\text{cost}_{G_t}^{\text{FP}}(\mathcal{C}) \stackrel{\text{def}}{=} \sum_{\substack{uv \in E \\ u \not\sim_{\mathcal{C}} v}} w_t^+(uv) + \mathbb{I} \left(\sum_{\substack{uv \in E \\ u \sim_{\mathcal{C}} v}} w_t^-(uv) \right).$$

Intuitively, the value of $\text{cost}^{\text{FP}}(\mathcal{C})$ is either the number of incorrect positive answers (+1) that the responder would have to have given if $\mathcal{C}$ were the correct partition, or the value is infinity if the responder would have given any incorrect negative (−1) answer.

The rules of the game require that at each iteration t , the responder must reply in such a way that guarantees the existence of some k -partition $\mathcal{C}^*$ of V such that

$$\text{cost}_{G_t}^{\text{FP}}(\mathcal{C}^*) \leq \ell.$$

The game terminates when there exists a unique k -partition $\mathcal{C}^*$ such that $\text{cost}_{G_t}^{\text{FP}}(\mathcal{C}^*) \leq \ell$.

The query complexity of the ℓ -PL^{FP} problem can be characterized as

$$\min_{Q \in \mathcal{Q}} \max_{R \in \mathcal{R}} \text{Game}^{\text{FP}}(Q, R),$$

where $\text{Game}^{\text{FP}}(Q, R)$ indicates the number of iterations in the RUCC^{FP} game when questioner strategy Q plays against responder R . As in Section 2.11.1, this characterization

allows us to lower bound the query complexity of the problem by analyzing the RUCC^{FP} game:

Lemma 2.11.4. *Let f be a non-negative real valued function. If there exists a responder strategy R^* such that*

$$\forall n, k, \ell \in \mathbb{N}_0 : \quad \min_{Q \in \mathcal{Q}} \text{Game}^{\text{FP}}(Q, R^*) \geq f(n, k, \ell)$$

then the query complexity of learning partitions with ℓ false positives is at least $f(n, k, \ell)$.

The proof is entirely analogous to that of Theorem 2.11.2.

In Theorem 2.12.3 from Section 2.12.2 we show that there exists a responder strategy R^* which satisfies:

$$\min_{Q \in \mathcal{Q}} \text{Game}^{\text{FP}}(Q, R^*) \geq \frac{(\ell + 1)(n - k + 1)}{2} = \Omega(\ell(n - k)).$$

Together with the lower bounds of $(k - 1)n - \binom{k}{2}$ (for the case when k is known) and $kn - \binom{k+1}{2}$ (for the case when k is unknown) in Liu and Mukherjee's work for the problem without errors [99], Lemma 2.11.4 implies Theorem 2.11.3.

2.12 One-sided error lower bounds strategies

2.12.1 A responder strategy for the RUCC^{FN} game

In this section we show that there exists a responder strategy for the RUCC^{FN} game that guarantees the game lasts at least $\Omega(\ell k^2)$ iterations. We consider a responder strategy which we call the $(k + 1)$ -groups responder strategy, denoted R_k^{FN} . In this strategy, the responder fixes an arbitrary $(k + 1)$ -partition of V , denoted $\mathcal{C}^* = \{C_i^*\}_{i=1}^{k+1}$. We define R_k^{FN} as the responder strategy in which the responder replies consistently with $\mathcal{C}^*$ whenever the rules of the RUCC^{FN} game allow, i.e. whenever this does not cause the cost of every k -partition

to exceed ℓ . We lower bound the number of iterations needed for any questioner strategy to terminate the RUCC^{FN} game when playing against R_k^{FN} :

Theorem 2.12.1. *The $(k+1)$ -groups responder strategy R_k^{FN} satisfies*

$$\min_{Q \in \mathcal{Q}} \text{Game}^{\text{FN}}(Q, R_{(k+1)}) \geq (\ell + 1) \left(\binom{k+1}{2} - 1 \right) = \Omega(\ell k^2)$$

queries.

Proof. For $i, j \in [k+1]$ such that $i < j$, consider the k -partitions $\mathcal{C}_{ij}$ obtained by starting with $\mathcal{C}^*$ and merging C_i^* and C_j^* i.e.:

$$\mathcal{C}_{ij} = \{C_a^*\}_{a \in [k+1] \setminus \{i,j\}} \cup \{C_i^* \cup C_j^*\}.$$

We show that the questioner cannot increase the cost of more than one of these partitions $\mathcal{C}_{ij}$ on any single iteration (a claim which we prove afterwards):

Claim 2.12.2. *Suppose that $\text{cost}_{G_t}^{\text{FN}}(\mathcal{C}_{ij}) \leq \ell$. Then on iteration $t+1$ the cost of $\mathcal{C}_{ij}$ increases only if the questioner submits a query uv for $u \in C_i^*$ and $v \in C_j^*$ or $u \in C_j^*$ and $v \in C_i^*$.*

Before proving Claim 2.12.2, let us prove Theorem 2.12.1. Consider the potential function:

$$\phi^{(t)} := \sum_{\substack{i,j \in [k+1] \\ i \neq j}} \min\{\text{cost}_{G_t}^{\text{FN}}(\mathcal{C}_{ij}), \ell + 1\}. \quad (2.12.1)$$

Note that, $\phi^{(0)} = 0$ and, by Claim 2.12.2, at every step t of the game we have:

$$\phi^{(t+1)} \leq \phi^{(t)} + 1. \quad (2.12.2)$$

Suppose that the game terminates at iteration T . Then by the rule of the RUCC^{FN} game, for all but one k -partitions $\mathcal{C}$, it must be the case that $\text{cost}_{G_T}^{\text{FN}}(\mathcal{C}) > \ell$. In particular, all but

at most one of the partitions $\mathcal{C}_{ij}$ must have cost at least $\ell + 1$, implying:

$$\phi^{(T)} \geq (\ell + 1) \left(\binom{k+1}{2} - 1 \right), \quad (2.12.3)$$

and hence:

$$T \geq (\ell + 1) \left(\binom{k+1}{2} - 1 \right) = \Omega(\ell k^2),$$

as needed. $\square$

We conclude this section by proving Claim 2.12.2.

Proof of Claim 2.12.2. Consider G_t such that $\text{cost}_{G_t}^{\text{FN}}(\mathcal{C}_{ij}) \leq \ell$ and such that the RUCC^{FN} Game has not yet terminated. Assume the query submitted by the questioner is not of the form uv for any $u \in C_i^*$, $v \in C_j^*$. For any such uv , by the definition of $\mathcal{C}_{ij}$ the pair u, v is in the same cluster with respect to partition $\mathcal{C}_{ij}$ if and only if the pair is in the same cluster with respect to partition $\mathcal{C}^*$. Thus if the responder is able to return an answer consistent with $\mathcal{C}^*$, this response will also be consistent with $\mathcal{C}_{ij}$, and as a result $\text{cost}_{G_t}^{\text{FN}}(\mathcal{C}_{ij}) = \text{cost}_{G_{t+1}}^{\text{FN}}(\mathcal{C}_{ij})$.

It remains to show that the responder is able to reply consistently with $\mathcal{C}^*$ on such a query. By assumption $\text{cost}_{G_t}^{\text{FN}}(\mathcal{C}_{ij}) \leq \ell$, and as shown above responding consistently to the query uv in a manner consistent with $\mathcal{C}^*$ will ensure $\text{cost}_{G_t}^{\text{FN}}(\mathcal{C}_{ij}) = \text{cost}_{G_{t+1}}^{\text{FN}}(\mathcal{C}_{ij}) \leq \ell$. Thus if the responder replies to query uv consistently with $\mathcal{C}^*$, there will still exist a k -partition—namely, $\mathcal{C}_{ij}$ —such that $\text{cost}_{G_{t+1}}^{\text{FN}}(\mathcal{C}_{ij}) \leq \ell$. Thus the rules of the RUCC^{FN} game allow the responder to reply to query uv consistently with $\mathcal{C}^*$, and by definition of strategy R_k^{FN} the responder will do so. Thus the cost of $\mathcal{C}_{ij}$ at iteration $(t + 1)$ does not increase when the questioner plays such a pair uv . $\square$

2.12.2 A responder strategy for the RUCC^{FP} game

In this section, we show that there exists a responder strategy for the RUCC^{FP} game that guarantees that the game lasts at least $\Omega(\ell(n - k))$ iterations. In particular, we consider a

responder strategy which we call the $(k - 1)$ -groups responder strategy, denoted R_k^{FP} . This strategy fixes $k - 2$ arbitrary elements $v_1^*, \dots, v_{k-2}^*$ and then fixes the following $(k - 1)$ -partition:

$$\mathcal{C}^* := \{V \setminus \{v_1^*, \dots, v_{k-2}^*\}, \{v_1^*\}, \dots, \{v_{k-2}^*\}\}.$$

(See Figure 2.2). Note that when $k = 2$, the set of $\{v_i^*\}$ is just empty and the partition

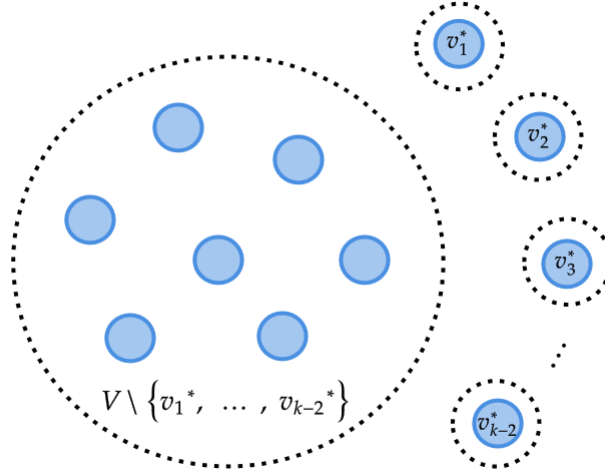


Figure 2.2: The partition $\mathcal{C}^*$ used in the $(k - 1)$ responder strategy for the RUCC^{FP} game.

$\mathcal{C}^*$ consists of a single set equal to V . We define the $(k - 1)$ -groups responder strategy, as the strategy in which the responder replies consistently with $\mathcal{C}^*$ whenever the rules of the RUCC^{FP} game allow them to.

Theorem 2.12.3. *The $(k - 1)$ -groups responder strategy R_k^{FP} satisfies:*

$$\min_{Q \in \mathcal{Q}} \text{Game}^{\text{FP}}(Q, R_{(k-1)}) \geq \frac{(\ell + 1)(n - k + 1)}{2} = \Omega(\ell(n - k)).$$

Proof. We will prove the theorem by considering a subset of the partitions which are hard for the questionner to differentiate among when playing against this responder strategy. Let $S^* := V \setminus \{v_1^*, \dots, v_{k-2}^*\}$, and for any $v \in S^*$ let $\mathcal{C}_v$ be the partition:

$$\mathcal{C}_v := \{\{v\}, S^* \setminus \{v\}, \{v_1^*\}, \dots, \{v_{k-2}^*\}\}.$$

By definition of the RUCC^{FP} game, at the end of the final iteration T , we must have $\text{cost}_{G_T}^{\text{FP}}(\mathcal{C}_v) > \ell$ for all but one of the partitions $\{\mathcal{C}_v\}_{v \in S^*}$. In particular, consider the potential function:

$$\phi^{(t)} \stackrel{\text{def}}{=} \sum_{v \in S^*} \min\{\text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_v), \ell + 1\}.$$

Then, if the game ends at iteration T , we must have:

$$\phi^{(T)} \geq (\ell + 1)(|S^*| - 1) = (\ell + 1)(n - k + 1). \quad (2.12.4)$$

On the other hand, we have:

$$\phi^{(0)} = 0. \quad (2.12.5)$$

We show that this potential $\phi^{(t)}$ changes by a small amount on any given iteration of the RUCC^{FP} game:

Claim 2.12.4. *If $\phi^{(t)} < (n - k + 1)(\ell + 1)$, then $\phi^{(t+1)} \leq \phi^{(t)} + 2$.*

An immediate consequence of Claim 2.12.4 and Equations (2.12.4) and (2.12.5) is that the number of iterations required for the game to terminate is at least $T \geq (n - k + 1)(\ell + 1)/2$, yielding the statement of Theorem 2.12.3. $\square$

We conclude by proving Claim 2.12.4.

Proof of Claim 2.12.4. Let uv be the query submitted by the questioner on iteration $t+1$.

We will split the proof of the theorem into cases.

Case 1. Suppose $\{u, v\} \not\subseteq S^*$. Then, if the responder answers in a way that is consistent with $\mathcal{C}^*$, their answer will also be consistent with $\mathcal{C}_w$ for every $w \in S^*$. By consequence of the assumption that $\phi^{(t)} < (n - k + 1)(\ell + 1)$, a simple counting argument implies that there exists an element $w_t \in S^*$ such that:

$$\text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_{w_t}) \leq \ell.$$

Since w_t as defined above satisfies $\text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_{w_t}) \leq \ell$, this implies that the responder can answer in a way consistent with $\mathcal{C}^*$ without violating the rules of the game. By the definition of the strategy, the responder then will answer in a way consistent with $\mathcal{C}^*$ (and with all partitions $\mathcal{C}_w$) and hence, for all $w \in S^*$: $\text{cost}_{G_{t+1}}^{\text{FP}}(\mathcal{C}_w) = \text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_w)$, giving $\phi^{(t+1)} = \phi^{(t)}$.

Case 2.a. Suppose that $\{u, v\} \subseteq S^*$, and that the responder replies in a way that is consistent with $\mathcal{C}^*$. In this scenario, for any $w \in S^* \setminus \{u, v\}$ we have $\text{cost}_{G_{t+1}}^{\text{FP}}(\mathcal{C}_w) = \text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_w)$, while $\text{cost}_{G_{t+1}}^{\text{FP}}(\mathcal{C}_u) \leq \text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_u) + 1$ and $\text{cost}_{G_{t+1}}^{\text{FP}}(\mathcal{C}_v) \leq \text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_v) + 1$. This immediately implies $\phi^{(t+1)} \leq \phi^{(t)} + 2$.

Case 2.b. Suppose that $\{u, v\} \subseteq S^*$, and that the responder replies in a way that is **not** consistent with $\mathcal{C}^*$, i.e., the responder replies to the query uv with -1 . Then by the definition of the responder strategy, it must be the case that answering $+1$ to the query would violate the rules of the game, i.e., it would cause every k -partition $\mathcal{C}$ of V to have $\text{cost}_{G_{t+1}}^{\text{FP}}(\mathcal{C}) \geq \ell + 1$. But, for every $w \in S^* \setminus \{u, v\}$, the cost of the partition $\mathcal{C}_w$ does not increase if the response to the query $\{u, v\}$ is $+1$. Hence, it must be the case that $\text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_w) \geq \ell + 1$ for all such w . Hence, for any $w \in S^* \setminus \{u, v\}$:

$$\min\{\ell + 1, \text{cost}_{G_{t+1}}^{\text{FP}}(\mathcal{C}_w)\} = \ell + 1 = \min\{\ell + 1, \text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_w)\}. \quad (2.12.6)$$

On the other hand, since the responder is replying to the query uv with -1 , their answer is simultaneously consistent with both $\mathcal{C}_u$ and $\mathcal{C}_v$. We then have:

$$\text{cost}_{G_{t+1}}^{\text{FP}}(\mathcal{C}_u) = \text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_u) \text{ and } \text{cost}_{G_{t+1}}^{\text{FP}}(\mathcal{C}_v) = \text{cost}_{G_t}^{\text{FP}}(\mathcal{C}_v). \quad (2.12.7)$$

In particular, Equations (2.12.6) and (2.12.7) directly imply $\phi^{(t+1)} = \phi^{(t)}$.

This concludes the proof of Claim 2.12.4 (and in turn the proof of Theorem 2.12.3).

□

2.13 Conclusions, Limitations, and Future Directions

In this chapter we initiated the study of learning partitions from same-cluster queries subject to bounded adversarial errors. We completely characterize the query complexity of the core problem as well as some of its relevant variants. To do so, we introduce a novel Rényi–Ulam style game based on correlation clustering.

The aim of this chapter is to resolve the query complexity of exact recovery with a fixed error tolerance. In particular, the results in this chapter do not apply to any setting in which the error may exceed the tolerance ℓ , e.g., when errors occur probabilistically and independently on every query. This limitation is analogous to those encountered when considering the Hamming (worst-case, bounded) model of error in the theory of error-correcting codes (see e.g. [82]). Thus, like the theory for bounded-error codes, results in this chapter should be assumed to hold verbatim in stochastic error settings.

The results in this chapter open up several avenues for future research. First, while we settle the complexity of all the variants of the problem in the k -known setting, the exact complexity of weighted ℓ -PL problem in the setting of k -unknown remains to be determined. Specifically, the lower bounds of $\Omega(k^2\ell_{\text{no}})$ and $\Omega((n-k)\ell_{\text{yes}})$ that we prove for k -known directly extend to the k -unknown setting. However, the analysis of Algorithm 9 can be used to establish upper bounds with terms $O(k^2\ell_{\text{no}} + (n-k)\max\{\ell_{\text{yes}}, \ell_{\text{no}}\})$. The question of whether the upper bounds can be tightened to $O(k^2\ell_{\text{no}} + (n-k)\ell_{\text{yes}})$, or whether the corresponding lower bound can be strengthened to, say, $\Omega((n-k)\max\{\ell_{\text{yes}}, \ell_{\text{no}}\})$ in the k -unknown setting, remains open. Another interesting open question is whether there exist algorithms that can achieve exact recovery in the setting of k -known when the number of false-positives, ℓ_{yes} , is *not* known to the algorithm a priori. In this work we established that such algorithms exist without knowing ℓ_{no} in advance, but it is unclear whether ℓ_{yes} has analogous properties.

We focus on same-cluster queries due to their popularity and simplicity, but there are other practical query models that may interest the community. One such example includes

the triangle queries introduced by Vinayak and Hassibi [151] in which a worker is asked to input all pairwise same-cluster responses for three data points at a time. It would be interesting to explore whether the analysis techniques introduced in this work can determine the query complexity of learning partitions under other query models.

CHAPTER 3

EPSILON-SATURATION FOR STABLE GRAPHS AND LITTLESTONE CLASSES

JOINT WITH MARYANTHE MALLIARIS AND SHAY MORAN

Any Littlestone class, or stable graph, has finite sets which function as “virtual elements”: these can be seen from the learning side as representing hypotheses which are expressible as weighted majority opinions of hypotheses in the class, and from the model-theoretic side as an approximate finitary version of realizing types. We introduce and study the *epsilon-saturation* of a Littlestone class, or stable graph, which is essentially the closure of the class under inductively adding all such virtual elements. We characterize this closure and prove that under reasonable choices of parameters, it remains Littlestone (or stable), though not always of the same Littlestone dimension. This highlights some surprising phenomena having to do with regimes of epsilon and the relation between Littlestone/stability and VC dimension.

3.1 Introduction

In infinite model theory, models give rise to types, which describe elements finitely approximated in the model but possibly not realized there. In learning theory, some learning algorithms (called improper learners) necessarily choose hypotheses outside the given class, which in various examples arise naturally from notions of dimension or of majority. Motivated jointly by these ideas we develop a quantitative notion of “closure under aggregation”, which we call ϵ -saturation.

These definitions make sense for any graph or hypothesis class a priori. However, a characteristic of Littlestone, or stable, classes is that every finite subset of elements or hypotheses contains a “strongly opinionated aggregation” (ϵ -good or ϵ -excellent set) of linear size. So

this is our central case. Let $\ell := \text{Ldim}(\mathcal{H}) \geq d := \text{VC}(\mathcal{H})$ (defined in 3.3.13, 3.3.15 below). We define and characterize the ϵ -saturation $\mathcal{H}_\infty(\epsilon)$ of a Littlestone class $\mathcal{H}$ for any $\epsilon < 1/2$, and we prove that ϵ modulates the complexity of $\mathcal{H}_\infty(\epsilon)$ in a strong way. When $\epsilon < 1/(2^{\ell+2})$, $\text{Ldim}(\mathcal{H}_\infty(\epsilon)) = \text{Ldim}(\mathcal{H}) = \ell$. For some numerical constant $C < 20$, if $\epsilon < 1/(Cd)$, $\mathcal{H}_\infty(\epsilon)$ has possibly larger, but finite, Littlestone dimension. When $1/(d+1) < \epsilon < 1/2$, $\mathcal{H}_\infty(\epsilon)$ may have unbounded VC dimension. We then develop a parallel theory for stable graphs with the more powerful notion of ϵ -excellent replacing ϵ -good.

The proofs involve theorems and methods from combinatorics, statistical learning, logic, Stable Regularity, and Littlestone Minimax.

3.2 Notation

We will deal with several closely related kinds of objects: graphs, bipartite graphs, and hypothesis classes. These need not necessarily be finite, though some constructions of the chapter ask that they are at most countable. To fix notation,

- a graph $G = (V, E)$ has a vertex set and an edge set, $|G|$ means the size of the vertex set, and E is symmetric and irreflexive (i.e., edges are not directed and there are no loops from a vertex to itself).
- we may write “ $a \in G$ ” for $a \in V(G)$ or “ $A \subseteq G$ ” for $A \subseteq V(G)$.
- for edges and nonedges, we may write “ $a \sim b$ ” to abbreviate $(a, b) \in E$ and “ $a \not\sim b$ ” to abbreviate $(a, b) \notin E$.
- a bipartite graph $G = (V_1 \cup V_2, E)$ is given with a partition of the vertex set, and here $E \subseteq V_1 \times V_2$ is neither symmetric nor irreflexive; it is also called a bigraph sometimes, and denoted $G = (V_1, V_2, E)$, with $E \subseteq V_1 \times V_2$ as before.
- a hypothesis class can be denoted either $\mathcal{H}$ or $(X, \mathcal{H})$. where X is a set and $\mathcal{H}$ is a set of subsets of X . It is also usual to identify sets $h \in \mathcal{H}$ with their characteristic

functions, writing $h(a) = 1$ for $a \in h$ and $h(a) = 0$ for $a \notin h$.

- a hypothesis class $\mathcal{H}$ can be seen as a bipartite graph in the natural way, letting X and $\mathcal{H}$ be the two sides of the partition of the vertex set and putting an edge between $a \in X$ and $h \in \mathcal{H}$ if and only if $h(a) = 1$.

Notation used locally in sections of the chapter will be introduced as needed.

3.3 Preliminaries: good, excellent, stable, Littlestone

In this section we briefly review and motivate good and excellent sets and stable graphs following [107], which gives one motivation for our work, and the parallel ideas in Littlestone classes. For further details, see [107], [108], [143], [105]. *Note: in the full generality of the chapter below, we will allow weighted good and excellent sets; but for this introductory section, the main ideas are communicated without this extra notation.* The reader familiar with the area may refer back to this section as needed, or may look ahead to Section 3.11 for examples relevant to the chapter.

Good sets and excellent sets were introduced by Malliaris and Shelah in [107] in order to prove the Stable Regularity Lemma, which is a strong version of Szemerédi Regularity Lemma that holds for stable graphs (see below). In this introduction, good and excellent sets are always finite, although the graph G need not be. Informally, good sets are almost-atoms:

Definition 3.3.1 (Good). *Given a graph G , $\epsilon > 0$, and a finite $B \subseteq G$, B is ϵ -good if for any $a \in G$, either $\{b \in B : a \sim b\}$ or $\{b \in B : a \not\sim b\}$ has size $< \epsilon|B|$. In the first case,¹ write $\mathbf{t}(a, B) = 0$, and in the second, $\mathbf{t}(a, B) = 1$.*

For any ϵ -good set $B \subseteq G$ and any $a \in G$, B behaves like a ‘virtual’ element in the sense that a ‘virtual’ edge or nonedge to a is well defined² (according to $\mathbf{t}(a, B)$). The following

1. Here “ $\mathbf{t}$ ” stands for “truth value,” expressing the majority opinion.

2. The astute reader may wonder about symmetry in the inputs of $\mathbf{t}$, which we address presently.

definition, also from [107], allows for a similar conclusion between virtual elements.³

Definition 3.3.2 (Excellent). *Given a graph G , $\epsilon > 0$, $\delta > 0$, and a finite $A \subseteq G$, A is (ϵ, δ) -excellent if for every δ -good $B \subseteq G$, either $\{a \in A : \mathbf{t}(a, B) = 0\}$ or $\{a \in A : \mathbf{t}(a, B) = 1\}$ has size $< \epsilon|A|$. In the first case, write $\mathbf{t}(A, B) = 1$, and in the second, $\mathbf{t}(A, B) = 0$. When $\epsilon = \delta$, write simply A is ϵ -excellent.*

Excellent sets are good, but the reverse need not be true: see Section 3.12.2 below. The existence of good and excellent sets is characteristic of stable graphs, so-called because of stability in model theory, an important concept in Shelah’s classification theory [135]. To explain this, first recall the connection of various combinatorial definitions:

Definition 3.3.3 (Stable). *A graph G contains a k -half graph if there exist disjoint sets of vertices $\{a_1, \dots, a_k\}$, $\{b_1, \dots, b_k\}$ so that $a_i \sim b_j$ if $i < j$ and $a_i \not\sim b_j$ if $j \geq i$. G is called k -stable, or sometimes k -edge stable for emphasis, if it contains no k -half graphs.⁴*

Convention 3.3.4. *We say “ G is a stable graph” to mean that G is k -stable for some finite k .*

The next definition is also a finite version of a configuration from Shelah’s classification theory, used to sound out the complexity of a formula [such as the edge relation $E(x, y)$] by asking whether the family of instances of the formula [such as the set of neighborhoods] allowed for a certain kind of repeated partitioning. We say “special tree” after [107], others say Shelah tree or Hodges tree.⁵

Definition 3.3.5 (Special trees). *A graph G contains a special tree of height m if there exist disjoint sets of vertices $\{a_\eta : \eta \in {}^{m>}2\}$, $\{b_\nu : \nu \in {}^m2\} \subseteq G$, so that:*

3. If A and B are ϵ -good, they are already in some sense close to being a regular pair, at the cost of changing ϵ ; as we prefer to keep ϵ fixed, excellence allows for a stronger conclusion.

4. Not all half-graph edges are specified, so k -stable forbids a family of configurations.

5. The motivation for these names will become evident shortly. Recall that in set theoretic notation, $2 = \{0, 1\}$, so ${}^{m>}2$ is the set of binary sequences of length strictly less than m , m2 is the set of binary sequences of length exactly m , and $\eta \smallfrown \langle t \rangle \leq \nu$ means that the string η followed by t is an initial segment of ν .

- if $\eta \frown \langle 0 \rangle \sqsubseteq \nu$, then $a_\eta \not\sim b_\nu$
- if $\eta \frown \langle 1 \rangle \sqsubseteq \nu$, then $a_\eta \sim b_\nu$

(and there are no constraints on other edges).

In Shelah's classification theory, one of the consequences of his celebrated Unstable Formula Theorem ([135], Theorem II.2.2, circa 1978) is that special trees and half-graphs are either both bounded or both unbounded.⁶ Around 1981, Hodges looked for a quantitative version, and proved (in our language) that there is a log bound in each direction. In the next Fact, (1) is essentially sharp but (2) is not known to be. Given its role in our argument below, we sketch a proof.

Fact 3.3.6 (Hodges–Shelah lemma, see [88]). *In a graph or bipartite graph G :*

1. *if there is a 2^{m+1} -half graph, there is an m -tree*
2. *if there is a $2^{k+2} - 2$ -special tree, there is a k -half graph.*

Proof Sketch. First, from a k -half graph $a_1, \dots, a_k, b_1, \dots, b_k$, we find a tree of height about $\log k$. We will let the b 's be the leaves of our tree and choose a 's for the internal nodes. For the root $a_\emptyset$, we choose the midpoint of the a 's, call it $a_{\lceil k/2 \rceil}$, noting that this element connects to the right half of the b 's and not to the left half, as befits the root of the tree. Continue inductively, choosing the 2^ℓ internal nodes needed for level ℓ as the midpoints of the segments with endpoints $\lceil ki/2^\ell \rceil$.

From an m -tree $A = \{a_\eta : \eta \in {}^{m>}2\}$, $B = \{b_\nu : \nu \in {}^m2\}$ we can obtain a half-graph of length about $\log m$ by induction. There are two preliminary steps. First, by induction on ℓ , we define what it means to be a subtree of height ℓ . A subtree of height 0 is a leaf, and a subtree of height $\ell+1$ consists of an internal node a_η which has two distinct proper extensions

6. That is, in the class of models of a complete first-order theory, in the bipartite graph given by fixing a first-order formula $\varphi(\bar{x}; \bar{y})$ and connecting two tuples of the correct length precisely when φ holds, special trees and half-graphs are either both bounded or both unbounded. For a discussion of Shelah's unstable formula theorem and learning theory, see [106].

a_ρ and a_ν , which are each the root of a subtree of height ℓ (here η is the common initial segment of ρ and ν , but ρ and ν need not have the same length; what happens necessarily, is that $\eta \frown \langle 1 \rangle$ is a prefix of one of ρ or ν , and $\eta \frown \langle 0 \rangle$ is a prefix of the other). Second, one proves a Ramsey-type lemma saying that if we 2-color the internal nodes of an m -tree then there is a monochromatic subtree of height about $m/2$.

Now let us choose a first pair of elements for our half-graph. (We will either be choosing a_1, b_1 or a_k, b_k .) Start by choosing any $b \in B$, which induces a 2-coloring of A according to whether or not $b \sim a$. By the Ramsey lemma we can find a large monochromatic subtree. Call it A_* , and suppose that $b \sim a$ for all $a \in A_*$. In this case our b will become b_k . Define a_k to be the root of the tree A_* . Let A_{**} be the subtree whose root is the left successor of a_k in A_* , and let B_{**} be its set of leaves. Now we have found a_k, b_k , and we have a slightly smaller tree A_{**}, B_{**} with the property that $b_k \sim a$ for all $a \in A_{**}$, and $a_k \not\sim b$ for all $b \in B_{**}$. Note also that had our Ramsey lemma returned $\not\sim$, we would have set $b_1 = b$, a_1 to be the root of A_* , A_{**} to be the subtree whose root is the right successor of a_1 in A_* , and B_{**} its leaves. We would then have found a_1, b_1 and a slightly smaller tree (A_{**}, B_{**}) with the property that $b_1 \not\sim a$ for all $a \in A_{**}$, and $a_1 \sim b$ for all $b \in B_{**}$. Either way, we can clearly focus on (A_{**}, B_{**}) and repeat. Each iteration reduces the height of the tree by about half. $\square$

Conclusion 3.3.7. *If G is a k -stable graph, hence has no k -half graphs, there is $m = m(k) \leq 2^{k+2} - 2$ such that G has no m -special trees.*

This connection of orders and trees was leveraged by Malliaris and Shelah in the stable regularity lemma to prove existence of good and excellent sets. These proofs will be central to our work in this chapter so we review them here. Since sets of size one are good, of course Lemma 3.3.9 implies Claim 3.3.8 and the inclusion of both proofs is pedagogical; also, if the X with which we begin is too small, say $|X| < 2^m$, we can choose any singleton and finish (but the proof works regardless and will find a singleton).

Claim 3.3.8 (Good sets in stable graphs, [107]). *Let G be a k -stable graph, and let $m = m(k)$ be from Conclusion 3.3.7. Then for any $0 < \epsilon < 1/2$, and any finite $X \subseteq G$, there is $Y \subseteq X$, with $|Y| \geq \epsilon^m |X|$ such that Y is ϵ -good.*

Proof. We proceed by induction⁷ on $\ell < m$. Let $X_\emptyset = X$. For each $\eta \in {}^\ell 2$, if X_η is not ϵ -good, let $a_\eta \in G$ be such that $X_{\eta \frown \langle 0 \rangle} := \{x \in X_\eta : x \not\sim a_\eta\}$ and $X_{\eta \frown \langle 1 \rangle} := \{x \in X_\eta : x \sim a_\eta\}$ are both nonempty of size $\geq \epsilon |X_\eta|$. Suppose the construction continues through stage $m-1$ so we have defined $\{X_\rho : \rho \in {}^m 2\}$ (which are by construction disjoint) and $\{a_\nu : \nu \in {}^{m>} 2\}$. By hypothesis each $X_\rho \neq \emptyset$, so we may choose $x_\rho \in X_\rho$ for each $\rho \in {}^m 2$. Then $\{a_\nu : \nu \in {}^{m>} 2\}$, $\{x_\rho : \rho \in {}^m 2\}$ form an m -tree, contradiction. So one of the X_η 's (for η of length $\leq m-1$) must have been ϵ -good, and its size will be $\geq \epsilon^m |X|$. $\square$

Lemma 3.3.9 (Excellent sets in stable graphs, [107]). *Let G be a k -stable graph, and let $m = m(k)$ be from Conclusion 3.3.7. Then for any $0 < \epsilon < 1/2^m$, and any finite $X \subseteq G$, there is $Y \subseteq X$, $|Y| \geq \epsilon^m |X|$ such that Y is ϵ -excellent.*

Proof. We also proceed by induction on $\ell < m$. Let $X_\emptyset = X$. For each $\eta \in {}^\ell 2$, if X_η is not ϵ -excellent, let $A_\eta \subseteq G$ be an ϵ -good set such that $X_{\eta \frown \langle 0 \rangle} := \{x \in X_\eta : \mathbf{t}(x, A_\eta) = 0\}$ and $X_{\eta \frown \langle 1 \rangle} := \{x \in X_\eta : \mathbf{t}(x, A_\eta) = 1\}$ are both nonempty of size $\geq \epsilon |X_\eta|$. Suppose the construction continues through stage $m-1$ so we have defined $\{X_\rho : \rho \in {}^m 2\}$ (which are by construction disjoint) and $\{A_\nu : \nu \in {}^{m>} 2\}$.

We now have a ‘virtual’ tree from which we would like to extract an ‘actual’ special tree. Since $X_\rho \neq \emptyset$ for every $\rho \in {}^m 2$, we first choose any $x_\rho \in X_\rho$ for each such ρ . For each $\nu \in {}^{m>} 2$, let $\mathbf{t}_\rho$ be such that $\mathbf{t}(x_\rho, A_\nu) = \mathbf{t}_\rho$. This can be considered as a virtual edge, and we want to find an actual element $a_\nu \in A_\nu$ which has an actual edge of the correct kind to x_ρ , for all ρ extending ν . More precisely, the set $\{\rho \in {}^m 2 : \nu \triangleleft \rho\}$ has size $\leq 2^m$. For any such ρ , the set $\{a \in A_\nu : \mathbf{t}(x_\rho, a) \neq \mathbf{t}(x_\rho, A_\nu)\}$ of incorrect elements has size $< \epsilon |A_\nu|$. By a union bound and the upper bound on the value of ϵ , after each x_ρ has ruled

7. If X is not ϵ -good, letting $a_\emptyset \in G$ be such that $B_{\langle 0 \rangle} := \{b \in X : b \not\sim a_\emptyset\}$ and $B_{\langle 1 \rangle} := \{b \in X : b \sim a_\emptyset\}$ are both nonempty of size $\geq \epsilon |X|$ starts building a tree.

out its small set, some a_ν remains. Then $\{a_\nu : \nu \in {}^{m>}2\}$, $\{x_\rho : \rho \in {}^m2\}$ form a depth m special tree, contradiction. So one of the X_η 's (for η of length $\leq m - 1$) must have been ϵ -excellent, and its size is at least $\epsilon^m |X|$. $\square$

Note that this result can be extended to any $\epsilon < 1/2$ by a theorem of Malliaris and Moran [105]. However, in that result the guarantee on the size of the excellent set, while still linear, may be smaller. The proof is beyond the scope of this introduction and involves multiplicative-weights algorithms.

Conclusion 3.3.10 ([105] Claim 2.11, in our language). *For a graph G , the following are equivalent:*

1. G is stable,
2. for every $\epsilon < 1/2$ there is a constant $c = c(\epsilon) > 0$ such that for every finite $A \subseteq G$ there is $B \subseteq A$, $|B| \geq c|A|$, such that $|B|$ is ϵ -good.

As mentioned above, a hypothesis class $(X, \mathcal{H})$ can naturally be seen as a bipartite graph. The natural analogue in learning of half-graphs is called threshold dimension (as was used in Alon–Livni–Malliaris–Moran [17]). Note that this dimension is a priori one more than stability as $\text{Thr}(G) = k$ means the largest half-graph has size k , i.e., G is $(k + 1)$ -stable.

Definition 3.3.11 (Thresholds). *The threshold dimension of a hypothesis class $\mathcal{H}$, $\text{Thr}(\mathcal{H})$, is defined to be the greatest natural number ℓ , if it exists, such that there exist $x_1, \dots, x_\ell \in X$ and $h_1, \dots, h_\ell \in \mathcal{H}$ with $h_i(x_j) = 1$ iff $i < j$. If no such ℓ exists⁸, $\text{Thr}(\mathcal{H}) = \infty$.*

The natural analogue in learning of special trees are Littlestone mistake trees, as pointed out by Chase and Freitag [38], who used this to give new examples of Littlestone classes. Note that here too we record the largest d where a tree occurs rather than the smallest d where it does not; also, following convention, the root of a mistake tree is at level 1, not level 0.

8. One can also define ordinal threshold dimension, see [29]. A subtlety in allowing ordinal values is to distinguish between the case of having arbitrarily large finite half-graphs and infinite half-graphs.

Definition 3.3.12 (Littlestone mistake tree). *Let $(X, \mathcal{H})$ be a hypothesis class. A Littlestone mistake tree of height d is a full binary tree whose internal nodes are named by elements of X , and which is shattered by $\mathcal{H}$ in the following sense.*

- (a) *We can represent the process of traversing a root-to-leaf path (i.e., a branch) in the tree as a sequence of pairs $\langle (x_i, y_i) : 1 \leq i \leq d \rangle$, where (x_i, y_i) records that the name of the current node is $x_i \in X$ and from this node we travel right (if $y_i = 1$) or left (if $y_i = 0$).*
- (b) *We say $h \in \mathcal{H}$ realizes a given branch $\langle (x_i, y_i) : 1 \leq i \leq d \rangle$ if for all $1 \leq i \leq d$, $h(x_i) = y_i$.*
- (c) *We say that $\mathcal{H}$ shatters the tree if every branch is realized by some $h \in \mathcal{H}$.*

Definition 3.3.13 (Littlestone). *The Littlestone dimension of a hypothesis class, $\text{Ldim}(\mathcal{H})$, is defined to be the greatest natural number d , if it exists, so that there exists a Littlestone mistake tree of height d which is shattered by $\mathcal{H}$. If no such d exists (i.e., if there are Littlestone mistake trees shattered by $\mathcal{H}$ of arbitrarily large finite height), we set $\text{Ldim}(\mathcal{H}) = \infty$. $\mathcal{H}$ is a Littlestone class if $\text{Ldim}(\mathcal{H}) < \infty$.*

Littlestone classes are well-studied in the learning literature, and are characterized by being *online learnable* (see Littlestone [97] and Ben-David, Pál, Shalev-Schwartz [26]). For more on Littlestone classes and online learning see [130] §18.1-18.2.⁹

In what follows, we shall use the analogue of goodness, excellence, and the Hodges-Shelah lemma in hypothesis classes in the natural way. This takes up a line of work beginning with Alon–Livni–Malliaris–Moran and Bun–Livni–Moran [17, 35] showing that the property of being Littlestone, i.e. stable, gave a qualitative characterization of when a hypothesis class was differentially-privately PAC learnable. This connection of stability/Littlestone/online learning to statistical learning was part of the motivation for the present chapter.

For completeness, recall the Vapnik–Chervonenkis dimension of a class $(X, \mathcal{H})$.

9. Note also that $\text{Ldim}(\mathcal{H}) = \infty$ a priori does not distinguish between shattered trees of arbitrarily large finite height and shattered trees of infinite height. See [29].

Definition 3.3.14 (Shattering). *In a hypothesis class $(X, \mathcal{H})$, say a set $S \subseteq X$ is shattered if, for every $S' \subseteq S$, there exists $h \in \mathcal{H}$ so that $\{x \in X : h(x) = 1\} \cap S = S'$.*

Definition 3.3.15 (VC dimension). *The Vapnik–Chervonenkis dimension of a hypothesis class, $\text{VC}(\mathcal{H})$, is defined to be the greatest natural number m , if it exists, so that there exists a set S of size m which is shattered by $\mathcal{H}$. If no such m exists (i.e., if there are VC mistake trees shattered by $\mathcal{H}$ of arbitrarily large finite height), we set $\text{VC}(\mathcal{H}) = \infty$. $\mathcal{H}$ is called a VC class if $\text{VC}(\mathcal{H}) < \infty$.*

Note that $\text{Ldim}(\mathcal{H}) \geq \text{VC}(\mathcal{H})$ for every hypothesis class $\mathcal{H}$ (and in the graph case, for every graph $\mathcal{G}$). This is because shattering a set in the way described above corresponds to a mistake tree with the same structure as a Littlestone mistake tree, with the additional condition that all internal nodes which lie in the same level must be the same element of X .

Finally, it will be useful to define the dual of a hypothesis class. Informally, any hypothesis class corresponds to a bipartite graph with X on the left, $\mathcal{H}$ on the right, and an edge between x and h if and only if $h(x) = 1$. The dual class corresponds to the bipartite graph we would get by swapping the left and right sides without changing any edges. More formally:

Definition 3.3.16 (Dual hypothesis class). *For a hypothesis class $(X, \mathcal{H})$, define the dual hypothesis class $\text{dual}(\mathcal{H})$ to be $(Y, \mathcal{G})$ where $Y = \mathcal{H}$ is the domain and $\mathcal{G} = \{g_x : x \in X\}$ is a set of functions from $\mathcal{H}$ to $\{0, 1\}$ given by: $g_x(h) = 1$ if and only if $h(x) = 1$.*

$\mathcal{H}$ is a VC class if and only if $\text{dual}(\mathcal{H})$ is a VC class, possibly of larger dimension, and $\mathcal{H}$ is Littlestone if and only if $\text{dual}(\mathcal{H})$ is Littlestone, possibly of larger dimension. For the first fact, see e.g. Lemma 10.3.4 in [113]. The second fact is proved by using a Hodges–Shelah tree to say that Littlestone dimension is finite if and only if Threshold dimension is finite:

Conclusion 3.3.17 (Hodges–Shelah lemma translated to hypothesis classes). *Given a hypothesis class $(X, \mathcal{H})$:*

1. *If $\text{Thr}(\mathcal{H}) \geq 2^{m+1}$, then $\text{Ldim}(\mathcal{H}) \geq m$.*

2. If $\text{Ldim}(\mathcal{H}) \geq 2^{k+2} - 2$, then $\text{Thr}(\mathcal{H}) \geq k$.

since Threshold dimension is clearly symmetric between a class and its dual.

3.4 Definition of ϵ -saturation for $\mathcal{H}$

In this section we give the central definitions of the chapter for hypotheses classes, corresponding to saturation and to types. Graphs are considered in Section 3.9.

These definitions work when $\mathcal{H}$ is finite, but they don't require it. The sets of aggregates we take, however, are always finite.

Convention 3.4.1. $(X, \mathcal{H})$, which we may abbreviate $\mathcal{H}$, will usually denote a Littlestone class, with

- finite Littlestone dimension ℓ ,
- and hence finite VC dimension d .

$\mathcal{H}$ can be finite, but need not be.

We will state “ $\mathcal{H}$ is Littlestone” when it is used in the main definitions and theorems, and will keep the convention of ℓ and d .

Convention 3.4.2. Below, ϵ and δ (possibly with subscripts or other decorations) are always in the interval $(0, 1/2)$, subject to further restrictions as stated.

Notation 3.4.3. A notational convention: we think of integers set-theoretically, so $n = \{0, \dots, n-1\}$ and “ $i < n$ ” means i ranges over the n values $0, \dots, n-1$.

In the present chapter, we allow good and excellent sets to be weighted. By convention we ask the weights to be positive, although we could allow zero weights at the cost of modifying many instances of “thus there exists g ” to “thus there exists g of nonzero weight” below.

Definition 3.4.4. Given ϵ and $\mathcal{H}$, an ϵ -hypothesis (or a weighted ϵ -good set) of $\mathcal{H}$ is a finite set of pairs:

$$H = \{(h_i, \gamma_i) : i < k\}$$

where $k \in \mathbb{N}$, and:

1. $\{h_i : i < k\} \subseteq \mathcal{H}$, and $\langle h_i : i < k \rangle$ is without repetition,
2. each $\gamma_i \in (0, 1]_{\mathbb{R}}$, and $\sum \{\gamma_i : i < k\} = 1$,
3. (ϵ -goodness) For each $x \in X$, for some $\mathbf{t} = \mathbf{t}(x, H) = \mathbf{t}_H(x) \in \{0, 1\}$,

$$\sum \{\gamma_i : h_i(x) = \mathbf{t}\} \geq 1 - \epsilon \text{ and hence } \sum \{\gamma_i : h_i(x) = 1 - \mathbf{t}\} < \epsilon.$$

Here $\mathbf{t}$ represents the “truth value” of the majority opinion between x and H .

Remark 3.4.5. As the definition records, any ϵ -hypothesis H determines a function $\mathbf{t}_H : X \rightarrow \{0, 1\}$. Say that H represents $\mathbf{t}_H$. A priori, $\mathbf{t}_H$ need not be in $\mathcal{H}$. Also, a priori, different ϵ -hypotheses may represent the same function.

We will use several different notions of approximation. Since these are well-studied we keep the usual names:

Definition 3.4.6. Suppose we are given $\mathcal{H}$ and $f : X \rightarrow \{0, 1\}$. Say f is:

1. k -realizable in $\mathcal{H}$, for a fixed $k \in \mathbb{N}$, when for every $Y \subseteq X$, $|Y| \leq k$, there exists $h \in \mathcal{H}$ such that $h \upharpoonright Y = f \upharpoonright Y$.
2. finitely realizable in $\mathcal{H}$ if f is k -realizable in $\mathcal{H}$ for every finite k .
3. realized if $f \in \mathcal{H}$.

Write $\mathcal{H}^{k\text{-real}}$ for the set of functions $f : X \rightarrow \{0, 1\}$ which are k -realizable in $\mathcal{H}$, and $\mathcal{H}^{(<\omega)\text{-real}} = \bigcap \{\mathcal{H}^{k\text{-real}} : k < \omega\}$ for the set of finitely realizable functions.

Remark 3.4.7. *Note that in the definition above, $\mathcal{H}^{k\text{-real}}$ could be defined as well if k is a positive real number. Then, $\mathcal{H}^{k\text{-real}} = \mathcal{H}^{\lfloor k \rfloor\text{-real}}$. This will be used in later sections.*

The next will play a key role. To see why Definition 3.4.8 below is preferable to finite realizability here, consider that when X is finite, the only finitely realizable functions are those already in $\mathcal{H}$, but for $\epsilon > 1/|X|$ there may be ϵ -types which are not realized, as in Section 3.11.2 below.

Definition 3.4.8 (ϵ -types). *Suppose we are given $\mathcal{H}$, $0 < \epsilon < 1/2$, and $f : X \rightarrow \{0, 1\}$. We call f an ϵ -type of $\mathcal{H}$, or just ϵ -type, if f is finitely realized in the class*

$$\{\mathbf{t}_H : H \text{ is an } \epsilon\text{-hypothesis of } \mathcal{H} \}.$$

In a slogan, the function f is an ϵ -type if its restriction to any finite set agrees with some ϵ -hypothesis (that is, with some finite weighted ϵ -good set of hypotheses).

The central definition of this chapter, in the case of hypothesis classes, is:

Definition 3.4.9 (The ϵ -saturation of $\mathcal{H}$). *Fix $0 < \epsilon < 1/2$ and $(X, \mathcal{H})$ a hypothesis class. Define by induction on $n \leq \omega$ a hypothesis class $\mathcal{H}_n(\epsilon)$ over X :*

1. $\mathcal{H}_0(\epsilon) = \mathcal{H}$,
2. $\mathcal{H}_{n+1}(\epsilon) = \mathcal{H}_n \cup \{\mathbf{t}_H : H \text{ is an } \epsilon\text{-hypothesis of } \mathcal{H}_n\}$,
3. $\mathcal{H}_\infty(\epsilon) := \mathcal{H}_\omega(\epsilon) := \bigcup \{\mathcal{H}_n(\epsilon) : n < \omega\}$.

We call $\mathcal{H}_\infty(\epsilon)$ “the ϵ -saturation of $\mathcal{H}$.”

Note that in condition (2) we define $\mathcal{H}_{n+1}$ as a set of functions, not names for functions; so in particular, if H_1 and H_2 are two different ϵ -hypotheses (that is, different finite weighted good sets) but $\mathbf{t}_{H_1} = \mathbf{t}_{H_2}$ as functions, we only add one copy of this function. (In particular, we do not re-add copies of functions already in $\mathcal{H}$.) In later calculations, it will not matter which representation we choose.

The words “saturation” and “type” come from model theory. In model theoretic language, saturation describes a structure where all types of a suitable size are realized. So this notation suggests a connection of ϵ -saturation and ϵ -types, which will be justified in due course.

Remark 3.4.10. *In Definition 3.4.9, note X stays fixed. If we were to increase both X and H by adding majorities, then in order to retain total functions, we would need (in the language of stable regularity) to upgrade “ ϵ -good” in the definition of ϵ -hypothesis above to “ ϵ -excellent.” This will be a consideration when we deal with graphs rather than hypothesis classes, see Definition 3.9.4 below.*

Remark 3.4.11. *In Definition 3.4.9 we could have defined $\mathcal{H}_\alpha(\epsilon)$ for ordinals α by transfinite induction, but this will not be used here.*

Remark 3.4.12. *Observe that as $n \rightarrow \infty$, the set of ϵ -types may change even though X stays the same, because the ϵ -hypotheses available in $\mathcal{H}_n(\epsilon)$ to witness finite realizability may be changing. See for instance Section 3.11.5.*

3.5 Preservation of dimension for small ϵ

This section proves Theorem 3.5.6 which says that Littlestone, threshold, and VC dimension are preserved from $\mathcal{H}$ to $\mathcal{H}_\infty(\epsilon)$ provided ϵ is sufficiently small relative to the dimension in question, by applying ideas from stable regularity (as outlined in Section 3.3) to weighted good and excellent sets in the natural way. The theorem will rely on several lemmas.

For orientation, recall “ $\text{Ldim} \geq \ell$ ” and “ $\text{VC} \geq d$ ” and “ $\text{Thr} \geq k$ ” are witnessed by the existence of certain finite configurations: the nodes and leaves of a shattered tree of height ℓ in the first case, the shattered set of size d (that is, d elements and 2^d hypotheses realizing all possible functions over them) in the second, and a half-graph of length k in the third. (We will also use that VC dimension can be witnessed by a shattered tree where all the labels at a given level are the same.) So in each case, the Littlestone, VC, or threshold dimension of $\mathcal{H}_n$

is nondecreasing with n . Also, since these are finite configurations, if the given configuration exists in $\mathcal{H}_\infty$ then already it exists in $\mathcal{H}_n$ for some n .

We will say “special tree” rather than Littlestone tree to emphasize the parallelism, since in some cases we will want to choose the nodes from the X side and the leaves from the hypothesis side, and in other cases the reverse.

Remark 3.5.1. *Recall our set-theoretic notation: for $s \in \mathbb{N}$, “ $i < s$ ” means that i takes the s values $0, \dots, s-1$. Recall also from Definition 3.3.5 that $^{s>}2$ is the set of binary sequences of length strictly less than s , and s2 is the set of binary sequences of length s .*

For Lemmas 3.5.2–3.5.5 let $(X, \mathcal{H})$ be an arbitrary but fixed hypothesis class. Here we can allow “ $\epsilon \leq \dots$ ” instead of “ $\epsilon < \dots$ ” just because the definition of ϵ -hypothesis has a strict inequality on the smaller side.

Lemma 3.5.2. *For any $s \in \mathbb{N}$ and $\epsilon \leq 1/s$, if there is a special tree of height s with the internal nodes chosen from X and the leaves chosen from $\mathcal{H}_\infty$, then there is a special tree of height s with the same internal nodes, and the leaves chosen from $\mathcal{H}$.*

Proof. Consider a special tree given by

$$\langle x_\eta : \eta \in {}^{s>}2 \rangle \text{ from } X \text{ and } \langle y_\rho : \rho \in {}^s2 \rangle \text{ from } \mathcal{H}_\infty.$$

Let $n < \omega$ be minimal such that $\{y_\rho : \rho \in {}^s2\} \subseteq \mathcal{H}_{n+1}$. For each y_ρ choose an ϵ -hypothesis H_ρ of $\mathcal{H}_n$ representing the same function over X as y_ρ . So we now have $\langle x_\eta : \eta \in {}^{s>}2 \rangle$ and $\langle H_\rho : \rho \in {}^s2 \rangle$ with $\mathbf{t}(H_\rho, x_\eta) = 1$ if $\eta \frown \langle 1 \rangle \leq \rho$ and $\mathbf{t}(H_\rho, x_\eta) = 0$ if $\eta \frown \langle 0 \rangle \leq \rho$. For each $\rho \in {}^s2$, define the error set

$$E_\rho = \bigcup \{(h_i, \gamma_i) \in H_\rho : \mathbf{t}(h_i, x_\eta) \neq \mathbf{t}(H_\rho, x_\eta) \text{ for some } \eta \text{ with } \eta \triangleleft \rho\}.$$

Observe that $\{\eta : \eta \triangleleft \rho\}$ has size s and that for each η , the “error weight” is $< \epsilon$. In other words, $\sum \{\gamma_i : (h_i, \gamma_i) \in E_\rho\} < \epsilon s \leq (1/s)s = 1$. So there is some $(h, \gamma) \in H_\rho \setminus E_\rho$. Set

$a_\rho := h$. Having thus defined each a_ρ , we have found a special tree with internal nodes $\langle x_\eta : \eta \in {}^{s>}2 \rangle$ from X and leaves $\langle a_\rho : \rho \in {}^s2 \rangle$ from $\mathcal{H}_n$. Either $n = 0$, or we contradict our choice of n , so either way we are done. $\square$

Lemma 3.5.3. *For any $s \in \mathbb{N}$ and $\epsilon \leq 1/2^s$, if there is a special tree of height s where the internal nodes are from $\mathcal{H}_\infty$ and the leaves are from X , there is also a special tree of height s with internal nodes from $\mathcal{H}$ and the same leaves from X .*

Proof. Consider $\langle z_\eta : \eta \in {}^{s>}2 \rangle$ from $\mathcal{H}_\infty$ and $\langle x_\rho : \rho \in {}^s2 \rangle$ from X which form a special tree. Let $n < \omega$ be minimal so that $\{z_\eta : \eta \in {}^{s>}2\} \subseteq \mathcal{H}_{n+1}$. For each $\eta \in {}^{s>}2$ choose an ϵ -hypothesis H_η of $\mathcal{H}_n$ representing the same type over X as z_η . For orientation, this means $\mathbf{t}(H_\eta, x_\rho) = 1$ if $\eta \frown \langle 1 \rangle \leq \rho$ and $\mathbf{t}(H_\eta, x_\rho) = 0$ if $\eta \frown \langle 0 \rangle \leq \rho$. For each of the internal indices $\eta \in {}^{s>}2$ there are $2^{1-\lg(\eta)} \leq 2^s$ leaf¹⁰ indices $\rho \in {}^s2$ with $\eta \leq \rho$. For each $\eta \in {}^{s>}2$, define the error set

$$E_\eta = \{(h_i, \gamma_i) \in H_\eta : \mathbf{t}(h_i, x_\rho) \neq \mathbf{t}(H_\eta, x_\rho) \text{ for some } \rho \text{ with } \eta \triangleleft \rho\}.$$

Then $\sum \{\gamma_i : (h_i, \gamma_i) \in E_\eta\} < \epsilon 2^s \leq 1$ so there is some $(h, \gamma) \in H_\eta \setminus E_\eta$. Set $c_\eta = h$. Having chosen all such c_η s, $\langle c_\eta : \eta \in {}^{s>}2 \rangle$ from H_n and $\langle x_\rho : \rho \in {}^s2 \rangle$ from X give a special tree of height s . $\square$

Lemma 3.5.4. *In the notation of the previous proof: suppose that our original tree $\langle z_\eta : \eta \in {}^{s>}2 \rangle$, $\langle x_\rho : \rho \in {}^s2 \rangle$ had the property that for all $\eta_1, \eta_2 \in {}^{s>}2$, if $\lg(\eta_1) = \lg(\eta_2)$ then $z_{\eta_1} = z_{\eta_2}$. Then we can ensure that for all $\eta_1, \eta_2 \in {}^{s>}2$, if $\lg(\eta_1) = \lg(\eta_2)$ then $c_{\eta_1} = c_{\eta_2}$.*

Proof. The property says that for each $t < s$, for every η of length t we chose the same value. For each such t , then, make sure to choose the same H_η representing each z_η of length t , and denote it by H_t . Then we simply let the error set for H_t collect the wrong votes from all of the x_ρ 's, of which there are 2^s -many. $\square$

10. Here, $\lg(\eta)$ represents the length of η .

Lemma 3.5.5. *Suppose there is a half-graph consisting of $\langle x_i : i < s \rangle$ from X and $\langle y_i : i < s \rangle$ from $\mathcal{H}_\infty$. If $\epsilon \leq 1/s$, there is a half-graph consisting of the same $\langle x_i : i < s \rangle$ from X and $\langle z_i : i < s \rangle$ from $\mathcal{H}$.*

Proof. Again, the error set picks up an error of weight strictly less than ϵ from each of the at most s elements on the opposite side. $\square$

Summarizing:

Theorem 3.5.6 (Preservation theorems for small ϵ). *Let $\mathcal{H}$ be any hypothesis class and $s \in \mathbb{N}$.*

1. *If $\epsilon \leq 1/s$,*

$$\text{Ldim}(\mathcal{H}) \geq s \iff \text{Ldim}(\mathcal{H}_\infty(\epsilon)) \geq s$$

and also

$$\text{VC}(\mathcal{H}) \geq s \iff \text{VC}(\mathcal{H}_\infty(\epsilon)) \geq s.$$

2. *If $\epsilon \leq 1/2^s$,*

$$\text{Ldim}(\text{dual}(\mathcal{H})) \geq s \iff \text{Ldim}(\text{dual}(\mathcal{H}_\infty(\epsilon))) \geq s$$

and also

$$\text{VC}(\text{dual}(\mathcal{H})) \geq s \iff \text{VC}(\text{dual}(\mathcal{H}_\infty(\epsilon))) \geq s.$$

3. *Hence if $\mathcal{H}$ is a Littlestone class and $\epsilon \leq 1/(\text{Ldim}(\mathcal{H}) + 1)$ then*

$$\text{Ldim}(\mathcal{H}) = \text{Ldim}(\mathcal{H}_\infty(\epsilon)).$$

4. *Hence if $\mathcal{H}$ is a VC class and $\epsilon \leq 1/(\text{VC}(\mathcal{H}) + 1)$ then*

$$\text{VC}(\mathcal{H}) = \text{VC}(\mathcal{H}_\infty(\epsilon)).$$

5. Note: regardless of ϵ , if $\mathcal{H}$ is not Littlestone, $\text{Ldim}(\mathcal{H}) = \text{Ldim}(\mathcal{H}_\infty(\epsilon)) = \infty$, and if $\mathcal{H}$ is not a VC class, $\text{VC}(\mathcal{H}) = \text{VC}(\mathcal{H}_\infty(\epsilon)) = \infty$.
6. In particular, if $\text{Ldim}(\mathcal{H}) = \ell \in \mathbb{N}$ and $\epsilon < 1/2^{\ell+2}$ then we have that $\text{Ldim}(\mathcal{H}) = \text{Ldim}(\mathcal{H}_\infty(\epsilon))$, $\text{Thr}(\mathcal{H}) = \text{Thr}(\mathcal{H}_\infty(\epsilon))$, $\text{Ldim}(\text{dual}(\mathcal{H})) = \text{Ldim}(\text{dual}(\mathcal{H}_\infty(\epsilon)))$, $\text{VC}(\mathcal{H}) = \text{VC}(\mathcal{H}_\infty(\epsilon))$ and $\text{VC}(\text{dual}(\mathcal{H})) = \text{Ldim}(\text{VC}(\mathcal{H}_\infty(\epsilon)))$.

Proof. (1) The first equation is Lemma 3.5.2. The second is also Lemma 3.5.2, with the extra observation that a hypothesis class $(X, \mathcal{G})$ shatters a subset $\{x_0, \dots, x_{s-1}\}$ from X if and only if we can find a special tree of height s with internal nodes from X and leaves from $\mathcal{G}$ where, for each $t < s$, every node at level t is equal to x_t .

(2) As in (1) using Lemmas 3.5.3 and 3.5.4.

(3) By monotonicity, $\text{Ldim}(\mathcal{H}) \leq \text{Ldim}(\mathcal{H}_\infty(\epsilon))$. Let $\ell = \text{Ldim}(\mathcal{H})$. By the choice of ϵ and Lemma 3.5.2, if $\text{Ldim}(\mathcal{H}_\infty(\epsilon)) \geq \ell + 1$ then also $\text{Ldim}(\mathcal{H}) \geq \ell + 1$, contradiction.

(4) As in (3) with VC in place of Ldim.

(5) By monotonicity.

(6) We know that $\text{Ldim}(\mathcal{H}) \geq \text{VC}(\mathcal{H})$ so the results on Ldim and VC of $\mathcal{H}_\infty(\epsilon)$ and its dual follow from the earlier statements. The minor new point is that a small ϵ in terms of $\text{Ldim}(\mathcal{H})$ helps with the threshold dimension also. Let $k = \text{Thr}(\mathcal{H})$. Taking the contrapositive of Conclusion 3.3.17(1) in the case $m = \ell + 1$ we see that $k < 2^{\ell+2}$ and so the contrapositive of Lemma 3.5.5 applies.

This completes the proof. □

Note that these results could have been stated in the language of k -realizability, which has other advantages, but then they would not as obviously extend to graphs.

Before continuing, the reader may wish to look at the section of motivating examples, Section 3.11 below.

3.6 Characterization of ϵ -saturation for Littlestone classes

Definition 3.6.1. *Call a hypothesis class $\mathbf{H}$ ϵ -saturated if every ϵ -type of $\mathbf{H}$ is realized in $\mathbf{H}$.*

The aim of this section is to prove:

Theorem 3.6.2 ($\mathcal{H}_\infty(\epsilon)$ is ϵ -saturated). *Suppose $\mathcal{H}$ and $\epsilon < 1/2$ are given. Suppose $\mathcal{H}_\infty(\epsilon)$ is a Littlestone class (hence also $\mathcal{H}$ is a Littlestone class). Then every ϵ -type of $\mathcal{H}_\infty(\epsilon)$ is realized in $\mathcal{H}_\infty(\epsilon)$, i.e. $\mathcal{H}_\infty(\epsilon)$ is ϵ -saturated.*

We will invoke the Littlestone Minimax Theorem, recently proved by Hanneke, Livni and Moran, which we state here in the language of the present chapter.¹¹

Theorem A (Littlestone Minimax Theorem; Hanneke, Livni, Moran [85]). *Suppose $\mathbf{H}$ is a Littlestone class and $\epsilon_L < 1/2$. Let $g : X \rightarrow \{0, 1\}$ be any function. If (A) then (B).*

(A) *For every set $\{(x_t, w_t) : t < s\}$ where $\{x_t : t < s\} \subseteq X$, $0 < w_t \leq 1$ for each $t < s$, and $\sum_t w_t = 1$, there is $h \in \mathbf{H}$ so that*

$$\sum \{w_t : h(x_t) \neq g(x_t), t < s\} < \epsilon_L.$$

Informally, “for every finitely supported probability measure on X there is some $h \in \mathbf{H}$ whose error against g is $< \epsilon_L$.”

(B) *For every ϵ_M with $\epsilon_L < \epsilon_M < 1/2$, there exists a finite weighted subset $\{(h_i, \gamma_i) : i < m\}$ of $\mathbf{H}$ such that: for every $x \in X$,*

$$\sum \{\gamma_i : h_i(x) = g(x)\} \geq 1 - \epsilon_M.$$

11. See [85, Corollary 7], where finite support is explicitly mentioned; also relevant for us are Lemma 11 and Lemma 12 of that paper.

In our language, “there is an ϵ_M -hypothesis H of $\mathbf{H}$ representing g .”

We work towards Theorem 3.6.2.

Claim 3.6.3. *Assume $\mathcal{H}_\infty(\epsilon)$ is Littlestone. For any $n < \omega$, if $g : X \rightarrow \{0, 1\}$ is finitely realized in $\mathcal{H}_n(\epsilon)$, then $g \in \mathcal{H}_{n+1}(\epsilon)$.*

Proof. In this case we can satisfy condition (A) of Littlestone Minimax for any $\epsilon_L > 0$, simply by choosing $h \in \mathcal{H}_n(\epsilon)$ to be the function which agrees with g on $\{x_i : i < s\}$ for some $s < \omega$. In particular, we can take $\epsilon_M = \epsilon$. Let H be the ϵ -hypothesis given by condition (B) where the h_i s belong to $\mathcal{H}_n(\epsilon)$. Then $\mathbf{t}_H$ belongs to $\mathcal{H}_{n+1}(\epsilon)$, or in other words, $g \in \mathcal{H}_{n+1}(\epsilon)$. $\square$

Claim 3.6.4. *Assume $\mathcal{H}_\infty(\epsilon)$ is Littlestone. For any $n < \omega$, if $\mathbf{t} : X \rightarrow \{0, 1\}$ is an ϵ -type of $\mathcal{H}_n(\epsilon)$, then $\mathbf{t}$ is realized in $\mathcal{H}_{n+2}(\epsilon)$.*

Proof. By construction, $\mathbf{t}$ will be finitely realized in $\mathcal{H}_{n+1}(\epsilon)$, hence realized in $\mathcal{H}_{n+2}(\epsilon)$ by Claim 3.6.3. $\square$

Claim 3.6.5. *If H is an ϵ -hypothesis of $\mathcal{H}_\infty(\epsilon)$, then $\mathbf{t}_H \in \mathcal{H}_\infty(\epsilon)$.*

Proof. Suppose $H = \{(h_i, \gamma_i) : i < n\}$. Since it only involves finitely many h_i 's, let $m < \omega$ be such that $\{h_i : i < n\} \subseteq \mathcal{H}_m(\epsilon)$. Clearly H remains an ϵ -hypothesis of $\mathcal{H}_m$, hence $\mathbf{t}_H \in \mathcal{H}_{m+1}(\epsilon) \subseteq \mathcal{H}_\infty(\epsilon)$. $\square$

Claim 3.6.6. *Assume $\mathcal{H}_\infty(\epsilon)$ is Littlestone. If $\mathbf{t}$ is an ϵ -type of $\mathcal{H}_\infty(\epsilon)$, then $\mathbf{t}$ is finitely realized in $\mathcal{H}_\infty(\epsilon)$, hence realized in $\mathcal{H}_\infty(\epsilon)$.*

Proof. By definition of ϵ -type, $\mathbf{t}$ is finitely realized by ϵ -hypotheses in $\mathcal{H}_\infty(\epsilon)$. So by Claim 3.6.5 it is finitely realized in $\mathcal{H}_\infty(\epsilon)$. In general, let $g : \mathcal{H}_\infty(\epsilon) \rightarrow \{0, 1\}$ be any function which is finitely realized in $\mathcal{H}_\infty(\epsilon)$. Then Littlestone Minimax (A) is always satisfied (for any $0 < \epsilon_L < \epsilon_M$, in particular for $\epsilon_M = \epsilon$, since in condition (A) we can find h which agrees exactly with g on the given finite domain, hence has zero error whatever the weights). So there is an ϵ -hypothesis H of $\mathcal{H}_\infty(\epsilon)$ so that $\mathbf{t}_H$ represents g . By Claim 3.6.5, $\mathbf{t}_H \in \mathcal{H}_\infty(\epsilon)$. In particular, $\mathbf{t}$ is realized. $\square$

Proof of Theorem 3.6.2. Follows from Claim 3.6.6. □

In the next lemma, the intent is that all the classes are over the same set X .

Lemma 3.6.7. *Suppose $\mathcal{H}_\infty(\epsilon)$ is Littlestone. Suppose $\mathcal{H} \subseteq \mathbf{H}$. If $\mathbf{H}$ is ϵ -saturated then $\mathcal{H}_\infty(\epsilon) \subseteq \mathbf{H}$. Hence $\mathcal{H}_\infty(\epsilon)$ is the smallest (under inclusion) ϵ -saturated hypothesis class containing $\mathcal{H}$.¹²*

Proof. By induction on $n < \omega$ we build an increasing sequence of functions $\langle F_n : n < \omega \rangle$ such that:

- (a) for $n < \omega$, F_n is an injective map from $\mathcal{H}_n(\epsilon)$ into $\mathbf{H}$,
- (b) for $n < m < \omega$, $F_m \upharpoonright \mathcal{H}_n(\epsilon) = F_n$, and
- (c) for $n < \omega$, for every $x \in X$ and $h \in \text{dom}(F_n)$, $h(x) = 1$ if and only if $(F_n(h))(x) = 1$.

[Since we identify two functions which are the same on X , condition (c) amounts to saying the map will be the inclusion map.] For $n = 0$, let F_0 act identically as $\mathcal{H} \subseteq \mathbf{H}$. Consider $n + 1$. Enumerate the elements of $\mathcal{H}_{n+1}(\epsilon) \setminus \mathcal{H}_n(\epsilon)$ by the appropriate cardinal α as $\langle h_\beta : \beta < \alpha \rangle$. The functions h_β in this list are necessarily distinct functions, and, by definition of $\mathcal{H}_{n+1}(\epsilon)$, they are distinct from the elements of $\mathcal{H}_n(\epsilon)$. It suffices to specify F_{n+1} on each h_β individually, so fix h_β . By definition of $\mathcal{H}_{n+1}(\epsilon)$, we have $h_\beta = \mathbf{t}_H$ for some finite weighted ϵ -good set $H = \{(g_i, \gamma_i) : i < k\}$ where the g_i belong to $\mathcal{H}_n(\epsilon)$, but $\mathbf{t}_H \notin \mathcal{H}_n$. By condition (c), $Y = \{(F_n(g_i), \gamma_i) : i < k\}$ is also a weighted ϵ -good set, where the $F_n(g_i)$ belong to $\mathbf{H}$, and Y represents the same function on X as H . So $g = \mathbf{t}_H = \mathbf{t}_Y$ is realized in the class $\{\mathbf{t}_Y : Y \text{ is an } \epsilon\text{-hypothesis of } \mathbf{H}\}$, hence it is a fortiori finitely realized in this class, thus g is itself a type of $\mathbf{H}$, hence it is realized by some (and indeed exactly one) $f \in \mathbf{H}$ by the assumption of saturation. Note that $f \notin \text{range}(F_n)$ because we assumed that $\mathbf{t}_H \notin \mathcal{H}_n$ and F_n satisfied condition (c). Set $F_{n+1}(h_\beta) = f$. □

12. There can of course be larger ϵ -saturated classes containing a given $\mathcal{H}$, as we can increase the class before taking the saturation.

3.7 Intermediate ϵ

We continue with the assumption that $\mathcal{H}$ is a Littlestone class, of Littlestone dimension ℓ and VC dimension d . In Theorem 3.5.6 we saw that if ϵ is sufficiently small as a function of d , then the VC dimension of $\mathcal{H}_\infty$ does not change, and if ϵ is sufficiently small as a function of ℓ , then the Littlestone dimension of $\mathcal{H}_\infty$ also does not change. In this section we prove the a priori quite surprising theorem that if ϵ is bounded as a function of d (here we do not refer to ℓ) then $\mathcal{H}_\infty$ remains a Littlestone class possibly of larger Ldim. An upper bound on this larger Ldim can be obtained from the proof as we will discuss at the end of the section.

In the next theorem, for definiteness we could have said “For any $C \geq 28$ ” but we give an approximate value, whose calculation is justified in 3.7.8.

Theorem 3.7.1. *There exists a constant $C \approx 27.7259$ that satisfies the following. Let $\mathcal{H}$ be a Littlestone class, with $\text{Ldim}(\mathcal{H}) = \ell$ and $\text{VC}(\mathcal{H}) = d$. If $0 < \epsilon < \frac{1}{Cd}$, then $\mathcal{H}_\infty(\epsilon)$ is Littlestone.*

More precisely: given d , we are free to choose:

1. any constant $\mathbf{C} \geq C$
2. based on $\mathbf{C}$, any $\epsilon_1 \in (0, 1/2)_{\mathbb{R}}$ such that $\mathbf{C} > 20 \cdot (1/\epsilon_1) \cdot \ln(1/\epsilon_1)$
3. based on ϵ_1 , any $\epsilon_2 \in (0, 1/2)_{\mathbb{R}}$ such that $\epsilon_1 < \epsilon_2$
4. based on ϵ_2 , any $\epsilon_3 \in (0, 1/2 - \epsilon_2)_{\mathbb{R}}$

Then, the Littlestone dimension of the class $\mathcal{H}_\infty(\epsilon)$ is upper bounded in terms of $d/(\epsilon_3)^2$ and the original Littlestone dimension of $\mathcal{H}$, i.e. ℓ .

In the statement, “20” in item (2) is an upper bound for the constant in the ϵ -net theorem, and the function bounding the final Littlestone dimension depends on the first part of the proof of Theorem 3 in [10] (bounding Littlestone dimension of fixed finite boolean combinations of elements of a Littlestone class) as discussed at the end of the section.

Before giving the proof, we need to prove several claims. First, we show that the functions in $\mathcal{H}_\infty(\epsilon)$ are all k -realizable, as long as k is not too large in terms of ϵ . (Compare Theorem 3.5.6 and Example 3.12.4 below.)

Claim 3.7.2. *Let $k < \lfloor 1/\epsilon \rfloor$. Then $\mathcal{H}_\infty(\epsilon) \subseteq \mathcal{H}^{k\text{-real}}$.*

Proof. The claim asserts that any $h \in \mathcal{H}_\infty(\epsilon)$ is already k -realizable in $\mathcal{H}$. If $h \in \mathcal{H}_\infty(\epsilon)$ then we can choose a minimal m so that $h \in \mathcal{H}_m(\epsilon)$. Clearly h is k -realizable in $\mathcal{H}_m$, so if $m = 0$ we are done. Otherwise, suppose for a contradiction that $m = n + 1$ and h is k -realizable in $\mathcal{H}_{n+1}$ but not in $\mathcal{H}_n$.

As $h \in \mathcal{H}_{n+1}$ there is a weighted ϵ -good set $H = \{(g_i, \gamma_i) : i < s\}$ which represents h , hence for each $x \in X$, $\sum \{\gamma_i : i < s, g_i(x) \neq h(x)\} < \epsilon$. Hence given any $x_0, \dots, x_{k-1}$ from X , we have that $\sum \{\gamma_i : i < s, j < k, g_i(x_j) \neq h(x_j)\} < \epsilon k \leq 1$ (by choice of ϵ). So there exists $(g, \gamma) \in H$ such that $g(x_j) = h(x_j)$ for $j = 0, \dots, k-1$. By definition $g \in \mathcal{H}_n$. This shows h is k -realizable in $\mathcal{H}_n$, contradicting our choice of n . $\square$

In the following claim, we fix C and ϵ from Theorem 3.7.1. Note that this background ϵ is not used in this claim.

The claim analyzes a single function f from $\mathcal{H}^{k\text{-real}}$ in the case where $k = Cd$. Recall that a priori we do not know the complexity of $\mathcal{H}^{Cd\text{-real}}$, but we do know that $\mathcal{H}$ is Littlestone and indeed of VC dimension d . Let $\mathcal{H}_f$ denote the class of symmetric differences of f with each element of $\mathcal{H}$, which remains a VC class of dimension d . Given any measure (we will only use finitely supported probability measures) μ on X and $h_f \in \mathcal{H}$, let $\mu(h_f)$ mean $\mu(\{x \in X : h_f(x) = 1\})$. There are two key ideas. First, since $\mathcal{H}_f$ is a VC class, the ϵ -net theorem [note: we should call it the ϵ_1 -net theorem since we are applying it to an ϵ_1 we shall choose presently and not our background ϵ] applies: for some $M = M(\epsilon_1, d)$, for any measure μ on X there is $S \subseteq X$, $|S| \leq M$ which “sees” all nontrivial hypotheses (i.e. for every h_f of measure $\geq \epsilon_1$ there exists $s \in S$ so that $h_f(s) = 1$). Since $\mathcal{H}_f$ measures symmetric difference with f , this amounts to saying that if $h \in \mathcal{H}$ differs from f by more than ϵ_1 , the set S sees it.

Second, note this along with Claim 3.7.2 tells us that for any choice of ϵ_1 satisfying Theorem 3.7.1(2), since $Cd > M(\epsilon_1, d)$, when μ is any finitely supported probability measure on X and S is an ϵ_1 -net, $Cd > |S|$ hence the assumption that f is Cd -realizable in $\mathcal{H}$ means that there is some $h \in \mathcal{H}$ so that h and f agree on S , hence the global difference between h and f is $< \epsilon_1$.

Claim 3.7.3. *Let $\mathcal{D}$ be a distribution over X with finite support, and let $\mathcal{H}$ be [a Littlestone class] of $\text{VC}(\mathcal{H}) = d$. Let C be as in Theorem 3.7.1. If $\epsilon_1 \in (0, 1/2)_{\mathbb{R}}$ satisfies the inequality $C > 20 \cdot (1/\epsilon_1) \cdot \ln(1/\epsilon_1)$ then for every $f \in \mathcal{H}^{Cd\text{-real}}$, there exists $h \in \mathcal{H}$ such that*

$$\Pr_{x \sim \mathcal{D}}[h(x) \neq f(x)] < \epsilon_1.$$

Proof. Define $\mathcal{H}_f := \{h_f : h \in \mathcal{H}\}$, where for each $h \in \mathcal{H}$ and $x \in X$ we define $h_f(x) := \delta_{f(x) \neq h(x)}$, as the symmetric difference of h and f . Since the map $h \mapsto h_f$ is an involution, shattered sets are the same for both classes, and $\text{VC}(\mathcal{H}) = \text{VC}(\mathcal{H}_f) = d$.

Then, by ϵ -net Theorem on $\mathcal{H}_f$ (see e.g. Theorem 10.2.4 in [113]), there exists a constant $C_{\text{net}} < 20$, and a number $M := m(d, \epsilon_1) \leq C_{\text{net}} \cdot d \cdot (1/\epsilon_1) \cdot \ln(1/\epsilon_1)$ such that a sample $\mathcal{S}$ of size M from $X \sim \mathcal{D}$, satisfies with positive probability that:

$$\forall h_f \in \mathcal{H}_f, \quad \Pr_{x \sim \mathcal{D}}[h_f(x) = 1] \geq \epsilon_1 \Rightarrow \exists s \in \mathcal{S} : h_f(s) = 1.$$

Following the correspondence between $\mathcal{H}$ and $\mathcal{H}_f$, the above is equivalent to

$$\forall h \in \mathcal{H}, \quad \Pr_{x \sim \mathcal{D}}[h(x) \neq f(x)] \geq \epsilon_1 \Rightarrow \exists s \in \mathcal{S} : h(s) \neq f(s).$$

The condition on ϵ_1 implies $M \leq Cd$. Let $\mathcal{S}$ be a sample with the above property. Since $|\mathcal{S}| = M \leq Cd$, then $f \in \mathcal{H}^{Cd\text{-real}}$ implies there is some $h \in \mathcal{H}$ such that for all $x \in \mathcal{S}$, $f(x) = h(x)$. By contrapositive, then, $\Pr_{x \sim \mathcal{D}}[h(x) \neq f(x)] < \epsilon_1$. $\square$

We continue to analyze a given $f \in \mathcal{H}^{Cd\text{-real}}$. The next claim applies Littlestone Minimax

to conclude that any such f can be represented by a weighted ϵ_2 -majority vote of elements of $\mathcal{H}$, for any $\epsilon_2 \in (\epsilon_1, 1/2)_{\mathbb{R}}$. Why? Condition (A) of the Littlestone Minimax, Theorem A above, is exactly what is guaranteed by Claim 3.7.3.

Claim 3.7.4. *Let $\mathcal{D}, \mathcal{H}, d, C, \epsilon_1$ be defined as before. In particular, recall that $\mathcal{H}$ is a Littlestone class with $\text{VC}(\mathcal{H}) = d$. Fix $f \in \mathcal{H}^{Cd\text{-real}}$. For any $\epsilon_2 \in (0, 1/2)$ such that $\epsilon_2 > \epsilon_1$, there exists a finite weighted sequence $\{(h_i, \gamma_i) : i < n\}$ in $\mathcal{H}$ such that, for every $x \in X$:*

(i) *there is an ϵ_2 -majority vote $\mathbf{t}(x) \in \{0, 1\}$, i.e., a value for which*

$$\sum \{\gamma_i : h_i(x) = 1 - \mathbf{t}(x), i < n\} < \epsilon_2;$$

(ii) $\mathbf{t}(x) = f(x)$.

Proof. Suppose we are given $\{(x_t, w_t) : t < s\}$ as in condition (A) of Littlestone Minimax, Theorem A. By Claim 3.7.3 there is some $h \in \mathcal{H}$ whose weighted error against f is $< \epsilon_1$. So by condition (B) of Littlestone Minimax there is an ϵ_2 -hypothesis $H = \{h_i, \gamma_i : i < n\}$ of H so that $\mathbf{t}_H = f$. $\square$

We have seen that our $f \in \mathcal{H}^{Cd\text{-real}}$ is expressible as a weighted ϵ_2 -majority vote of some $H \subseteq \mathcal{H}^{Cd\text{-real}}$. H is finite, but a priori we have no bound on its size. Our last claim shows that we may assume a uniform finite bound on the size of H , as a function of d and any fixed $\epsilon_3 \in (0, 1/2 - \epsilon_2)_{\mathbb{R}}$. (Why? Use the fact that the dual of any VC class is also a VC class, take the finitely supported measure on $\mathcal{H}$ corresponding to the weighted ϵ_2 -good set H , and choose a representative sample from it of size about $d/(\epsilon_3)^2$, as the proof describes.)

Definition 3.7.5. *Given a hypothesis class $\mathcal{H}$ and an odd $p \in \mathbb{N}$, let $\langle g_j : j < p \rangle$ be a sequence of elements of $\mathcal{H}$, possibly with repetition. The majority vote of this sequence is defined to be the function $g : X \rightarrow \{0, 1\}$ given by: $g(x) = 1$ if $|\{j < p : g_j(x) = 1\}| > (p - 1)/2$ and $g(x) = 0$ otherwise. Let*

$$\text{Maj}_p(\mathcal{H})$$

be the set of functions arising as majority votes of sequences $\langle g_j : j < p \rangle$ in the p -fold Cartesian power $\mathcal{H} \times \cdots \times \mathcal{H}$.

In particular, the majority in Definition 3.7.5 is the counting majority and is not directly related to a background epsilon.

Claim 3.7.6. *Let $\mathcal{D}, \mathcal{H}, d, C, \epsilon_2$ be defined as before. For some odd $p \in \mathbb{N}$ upper bounded $p \leq c_* + 1$, where $c_* = c_*(\epsilon_3, d_*)$ is the constant obtained from the convergence property, for every $f \in \mathcal{H}^{Cd\text{-real}}$, there exists a sequence $\langle f_i : i < p \rangle$ of elements of $\mathcal{H}$, possibly with repetition, whose majority vote is f . In other words, $\mathcal{H}^{Cd\text{-real}} \subseteq \text{Maj}_p(\mathcal{H})$.*

Proof. Let $f \in \mathcal{H}^{Cd\text{-real}}$ be given, and let $H = \{(h_i, \gamma_i) : i < n\}$ be the corresponding ϵ_2 -hypothesis given by Claim 3.7.4. Recall that $\mathcal{H}_* := \text{dual}(\mathcal{H})$ (see Definition 3.3.16) is also a VC class, of VC dimension $d_* \leq 2^{d+1} - 1$. So it satisfies uniform convergence: namely, there is a constant $c_* = c_*(\epsilon_3, d_*)$ and a sample size $m_* = m_*(\epsilon_3, d_*) \leq c_*$ with the following property. For any probability measure on $\mathcal{H}$ [recall that in the dual $\mathcal{H}_*$, the domain is $\mathcal{H}$], and in particular for that corresponding to $H = \{(h_i, \gamma_i) : i < m\}$, for any $p \geq m_*$ [for definiteness, for us, let p be the smallest odd number $\geq m_*$], with high probability an i.i.d. sample $\langle g_t : t < p \rangle$ satisfies: for every $x \in X$, the “empirical and actual measures are close,” meaning that

$$\left| \left(\frac{|\{t < p : g_t(x) = 1\}|}{|p|} \right) - \left(\sum \{\gamma_i : i < n, h_i(x) = 1\} \right) \right| < \epsilon_3.$$

(Informally, each $x \in \mathcal{H}_*$ determines the set of $h \in \mathcal{H}$ which label it by 1; the size of that set according to H should be close to the size of that set according to the sample, which is of course sub-sampled from $\{h_i : i < n\}$ by our choice of measure.)

Since this happens with high probability, in particular we can fix *some* sequence $\langle g_t : t < p \rangle$ on which this happens. Now $\sum \{\gamma_i : i < n, h_i(x) = 1\}$ is either $< \epsilon_2$ or $\geq 1 - \epsilon_2$, so $|\{t < p : g_t(x) = 1\}|/|p|$ is either $< \epsilon_2 + \epsilon_3$ or $\geq 1 - \epsilon_2 - \epsilon_3$. In other words, since

$(\epsilon_2 + \epsilon_3) < 1/2$, we have that for every x , $\langle g_t(x) : t < p \rangle$ has a numerical majority which agrees with H and hence represents f . $\square$

Now we can conclude by applying known bounds on the Littlestone dimension of fixed finite boolean combinations of elements of a Littlestone class. We will use:

Fact 3.7.7 ([10], Theorem 3). *If $\mathcal{H}$ is Littlestone then for any fixed odd $p \in \mathbb{N}$, $\text{Maj}_p(\mathcal{H})$ is also Littlestone. Concretely, we have the following upper bound:*

$$\text{Ldim}(\text{Maj}_p(\mathcal{H})) \leq \tilde{O}(p^2 \cdot \text{Ldim}(\mathcal{H})).$$

The proof of this fact uses a technique named Online boosting. In model theoretic language, the familiar fact is that if φ is a stable formula, then any fixed finite boolean combination of φ 's is also stable, though of a possibly larger rank.

We now formally assemble the pieces to prove Theorem 3.7.1:

Proof of Theorem 3.7.1. By Observation 3.7.2, regardless of what our constant C is, $\epsilon < 1/(Cd)$ implies that $\mathcal{H}_\infty(\epsilon) \subseteq \mathcal{H}^{Cd\text{-real}}$. Also, as long as $Cd \geq d + 1$, Theorem 3.5.6(4), $\text{VC}(\mathcal{H}_\infty(\epsilon)) = \text{VC}(\mathcal{H}) = d < \infty$.

For the rest of the proof, fix C satisfying the statement of the theorem (recall $C \approx 27.7259$) and choose some $\epsilon_1 > 0$ satisfying $C > 20 \cdot (1/\epsilon_1) \cdot \ln(1/\epsilon_1)$. Then Claim 3.7.3 will allow us to find, for every $f \in \mathcal{H}^{Cd\text{-real}}$, some $h \in \mathcal{H}$ with error less than ϵ_1 with respect to f , by realizability and the ϵ -net property. For any $\epsilon_2 > \epsilon_1$, Claim 3.7.4 yields an ϵ_2 -hypothesis H on $\mathcal{H}$ such that $\mathbf{t}_H$ equals $f(x)$, for all $x \in X$. Then, Claim 3.7.6 allows us to find a uniform size for these representations: it shows there is $p \in \mathbb{N}$ so that, sampling p times from the distribution given by H , and then taking the p -wise majority vote, we obtain $f(x)$ for all $x \in X$.

We have shown that under the assumptions of the theorem

$$\mathcal{H}_\infty(\epsilon) \subseteq \mathcal{H}^{Cd\text{-real}} \subseteq \text{Maj}_p(\mathcal{H})$$

Concluding by Fact 3.7.7, $\mathcal{H}_\infty(\epsilon)$ is Littlestone since $\text{Maj}_p(\mathcal{H})$ is. □

Remark 3.7.8. *The choice $C \approx 27.7259$ comes from the following considerations:*

- *The decreasing and injective behavior of $f(x) = \frac{1}{x} \cdot \ln(\frac{1}{x})$ over $(0, 1/2)_{\mathbb{R}}$.*
- *The argument in Claim 3.7.3 needs the inequality $Cd > d \cdot f(\epsilon_1)$ to hold*
- *Continuity of $f(x)$ at $1/2$, and $27.7259 \approx 40 \ln(2) = \lim_{x \rightarrow 1/2^-} f(x)$.*

3.8 Conclusions for Littlestone classes

In this section, we summarize the main results of this chapter so far into the following theorem, also visible in Figure 3.1.

Theorem 3.8.1. *Let $(X, \mathcal{H})$ be a hypothesis class such that $\text{Ldim}(\mathcal{H}) = \ell$ and $\text{VC}(\mathcal{H}) = d$.*

The saturations of these classes satisfy the following:

- (a) *When $\epsilon \leq 1/(\ell + 1)$, $\text{Ldim}(\mathcal{H}_\infty(\epsilon)) = \text{Ldim}(\mathcal{H}) = \ell$;*
- (b) *When $\epsilon \leq 1/(d + 1)$, $\text{VC}(\mathcal{H}_\infty(\epsilon)) = \text{VC}(\mathcal{H}) = d$.*
- (c) *For $\epsilon < 1/(Cd)$, for some $C \gtrapprox 27.7259$, $\text{Ldim}(\mathcal{H}_\infty(\epsilon)) < \infty$ although it need not stay the same as $\text{Ldim}(\mathcal{H})$;*
- (d) *For $\epsilon > 1/(d + 1)$, it is possible for both $\text{Ldim}(\mathcal{H}_\infty)$ and $\text{VC}(\mathcal{H}_\infty)$ to be infinite.*

Proof. Parts (a) and (b) are both shown in Theorem 3.5.6. Part (c) is shown in Theorem 3.7.1. Part (d) is shown by Example 3.12.1. □

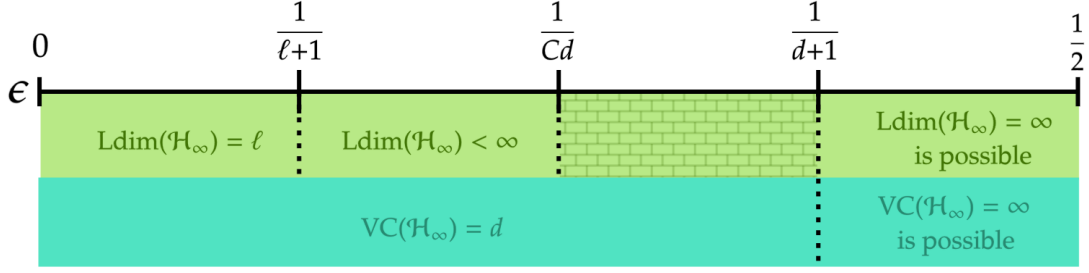


Figure 3.1: Regimes of $\epsilon \in (0, 1/2)_{\mathbb{R}}$ and the corresponding statements of what we showed about $\text{Ldim}(\mathcal{H}_{\infty})$ and $\text{VC}(\mathcal{H}_{\infty})$, given that $\text{Ldim}(\mathcal{H}) = \ell$ and $\text{VC}(\mathcal{H}) = d$. Here C is from Theorem 3.7.1.

3.9 Definition of ϵ -saturation for $\mathcal{G}$

We now give the definitions in the case of graphs. A crucial role will be played by excellent sets. Remember these can be different from good sets; see Section 3.12.2.

Convention 3.9.1. *When dealing with graphs we will always assume $\epsilon < 1/4$.*

Convention 3.9.2. $\mathcal{G} = (V, E)$, which we may abbreviate $\mathcal{G}$, will usually denote a stable graph with:

- *finite edge stability, also called threshold dimension, s ,*
- *hence finite Littlestone dimension ℓ ,*
- *hence finite VC dimension d .*

The vertex set V can be finite, but need not be. We allow self-loops, that is, we allow (v, v) as an edge for any vertex v .¹³

As before, we will repeat the assumption of stability in the main definitions and theorems which use it. We will often write “ $v \in \mathcal{G}$ ” to mean $v \in V(\mathcal{G})$, and “ $v \sim w$ ” to mean $(v, w) \in E(\mathcal{G})$.

¹³. Briefly, this is because we will add ϵ -excellent sets of vertices as new “virtual” vertices and any such set A will have a well-defined edge or nonedge to any other such set, including itself. There is a priori no reason to erase the information of which of the two it is.

Compared to the definition of ϵ -hypothesis, the definition of ϵ -vertex will require an extra step due to the fact that we would like edges between ϵ -vertices to be well defined (this corresponds to ϵ -excellent in the stable regularity lemma). As before, we will use weighted good and excellent sets.

Definition 3.9.3. *Given ϵ and $\mathcal{G}$, a weighted ϵ -good set of $\mathcal{G}$ is a finite set of pairs:*

$$A = \{(v_i, \gamma_i) : i < k\}$$

where $k \in \mathbb{N}$, and:

1. $\{v_i : i < k\} \subseteq \mathcal{G}$, and $\langle v_i : i < k \rangle$ is without repetition.
2. each $\gamma_i \in (0, 1]_{\mathbb{R}}$, and $\sum \{\gamma_i : i < k\} = 1$.
3. (ϵ -goodness) For each $v \in V$, either

$$\sum \{\gamma_i : v_i \sim v\} \geq 1 - \epsilon \text{ or } \sum \{\gamma_i : v_i \not\sim v\} \geq 1 - \epsilon.$$

In the first case, write $\mathbf{t}(A, v) = 1$, and in the second $\mathbf{t}(A, v) = 0$.

Definition 3.9.4. *Given ϵ and $\mathcal{G}$, an ϵ -vertex (or a weighted ϵ -excellent set) of $\mathcal{G}$ is a finite set of pairs:*

$$B = \{(v_i, \gamma_i) : i < k\}$$

where $k \in \mathbb{N}$, B is a weighted ϵ -good set, and moreover:

- (4) (ϵ -excellence) For every weighted ϵ -good set A , either

$$\sum \{\gamma_i : \mathbf{t}(A, v_i) = 1\} \geq 1 - \epsilon \text{ or } \sum \{\gamma_i : \mathbf{t}(A, v_i) = 0\} \geq 1 - \epsilon.$$

In the first case, write $\mathbf{t}(B, A) = 1$, and in the second $\mathbf{t}(B, A) = 0$.

That Definition 3.9.4 is sufficiently symmetric for our purposes is justified by the following (which also explains our convention that in working with graphs, $\epsilon < 1/4 < 2 - \sqrt{3}$).

Remark 3.9.5. *Let $\epsilon \in (0, 2 - \sqrt{3})$, and let A and B be two ϵ -excellent weighted sets of G . Then, the values $\mathbf{t}(A, B)$ and $\mathbf{t}(B, A)$ are well-defined and equal.*

Proof. Suppose for a contradiction that $\mathbf{t}(A, B) = 1$ and $\mathbf{t}(B, A) = 0$. This implies that, if the weight for each a_i is γ_i and the weight for each b_i is ρ_i , then:

$$\sum \{\gamma_i : \mathbf{t}(a_i, B) = 1\} \geq 1 - \epsilon, \quad \sum \{\rho_i : \mathbf{t}(A, b_i) = 1\} \leq \epsilon.$$

The first bound above implies that the total sum of the products $\gamma_i \rho_j$ over all edges from A to B is lower bounded by

$$\sum_{a_i \in A: \mathbf{t}(a_i, B)=1} \gamma_i \cdot (1 - \epsilon) + \sum_{a_i \in A: \mathbf{t}(a_i, B)=0} \gamma_i \cdot 0 \geq (1 - \epsilon)^2.$$

The second bound above implies that the same total sum of products of weights described above is upper bounded by

$$\sum_{b_i \in B: \mathbf{t}(A, b_i)=1} \rho_i \cdot 1 + \sum_{b_i \in B: \mathbf{t}(A, b_i)=0} \rho_i \cdot \epsilon \leq 2\epsilon.$$

The symmetry of edges in G and these two bounds imply that

$$(1 - \epsilon)^2 < \sum_{(a_i, b_j) \in E(A, B)} \gamma_i \rho_j < 2\epsilon.$$

$$\Rightarrow \epsilon \in (2 - \sqrt{3}, 2 + \sqrt{3})$$

These values for ϵ are ruled out by the hypothesis that $\epsilon < 2 - \sqrt{3}$. □

Definition 3.9.6. *Given ϵ and $\mathcal{G}$, whenever A, B are ϵ -vertices, say there is an ϵ -edge between A, B , in symbols $A \sim B$ or $A \sim_\epsilon B$, to mean that $\mathbf{t}(A, B) = \mathbf{t}(B, A) = 1$, and say*

there is an ϵ -nonedge, in symbols $A \not\sim B$ or $A \not\sim_\epsilon B$, to mean that $\mathbf{t}(A, B) = \mathbf{t}(B, A) = 0$.

Notation 3.9.7. Whenever A is an ϵ -vertex, let $\mathbf{t}_A$ be the function whose domain is the set of ϵ -vertices of $\mathcal{G}$ and whose range is $\{0, 1\}$, given by $B \mapsto \mathbf{t}(A, B) = \mathbf{t}(B, A)$.

Convention 3.9.8. Since for any $v \in V$, $\{v\}$ is an ϵ -vertex, we may drop parentheses, writing v for $\{v\}$ or w for $\{w\}$ in expressions like $\mathbf{t}$ or $\sim_\epsilon$.

To define the ϵ -saturation, we will formally extend our graphs by adding ϵ -vertices, and so all our edges are a priori ϵ -edges. Note, however, that in Definition 3.4.9 in some sense we constantly forget this: the graphs $\mathcal{G}_n$ at each stage in the construction are *actual graphs* (which come from formally adding ϵ -vertices as actual new vertices, then “rounding up” $\sim_\epsilon$ to $\sim$ and “rounding down” $\not\sim_\epsilon$ to $\not\sim$). Since there will often be many ϵ -vertices with the same $\sim_\epsilon$ -neighborhood, in order to avoid triviality, we only add one representative of each such neighborhood class.¹⁴ This is the analogue of not repeating functions in the case of $\mathcal{H}$ above.

Definition 3.9.9 (ϵ -types). Given $\mathcal{G} = (V, E)$ and a function $\mathbf{t} : V \rightarrow \{0, 1\}$:

1. $\mathbf{t}$ is k -realized in $\mathcal{G}$, for some given $k \in \mathbb{N}$, if for any $W \subseteq V$, $|W| \leq k$ there is $v \in V$ such that $\mathbf{t}_{\{v, 1\}} \upharpoonright W = \mathbf{t} \upharpoonright W$.
2. $\mathbf{t}$ is finitely realized in $\mathcal{G}$ if it is k -realized in $\mathcal{G}$ for all $k \in \mathbb{N}$.
3. Given also $\epsilon < 1/4$, an ϵ -type for $\mathcal{G}$ is a function $\mathbf{t} : V \rightarrow \{0, 1\}$ such that for any finite $W \subseteq V$, there is an ϵ -vertex A of $\mathcal{G}$ such that¹⁵

$$\mathbf{t}_A \upharpoonright W = \mathbf{t} \upharpoonright W.$$

The central definition of this chapter, in the case of graphs, is:

14. Indeed, since $\mathbf{t}(A, B) = \mathbf{t}(A', B')$ whenever A, A' and B, B' have the same $\sim_\epsilon$ -neighborhoods, when we will want to remember these arose from ϵ -vertices, we are free to calculate by choosing some possibly different representative of the class.

15. informally, we could say the function is finitely realized by $\{\mathbf{t}_A : A \text{ an } \epsilon\text{-vertex of } \mathcal{G}\}$.

Definition 3.9.10 (The ϵ -saturation of $\mathcal{G}$). Suppose we are given a graph $\mathcal{G} = (V, E)$ and $0 < \epsilon < 1/4$. Define by induction on $n \leq \omega$ a graph $\mathcal{G}_n(\epsilon) = (V_n, E_n)$:

1. $\mathcal{G}_0(\epsilon) = \mathcal{G} = (V, E)$.

2. $\mathcal{G}_{n+1}(\epsilon) = (V_{n+1}, E_{n+1})$ where:¹⁶

(a) $V_{n+1} = V_n \cup W$ where W is a representative set of ϵ -vertices of V_n meaning: for every ϵ -vertex A of V_n with $|A| > 1$, there is precisely one ϵ -vertex B of V_n such that $B \in W$ and $\mathbf{t}_B = \mathbf{t}_A$.

(b) E_{n+1} is given by:

i. for $(v, w) \in V_n \times V_n$, $(v, w) \in E_{n+1}$ iff $(v, w) \in E_n$;

ii. otherwise, for $(v, w) \in (V_{n+1} \times V_{n+1}) \setminus (V_n \times V_n)$,

$(v, w) \in E_{n+1}$ iff $v \sim_\epsilon w$.

3. $\mathcal{G}_\infty(\epsilon) := \mathcal{G}_\omega(\epsilon) := (V_\infty, E_\infty)$ where $V_\infty := V_\omega := \bigcup \{V_n : n < \omega\}$ and $E_\infty := E_\omega := \bigcup \{E_n : n < \omega\}$.

We call $\mathcal{G}_\infty(\epsilon)$ “the ϵ -saturation of $\mathcal{G}$.”

Here also the name, and the connection to ϵ -types, will be justified in due course. Also, we could have continued the definition by transfinite induction, but this is not used here.

3.10 Characterization of ϵ -saturation for stable graphs

The following definition and lemma have a natural analogue for hypothesis classes, which we do not state separately. Note that in the case of a hypothesis class $(X, \mathcal{H})$, the domain X did not change, we simply added new functions; whereas here the domain is a priori constantly changing, so things have a slightly different character.

16. Why do we not ask that for every ϵ -vertex A of V_n , there is precisely one ϵ -vertex B of V_n such that $B \in V_n \cup W$ and $\mathbf{t}_B = \mathbf{t}_A$? This would impose a retroactive restriction on G_n saying that there cannot be two singletons with the same neighborhood. There is a priori no reason this should be true of G_0 , that is, we do not require G_0 to be twin-free, even though we take care not to add instances of non-primalty later on.

Definition 3.10.1. A partial ϵ -type of a graph or hypothesis class means that we allow the function $\mathbf{t}$ to be partial.

Lemma 3.10.2 (Types can be extended). *Let $\mathbf{s}$ be a partial ϵ -type of a graph G . Then $\mathbf{s}$ can be extended to an ϵ -type, that is, there is an ϵ -type $\mathbf{t} \supseteq \mathbf{s}$. In fact, $\mathbf{s}$ can be extended to $V \cup \{B : B \text{ is a weighted } \epsilon\text{-good subset of } G\}$.*

Proof. It will be notationally convenient to identify functions with their graphs. Enumerate the elements of

$$V \cup \{B : B \text{ is a weighted } \epsilon\text{-good subset of } G\} \setminus \text{dom}(\mathbf{s})$$

by the cardinal $\kappa \geq |G|$) as

$$\langle c_\alpha : \alpha < \kappa \rangle.$$

Let us show by induction on $\alpha < \kappa$ that we can build an increasing continuous chain of partial ϵ -types $\mathbf{t}_\alpha$ such that:

1. $\mathbf{t}_0 = \mathbf{s}$,
2. each $\mathbf{t}_\alpha$ is an ϵ -type, i.e., finitely realized by ϵ -vertices,
3. $\beta < \alpha \implies \mathbf{t}_\beta \subseteq \mathbf{t}_\alpha$,
4. if α is a limit ordinal then $\mathbf{t}_\alpha = \bigcup \{\mathbf{t}_\beta : \beta < \alpha\}$,
5. if $\alpha = \beta + 1$ then $c_\beta \in \text{dom}(\mathbf{t}_\alpha)$.

For $\alpha = 0$ and α limit this is easy as being finitely realized by ϵ -vertices is preserved under unions of chains.

Let $\alpha = \beta + 1$. If $c_\beta \in \text{dom}(\mathbf{t}_\beta)$, set $\mathbf{t}_\alpha = \mathbf{t}_\beta$ and we are done. Otherwise, we ask if $\mathbf{t}_\beta \cup \{(c_\beta, 1)\}$ is finitely realizable by ϵ -vertices. If not, there is a finite $\mathbf{t}' \subseteq \mathbf{t}_\beta$ so that $\mathbf{t}' \cup \{(c_\beta, 1)\}$ is not realized by any ϵ -vertex of $\mathcal{G}$. Likewise if $\mathbf{t}_\beta \cup \{(c_\beta, 0)\}$ is not finitely

realizable by ϵ -vertices, there is a finite $\mathbf{t}'' \subseteq \mathbf{t}_\beta$ such that $\mathbf{t}'' \cup \{(c_\beta, 0)\}$ is not finitely realizable by ϵ -vertices. But then $\mathbf{t}' \cup \mathbf{t}''$ cannot be finitely realized by ϵ -vertices, because if it were, say by A , this same A would have to take some value on c_β which would contradict one of our two assumptions. Since $\mathbf{t}_\beta$ is inductively assumed to be finitely realized by ϵ -vertices, one of the two additions must be possible, completing this step and hence the proof. $\square$

Theorem 3.10.3. *Let $\mathcal{G}$ be a k -stable graph, $\epsilon < 1/4$ and $\mathcal{G}_\infty(\epsilon)$ its saturation. Suppose $\mathcal{G}_\infty(\epsilon)$ is also Littlestone. Then $\mathcal{G}_\infty(\epsilon)$ realizes all of its ϵ -types.*

The proof will depend on several claims, slightly different than Theorem 3.6.2.

Claim 3.10.4. *If $V = \{(v_i, \gamma_i) : i < m\}$ is an ϵ -vertex of $G_\infty(\epsilon)$ then it is also an ϵ -vertex of $G_n(\epsilon)$ for some $n < \omega$.*

Proof. Let $n < \omega$ be such that $\{v_i : i < m\} \subseteq V_n$. So V is a weighted set of elements of $G_n(\epsilon)$. Considered as a weighted set of elements of $G_\infty(\epsilon)$, V is good, and indeed excellent. We need to check that both remain true in $G_n(\epsilon)$. Goodness is easy: for any $x \in G_\infty(\epsilon)$, $\sum\{\gamma_i : \mathbf{t}(x, V) = 1\}$ is either $< \epsilon$ or $\geq 1 - \epsilon$, and so a fortiori this is true for the $x \in G_n(\epsilon)$. In other words:

($\star$) if A is a weighted set of elements of $G_n(\epsilon)$, and A is a weighted ϵ -good set in $G_\infty(\epsilon)$, then also A is a weighted ϵ -good set in $G_n(\epsilon)$.

The subtlety about excellence is that we need to quantify over all ϵ -good weighted sets of the given graph, and a priori those may be different in $G_n(\epsilon)$. Suppose B were an ϵ -good weighted set in $G_n(\epsilon)$ which ceased to be good in $G_{n+1}(\epsilon)$. (Since we made no assumptions about the minimality of n this is the general case.) This would mean that we added some new vertex which split B in an balanced way. But the vertices we add are all ϵ -vertices, that is, they arise as weighted ϵ -excellent sets of $G_n(\epsilon)$. In particular they have ϵ -majority opinions on any weighted ϵ -good set. We have shown that:

($\star\star$) If C is a weighted set of elements of $G_n(\epsilon)$, then C is an ϵ -good weighted set of $G_\infty(\epsilon)$ if and only if it is an ϵ -good weighted set of $G_n(\epsilon)$.

So V is a weighted ϵ -excellent set of $G_n(\epsilon)$, i.e., an ϵ -vertex of $G_n(\epsilon)$. $\square$

Corollary 3.10.5. *If v is an ϵ -vertex of $\mathcal{G}_\infty(\epsilon)$ then $v \in V_\infty$.*

Corollary 3.10.6. *If $\mathbf{t}$ is an ϵ -type of $\mathcal{G}_\infty(\epsilon)$ then $\mathbf{t}$ is finitely realized in $\mathcal{G}_\infty(\epsilon)$.*

The subtlety of the next claim is that simply applying Littlestone Minimax to an ϵ -type returns some weighted ϵ -good A , but Claim 3.10.6 only tells us that weighted ϵ -*excellent* sets belong to B .

Claim 3.10.7. *If $\mathbf{t}$ is finitely realized in $\mathcal{G}_\infty(\epsilon)$ then $\mathbf{t}$ is realized in $\mathcal{G}_\infty(\epsilon)$.*

Proof. Recall that V_∞ denotes the vertex set of $\mathcal{G}_\infty(\epsilon)$. Let $\mathbf{t}$ be an ϵ -type of $\mathcal{G}_\infty(\epsilon)$ and by 3.10.2, let $\mathbf{t}_*$ be an extension to the domain

$$V_\infty \cup \{B : B \text{ is a weighted } \epsilon\text{-good set of } \mathcal{G}_\infty(\epsilon) \}.$$

[Recall that weighted ϵ -good sets are always finite.] Since by construction $\mathbf{t}_*$ is finitely realized by ϵ -vertices of $\mathcal{G}_\infty(\epsilon)$, by Corollary 3.10.5, $\mathbf{t}_*$ is finitely realized in $\mathcal{G}_\infty(\epsilon)$. Hence by Littlestone Minimax $\mathbf{t}_*$ is realized by some weighted ϵ -good A . But now by construction A has an opinion about every weighted ϵ -good B of $\mathcal{G}_\infty(\epsilon)$. Hence A is a weighted ϵ -excellent set, in other words an ϵ -vertex, which realizes the type. Invoke Corollary 3.10.5 one more time to conclude $A \in V_\infty$, which completes the proof. $\square$

We can now complete the proof of the theorem:

Proof of 3.10.3. By Corollary 3.10.6 and Claim 3.10.7. $\square$

Theorem 3.5.6 goes through for graphs with a similar proof: we need to make the natural updates to use “excellent” rather than “good” and to deal with the fact that things are no longer bipartite.

Lemma 3.10.8. *Let $s \in \mathbb{N}$, $s \geq 2$, and $\epsilon < 1/2^s$. Let $\langle a_\eta : \eta \in {}^{s>}2 \rangle$ and $\langle b_\rho : \rho \in {}^s2 \rangle$ be a special tree of height s in $G_\infty(\epsilon)$. Then there is also a special tree of height s in G .*

Proof. Let $m < \omega$ be minimal so that $\{a_\eta : \eta \in {}^{s>}2\} \cup \{b_\rho : \rho \in {}^s2\} \subseteq G_m(\epsilon)$; if $m = 0$ we are done, so assume $m = n + 1$ for some $n \in \mathbb{N}$. For each $\eta \in {}^{s>}2$ choose a corresponding ϵ -vertex A_η of G_n and for each $\rho \in {}^s2$ choose a corresponding ϵ -vertex B_ρ of G_n , so for $\eta \leq \rho$, $\mathbf{t}(A_\eta, B_\rho) = 1$ if and only if there is an edge between a_η and b_ρ .

Since these are virtual vertices and virtual edges, there are two steps. First, for each $\rho \in {}^s2$, define an error set $E_\rho = \{(b_i, \gamma_i) \in B_\rho : \mathbf{t}(b_i, A_\eta) \neq \mathbf{t}(B_\rho, A_\eta)\}$ which will have weight $< \epsilon s$. So there is $(b, \gamma) \in B_\rho \setminus E_\rho$ and we can set $b'_\rho := b$. Second, now we are in the setting of having a special tree in G_n with internal nodes $\langle A_\eta : \eta \in {}^{s>}2 \rangle$ which may be virtual, and leaves $\langle b'_\rho : \rho \in {}^s2 \rangle$ which are vertices of G_n , so by Lemmas 3.5.3 and 3.5.4, we find a special tree of height s , $\langle a'_\eta : \eta \in {}^{s>}2 \rangle$, $\langle b'_\rho : \rho \in {}^s2 \rangle$ in the vertices of G_n , contradicting the choice of n . $\square$

Conclusion 3.10.9. *Let $\mathcal{G}$ be a graph.*

1. *If $\text{Ldim}(\mathcal{G}) := \ell < \infty$, and $\epsilon < 1/2^{\ell+1}$, then*

$$\text{Ldim}(\mathcal{G}) = \text{Ldim}(\mathcal{G}_\infty(\epsilon)) = \ell.$$

2. *If in addition $\epsilon < 1/2^{\ell+2}$, then*

$$\text{Thr}(\mathcal{G}) = \text{Thr}(\mathcal{G}_\infty(\epsilon)).$$

3. *If $\text{VC}(\mathcal{G}) := d < \infty$, and $\epsilon < 1/2^{d+1}$, then*

$$\text{VC}(\mathcal{G}) = \text{VC}(\mathcal{G}_\infty(\epsilon)).$$

Proof. To prove this, we can use a parallel series of arguments as in Theorem 3.5.6, to

transfer a full special tree, half graph, or VC-tree, from $\mathcal{G}_{n+1}$ to $\mathcal{G}_n$, and inductively, all the way back to $\mathcal{G}$. $\square$

We have seen that ϵ -saturation for graphs behaves very well for small ϵ . What about intermediate ϵ ? Can it still happen that bounding ϵ in terms of the VC dimension gives us some control over the Littlestone dimension of $\mathcal{G}_\infty(\epsilon)$?

Remark 3.10.10. *In considering extending the arguments of Section 3.7 to graphs, proving an analogue of Claim 3.7.2 does not make sense globally since the domain of a function (that is, the neighborhood of a vertex) in G_∞ includes many vertices which are not in G , hence k -realizability does not make sense out of the box. We leave it as an interesting open question whether it is possible to conclude that $\mathcal{G}_\infty(\epsilon)$ remains Littlestone based only on restricting ϵ as a function of Littlestone dimension d .*

3.11 Examples

In this section and the next we work out some simple and some less simple examples relevant to the chapter. Any singleton is a good set, in any graph (and for any value of $\epsilon \in (0, 1/2)_{\mathbb{R}}$).

3.11.1 Cliques

There is a slight asymmetry between cliques and anticliques since we assume that graphs have no loops. If G is an anticlique, every finite subset is ϵ -good. But also notice that every good subset of G realizes a type which is already realized by a vertex in G . So $G_\infty(\epsilon) = G$.

If G is a clique, every singleton and every sufficiently large finite set $X \subseteq G$, i.e. of size at least ϵ^{-1} , is ϵ -good, indeed ϵ -excellent. There exist excellent sets which realize types not already realized in G , namely that of connecting to every $v \in G$ (since no vertex of G is connected to itself). Thus, if $\epsilon < 1/2$ and $|G| > 1/\epsilon$, we will have that $G_1(\epsilon)$ is G along with one new vertex connected to all elements of G and to itself, and $G_n(\epsilon) = G_{n+1}(\epsilon)$ for all $n \geq 1$.

3.11.2 Matchings

Suppose $\mathcal{H}$ is a hypothesis class representing an infinite matching, i.e. $\mathcal{H} = \{h_x : x \in X\}$ and $h_x(z) = 1$ iff $z = x$. If $H \subseteq \mathcal{H}$ is such that $|H| > \epsilon^{-1}$ then H is ϵ -good. Thus, any singleton and any sufficiently large finite subset of $\mathcal{H}$ is good. Note that, for sufficiently small ϵ , all good sets which are not singletons represent the constant-zero function.

Similarly to Section 3.11.1, in an infinite matching, for any ϵ , $\mathcal{H}_1$ is $\mathcal{H}$ adjoin one new hypothesis, the constant-zero function; and $\mathcal{H}_n = \mathcal{H}_{n+1}$ for all $n \geq 1$.

A similar analysis holds for finite matchings, where say $|X| = |\mathcal{H}| \in \mathbb{N}$. If $|\mathcal{H}| \leq 1/\epsilon$, there are no new good sets, and if $|\mathcal{H}| > 1/\epsilon$, the constant-zero function appears. Note that in this second case, the constant-zero function is what we have called an ϵ -type, but it is *not* finitely realizable in $\mathcal{H}$ since there is no function $h \in \mathcal{H}$ which agrees with it on the finite set X . This example continues in Section 3.12.4.

3.11.3 Random graphs

We say “ G is an infinite random graph” to mean that G satisfies the model-theoretic extension axioms T_{rg} .¹⁷

Suppose G is an infinite random graph. Then:

1. any singleton is a good set,
2. any finite $X \subseteq G$ of even size is not a good set,
3. any finite $X \subseteq G$ of odd size $2k + 1$, $k \geq 1$ is not $k/(2k + 1)$ -good.

17. The axioms T_{rg} say: for each n , there is an axiom saying there are at least n elements, and for each n, m there is an axioms saying that for any two disjoint sets of n and m elements respectively, there is an element connected to all vertices in the first set and none in the second. There is a unique countable model up to isomorphism, which is isomorphic with probability 1 to the graph obtained by starting with countably many vertices and a fair coin and flipping the coin for each pair to determine whether or not this is an edge. This explains the model-theoretic name “random graph”; however, with the axiomatic approach, one has models of arbitrarily large infinite sizes.

Thus, if $\epsilon \leq 1/3$, the only finite sets which are ϵ -good are the singletons. This follows by the extension axioms, and the fact that $k/(2k+1)$ attains its minimum value at $k = 1$. Hence for $\epsilon \leq 1/3$, we have that $G = G_\infty(\epsilon)$.

3.11.4 Half-graphs

Suppose G is an infinite half-graph, that is, V is the disjoint union $A = \{a_i : i \in \mathbb{N}\} \cup B = \{b_j : j \in \mathbb{N}\}$, and $a_i \sim b_j$ iff $i < j$. We make no assumptions about other edges.¹⁸ Then:

1. any singleton is a good set,
2. if $X \subseteq G$ is finite but not a singleton, and $X \subseteq A$ or $X \subseteq B$, then X is not $1/3$ -good,
3. if $X \subseteq G$ is finite, and $X \cap A \neq \emptyset$ and $X \cap B \neq \emptyset$, then X is not $1/6$ -good.

Thus, if $\epsilon \leq 1/6$, the only finite sets which are ϵ -good are the singletons. Here also $G(\epsilon) = G_\infty(\epsilon)$.

Proof. Since Littlestone is central to the chapter and is characterized via half-graphs, we work this out precisely. For (1), we use the fact that all singletons are ϵ -good.

For (2), if $X \subseteq A$, let i_m be the median of $\{i : a_i \in X \cap A\}$. Then, b_{i_m} partitions X in a $1/3$ -balanced way, since the smallest part has size

$$\left\lfloor \frac{|X|}{2} \right\rfloor = \begin{cases} \frac{1}{2} \cdot |X| & |X| \text{ is even} \\ \left(\frac{1}{2} - \frac{1}{2|X|} \right) \cdot |X| \geq \frac{|X|}{3} & |X| \text{ is odd} \end{cases}$$

Therefore, X is not $1/3$ -good. The case where $X \subseteq B$ is analogous.

For (3), one of $A \cap X$ and $B \cap X$ has size at least $|X|/2$. Without loss of generality, let it be $A \cap X$. As in (2), there some $b \in B$ splitting $A \cap X$ in an $(1/3)$ -balanced way. Then, b

18. By convention, “half-graph” we will mean that the bi-induced pattern is a half graph, and will make no assumptions about the edges among the a ’s or among the b ’s.

partitions X in a $1/6$ -balanced way, since the minimum of

$$\{|N(b, B \cap X)| + |N(b, A \cap X)|, \quad |N(b, B \cap X)^c| + |N(b, A \cap X)^c|\}$$

has size at least $|A \cap X|/3 \geq |X|/6$. $\square$

In the next examples, it will be convenient to display types as (A, B) where $A \cup B = X$, A are the elements labeled 1 and B are the elements labeled 0.

3.11.5 An example where $\mathcal{H}_0 \neq \mathcal{H}_1 \neq \mathcal{H}_2$.

We work out a basic example to show that saturation can take more than one step. This will be generalized in the next example, but may be simpler to read in this first case.

Let $\epsilon \in (1/3, 1/2)$, and consider the following hypothesis class $\mathcal{H}$:

- $\mathcal{X} = \{x_1, x_2, x_3, x_4\}$
- $\mathcal{H} = \{h_{(i,j)} : 1 \leq i < j \leq 4\}$
- $h_{(i,j)}(a_\ell) = 1$ iff $\ell \in \{i, j\}$.

In words, the three hypotheses with i in the subscript connect to x_i , and to exactly one of the other x_j 's. From $\mathcal{H}_0$ to $\mathcal{H}_1$ to $\mathcal{H}_2$, the types realized over $\mathcal{X}$ progress from 2-element subsets, to 1-element subsets, to the empty subset.

- Analysis of $\mathcal{H}_0 = \mathcal{H}$: types realized by $\mathcal{H}$ over $\mathcal{X}$ are $\{(A, B) : |A| = 2\}$.
- Analysis of $\mathcal{H}_1$: Additional to the previously realized types,
 - The elements $H_\ell = \{h_{(i,j)} : \ell \in \{i, j\}\}$, for $1 \leq \ell \leq 4$ were added, and each realizes some type (A, B) with $|A| = 1$. If $\ell = k$, then $e(x_\ell, H_\ell) = |H_\ell|$ and $H_\ell(x_k) = 1$. Otherwise, $e(x_\ell, H_\ell) = |H_\ell|/3 < \epsilon|H_\ell|$, and $H_\ell(a_k) = 0$. Therefore, H_ℓ realizes $(\{x_\ell\}, \{x_i : i \neq \ell\})$.

– No element of $\mathcal{H}_1$ realizes $(\emptyset, X)$. Indeed, if some ϵ -good set $S \subseteq \mathcal{H}_0$ of size m realized this type, then $e(S, \mathcal{X}) = 2m$. By Pigeonhole principle, there is $x \in \mathcal{X}$ such that $e(x, S) \geq 2m/4 \geq 1/2$, and $\mathbf{t}(x, S) = 1$ must hold.

- Analysis of $\mathcal{H}_2$: $h = \{H_1, H_2, H_3, H_4\}$ was added. Additional to the previously realized types, it realizes $(\emptyset, X)$ because $e(x_i, h) = |h|/4 < \epsilon|h|$ for all $1 \leq i \leq 4$.

To summarize this example: we started with a set X of size 4, our hypotheses corresponded to all 2-element subsets of X ; so notice that any $Y \subseteq X$ of size two is shattered. At stage zero, every $h \in \mathcal{H}$ has degree two. At stage one, we acquire some hypotheses of degree one, and at stage two, we acquire a hypothesis of degree zero. This descent of degree suggests why the VC and Littlestone dimension don't increment even though we are adding new sets. This becomes quite a bit more general in the next example.

3.11.6 An example where k extension steps are needed before saturation

Analogous to the previous example, we can construct finite classes $(X, \mathcal{H})$ that need an arbitrary finite number of steps to occur until saturation.

Lemma 3.11.1. *Let $k \in \mathbb{N}$, and $\epsilon \in (0, 2/(2k-1))_{\mathbb{R}}$. Then, there exists $n \in \mathbb{N}$ and $(X, \mathcal{H}_0)$ such that $|X| = n$ and $\mathcal{H}_0 \subsetneq \mathcal{H}_1 \subsetneq \dots \subsetneq \mathcal{H}_k$.*

Proof. First, we can find some $n > 2k-2$ such that $\epsilon \in (1/(n-k+1), 2/n)$, because

$$\bigcup_{n > 2k-2} \left(\frac{1}{n-k+1}, \frac{2}{n} \right) = \left(0, \frac{2}{2k-1} \right).$$

Having chosen n , we can let $X = \{a_i : i \in [n]\}$ and $\mathcal{H}_0 = \{b_S^k : S \in \binom{[n]}{k}\}$, where $b_S^k(a_i) = [[i \in S]]$. The types realized by $\mathcal{H}_0$ over X are $\{(A, B) : |A| = k\}$. By the choice of ϵ , we have $\mathcal{H}_1 \supseteq \mathcal{H}_0 \cup \{b_S^{k-1} : S \in \binom{[n]}{k-1}\}$, where $b_S^{k-1} := \{b_{S'}^k : S \subset S'\}$.

The elements b_S^{k-1} are ϵ -good sets of size $n-k+1$, realizing over X the types $\{(A, B) :$

$|A| = k - 1\}$, because $\epsilon > 1/(n - k + 1)$ allows for

$$e(a_i, b_S^{k-1}) = \begin{cases} n - k + 1 & i \in S \\ 1 & i \notin S \end{cases} \Rightarrow \mathbf{t}(a_i, b_S^{k-1}) = \begin{cases} 1 & i \in S \\ 0 & i \notin S \end{cases}$$

Hence, no b_S^{k-1} is in $\mathcal{H}_0$, implying $\mathcal{H}_0 \subsetneq \mathcal{H}_1$. The proof of $\mathcal{H}_{\ell-1} \subsetneq \mathcal{H}_\ell$ is similar for each $2 \leq \ell \leq k$, and reveals which types are realized at each step:

- We can define $b_S^{k-\ell} := \{b_{S'}^{k-\ell+1} : S' \in \binom{n}{k-\ell+1}, S \subset S'\}$ for each $S \in \binom{n}{k-\ell}$, which is an ϵ -good set of $(n - k + \ell)$ elements in $\mathcal{H}_{\ell-1}$, by $\epsilon > 1/(n - k + \ell)$ and

$$e(a_i, b_S^{k-\ell}) = \begin{cases} n - k + \ell & i \in S \\ 1 & i \notin S \end{cases} \Rightarrow \mathbf{t}(a_i, b_S^{k-\ell}) = \begin{cases} 1 & i \in S \\ 0 & i \notin S \end{cases}$$

- Hence $\mathcal{H}_\ell \supseteq \mathcal{H}_{\ell-1} \cup \{b_S^{k-\ell} : S \in \binom{n}{k-\ell}\} \supsetneq \mathcal{H}_{\ell-1}$, by the fact that elements in $\mathcal{H}_\ell \setminus \mathcal{H}_{\ell-1}$ realize $\{(A, B) : |A| = k - \ell\}$, and by the claim below.

Claim 3.11.2. *For every $1 \leq \ell \leq k$, the elements in $\mathcal{H}_{\ell-1}$ only realize types $\{(A, B) : |A| \geq k - \ell + 1\}$ over X .*

Proof. We use induction over ℓ . For $\ell = 1$, $\mathcal{H}_0$ indeed realizes types in $\{(A, B) : |A| \geq k\}$, since $\mathcal{H}_0 = \{b_S^k : S \in \binom{[n]}{k}\}$ and $e(b_S^k, X) = k$ for all such S . For the inductive step, any element of $\mathcal{H}_{\ell-1}$ is of one of three kinds. For each, we argue the type realized is of size $\geq k - \ell + 1$.

1. Elements $b_S^k, \dots, b_S^{k-\ell+1}$ have $e(b_S^p, X) \geq k, \dots, k - \ell + 1$, respectively.
2. Element $b_S^p \in \mathcal{H}_p$ added at step $p \leq \ell - 2$, satisfies $e(b_S^p, X) \geq k - \ell + 2$, by inductive hypothesis.
3. An ϵ -good set $S \subseteq \mathcal{H}_{\ell-2}$ added as $b \in \mathcal{H}_{\ell-1}$. If $|S| = m$, then $e(X, S) \geq (k - \ell + 2)m$, since types realized by $\mathcal{H}_{\ell-2}$ are of size $\geq k - \ell + 2$. There must

exist $a_{i_r} \in X$ for $r \in [k - \ell + 1]$, such that $e(a_{i_r}, S) \geq 2m/(n - k + \ell) > 2/n$ for each r . Else, from counting in two ways, we get a contradiction

$$(k - \ell + 2)m \leq e(X, S) < m(k - \ell) + \frac{2m}{n - k + \ell} \cdot (n - k + \ell)$$

Then, $\epsilon < 2/n$ implies $\mathbf{t}(a_{i_r}, b) = \mathbf{t}(a_{i_r}, S) = 1$ for all $r \in [k - \ell + 1]$.

□

This completes our proof that $\mathcal{H}_0 \subsetneq \cdots \subsetneq \mathcal{H}_k$ holds for the constructed $(X, \mathcal{H}_0)$. □

These examples give a family of Littlestone classes where the saturation construction does not stabilize until level k . Note that this is arranged at the cost of higher and higher initial, though obviously still finite, VC dimension.

Also note that $\text{VC}(\mathcal{H}_i) = \text{Ldim}(\mathcal{H}_i) = k$ for all $i < \omega$, and in particular $\text{VC}(\mathcal{H}_\infty(\epsilon)) = \text{Ldim}(\mathcal{H}_\infty(\epsilon)) = k$. On one hand, $\text{VC}(\mathcal{H}_i) \leq \text{Ldim}(\mathcal{H}_i) \leq k$ can be shown by noticing that all virtual elements realize types of size at most k . On the other hand, $\text{Ldim}(\mathcal{H}_i) \geq \text{VC}(\mathcal{H}_i) \geq \text{VC}(\mathcal{H}_0) \geq k$, because any subset of X of size k can be shattered by elements $b_S^k \in \mathcal{H}_0$ as long as $|X| \geq 2k$ (a property of the above constructions, as long as $\epsilon < 1/k$).

3.12 Examples for large ϵ

In this section we prove by example that for “large” ϵ , the saturation of a Littlestone class can even have infinite VC dimension.

3.12.1 An example where VC-dimension goes to infinity

The following claim shows that if $\epsilon > 0$ is large enough, it is possible for VC-dimension to increment at every step of saturation:

Example 3.12.1. Consider the class $(X, \mathcal{H})$ where $X := \mathbb{N}$, $\mathcal{H}_0 := \{b_S \subseteq \mathbb{N} : |S| = d\}$, and $b_S(i) := [[i \in S]]$ for any $S \subset \mathbb{N}$. Then, $\text{VC}(\mathcal{H}_0) = \text{Ldim}(\mathcal{H}_0) = d$. What's more, if $\epsilon > 1/(d+1)$, then $\text{Ldim}(\mathcal{H}_n) \geq \text{VC}(\mathcal{H}_n) \geq n+d$ for all $n < \omega$. In particular, $\text{VC}(\mathcal{H}_\infty(\epsilon)) = \omega$.

Proof. To show $\text{VC}(\mathcal{H}_0) = \text{Ldim}(\mathcal{H}_0) = d$, all elements of $\mathcal{H}$ have degree d and infinite codegree, so any set of d elements in X can be shattered. We show by induction on $n < \omega$ that $\mathcal{H}_n \supseteq \{b_S \subseteq \mathbb{N} : d \leq |S| \leq n+d\}$, which would imply $\text{Ldim}(\mathcal{H}_n) \geq \text{VC}(\mathcal{H}_n) \geq n+d$. The case $n = 0$ holds by construction. Suppose $\mathcal{H}_n \supseteq \{b_S \subseteq \mathbb{N} : d \leq |S| \leq n+d\}$ for some $n < \omega$. Then, for all $A \subseteq \mathbb{N}$ with $|A| = n+d+1$, the set

$$B_A := \{b_S : |S| = n+d, S \subseteq A\} \subseteq \mathcal{H}_n$$

is of size $(n+d+1)$, and for all $i \in \mathbb{N}$, either $e(i, B_A) = 0$ when $i \notin A$, or $e(i, B_A) = n+d$ when $i \in A$. Since $\epsilon > 1/(d+1) \geq 1/(n+d)$, then $\mathbf{t}(i, B_A) = [[i \in A]]$, is defined, B_A is ϵ -good, and $b_A \in \mathcal{H}_{n+1}$ is added. Thus, $\mathcal{H}_{n+1} \supseteq \{b_S : d \leq |S| \leq (n+d+1)\}$. To show that $\text{Ldim}(\mathcal{H}_n) \geq \text{VC}(\mathcal{H}_n) \geq n+d$, we use the same reasoning as for $\mathcal{H}_0$. $\square$

To summarize a key idea of this example: Suppose X is infinite and $\mathcal{H}$ is the set of (characteristic functions of) k -element subsets of X . Consider $\{a_1, \dots, a_{n+1}\} \subseteq X$. We would like to realize some pattern of zeros and ones over it, call this pattern $\sigma : \{a_1, \dots, a_{n+1}\} \rightarrow \{0, 1\}$. By assumption on $\mathcal{H}$, for every $1 \leq i \leq n+1$, there is a function h_i which agrees with σ on $\{x_1, \dots, x_{n+1}\} \setminus \{x_i\}$. Consider $H = \{h_i : 1 \leq i \leq n+1\}$. (We can choose the h_i 's to be distinct, or we can make this a weighted set.) For every i , at least $n/(n+1)$ of the functions vote correctly. So if $\epsilon > 1/n$, the majority vote of H is well defined and agrees with σ . In this way we progressively extend the shattering to larger sets.

3.12.2 An example of a good set which is not an excellent set

Excellent sets play an important role in the sections on graphs. For more discussion and examples on this see [105]. Let $\epsilon = 21/60$, and $G = (V, E)$ be the graph in the figure below,

with $V = A \cup B$, $A = \{a_i\}_{i \in [6]}$, and $B = \{b_j\}_{j \in [6]}$.

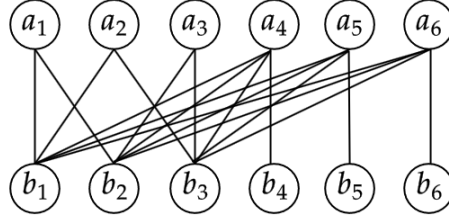


Figure 3.2: A diagram showing a bigraph with $V = A \cup B$, $A = \{a_i\}_{i \in [6]}$, and $B = \{b_j\}_{j \in [6]}$, which are connected in a way so that while A and B are both ϵ -good, neither is ϵ -excellent, for $\epsilon = 21/60$.

While A and B are both ϵ -good, neither is ϵ -excellent. Indeed, all vertices have 0, 1, or 5 neighbors in A , and 0, 2 or 4 neighbors in B , and $\{1/6 : i \in \{0, \dots, 6\} \setminus \{3\}\} \subseteq [0, \epsilon) \cup (1 - \epsilon, 1]$. Moreover, $\mathbf{t}(A, B)$ is not defined because B is ϵ -good and $|\{a \in A : \mathbf{t}(a, B) = 0\}| = |\{a \in A : \mathbf{t}(a, B) = 1\}| = |A|/2$. Similarly, $\mathbf{t}(B, A)$ is not defined because A is ϵ -good and $|\{b \in B : \mathbf{t}(b, A) = 0\}| = |\{b \in B : \mathbf{t}(b, A) = 1\}| = |B|/2$.

3.12.3 Extensions

Observe that in both cliques (see Section 3.11.1) and matchings (see Section 3.11.2), given any two good sets, there is a larger good set containing both of them. This is a property that all bounded degree graphs enjoy:

Proposition 3.12.2. *Let $G = (V, E)$ be an arbitrary infinite graph such that every vertex has degree at most $d \in \mathbb{N}$. Then, every finite $X \subseteq V$ with $|X| > d/\epsilon$ is ϵ -good. Moreover, for any two ϵ -good sets X_1 and X_2 , there exists an ϵ -good set X_3 such that $X_1 \cup X_2 \subseteq X_3$.*

Proof. For the first part, consider such vertex set X , and any $v \in V$. Since $\deg(v) \leq d$, v partitions X in an ϵ -unbalanced way, since the smaller size of this partition is less than or equal than $d < \epsilon \cdot |X|$.

For the second part, consider two ϵ -good sets X_1 and X_2 . Infiniteness of $|V|$ implies there is a set $X_3 \subset V$ containing $X_1 \cup X_2$, possibly with some vertices added in order to

have $|X_3| > d/\epsilon$ hold. By the first claim, X_3 is ϵ -good. $\square$

3.12.4 Realizability and the value of epsilon

Suppose $\mathcal{H}$ is a hypothesis class and $r < 1/\epsilon$. Observe that any function which is r -realized in $\mathcal{H}_1(\epsilon)$ is already realized in $\mathcal{H}$. [Why? By a union bound, see 3.7.2 for a more general result.] On the other hand, consider the class of singletons over $X = \{1, 2, 3, 4, 5\}$, $\epsilon = 1/4$, and $r = 5 > 1/\epsilon$. In this example, $H = \mathcal{H}$ (with the counting measure) is an ϵ -good set, and $\mathbf{t}_H$ is the constant zero function, which is realized in $\mathcal{H}_1(\epsilon)$ and even 4-realized in $\mathcal{H}$, but not realized, indeed not even 5-realized, in $\mathcal{H}$.

3.13 Open questions

There are some very interesting combinatorial questions arising from this analysis around the regimes of ϵ , which don't have obvious analogues in infinitary notions of saturation. Theorem 3.7.1 draws our attention to the case $1/(Cd) \geq \epsilon \geq 1/(d+1)$. The first few questions are about different aspects of this gap. In each case we assume $\text{Ldim}(\mathcal{H}) = \ell < \infty$ and hence $\text{VC}(\mathcal{H}) = d < \infty$, and in each case $\mathcal{H}_\infty$ means $\mathcal{H}_\infty(\epsilon)$. The constant C is from Theorem 3.7.1.

Question 3.13.1. *If $\text{Ldim}(\mathcal{H}) < \infty$, can it happen that $\text{Ldim}(\mathcal{H}_\infty) = \infty$ but $\mathcal{H}_\infty$ still has the property that $\text{VC}(\mathcal{H}_\infty) < \infty$?*

Question 3.13.2. *In general, what happens to $\text{Ldim}(\mathcal{H}_\infty(\epsilon))$ when $1/(Cd) \leq \epsilon$?*

It follows from the definitions that if ever $\mathcal{H}_n = \mathcal{H}_{n+1}$ then $\mathcal{H}_{n+1} = \mathcal{H}_m$ for all $m > n$. However, our only example above where $\mathcal{H}_n \neq \mathcal{H}_{n+1}$ for all $n < \omega$ is in the case where $\text{VC}(\mathcal{H}_\infty) = \infty$. The next question asks if this can happen in a tamer setting. (By the above, sufficiently small values of ϵ can be ruled out.)

We can also ask the parallel question for graphs G .

Question 3.13.3. *What happens to the Littlestone dimension of $\mathcal{G}_\infty(\epsilon)$ when $\mathcal{G}$ is a Littlestone class and ϵ is of intermediate range bounded as a function of the VC dimension d , per Remark 3.10.10?*

Question 3.13.4. *Are there any other preservation theorems for the saturation with respect to other combinatorial complexity measures?*

CHAPTER 4

LOCAL COMBINATORIAL ANALOGUES FOR BOUNDED VC DIMENSION

Stable graphs, or equivalently Littlestone classes, were characterized by existence of linear-sized ‘good’ sets, a kind of strongly homogeneous set, in work of Malliaris–Shelah and Malliaris–Moran. We prove a parallel result for VC classes, showing these are characterized by existence of linear-sized symmetric or asymmetric good pairs (which we define). We give several proofs, each drawing from methods and results from different areas, and resulting in different kinds of bounds. We finish with a few words on our learning theory motivation for these investigations and state some further research directions.

4.1 Introduction

The context of studying classes $(X, \mathcal{H})$ as bigraphs has allowed for connections and applications among areas including learning theory, model theory, and extremal graph theory.

Important work has been done for the class of graphs that have bounded VC dimension, which are characterized by *PAC learnability* due to the Fundamental Theorem of Statistical Learning [149]. In extremal combinatorics, VC classes have a regularity property stronger than the one for the class of all graphs (see [68, 16, 101]).

Bigraphs with the property of *k-edge stability* correspond to Littlestone classes in learning theory [38], which are moreover *online learnable* [26] and *differentially private PAC learnable* [12]. In extremal combinatorics [107], the class of *k-edge stable* graphs has been proven to satisfy a stronger regularity condition than the class of all graphs. The work in [105] has found an application of the existing tools and techniques from learning theory into extremal combinatorics, proving that Littlestone classes can be shown to have ϵ -excellent sets of large size, not only for $\epsilon < 1/2^\ell$, where ℓ is the Littlestone dimension of the class, but for all $\epsilon \in (0, 1/2)$, by using regret bound methods from online learning.

In the present work we are interested in the question of to what extent the strong results on good sets and Littlestone classes can be extended to the larger set of structures which are VC classes. Since the existence of linear-sized good sets characterized stability, there will need to be a distinct definition for VC classes. We propose and study a partite analogue and show that the existence of linear-sized symmetric good pairs, in the sense of Definition 4.4.2, characterizes VC classes.

We give three different proofs of the above. The first one is a combinatorial proof by hand. The second one is a proof appealing to Haussler’s Packing Lemma [86]. The third one is a proof based on an application of the VC Regularity Lemma [68, 16, 101]. For our hope for applications in learning theory, there are different advantages to these proofs because the algorithmic and computational costs of the respective combinatorial tools used can be significant.

4.1.1 Prior and Related Work

The work done used the following existing results: the VC Regularity Lemma [68, 16, 101]; Haussler’s Packing Lemma [86]; the Stable Regularity Lemma [107]; the analysis of good and excellent sets in Littlestone classes [105].

The search for better Ramsey type results in VC classes has a long history of work. In particular, [118] recently established the Erdős-Hajnal property for VC classes. However, other work on the linear scale does not necessarily relate to this.

4.2 Chapter Overview

Section 4.3 defines VC and Littlestone dimension, and defines good and antigood sets, provided a parameter $\epsilon \in (0, 1/2)$. Section 4.4 defines the main objects we study in their symmetric and asymmetric versions, and states our main result, Theorem 4.4.4.

We will prove this theorem, or analogues of it, in several ways. Section 4.4.1 gives a

combinatorial proof, by hand. Section 4.9 compares Theorem 4.4.4 to the existing characterization for Littlestone classes. Section 4.7 expands on a key example, the half graph. Then, Section 4.8 gives an alternative proof of Theorem 4.4.4 using Haussler’s Packing Lemma, and Section 4.9 gives another alternative proof, by using the VC (also known as ultra strong) Regularity Lemma. Section 4.10 ends with some open questions.

4.3 Background

4.3.1 Notation

The following are some objects that we will work with in this paper:

Definition 4.3.1. A hypothesis class $(X, \mathcal{H})$, also denoted $\mathcal{H}$, consists of a set of subsets of the set X . We may several times identify each $h \in \mathcal{H}$ with its characteristic function, which for each $a \in h$ gives $h(a) = 1$, and for each $a \notin h$ gives $h(a) = 0$.

Definition 4.3.2. The bipartite graph $G = (V_1 \cup V_2, E)$ is given with a partition of the vertex set, and with an edge set $E \subseteq V_1 \times V_2$ which is neither symmetric nor irreflexive.

Definition 4.3.3. Given a hypothesis class $\mathcal{H}$, we can regard it as a bipartite graph, or bigraph, denoted $\text{Bip}(X, \mathcal{H})$. Here:

1. X and $\mathcal{H}$ are the two sides of the vertex set partition, and an edge relation is given by $E := \{(a, h) : h(a) = 1\}$; and
2. $\text{Bip}(X', \mathcal{H}')$ can be defined by letting $X' = \mathcal{H}$, $\mathcal{H}' = X$, and $E := \{(h, a) : h(a) = 1\}$
3. by letting $\mathcal{H}' = \{h \in \mathcal{H} : h(x) = 1\} : x \in X\}$, we obtain the dual class $(\mathcal{H}', X')$.

4.3.2 Littlestone and VC dimensions

Even though the notions presented herein have already been defined in Chapter 3, we will restate them for reader friendliness. The following combinatorial structure was introduced by Littlestone [97].

Definition 4.3.4 (Littlestone Mistake tree). *A Littlestone mistake tree is a binary decision tree whose nodes are $x_i \in X$, and whose branch edges are labeled respectively by $y_i \in \mathcal{Y} = \{\pm 1\}$, each node having its two outgoing edges with different labels. A hypothesis $h \in \mathcal{H}$ realizes a root-to-leaf path $\{(x_i, y_i)\}_{i=1}^d$ when $h(x_i) = y_i$ for every branch, and $\mathcal{H}$ shatters a tree when every root-to-leaf path is realized by some $h \in \mathcal{H}$.*

Definition 4.3.5 (Littlestone dimension). *The maximum depth of a Littlestone mistake tree that is shattered by $\mathcal{H}$ is defined as the Littlestone dimension of the hypothesis class, and denoted by $Ldim(\mathcal{H})$.*

Another complexity measure we work with is VC dimension:

Definition 4.3.6 (Vapnik–Chervonenkis (VC) dimension). *We say that a class $\mathcal{H}$ shatters a set $\{x_i\}_{i=1}^n \subseteq X$ if for every possible labeling $\{y_i\} \in \{0, 1\}^n$ of the elements of X , there exists some $h \in \mathcal{H}$ such that $h(x_i) = y_i$ for all $i \in [n]$. We then define the VC dimension, denoted by $VC(\mathcal{H})$, to be the supremum of the sizes of all finite sets in X shattered by $\mathcal{H}$ this way.*

By using mistake trees, we can give an equivalent definition for VC dimension:

Definition 4.3.7 (VC mistake tree). *A VC mistake tree is a binary decision tree with nodes $x_i \in X$ and branching edges labeled by $y_i \in \{\pm 1\}$, such that each node has two outgoing branches with different labels, and such that for every level $0 \leq \ell \leq d$ of the tree, the node labels of all elements of X in level ℓ are the same. A hypothesis $h \in \mathcal{H}$ realizes a root-to-leaf path $\{(x_i, y_i)\}_{i=1}^d$ when $h(x_i) = y_i$ for every $1 \leq i \leq d$, and a class $\mathcal{H}$ shatters a tree when every root-to-leaf path is realized by some $h \in \mathcal{H}$.*

Definition 4.3.8 (Vapnik–Chervonenkis dimension). *The maximum depth of a VC mistake tree with nodes from X that is shattered by $\mathcal{H}$ is defined as the VC dimension of the hypothesis class, $VC(\mathcal{H})$.*

4.3.3 Good sets

The notion of good and excellent sets from the work of Malliaris and Shelah [107] is key here. In this paper, we will only focus on the notion of ϵ -good sets.

Definition 4.3.9 (Good sets). *Let $G = (V, E)$ be a graph, and let $\epsilon \in (0, 1/2)$. An ϵ -good set $S \subseteq V$ is one such that for every vertex $v \in V$, one of the following holds:*

- (i) $|N(v, S)| < \epsilon|S|$; in this case, we denote $\mathbf{t}(v, S) = 0$, and v has a vote of 0 on S ;
- (ii) $|N(v, S)| > (1 - \epsilon)|S|$; in this case, we denote $\mathbf{t}(v, S) = 1$, and v has a vote of 1 on S .

Note that the property of being ϵ -good is monotonic with respect to ϵ : an ϵ_1 -good set is also ϵ_2 -good, if $\epsilon_1 < \epsilon_2$. Indeed, for each v , $|N(v, S)| < \epsilon_1|S|$ implies $|N(v, S)| < \epsilon_2|S|$, and $|N(v, S)| > (1 - \epsilon_1)|S|$ implies $|N(v, S)| > (1 - \epsilon_2)|S|$. The implication from ϵ_1 to ϵ_2 remains true if we quantify over all $v \in V$.

4.3.4 Antigood sets

Given a set S and a parameter ϵ , we introduce the notion of its antigood set $\text{Anti}(S, \epsilon)$, which denotes the set of vertices that prevent $S \subseteq V$ from being ϵ -good.

Definition 4.3.10. *Let $S \subseteq V$, and $\epsilon \in (0, 0.5)$. Its antigood set, $\text{Anti}(S, \epsilon)$, is defined as follows*

$$\text{Anti}(S, \epsilon) := \{v \in V : \epsilon|S| \leq |N(v, S)| \leq (1 - \epsilon)|S|\}$$

Note, that if a set $S \subseteq V$ is ϵ -good, then $\text{Anti}(S, \epsilon) = \emptyset$. Allowing for the antigood set to be nonempty, but requiring it to be small, we can still get relevant information, as will be elaborated below.

4.4 Contributions of this chapter

4.4.1 Characterization of classes of bounded VC dimension

We found a property equivalent to being of bounded VC dimension, and called this property (c_1, c_2, ϵ) -good sampling. Before defining it, we state two preliminary notions of having a pair of sets be ϵ -good. This could happen symmetrically or not.

Definition 4.4.1. [*Asymmetric ϵ -goodness*] Let $G = (V, E)$ be a graph, and $A, B \subset V$ be subsets of nodes in this graph. We say that A is ϵ -good for B if for every $b \in B$, the following holds:

$$\frac{|\{a \in A : (b, a) \in E\}|}{|A|} \in [0, \epsilon] \cup (1 - \epsilon, 1].$$

Definition 4.4.2. [*Symmetric ϵ -goodness*] Let $G = (V, E)$ be a graph, and $A, B \subset V$ be subsets of nodes in this graph. We say that A and B form a symmetric ϵ -good pair of sets if A is ϵ -good for B and B is ϵ -good for A .

These definitions are different a priori. The relevance of this difference will become clearer in this section and Section 4.4.1.

Note that Definitions 4.4.1 and 4.4.2 differ: there exist pairs A and B of sets which are ϵ -good, only in the asymmetric sense. Consider, for example, the bipartite graph with $V = A \cup B$, where half of B is connected to all of A , whereas the other half is not connected to all of A , as Figure 4.1 below shows. Then, A is ϵ -good for B , but B is not ϵ -good for A , for any choice of parameter $\epsilon \in (0, 1/2)$.

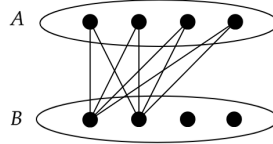


Figure 4.1: An example of a pair that is ϵ -good only in the asymmetric sense, for every $\epsilon \in (0, 1/2)_{\mathbb{R}}$.

Motivated by the above, we give the following definition:

Definition 4.4.3. We say that a pair $(X, \mathcal{H})$ has (c_1, c_2, ϵ) -good sampling if, for every $A \subseteq X$ and $B \subseteq \mathcal{H}$, there exist $A' \subset A$ and $B' \subset B$ such that (A', B') form a symmetric ϵ -good pair, and

$$|A'| \geq c_1|A| \quad \text{and} \quad |B'| \geq c_2|B|.$$

The following is the main contribution of this chapter, and states that the property of (c_1, c_2, ϵ) -good sampling is characteristic in the classes that have bounded VC dimension:

Theorem 4.4.4. Let $\mathcal{H} \subseteq \{0, 1\}^X$ be a hypothesis class. The following are equivalent:

- (a) $\text{VC}(\mathcal{H}) = d < \infty$;
- (b) for every $\epsilon \in (0, 1/2)$ there exist constants c_1, c_2 such that $(X, \mathcal{H})$ has (c_1, c_2, ϵ) -good sampling;
- (c) there exists $\epsilon_0 \in (0, 1/2)$ and constants c_1, c_2 such that $(X, \mathcal{H})$ has (c_1, c_2, ϵ_0) -good sampling.

As a remark, it will follow from the proof in Section 4.5 for Theorem 4.4.4 that there is a basic lower bound for each of the constants: $c_1 \geq \epsilon/2^{d+1}$ and $c_2 \geq 1/2^{d+1}$.

An example where we have the property of (c_1, c_2, ϵ) -good sampling is the half graph, whose VC dimension is 1. This example is discussed more in depth in Section 4.7.

4.4.2 Existence of sets with small antigood sets

In the previous subsection, Theorem 4.4.4 shows that classes of bounded VC dimension have the property of (c_1, c_2, ϵ) -good sampling, which in words means there exist linear-sized symmetric ϵ -good pairs. This subsection studies a reinterpretation of this characterization, using the language of antigood sets introduced earlier.

Indeed, an immediate corollary of Theorem 4.4.4 states that given a VC class and a parameter $\epsilon > 0$, it is possible to find a set with lower bounded size, with an antigood set whose size is upper bounded:

Corollary 4.4.5. *Let $\mathcal{H}$ be a hypothesis class with VC dimension d . Then, there exists a value $\epsilon_0 \in (0, 1/2)$ such that $(X, \mathcal{H})$ has (c_1, c_2, ϵ_0) -good sampling.*

Thus, for every pair $A \subseteq X$ and $B \subseteq \mathcal{H}$, there exists a pair $A' \subseteq A, B' \subseteq B$ of sets with respective sizes at least $c_1|A|$ and $c_2|B|$ respectively, such that (A', B') is a symmetric ϵ -good pair. This implies that the antigood set of A' is small in B :

$$|\text{Anti}(A', \epsilon_0)| = |B| - |B'| \leq |B| \cdot (1 - c_2).$$

4.5 Proof of Theorem 4.4.4

First we will show that bounded VC dimension implies the existence of asymmetric ϵ -good pairs:

Lemma 4.5.1. *If $\text{VC}(\mathcal{H}) \leq d < \infty$ then for every $\epsilon \in (0, 1/2)$ there exist constants $c_1 := \epsilon^d$ and $c_2 := 1/2^d$ such that for every two finite sets $A \subseteq \mathcal{H}$ and $B \subseteq X$, there exist $A' \subseteq A$ and $B' \subseteq B$ with $|A'| \geq c_1|A|$ and $|B'| \geq c_2|B|$, such that A' is ϵ -good to B' .*

Proof. We can construct a VC mistake tree in the following way. We begin with $A_\emptyset := A$ at level 0. If there is some $b_0 \in B$ that partitions $A_\emptyset$ into $A_0 \cup A_1$ in an ϵ -balanced way, then we place A_0 and A_1 at level 1.

Consider level ℓ , with the nodes in it being a partition of $A_\emptyset$ into $\bigcup_{i \in 2^\ell} A_i$, with $\ell \in \mathbb{N}_{\geq 0}$. Constructing level $\ell + 1$ would require finding $b_\ell \in B$ such that b_ℓ partitions every A_i in an ϵ -balanced way, into $A_{i \cap 0} := \{a \in A_i : a(b_\ell) = 0\}$ and $A_{i \cap 1} := \{a \in A_i : a(b_\ell) = 1\}$. If all the sets A_i for $i \in 2^\ell$ are nonempty, the VC tree can be obtained by extracting one $a_i \in A_i$ for $i \in \bigcup_{0 \leq m \leq \ell} 2^m$. This collection of hypotheses that shatter the set $\{b_k : 0 \leq k \leq \ell - 1\}$ would witness that $\text{VC}(\mathcal{H}) \geq \ell$, a contradiction.

The upper bound $\text{VC}(\mathcal{H}) \leq d$ implies that this VC tree construction can go only as far as d levels or less. Thus, for some $\ell_0 \leq d$, no $b \in B$ partitions simultaneously all parts in that level in a balanced way. Then, for every $b \in B$, there exists at least one A_i with $i \in 2^{\ell_0}$

that is partitioned in an ϵ -unbalanced way by b .

By Pigeonhole Principle, at least a fraction of $1/2^{\ell_0} \geq 1/2^d = c_1$ of the elements b of B partition the same part A_{i_0} in an ϵ -unbalanced way; define such set B' , and define $A_{i_0} = A'$. We have that $|B'| \geq c_2|B|$, with $c_2 = 1/2^d$, and that $|A'| \geq \epsilon^{\ell_0}|A| \geq \epsilon^d|A| = c_2|A|$. Moreover, A' is ϵ -good to B' , since this set is partitioned in an ϵ -unbalanced way by each $b \in B'$:

$$\forall b \in B', \quad \frac{|N(b) \cap A'|}{|A'|} \in [0, \epsilon) \cup (1 - \epsilon, 1]. \quad \square$$

With the lemma above, we could find an asymmetric ϵ -good pair, which is not symmetric a priori. Applying the following lemma will allow us to extract a symmetric ϵ -good pair from it.

Lemma 4.5.2. *Let $0 < \epsilon < 1/4$ and let $A, B \subset V$ be such that A is ϵ -good to B . Then, there exist $A' \subseteq A$ and $B' \subseteq B$ such that (A', B') is a symmetric 2ϵ -good pair and $|A'| \geq |A|/2$, and $|B'| \geq |B|/2$.*

Proof. Since A is ϵ -good to B , we can write $B = B_0 \cup B_1$, where $B_i := \{b \in B : \mathbf{t}(b, A) = i\}$. We have two cases, depending on which of the two subsets is larger.

If $|B_0| \geq |B_1|$, set $B' := B_0$. Each vertex in B' has less than $\epsilon|A|$ neighbors in A , and edges are symmetric, thus

$$e(A, B') = e(B', A) < |B'| \cdot \epsilon|A|.$$

Let $A = A^- \cup A^+$, where $A^+ := \{a \in A : |N(a) \cap B'| \geq 2\epsilon|B'|\}$ and $A^- := A \setminus A^+$. The left-hand side is bounded:

$$e(A, B') = e(A^+, B') + e(A^-, B') \geq |A^+| \cdot 2\epsilon|B'| + |A^-| \cdot 0$$

And thus, $\epsilon|A| \cdot |B'| > 2\epsilon|A^+| \cdot |B'|$, which finally implies $\frac{1}{2}|A| > |A^+|$, and that $|A^-| \geq |A|/2$. If we define $A' := A^-$, then $(A', B') = (A^-, B_0)$ is a symmetric 2ϵ -good pair. Indeed, every

$a \in A^-$ has at most $2\epsilon|A'|$ neighbors in B_0 , and moreover, every $b \in B_0$ satisfies the following:

$$|N(b) \cap A^-| = |N(b) \cap A| - |N(b) \cap A^+| < \epsilon|A| \leq 2\epsilon|A^-|.$$

If $|B_1| \geq |B_0|$, we set $B' := B_1$. Every vertex in B' has more than $(1 - \epsilon)|A|$ neighbors in A . Since edges are symmetric:

$$e(A, B') = e(B', A) > |B'| \cdot (1 - \epsilon)|A|.$$

If we now let $A = A^- \cup A^+$, where $A^- := \{a \in A : |N(a) \cap B'| \leq (1 - 2\epsilon)|B'|\}$ and $A^+ := A \setminus A^-$, we then get the bound:

$$e(A, B') = e(A^-, B') + e(A^+, B') \leq |A^-| \cdot (1 - 2\epsilon)|B'| + |A^+| \cdot |B'|$$

Joining and simplifying the above two inequalities:

$$(1 - \epsilon)|A| \cdot |B'| < e(A, B') < |A^-| \cdot (1 - 2\epsilon)|B'| + |A^+| \cdot |B'|$$

$$\Rightarrow (1 - \epsilon)|A| < (1 - 2\epsilon)|A^-| + |A^+|,$$

implying that $|A^+| \geq |A|/2 > |A^-|$. Then, the pair $(A', B') = (A^+, B_1)$ is symmetric 2ϵ -good: every $a \in A^+$ has at least $(1 - 2\epsilon)|B'|$ neighbors in B' , and every $b \in B'$ satisfies

$$|N(b)^c \cap A^+| = |N(b)^c \cap A| - |N(b)^c \cap A^-| < \epsilon|A| \leq 2\epsilon|A^+|$$

$$\Rightarrow |N(b) \cap A^+| = |A^+| - |N(b)^c \cap A^+| \geq (1 - 2\epsilon)|A^+|.$$

In either case, we obtained two subsets $A' \subseteq A$ and $B' \subseteq B$ of size at least half that of the original sets in the pair, that are symmetric 2ϵ -good. Note, $2\epsilon \in (0, 1/2)$ as long as $\epsilon \in (0, 1/4)$. □

Now we have all that we need in order to extract symmetric good pairs in the case that our hypothesis class is of finite VC dimension:

Corollary 4.5.3. *If $\text{VC}(\mathcal{H}) \leq d < \infty$ then for every $0 < \epsilon < 1/2$ there exist constants $c_1 := \epsilon^d/2^{d+1}$, and $c_2 := 1/2^{d+1}$ such that for every two finite sets $A \subseteq \mathcal{H}$ and $B \subseteq X$, there exist $A' \subseteq A$ and $B' \subseteq B$ with $|A'| \geq c_1|A|$ and $|B'| \geq c_2|B|$, such that (A', B') is a symmetric ϵ -good pair.*

Proof. By Lemma 4.5.1, but with parameter $\epsilon/2 > 0$, we obtain sets A'', B'' such that A'' is $(\epsilon/2)$ -good to B'' , with sizes at least

$$|A''| \geq \left(\frac{\epsilon}{2}\right)^d |A|, \quad |B''| \geq \frac{1}{2^d} |B|.$$

Then, by Lemma 4.5.2 there exist sets $A' \subseteq A''$ and $B' \subseteq B''$, such that (A', B') is a symmetric ϵ -good pair (since we have $2 \cdot (\epsilon/2) = \epsilon$), and such that

$$|A'| \geq \frac{1}{2} |A''| \geq \frac{\epsilon^d}{2^{d+1}} |A|, \quad |B'| \geq \frac{1}{2} |B''| \geq \frac{1}{2^{d+1}} |B|. \quad \square$$

The next part of our proof is that a large VC dimension implies that there is no (c_1, c_2, ϵ) -good sampling. We will first show that this happens in the case of infinite VC dimension.

Lemma 4.5.4. *If $\text{VC}(\mathcal{H}) = \infty$, then for any $\epsilon \in (0, 1/2)$, and for every constant values $c_1, c_2 > 0$ there exist two sets $A \subseteq X$ and $B \subseteq \mathcal{H}$ such that any $A' \subseteq A$ and $B' \subseteq B$ of sizes at least:*

$$|A'| \geq c_1|A| \quad \text{and} \quad |B'| \geq c_2|B|,$$

will not form an ϵ -good pair. That is, a large subset A' cannot be ϵ -good to a large subset B' .

Proof. Let $d \in \mathbb{N}$. Since $\text{VC}(\mathcal{H}) \geq d$, there exist sets $A \subseteq X$ and $B \subseteq \mathcal{H}$, of sizes d and 2^d , respectively, such that every subset of A is realized as a neighborhood of an element in B . We want to show the property stated above, namely that no sufficiently large subsets A' and B' can form an ϵ -good pair.

For the sake of contradiction, suppose there exists such pair (A', B') . Then, for any subset of A' of size $k' \geq c_1|A|$, every element of B' must connect to any number of elements on A' that lies in the range $[0, \epsilon k'] \cup (k'(1 - \epsilon), k']$. Thus,

$$|B'| \leq \sum_{s=0}^{\lfloor \epsilon k' \rfloor} \binom{k'}{s} \cdot 2^{d-k'} + \sum_{s=\lceil k'(1-\epsilon) \rceil}^{k'} \binom{k'}{s} \cdot 2^{d-k'} \leq \sum_{s=0}^{\lfloor \epsilon k' \rfloor} 2 \cdot \binom{k'}{s} \cdot 2^{d-k'}.$$

We now argue that $|B'|/|B| = o(1)$ as $k \rightarrow \infty$:

$$\frac{|B'|}{|B|} \leq \sum_{s=0}^{\lfloor \epsilon k' \rfloor} 2 \cdot \binom{k'}{s} \cdot 2^{-k'} = 2^{1-k'} \cdot \sum_{s=0}^{\lfloor \epsilon k' \rfloor} \binom{k'}{s}.$$

Since $\epsilon \in (0, 1/2)$, the above partial sum of binomial coefficients is upper bounded by

$$2^{H(\epsilon k'/k') \cdot k'} = 2^{H(\epsilon) \cdot k'},$$

where H is the binary entropy function (see e.g., [65]), defined as the entropy of a Bernoulli random trial with probabilities $\{p, 1 - p\}$:

$$H(p) := \begin{cases} 0 & p = 0, 1 \\ -p \log_2(p) - (1 - p) \log_2(1 - p) & p \in (0, 1) \end{cases}$$

We then have

$$\frac{|B'|}{|B|} \leq 2^{1-k'+k' \cdot H(\epsilon)}. \quad (4.5.1)$$

We have that $k' \rightarrow \infty$ as $d \rightarrow \infty$, since $k' \geq c_1 \cdot d$. Moreover, $H(\epsilon) < 1$ because $\epsilon < 1/2$. Thus, the exponent in 4.5 goes to $-\infty$ as $k' \rightarrow \infty$, and so $|B'|/|B|$ goes to zero. This would contradict that $|B'| \geq c_2|B|$ for any value c_2 that is constant with respect to d . $\square$

The following presents a finite version of the above conclusion.

Lemma 4.5.5. *For every $\epsilon \in (0, 1/2)$, for all constants $c_1, c_2 > 0$, and for every sufficiently large $d \in \mathbb{N}$, there exists a class $\mathcal{H}$ such that $\text{VCdim}(\mathcal{H}) = d$, with the property that any two sets $A \subseteq X$ and $B \subseteq \mathcal{H}$ and any two subsets $A' \subseteq A$ and $B' \subseteq B$ with*

$$|A'| \geq c_1|A| \quad \text{and} \quad |B'| \geq c_2|B|$$

cannot form an ϵ -good pair (A', B') , not even in the asymmetric sense.

Proof. Having $\text{VC}(\mathcal{H}) = d$ gives two sets $A \subseteq X$ and $B \subseteq \mathcal{H}$ of sizes d and 2^d respectively, B realizing at least once every subset of A . In analogy to the work done above, a counting argument gives that the following bound must hold if the pair (A', B') is (asymmetric) ϵ -good:

$$c_2 \leq \frac{|B'|}{|B|} \leq 2^{1+k' \cdot (H(\epsilon)-1)}.$$

Since $H(\epsilon) < 1$ and $k' \geq c_2 \cdot |B| = c_2 \cdot 2^d$, then for sufficiently large d we have that the bound doesn't hold, or equivalently, the following is not true for all $d \in \mathbb{N}$.

$$\log_2(c_2/2) \leq k' \cdot (H(\epsilon) - 1) \leq d \cdot (H(\epsilon) - 1)$$

$$\Leftrightarrow \frac{\log(2/c_2)}{1 - H(\epsilon)} \geq d.$$

□

We can now complete the proof of Theorem 4.4.4, as follows:

Proof of Theorem 4.4.4. The implication $(a) \Rightarrow (b)$ is given in the statement of Lemma 4.5.1. The implication $(b) \Rightarrow (c)$ only consists of the specification of the parameter ϵ_0 , and finding the constants c_1, c_2 corresponding to that ϵ_0 that give the property of (c_1, c_2, ϵ_0) -good sampling. Finally, the implication $(c) \Rightarrow (a)$ is precisely the contrapositive of Lemma 4.5.5. □

4.6 Comparison with Littlestone characterization

Let us restate in the language of this paper an existing characterization of Littlestone classes, from Claim 2.11 in the work of [105]. First we define an important object that appears both in this characterization and in Section 4.7.

Definition 4.6.1. *The k -half graph H_k is given by two disjoint sets of vertices $\{a_1, \dots, a_k\}$, $\{b_1, \dots, b_k\}$ so that*

$$(a_i, b_j) \in E \text{ if } i < j \text{ and } (a_i, b_j) \notin E \text{ if } j \geq i.$$

In a bigraph G , we say that it contains a k -half graph if there exist a pair of sets of vertices on each part of the vertex set partition, that follows the above configuration. Also, G is called k -edge stable, if it does not contain k -half graphs.

Above, note that not all edges or nonedges are specified between the pair of sets of vertices; therefore, the property of being k -edge stable forbids a family of configurations occurring in G , rather than a single one.

Having defined the k -half graph and k -edge stability, we state the following:

Theorem 4.6.2 ([105, Claim 2.11]). *Consider a hypothesis class $(X, \mathcal{H})$. The following properties are equivalent:*

- (a) *For all $\epsilon \in (0, 1/2)$, there is a constant $c(\epsilon)$ such that every finite $A \subseteq \mathcal{H}$ has $B \subseteq A$, with $|B| \geq c|A|$, that is ϵ -good.*
- (b) *For some $\epsilon < 1/2$ and some constant $c > 0$, every finite $A \subseteq \mathcal{H}$ has $B \subseteq A$ such that $|B| \geq c|A|$, that is ϵ -good.*
- (c) *$\text{Ldim}(\mathcal{H}) < \infty$; or, $\mathcal{H}$ is a Littlestone class.*
- (d) *The bigraph $(X, \mathcal{H})$ is k -edge stable for some finite $k > 0$.*

- (e) The bigraph $\text{Bip}(\mathcal{H}, X)$ is ℓ -edge stable for some finite $\ell > 0$.
- (f) The dual class $(\mathcal{H}, X)$ has finite Littlestone dimension.
- (g) For all $\epsilon < 1/2$, there is a constant $c(\epsilon)$ such that every finite $A \subseteq X$ has $B \subseteq A$, such that $|B| \geq c|A|$, that is ϵ -good.
- (h) For some $\epsilon < 1/2$ and some constant $c > 0$, every finite $A \subseteq X$ has $B \subseteq A$ such that $|B| \geq c|A|$, that is ϵ -good.

Thus, Theorem 4.4.4 provides a characterization similar to some of the equivalent conditions in the cited result above. The properties in Theorem 4.4.4 are weaker, since there exist hypothesis classes $\mathcal{H}$ where $\text{VC}(\mathcal{H})$ is bounded and $\text{Ldim}(\mathcal{H})$ is not.

4.7 Analysis of the half graph and some examples

To further explore the distinction between Theorem 4.4.4 and Theorem 4.6, let us consider some important basic examples in regard to a class with finite VC dimension: the one with a half graph structure. Before the end of the section we add a couple of examples for known hypothesis classes of bounded VC dimension and briefly study them.

We work explicitly on the following examples, in the hopes that understanding such properties will be constructive and lead to further understanding about the properties of VC classes.

4.7.1 Countable half graph

We begin with the following, analogous to Definition 4.6.1:

Definition 4.7.1. *The half graph $\mathcal{H}_\omega$ of countable size is given by two disjoint sets $\{a_i : i < \omega\}$, $\{b_i : i < \omega\}$ so that*

$$(a_i, b_j) \in E \text{ if } i < j \text{ and } (a_i, b_j) \notin E \text{ if } i \geq j.$$

Note that the VC dimension of H_ω is 1, and its Littlestone dimension is infinite. The observations we show still hold true if we just work with a finite k -half graph, which we denote $\mathcal{H}_k$. Its VC dimension is 1 and its Littlestone dimension is $\lfloor \log(k) \rfloor$.

4.7.2 Good sampling

The property of good sampling (see Definition 4.4.3) holds, as a corollary of Theorem 4.4.4, as we see below.

Example 4.7.2. *The half graph $\mathcal{H}_\omega$ has (c_1, c_2, ϵ) -good sampling for any $\epsilon \in (0, 1/2)$, where $c_1 = \epsilon/4$ and $c_2 = 1/4$.*

Proof. Consider any two finite sets $A \subseteq X$ and $B \subseteq \mathcal{H}$. Either (A, B) form an asymmetric $(\epsilon/2)$ -good pair, and we are done, or there exists $b_1 \in B$ which partitions $A = A_0 \sqcup A_1$ into vertices that connect and don't connect to it, and such that $|A_0| \geq (\epsilon/2) \cdot |A|$ and $|A_1| \geq (\epsilon/2) \cdot |A|$.

In the second case, the structure of $\mathcal{H}_\omega$ implies that all vertices in A_0 must have a smaller index than those of A_1 . Hence, there does not exist a second element h_2 of B that simultaneously induces a balanced partition in both A_0 and A_1 . Then, by Pigeonhole Principle, there is a set $B' \subseteq B$ of size at least $|B|/2$ with defined true value towards the same A_i ; let us take that set to be A_0 , without loss of generality. Then, (A_0, B') forms an asymmetric $(\epsilon/2)$ -good pair such that A_0 is $(\epsilon/2)$ -good to B' , $|A_0| \geq (\epsilon/2)|A|$, and $|B'| \geq |B|/2$.

Then, using the symmetrization argument from Lemma 4.5.2, we can extract two subsets $A'' \subseteq A_0$ and $B'' \subseteq B'$ such that (A'', B'') is symmetric ϵ -good, $|A''| \geq |A_0|/2 \geq \epsilon|A|/4$, and $|B''| \geq |B|/4$. $\square$

The result we just showed matches the following constructive proof of the existence of the ϵ -good symmetric pair (A', B') :

Proposition 4.7.3. *Consider $(X, \mathcal{H})$ with half graph structure $\mathcal{H}_\omega$ and two arbitrary subsets $A \subset X$ and $B \subset \mathcal{H}$. Let A_1 be the set given by the leftmost $c_1|A|$ vertices of A according to indexing. Let A_2 be the set given by the rightmost $c_2|A|$ vertices of A , correspondingly. Then, define B_1 and B_2 analogously. Then, at least one of the pairs (A_i, B_j) with $i, j \in \{1, 2\}$ is symmetric ϵ -good.*

Proof. Let $\ell(A_1)$ and $\ell(B_1)$ be the last indexes of elements of A_1 and B_1 , respectively. Also let $f(A_2)$ and $f(B_2)$ be the first indexes of the elements of A_2 and B_2 .

We claim that, either the sets of indexes of A_1 and B_2 are disjoint, or the sets of indexes of A_2 and B_1 are disjoint. By definition of the half graph structure, (A_1, B_2) is either a complete bigraph, or (A_2, B_1) is an empty bigraph, both of which are ϵ -symmetric good pairs.

Suppose this is not true for the sake of contradiction. Then, the following two inequalities must happen:

$$\ell(A_1) > f(B_2),$$

$$\ell(B_1) > f(A_2).$$

By the assumptions on the sizes of the subsets A_i, B_j , we have

$$f(B_2) \geq \ell(B_1) + (1 - 2c_2)|B|$$

$$f(A_2) \geq \ell(A_1) + (1 - 2c_1)|A|.$$

Threading all these inequalities together, we obtain the absurd:

$$\ell(A_1) > \ell(A_1) + (1 - 2c_1)|A| + (1 - 2c_2)|B|.$$

Therefore it must have been that one of (A_1, B_2) or (A_2, B_1) forms a complete or empty

bipartite graph, hence symmetric ϵ -good pair. $\square$

4.7.3 Interaction of B , ϵ , and $|\text{Anti}(B, \epsilon)|$

Next, we see that, given (A, B) a pair of finite subsets of X and $\mathcal{H}$, it is possible to have small subsets of B which have both *small* and *large* antigood sets in A , depending on the choice of ϵ and the *sparsity* of B .

One general observation is that there is a tradeoff between the value of ϵ and the size of antigood set of a set B of given size $|B|$. In particular, we study the nested sequence of sets:

$$\text{Anti}(B, \epsilon_0) \supseteq \text{Anti}(B, \epsilon_1) \supseteq \cdots \supseteq \text{Anti}(B, \epsilon_{\lceil |B|/2 \rceil}),$$

where the values of the parameters $\epsilon_1, \dots, \epsilon_{\lceil |B|/2 \rceil}$ are chosen within the respective intervals:

$$\begin{aligned} \epsilon_0 &\in \left(0, \frac{1}{|B|}\right) \\ \epsilon_1 &\in \left[\frac{1}{|B|}, \frac{2}{|B|}\right), \\ \epsilon_{\lceil |B|/2 \rceil - 1} &\in \left[\frac{\lceil |B|/2 \rceil - 1}{|B|}, \frac{\lceil |B|/2 \rceil}{|B|}\right), \\ &\vdots \\ \epsilon_{\lceil |B|/2 \rceil} &\in \left[\frac{\lceil |B|/2 \rceil}{|B|}, \frac{1}{2}\right). \end{aligned}$$

In particular, $\text{Anti}(B, \epsilon_{\lceil |B|/2 \rceil}) = \emptyset$ since no vertex can give an $\epsilon_{\lceil |B|/2 \rceil}$ -unbalanced partition of a set of size $|B|$.

The following gives a concrete expression for the size of the antigood set of any given subset of $\mathcal{H}_k$.

Proposition 4.7.4. *Let $(X, \mathcal{H})$ be a class forming the bipartite half graph H_ω . Let $B \subseteq \mathcal{H}$ defined as $B = \{h_m, \dots, h_M\}$, where m and M are the smallest and largest indexes. If*

$0 < \epsilon < 1/|B|$, then

$$\text{Anti}(B, \epsilon) = \{x_i : m \leq i \leq M - 1\} \subseteq A \text{ and } |\text{Anti}(B, \epsilon)| = M - m.$$

In particular, we can let B be so that the size of the antigood set is large, i.e., $M - m = \Theta(n)$ yet $|B| = o(n)$, by letting such set be sparse. We can also let B so that the size of the antigood set is small, i.e., $M - m = |B|$.

Proof. Any vertex set whose neighborhood is a proper subset of B is in the antigood set. By the half graph's structure, such set will contain precisely the vertices x_i , where the index i is at least m , and at most $M - 1$. $\square$

Example 4.7.5. Let $A = \{x_1, \dots, x_{15}\}$, $B = \{h_3, h_4, h_5, h_6, h_7\}$, and $\epsilon \in (0, 1/5)$. In this case, $\text{Anti}(B, \epsilon) = \{x_3, x_4, x_5, x_6\}$, which is the smallest possible nonzero size that the antigood set of a set in B of size 5 can have here.

If instead we had chosen $B = \{x_1, x_5, x_6, x_{11}, x_{15}\}$ and let $\epsilon \in (0, 1/5)$, then we have $\text{Anti}(B, \epsilon) = \{x_1, \dots, x_{14}\}$, which gives the largest possible size that the antigood set of a set in B can have.

One of the observations about the example above is that sparse sets have large antigood sets. In the following, we will show a stronger property than the existence of sets with large antigood sets. Moreover, it becomes highly likely that a subset of X in a half graph will have a large antigood set, due to its sparseness.

Proposition 4.7.6. Let $\epsilon > 0$ and consider the half graph $(X, \mathcal{H}_n)$. There exists $M = M(\epsilon) = O(1/\epsilon^2)$ such that the following holds. Let $H \subseteq \mathcal{H}_n$ be a subset of size at least M , sampled with respect to the uniform probability distribution $\mathcal{D}$ over $[n]$. With high probability of at least $(1 - \epsilon)$, H has a large antigood set, and more concretely:

$$\Pr_{H \sim \mathcal{D}^M} [|\text{Anti}(H, \epsilon)| \geq n - 2\epsilon n] \geq 1 - \epsilon.$$

Proof. Consider the set system $\mathcal{S} = \{S_1, \dots, S_{\lfloor 1/\epsilon \rfloor}\}$ on the ground set $[n]$, where each set S_i is one of $\lfloor 1/\epsilon \rfloor$ nearly equally sized parts $S_i := \{s_{i-1} + 1, \dots, s_i\}$, and the values s_i are given by:

$$s_0 := 0, \text{ and for all } i \geq 1, s_i := s_{i-1} + \left\lfloor \frac{n}{\lfloor 1/\epsilon \rfloor} \right\rfloor + \mathbb{1}[i \leq n \bmod \lfloor 1/\epsilon \rfloor].$$

Since $\text{VC}(\mathcal{S}) = 1$, the Fundamental Theorem of Statistical Learning (see Section 4.11.2) implies that this set system has the uniform convergence property. Namely, for any $\epsilon > 0$, there exists $M = M(\epsilon) = O(d/\epsilon^2)$ (which is $O(1/\epsilon^2)$ since $d = 1$), such that:

$$\Pr_{H \sim \mathcal{D}^M} \left[\sup_{i=1, \dots, n/\epsilon} |\mu_H(S_i) - \mu(S_i)| < \epsilon \right] \geq 1 - \epsilon,$$

where μ_H is the empirical counting measure on $[n]$ and μ is the counting measure on $[n]$. In particular, the above implies that for all $i = 1, \dots, \lfloor 1/\epsilon \rfloor$, it must happen that $H \cap S_i \neq \emptyset$. Otherwise:

$$\mu_H(S_i) = \frac{|H \cap S_i|}{n} = 0 \text{ whereas } \mu(S_i) \geq \frac{\epsilon n}{n} = \epsilon,$$

a contradiction. In particular, $H \cap S_1 \neq \emptyset$ and $H \cap S_{n/\epsilon} \neq \emptyset$, and by Proposition 4.7.4,

$$|\text{Anti}(H, \epsilon)| \geq (n - \lfloor n\epsilon \rfloor) - \lfloor n\epsilon \rfloor \geq n - 2\epsilon n. \quad \square$$

In particular, in examples where the graph has finite VC dimension and infinite Littlestone dimension, it is possible to find a large half graph, in which many instances of sets with large antigood sets can be found, as Proposition 4.7.6 shows.

4.7.4 Other examples of VC classes

Before we conclude this section and study bounds that can be obtained using different techniques, let us visit two more possible examples of VC classes in which, with an appropriate

choice of $\epsilon > 0$, antigood sets may be large.

The first example consists of the class of *intervals* in $\mathbb{R}$. Let us denote this class $(X, \mathcal{H}_I)$, and define it as the class on the input domain $X = \mathbb{R}$ of all closed interval memberships, namely:

$$\mathcal{H}_I = \{h_{a,b} := \mathbb{1}_{[a,b]} : a, b \in \mathbb{R}\}$$

We know that $\text{VC}(\mathcal{H}_I) = 2$, and have the following:

Example 4.7.7. Consider B to be the set of all functions $h_{a,b} := \mathbb{1}_{[a,b]}$ given by the intervals $[a, b]$, where a and b take integer values from 0 to N . Consider A to be the set of real numbers $\{-0.5, 0.5, \dots, N + 0.5\}$. Then, if $\epsilon < 1/|N|$, $\text{Anti}(B, \epsilon) = \{0.5, \dots, N - 0.5\}$, which is large in A , since

$$\frac{|\text{Anti}(B, \epsilon)|}{|A|} = \frac{N - 1}{N + 1}.$$

The second example consists of the class of *rectangles* in $\mathbb{R}^2$. Let us denote this class $(X, \mathcal{H}_R)$, and define it as the class on the input domain $X = \mathbb{R}^2$ of all closed rectangle memberships, namely:

$$\mathcal{H}_R = \{h_{a,b,c,d} := \mathbb{1}_{[a,b] \times [c,d]} : a, b, c, d \in \mathbb{R}\}$$

We know that $\text{VC}(\mathcal{H}_R) = 4$, and have the following:

Example 4.7.8. Consider B to be the set of all functions $h_{a,b,c,d}$ given by the rectangles $[a, b] \times [c, d]$, where a, b, c and d take integer values from 0 to N . Consider A to be the set of points $\{(a - 0.5, b - 0.5) : a, b \in \{0, \dots, N\}\}$. Then, if $\epsilon < N/\binom{|N|}{2}^2$, $\text{Anti}(B, \epsilon) = \{(a - 0.5, b - 0.5) : 1 \leq a, b \leq N\}$, which is large in A , since

$$\frac{|\text{Anti}(B, \epsilon)|}{|A|} = \frac{N^2}{(N + 1)^2}.$$

4.8 Comparison with proof via Haussler Packing

In this section we present an alternative way to find a large set with a small antigood set by applying directly Haussler's Packing Lemma [86]. We compare the outcome of this method if applied in a setup with the same initial conditions. Please see the cited results and lemmas in Section 4.11.3. In the setting we work on, we are given $A \subseteq X$ and $B \subseteq \mathcal{H}$ finite sets, and $\text{VC}(\mathcal{H}) = d$. Our goal for this section will be to show that a lower bounded sized portion of the elements in B lies within some ball of radius δ , which then has a small antigood set. In order to do so, we can apply the following implication of Haussler's Packing Lemma:

Lemma 4.8.1. *If (A, B) is a finite pair in $(X, \mathcal{H})$ whose VC dimension is d , then there exists some element $b \in B$ whose δ neighborhood $B_\delta(b)$ is of size at least $|B| \cdot ((d+1)e)^{-1} \cdot (\delta/2e)^d$.*

Proof. Let a set C be constructed by adding elements of B to it so that the elements added so far are always pairwise δ -separated. By Haussler's Packing Lemma, this process must stop after no more than $e(d+1)(2e/\delta)^d$ many steps. After doing this, no elements can be added, hence any other element in B has to be δ -close to at least one element of C . By Pigeonhole Principle, at least $|B|/(e(d+1)(2e/\delta)^d) = |B| \cdot \delta^d / ((d+1) \cdot 2^d \cdot e^{d+1})$ elements of B are δ -close to the same element $b \in B$. Thus, $|B_\delta(b)| \geq |B| \cdot ((d+1)e)^{-1} \cdot (\delta/2e)^d$. $\square$

In the remainder of this chapter, we will use the following lower bound for simplicity of notation, which is true for all values $d \geq 1$: $((d+1)e)^{-1} \cdot (\delta/2e)^d \geq (\epsilon/60)^d$. Namely, so long as $1 \leq \text{VC}(\mathcal{H}) = d < \infty$ (the left inequality only fails when $\mathcal{H} = \{X\}$ or $\mathcal{H} = \{\emptyset\}$), and $\delta < \epsilon/2$, Lemma 4.8 implies that there exists some $b \in B$ such that $|B_\delta(b)| \geq |B| \cdot (\epsilon/60)^d$.

Proposition 4.8.2. *For any two h_1, h_2 in $B_\delta(h_0) \subseteq \mathcal{H}$, the distance between these two functions is $d(h_1, h_2)$ less than 2δ .*

Proof. This fact is a direct application of the triangle inequality for the error distance:

$$d(h_1, h_2) = \frac{|S_{h_1} \Delta S_{h_2}|}{|X|} \leq \frac{|S_{h_1} \Delta S_{h_0}|}{|X|} + \frac{|S_{h_0} \Delta S_{h_2}|}{|X|} < \delta + \delta.$$

$\square$

Proposition 4.8.3. *Let $\epsilon \in (0, 1/2)$, and let $B = B_\epsilon(h_0)$ be any δ -ball centered at a hypothesis h_0 . Then the size of the set $\text{Anti}(B, \epsilon)$ is no larger than:*

$$\min \left\{ \frac{|B| - 1}{|B|} \cdot \frac{\delta}{\epsilon(1 - \epsilon)} \cdot |A|, |A| \right\}.$$

Note that $|A|$ is always larger than the antigood set's size: this statement is always true regardless of the choice of $\delta > 0$. As we will see later, certain particular choices of δ will be so that $|A|$ is significantly larger than $|\text{Anti}(B, \epsilon)|$.

Proof. In this proof, we will double count the size of the following set of triples

$$T := \{(h_1, h_2, x) \in B \times B \times A : h_1(x) \neq h_2(x)\}.$$

Consider any $x \in \text{Anti}(B, \epsilon) \subseteq A$. Since this element is in the antigood set, the size of both its neighborhood and its complement in B is between $\epsilon|B|$ and $(1 - \epsilon)|B|$. Since the function $x(1 - x)$ is decreasing with respect to $|x - 0.5|$, we have that:

$$|\{(h_1, h_2) \in B \times B : h_1(x) \neq h_2(x)\}| \geq 2 \cdot \epsilon|B| \cdot (1 - \epsilon)|B|.$$

If we add together over all elements x in the antigood set, we get

$$|T| \geq \epsilon|B| \cdot (1 - \epsilon)|B| \cdot |\text{Anti}(B, \epsilon)|.$$

Now, consider any pair of distinct elements $(h_1, h_2) \in B \times B$. By Proposition 4.8.2, any such tuple is 2δ -close and therefore

$$|\{x \in A : h_1(x) \neq h_2(x)\}| \leq 2\delta \cdot |A|.$$

If we add together over all the possible ordered pairs (h_1, h_2) in $B \times B$, we get that

$$|T| \leq 2\delta \cdot |A| \cdot |B| \cdot (|B| - 1).$$

The above inequalities are lower and upper bounds for the same count for the size of the set T . Threading the two together, and then simplifying, we obtain the claim, which is an upper bound on the antigood set size. $\square$

Note that the above upper bound is stronger than $|A|$ only if δ is not much larger than ϵ . In particular, $\delta \leq 2\epsilon$ since we need:

$$\frac{\delta}{\epsilon} < (1 - \epsilon) \cdot \frac{|B|}{|B| - 1} < 2.$$

One implication of the work done above is that we can now reprove Corollary 4.5.3, now using Proposition 4.8.3 and therefore obtaining distinct constant values. In particular, if we set $\delta = \epsilon/2$, we get (c_1, c_2, ϵ) -good sampling with:

$$c_1 = \frac{1}{2} \cdot \left(\frac{\epsilon}{60}\right)^d, \quad c_2 = \frac{1}{2} \cdot \left(1 - \frac{|B| - 1}{2 \cdot |B| \cdot (1 - \epsilon)}\right).$$

The reader may wonder how this result compares with the bounds from Corollary 4.5.3. We can point out in the case that $\delta = \epsilon/2$, that even if the bound on the size of the antigood set is better (or the symmetric good pair is larger in size):

$$|\text{Anti}(B, \epsilon)| \leq \frac{|B| - 1}{|B|} \cdot \frac{1}{2(1 - \epsilon)} \cdot |A|.$$

Still, the lower bound on the resulting set $B = B_\delta(h_0)$ is much smaller, as its size is lower bounded only by $B_\delta(b)$ is of size at least $|B| \cdot (\epsilon/60)^d$.

4.9 Comparison with proof via VC Regularity Lemma

Following up on the proof of Theorem 4.5.3, we can also find an alternative way to obtain good pairs by usage of the VC Regularity Lemma [16, 101, 68]. The statement of this regularity lemma can be found in Section 4.11.

In this setting, we are again given $A \subseteq X$ and $B \subseteq \mathcal{H}$ finite sets, and $\text{VC}(\mathcal{H}) = d$. We can then directly use the equipartition obtained from applying this regularity lemma in order to obtain the following bounds:

Proposition 4.9.1. *Consider a class $(X, \mathcal{H})$ such that $\text{VC}(\mathcal{H}) = d$, finite subsets $A \subseteq X$, $B \subseteq \mathcal{H}$, and a parameter $\epsilon \in (0, 1/204)$. It is possible to find a symmetric $(100^2\epsilon/99)$ -good pair (A', B') of subsets of A and B , respectively, such that, for some $m = O(1/\epsilon^{2d+1})$:*

$$|A'| \geq 99 \cdot |A|/(100 \cdot m),$$

$$|B'| \geq 99 \cdot |B|/(100 \cdot m).$$

Proof. Using the VC Regularity Lemma, let us take first an equipartition of the bigraph (A, B) such that each of A, B are partitioned into $m = m(\epsilon)$ nearly-equal sized parts, with the bound $m = O(1/\epsilon^{2d+1})$, and such that all but at most $\epsilon \cdot m^2$ of them are ϵ -homogeneous, i.e., they have edge density in $[0, \epsilon) \cup (1 - \epsilon, 1]$. In particular, let (A, B) be one of these ϵ -homogeneous pairs.

Without loss of generality, $e(A, B) < \epsilon|A| \cdot |B|$. We define the sets

$$S_A := \{a \in A : |N(a) \cap B| \geq 100\epsilon \cdot |B|\}, \text{ and}$$

$$S_B := \{b \in B : |N(b) \cap A| \geq 100\epsilon \cdot |A|\}.$$

For $e(A, B) < \epsilon|A| \cdot |B|$ to hold, it must be the case that $|S_A| \leq |A|/100$ and $|S_B| \leq |B|/100$.

Defining $A' := A \setminus S_A$ and $B' = B \setminus S_B$, we have that

$$|A'| \geq |A| \cdot \frac{99}{100} \quad \text{and} \quad |B'| \geq |B| \cdot \frac{99}{100};$$

$$\implies \forall a \in A', e(a, B') \leq e(a, B) \leq 100\epsilon \cdot |B| \leq \frac{100^2\epsilon}{99} \cdot |B'|.$$

Similarly, for every $b \in B'$, the number of edges from b to A' is at most $(100^2\epsilon/99) \cdot |A'|$, which implies that the pair (A', B') is symmetric $(100^2\epsilon/99)$ -good (and thus, 102ϵ -good, with $102\epsilon \in (0, 1/2)$). This finishes the proof. $\square$

Let us put into perspective the bounds on the size of the symmetric pair obtained from Proposition 4.9.1 with those of the pair we previously obtained. Since the number of parts was $m = O(1/\epsilon^{2d+1})$, then the $|A'| \geq 99|A|/(100m) = \Omega(|A| \cdot \epsilon^{2d+1})$ (and $|B'| = \Omega(|B| \cdot \epsilon^{2d+1})$). This gives a worse bound at least in the sense, that the constant values $c_1 = \epsilon^d/2^{d+1}$ and $c_2 = 1/2^{d+1}$ from Theorem 4.4.4 result in the existence of a pair with a higher lower bound in size.

The following is one more application of the VC Regularity Lemma towards showing the existence of good pairs or sets with bounded antigood sets in VC classes:

Corollary 4.9.2. *Let $n > 0$ be a positive integer, and $\epsilon > 0$ be a positive real such that $\epsilon < 1/n$. If $k \in \mathbb{N}$, then there exists a constant $c = c(k)$ such that: for every class $(X, \mathcal{H})$ with $\text{VC}(\mathcal{H}) = d < \infty$ and every finite subsets $A \subseteq \mathcal{H}$ and $B \subseteq X$, there is some set $A' \subseteq A$ of size $|A'| \geq c^{-1} \cdot |A| \cdot \epsilon^{2d+1}$ such that*

$$|\text{Anti}(A', \epsilon n) \cap B| \leq |B| \cdot \frac{2}{n}.$$

Proof. By the VC Regularity Lemma, there exists a near equipartition of each of A and B into $m \leq c \cdot (1/\epsilon)^{2d+1}$ parts, for some constant $c = c(d)$, such that all but at most $\epsilon \cdot m^2$ of the pairs of parts obtained are ϵ -homogeneous. Suppose that A is partitioned as $\bigcup_{i \in [m]} S_i$, and that B is partitioned as $\bigcup_{j \in [m]} T_j$. By Pigeonhole Principle, at least one of the resulting

parts, without loss of generality $A' := S_1$, does not form an ϵ -homogeneous pair with $\leq \epsilon \cdot m$ pairs, which must be among the T_j . For those T_j such that (A', T_j) is ϵ -homogeneous, either $e(A', T_j) < \epsilon \cdot |T_j| \cdot |A'|$, or $ne(A', T_j) < \epsilon \cdot |T_j| \cdot |A'|$ (here, ne stands for *non-edges*). In the first case, there are less than $|T_j|/n$ vertices $v \in T_j$ such that $e(v, A') \geq \epsilon n |A'|$. In the second case, there are less than $|T_j|/n$ vertices $v \in T_j$ such that $ne(v, A') \geq \epsilon n |A'|$. In either case, then, $|\text{Anti}(A', \epsilon n) \cap T_j| < |T_j|/n$. We can get the following upper bound by aggregating:

$$\begin{aligned}
|\text{Anti}(A', \epsilon n)| &= \sum_{i=1}^{m-1} |\text{Anti}(A', \epsilon n) \cap T_j| \\
&\leq \sum_{i: (T_j, A') \text{ } \epsilon\text{-hom.}} \frac{|T_j|}{n} + \sum_{i: (T_j, A') \text{ not } \epsilon\text{-hom.}} |T_j| \\
&\leq \frac{|B|}{n} + \left(1 - \frac{1}{n}\right) \cdot \sum_{i: (T_j, A') \text{ not } \epsilon\text{-hom.}} |T_j| \\
&= |B|(\epsilon - \epsilon/n + 1/n)
\end{aligned}$$

By the condition on ϵ , the above is less than $|B| \cdot (2/n)$. $\square$

The above shows that for finite bigraphs (A, B) embedded in a class of bounded VC dimension, there exists a large set $A' \subseteq A$ such that $|\text{Anti}(A', \epsilon n)| \leq 2|B|/n$. Namely, its antigood set with error parameter $n\epsilon$, has an upper bounded size.

It remains to explore how large $|\text{Anti}(A', \epsilon n)|$ is. In fact, the upper bound for the ratio is not small a priori, especially if ϵ is close to zero.

$$\frac{|\text{Anti}(A', n\epsilon)|}{|B|} \leq \frac{|B| \cdot 2/n}{|B| \cdot c^{-1} \cdot \epsilon^{2d+1}} = \frac{2c}{n \cdot \epsilon^{2d+1}}.$$

The above uses of the VC regularity lemma suggest that it is worth clarifying the interactions between the pair property of being ϵ -homogeneous and that of being symmetric ϵ -good.

4.10 Concluding remarks

This work provides a characterization of classes with bounded VC dimension. A few open questions include:

1. What results can be shown in the case that we work with graphs, instead of bigraphs?
2. How does the existence of ϵ -good pairs interact with learning theoretical questions and existing results? In particular, part of the Fundamental Theorem of Statistical Learning may have interactions with this characteristic of VC classes.
3. Let us denote this class $(X, \mathcal{H}_{H,n})$, and define it as the class on the input domain $X = \mathbb{R}^n$ of all closed halfspace memberships, namely:

$$\mathcal{H}_{H,n} = \{\{x \in \mathbb{R}^n | \langle x, a \rangle \geq 0\} : \vec{a} \in \mathbb{R}^n \setminus \{\vec{0}\}\}$$

Knowing that $\text{VC}(\mathcal{H}_{H,n}) = n + 1 < \infty$, a possible subject of further study is the good sampling property for this class.

4. Is there anything from our analysis that can allow us to find half-graphs by using the ϵ -good pairs shown in Corollary 4.5.3, or in any of the other analogous results that use different technologies?
5. Is there any other technology that could be used in order to improve the bounds obtained by using Haussler's Packing Lemma as done in Section 4.8?

4.11 Appendix: referenced and used results

4.11.1 Regularity Lemma

A large collection of work in extremal combinatorics concerns the large scale properties of certain classes of graphs. In particular, some classes with forbidden configurations are

found to satisfy stronger versions of regularity lemmas. Szemerédi's Regularity Lemma is a result in extremal combinatorics that allows us to approximate large graphs with smaller ones of fixed size. Below, an ϵ -regular pair (A, B) is one such that for every $A' \subseteq A$ and $B' \subseteq B$, such that $|A'| \geq \epsilon|A|$ and $|B'| \geq \epsilon|B|$, the difference between the edge densities $d(A, B) := e(A, B)/|A| \cdot |B|$ and $d(A', B') := e(A', B')/(|A'| \cdot |B'|)$ is less than ϵ .

Theorem 4.11.1 (Szemerédi's Regularity Lemma, [140]). *For every $\epsilon > 0$ there is $N(\epsilon) \in \mathbb{N}$ so that every finite graph G may be partitioned into m classes $V_1 \cup \dots \cup V_m$ where $m \leq N$ and*

- (i) *all of the pairs V_i, V_j satisfy $||V_i| - |V_j|| \leq 1$;*
- (ii) *all but at most ϵm^2 of the pairs (V_i, V_j) are ϵ -regular.*

One caveat of the above is that $N(\epsilon)$ may be a tower of exponents, as shown by Gowers in [74], and in the lower bound construction in [75]; such tower dependence was also shown in [66, 48, 116] and was shown to be characterized by infinite VC dimension. Another caveat is the existence of irregular pairs, which is characterized by the existence of half graphs.

For graphs of bounded VC-dimension, it is possible to find a smaller $N(\epsilon)$ than originally, concerning ϵ -homogeneous pairs (A, B) , which are such that $d(A, B) \in [0, \epsilon) \cup (1 - \epsilon, 1]$.

Theorem 4.11.2 (Bounded-VC Regularity Lemma [101, 16, 68]). *For any $\epsilon > 0$, there is a $N = N(\epsilon) \in \mathbb{N}$ such that the vertex set V of any finite graph G with VC-dimension $d < \infty$ has a partition into at most $m \leq N$ parts. Here, $N \in [8/\epsilon, c \cdot \epsilon^{-2d+1}]$, where $c = c(d)$ is a constant only depending on d , and*

- (i) *all of the pairs V_i, V_j satisfy $||V_i| - |V_j|| \leq 1$;*
- (ii) *all but at most $\epsilon \cdot N^2$ pairs are ϵ -homogeneous.*

An even stronger property holds if we focus on the class of k -edge stable graphs. Namely, there is a Regularity Lemma that gives stronger bounds and properties, as stated below.

Assumption	No. parts	Guarantees
General	$N(\epsilon)$	all but $m\epsilon^2$ pairs ϵ -regular
$VC(\mathcal{H}) = d$	$N = O(1/\epsilon^{2d+1})$	all but $k\epsilon^2$ pairs ϵ -homogeneous
k -stable	$N = O\left(1/\epsilon^{2^{k+3}-7}\right)$	all pairs ϵ -regular

Table 4.1: Three different Regularity Lemmas.

Theorem 4.11.3 (Stable Regularity Lemma [107]). *For each $\epsilon > 0$ and $k \in \mathbb{N}$ there is $N = N(\epsilon, k)$ such that for a sufficiently large finite k -edge stable graph $G = (V, E)$, there exists $\ell \leq N$ such that V can be partitioned into disjoint pieces $A_1, \dots, A_\ell$ such that*

- (i) $||A_i| - |A_j|| \leq 1$ for all i, j ,
- (ii) (A_i, A_j) is ϵ -regular for all i, j ,
- (iii) A_i is ϵ -excellent for all i ,
- (iv) $N < (4/\epsilon)^{2^{k+3}-7}$.

Above, the $\text{poly}(\epsilon^{-1})$ bound on the number of parts and the guarantee that all pairs will be ϵ -regular show that the class of stable graphs enjoy stronger regularity properties than the class of all graphs.

4.11.2 Fundamental Theorem of Statistical Learning

The work of [27] provides a characterization of PAC-learnability by finiteness of VC dimension:

Theorem 4.11.4. *Let $\mathcal{H} \subseteq \{\pm 1\}^X$ be a hypothesis class on the input domain X . The following properties are equivalent:*

1. $\mathcal{H}$ is (agnostic) PAC-learnable;
2. $\mathcal{H}$ is (realizable) PAC-learnable;

3. $\text{VC}(\mathcal{H}) = d$ is finite;
4. $\mathcal{H}$ has the uniform convergence property;
5. there exists an ERM algorithm that learns $\mathcal{H}$.

In particular, the quantitative statement (see e.g., [130]) states that the uniform convergence property of $\mathcal{H}$ is satisfied with sample complexity $m_{\mathcal{H}}^{\text{UC}}(\epsilon, \delta)$:

$$C_1 \cdot \frac{d + \log(1/\delta)}{\epsilon^2} \leq m_{\mathcal{H}}^{\text{UC}}(\epsilon, \delta) \leq C_2 \cdot \frac{d + \log(1/\delta)}{\epsilon^2},$$

where C_1 and C_2 are constants. In other words, the uniform convergence property is satisfied as long as $m_{\mathcal{H}}^{\text{UC}}(\epsilon, \delta)$ many samples were taken. This is $O(d/\epsilon^2)$ when the value of δ has been fixed.

4.11.3 Haussler's Packing Lemma

We cite the results and notions we used from the work in [86], beginning with the following distance notion among elements of $\mathcal{H}$ in the pair $(X, \mathcal{H})$. Here, we assume $|X|$ to be finite.

Definition 4.11.5. *For any two hypotheses $h_1, h_2 \in \mathcal{H}$, their distance is given by the counting measure*

$$d(h_1, h_2) := \frac{|S_{h_1} \Delta S_{h_2}|}{|X|},$$

where $S_h := \{x \in X : h(x) = 1\}$. Then, an ϵ -ball around an element h_0 is defined as follows:

$$B_{\epsilon}(h_0) := \{h \in \mathcal{H} : d(h, h_0) < \epsilon\}.$$

In other words, a pair of functions is ϵ -close if they roughly look the same to the sample X , and we can also refer to this pair of functions to be ϵ -close in the error measure.

In the remaining part of this section, we will consider mostly interchangeably the notions of the pair $(X, \mathcal{H})$ as a set system as in the sense that $\mathcal{H} \subseteq 2^X$, and as a hypothesis class in

the sense that each element $h \in \mathcal{H}$ is a function $h : X \rightarrow \{0, 1\}$, since they will allow us to work with the same combinatorial structure.

A next definition that will be needed to understand the statement and use of Haussler's Packing Lemma is the following:

Definition 4.11.6. *A set system $(X, \mathcal{H})$ on X is δ -separated when $|h_1 \Delta h_2| \geq \delta|X|$, for all $h_1, h_2 \in \mathcal{H}$.*

An alternative way of stating the above is that if a set system is δ -separated, then for every pair of elements h_1, h_2 in it, it has to happen that $h_1 \notin B_\epsilon(h_2)$, and vice versa. Haussler's Packing Lemma notes that a general set system of bounded VC dimension cannot be both large and δ -separated.

Theorem 4.11.7 (Haussler's Packing Lemma [86]). *Let $d, \delta > 0$. If $(X, \mathcal{H})$ are such that $VCdim(\mathcal{H}) \leq d$, there are at most*

$$e(d+1) \left(\frac{2e}{\delta} \right)^d$$

elements in $\mathcal{H}$ if it is a δ -separated system of X .

CHAPTER 5

OPEN QUESTIONS AND EXAMPLES

This chapter covers an initial exploration of further research directions and examples that follow from the material covered in the earlier chapters.

5.1 Follow ups from Chapter 3

In Chapter 3 we define ϵ -saturation of hypothesis classes and of graphs. A question that follows, is whether there is some degree of robustness, or error tolerance on the Littlestone, VC or threshold dimension of the initial object. Namely, is there a degree to which a given bigraph $(X, \mathcal{H})$ can be *close* to the a graph $(X, \mathcal{H}_0)$ such that $\text{Ldim}(\mathcal{H}_0) = \ell$, and have its saturation be of the same complexity as that of $\mathcal{H}_0$, namely $\text{Ldim}((\mathcal{H}_0)_\infty(\epsilon)) = \text{Ldim}(\mathcal{H}_\infty(\epsilon))$, as well as $\text{VC}((\mathcal{H}_0)_\infty(\epsilon)) = \text{VC}(\mathcal{H}_\infty(\epsilon))$ and $\text{Thr}((\mathcal{H}_0)_\infty(\epsilon)) = \text{Thr}(\mathcal{H}_\infty(\epsilon))$?

In the following first example, we begin to approach this question by noting that a single edge modification can cause a drop in threshold dimension, not to be recovered by the process of ϵ -saturation:

Proposition 5.1.1. *Consider a half graph $(X_k, \mathcal{H}_k)$ of finite length k . Then, let us modify one specific edge of $\mathcal{H}_k$ in order to obtain $\mathcal{H}'$. In particular, let $h_1(x_1) = 1$ in $\mathcal{H}'$ and the rest stay the same, i.e. $h_j(x_i) = \mathbb{1}[i < j]$. Then $(\mathcal{H}')_\infty(\epsilon)$ has threshold dimension at most $k - 1$, i.e., $(\mathcal{H}')_\infty(\epsilon)$ does not contain a half graph of length k .*

Proof. Note that X_k stays the same, so let us now show the threshold dimension of the ϵ -saturation $(\mathcal{H}')_\infty(\epsilon)$ can be at most k .

With this modification, all the hypotheses $h_j \in \mathcal{H}'$ are such that $h_j(x_1) = 1$. Thus, for every ϵ -hypothesis $H \in (\mathcal{H}')_1(\epsilon)$, it must happen that $H(x_1) = 1$, since no matter what finitely supported probability distribution H is given by, the sum of weights of $\{h_j : h_j(x_1) = 1\}$ is $1 > 1 - \epsilon$. Suppose for the sake of contradiction that for some $i \in \{1, \dots, k - 1\}$ we

have that $H(x_i) = 0$ and $H(x_{i+1}) = 1$. By the above argument we must have $H(x_1) = 1$, thus $i \geq 2$ is necessary for this. Let H be given by the probability distribution $\{(h_i, \gamma_i) : i < k\}$ where possibly some weights γ_i are zero. The assumed conditions in terms of weight inequalities are:

$$\begin{aligned} H(x_i) = 0 & \Rightarrow \gamma_{i+1} + \cdots + \gamma_k < \epsilon, \\ H(x_{i+1}) = 1 & \Rightarrow \gamma_1 + \cdots + \gamma_{i+1} < \epsilon. \end{aligned}$$

Adding the two together, $(\gamma_{i+1} + \cdots + \gamma_k) + (\gamma_1 + \cdots + \gamma_{i+1}) < 2\epsilon$, which gives a contradiction since the left hand side equals $(1 + \gamma_{i+1})$ and is greater than or equal to 1, while the right hand side equals $2\epsilon < 1$. With this we show our subclaim that $H(x_i) \geq H(x_{i+1})$ for all i .

This implies that every element $H \in (\mathcal{H}')_1(\epsilon)$ is such that

$$(H(x_1), H(x_2), \dots, H(x_k)) = (\underbrace{1, \dots, 1}_{>0}, 0, \dots, 0),$$

thus all the ϵ -hypotheses of $\mathcal{H}'$ are already in its set of hypotheses. This implies that $\mathcal{H}' = (\mathcal{H}')_1(\epsilon) = \cdots = \cup_{n < \omega} (\mathcal{H}')_n(\epsilon) = (\mathcal{H}')_\infty(\epsilon)$. Therefore, threshold dimension stays the same as $\text{Thr}(\mathcal{H}') = k - 1$, where a bi-induced length $(k - 1)$ half graph can be found among $\{x_2, \dots, x_k\} \cup \{h_2, \dots, h_k\}$, but one of length k cannot be found. $\square$

While it was simple to lower threshold dimension by 1 with a single and specific edge modification, a natural question to follow is whether we can do so by modifying *any* edge. A partial answer is the following:

Lemma 5.1.2. *Let $\mathcal{H}_k$ be a half graph of size k . Let $\mathcal{H}'$ be obtained from $\mathcal{H}_k$ by modifying the edge between h_i and x_i . Then, $(\mathcal{H}')_1(\epsilon) = (\mathcal{H}')_\infty(\epsilon)$, and in particular, $\text{Thr}((\mathcal{H}')_\infty(\epsilon)) = k - 1$.*

Proof. By symmetry, the cases $i = j = 1$ and $i = j = k$ can be handled in a similar way. We thus assume that $1 < i = j < k$ in this proof. The following must hold for any ϵ -hypothesis

$H \in (\mathcal{H}')_1(\epsilon)$:

1. For every $m \neq i, \ell \neq i$ such that $m \leq \ell$, it must hold that $H(x_m) \geq H(x_\ell)$,
2. For every $m < i$, it must hold that $H(x_m) \geq H(x_i)$,
3. For every $\ell > i$, it must hold that $H(x_i) \geq H(x_\ell)$,
4. $H(x_k) = 0$.

The first fact can be proved in a similar way as in the previous lemma, since the outgoing edges from x_m and x_ℓ were unaffected.

The second fact can be proved similarly. If the claimed inequality isn't true, then

$$\begin{aligned} H(x_m) = 0 & \Rightarrow \gamma_{m+1} + \dots + \gamma_k < \epsilon, \\ H(x_i) = 1 & \Rightarrow \gamma_1 + \dots + \gamma_{i-1} < \epsilon. \end{aligned}$$

And since $i - 1 \geq m$, adding these two inequalities we get $1 < 2\epsilon$, a contradiction.

The third fact can be proved similarly by assuming the contrary

$$\begin{aligned} H(x_i) = 0 & \Rightarrow \gamma_i + \dots + \gamma_k < \epsilon, \\ H(x_\ell) = 1 & \Rightarrow \gamma_1 + \dots + \gamma_\ell < \epsilon. \end{aligned}$$

Since $i \geq \ell + 1$, the addition of the two inequalities gives $1 < 2\epsilon$, a contradiction.

The fourth fact is a consequence of the fact that for every $h \in \mathcal{H}'$, $h(x_k) = k$.

These constraints together imply that there are k possible ϵ -hypotheses in $(\mathcal{H}')_1(\epsilon)$: these include $h_1, \dots, h_{i-1}$, then $h_{i+1}, h_{i+2}, \dots, h_k$, and a hypothesis H such that

$$(H(x_1), \dots, H(x_k)) = (\underbrace{1, \dots, 1}_{i-1}, \underbrace{0, \dots, 0}_{k-(i-1)}).$$

This last potential ϵ -hypothesis H is not possible: it would imply the two inequalities below, which added give $1 < 2\epsilon$, a contradiction:

$$\begin{aligned} H(x_{i-1}) = 1 &\Rightarrow \gamma_i + \cdots + \gamma_k < \epsilon, \\ H(x_i) = 0 &\Rightarrow \gamma_1 + \cdots + \gamma_{i-1} < \epsilon. \end{aligned}$$

Summarizing, $\mathcal{H}'_1(\epsilon) = \mathcal{H}'$. Then, every step of ϵ -saturation will leave $\mathcal{H}'$ the same, then $\mathcal{H}'_n(\epsilon) = \mathcal{H}'$, implying $\mathcal{H}'_\infty(\epsilon) = \mathcal{H}'$. $\square$

A possible follow up from this lemma would be that any edge modification of the half graph could induce a reduction in threshold dimension, not to be recovered by ϵ -saturation. This is not shown within this thesis, but it is stated as a conjecture:

Conjecture 5.1.3. *Let $\mathcal{H}_k$ be a half graph of size k . Let $\mathcal{H}'$ be obtained from $\mathcal{H}_k$ by modifying the edge between h_j and x_i , for some $1 \leq i, j \leq k$. Then, $(\mathcal{H}')_1(\epsilon)$ does not contain a half graph of length k .*

On a separate note, the following lemma hopes to provide a tool for further exploration of ϵ saturation of certain graphs that may be obtained from the disjoint union operation:

Lemma 5.1.4. *If $\mathcal{H} = \bigsqcup_{i < \omega} \mathcal{H}_i$, then $\mathcal{H}_\infty(\epsilon) = \bigsqcup_{i < \omega} (\mathcal{H}_i)_\infty(\epsilon)$.*

Proof. We will show this by induction over each of the partial steps $n < \omega$ of the saturation process of $\mathcal{H}$. At $n = 0$, the claim that $\mathcal{H}_0(\epsilon) = \bigsqcup_{i < \omega} (\mathcal{H}_i)_0(\epsilon)$ is true by hypothesis.

Now, we show that if the statement is true for some $n < \omega$ then it is also true for $n + 1$. We already know the containment $\mathcal{H}_{n+1}(\epsilon) \subseteq \bigsqcup_{i < \omega} (\mathcal{H}_i)_\infty(\epsilon)$ holds, because every ϵ -good set in each of the individual bigraphs is also an ϵ -good set in their union. Next, we will show this containment is actually an equality. In words, we want to show that each ϵ -hypothesis of $\mathcal{H}_n(\epsilon)$ vanishes on the elements of all but at most one of the bigraphs that are joined to form $\mathcal{H}$.

For the sake of contradiction, suppose there exists an ϵ -hypothesis $H = \{(x_m, \gamma_m) : m < i_H\}$, two distinct indexes $i, j < \infty$ and $x_1 \in (\mathcal{H}_i)_n(\epsilon)$, $x_2 \in (\mathcal{H}_j)_n(\epsilon)$ such that $H(x_1) = H(x_2) = 1$. Then,

$$\sum_{(h_m, \gamma_m) \in H : h_m(x_1)=1} \gamma_m > 1 - \epsilon, \quad \sum_{(h_m, \gamma_m) \in H : h_m(x_2)=1} \gamma_m > 1 - \epsilon.$$

The indices m cannot appear in both sums, because by hypothesis of induction, it cannot be true that both $h(x_1) = 1$ and $h(x_2) = 1$ for any $h \in \mathcal{H}_n(\epsilon)$. Therefore,

$$\sum_{(h_m, \gamma_m) \in H : h_m(x_1)=1 \vee h_m(x_2)=1} \gamma_m > 2(1 - \epsilon) > 1.$$

This is a contradiction, since $\{\gamma_m : m < i_H\}$ are defined to give a finitely supported probability distribution over the hypotheses in H . $\square$

A possible follow up from the previous lemma would be whether there are any examples where it could be applied. This would consist, for example, of looking for stable graphs or Littlestone classes, partitioned into countable instances in different ways, and studying the complexity measures of their ϵ -saturation (which a priori should be the same) by use of this lemma.

Another possible follow-up from the work in this chapter is whether the proof (or claim) of Theorem 3.6.7 holds for the graph case as well. Namely, can it also be said that $\mathcal{G}_\infty(\epsilon)$ is the smallest ϵ -saturated graph that contains $\mathcal{G}$ (under inclusion)?

5.2 Comparison of results in Chapter 4

In Chapter 4, we show that a class is of bounded VC dimension if and only if there exist *symmetric ϵ -good pairs* of element sets. We prove this characterization in three ways: first, we use combinatorial techniques including Pigeonhole Principle; second, we apply Haussler's Packing Lemma (see Haussler [86]); last, we apply the VC Regularity Lemma, (see Lovasz and

Szegedy [101], Alon, Fisher and Newman [16], and Fox, Pach and Suk [68]). The following table summarizes the results of these main proofs (see Theorem 4.4.4, Corollaries 4.5.3 and 4.8.3, and Proposition 4.9.1).

	(c_1, c_2, ϵ') -good sampling	Antigood set bound
Combinatorial	$c_1 = \epsilon^d / 2^{d+1}$ $c_2 = 1/2^{d+1}$ $\epsilon' = \epsilon$	$\epsilon' = \epsilon$ $c'_1 = 1 - (\epsilon/2)^d$ $c'_2 = 1/2^d$
Haussler Packing	$c_1 = (1/2) \cdot \left(1 - \frac{ B -1}{2 B (1-\epsilon)}\right)$ $c_2 = (1/2) \cdot (\epsilon/60)^d$ $\epsilon' = 2\epsilon$	$\epsilon' = 2\epsilon$ $c'_1 = \max \left\{1, \frac{(B -1)\delta}{ B \epsilon(1-\epsilon)}\right\}$ $c'_2 = (\delta/30)^d, \delta > 0$
VC Regularity	$c_1 = (99/100) \cdot O(\epsilon^{-(2d+1)})$ $c_2 = (99/100) \cdot O(\epsilon^{-(2d+1)})$ $\epsilon' = (100^2/99)\epsilon$	$\epsilon' = 102\epsilon$ $c'_1 = 1/51$ $c'_2 = \epsilon^{2d+1}/C(d)$

Table 5.1: A table summarizing the results on symmetric good pairs and large sets with small antigood sets which can be found characteristically in classes of VC dimension bounded by some fixed $d \in \mathbb{N}$. In the table below, (c_1, c_2, ϵ) -good sampling stands for the following property: *For any $A \subseteq X, B \subseteq \mathcal{H}$ finite, $\exists A' \subseteq A, B' \subseteq B$, such that $|A'| \geq c_1|A|$, $|B'| \geq c_2|B|$, and (A', B') forms a symmetric ϵ good pair.* The antigood set bound stands for the following: *For any $B \subseteq \mathcal{H}$, there is $B' \subseteq B$ with small ϵ' -antigood set: concretely, $|B'| \geq c'_2 \cdot |B|$, $|\text{Anti}(B', \epsilon)| \leq c'_1|A|$.*

Adjusting the parameters $\epsilon > 0$ to be the same accross the three methods, we obtain the following table that can be helpful to compare the results of the three proof methods:

A possible follow up in this directions is: What are the specific tradeoffs between the

	(c_1, c_2, ϵ) -good sampling	Antigood set bound
Combinatorial	$c_1 = \epsilon^d / 2^{d+1}$ $c_2 = 1/2^{d+1}$	$c'_1 = 1 - (\frac{\epsilon}{2})^d$ $c'_2 = 1/2^d$
Haussler Packing	$c_1 = (1/2) \cdot \left(1 - \frac{ B -1}{2 B (1-\epsilon/2)}\right)$ $c_2 = (1/2) \cdot (\epsilon/120)^d$	$c'_1 = \max \left\{1, \frac{(B -1)\delta \cdot 2}{ B \epsilon(1-\epsilon/2)}\right\}$ $c'_2 = (\delta/30)^d, \delta > 0$
VC Regularity	$c_1 = \frac{99^{-2d} \cdot O(\epsilon^{-(2d+1)})}{100^{-4d+1}}$ $c_2 = \frac{99^{-2d} \cdot O(\epsilon^{-(2d+1)})}{100^{-4d+1}}$	$c'_1 = 1/51$ $c'_2 = \epsilon^{2d+1} / (102^{2d+1} C(d))$

Table 5.2: A table summarizing the results on symmetric good pairs and large sets with small antigood sets characteristic of classes of VC dimension bounded by some fixed $d \in \mathbb{N}$. In this table, the parameter $\epsilon \in (0, 1/2)$ is scaled for. In the table below, (c_1, c_2, ϵ) -good sampling stands for the following property: *For any $A \subseteq X, B \subseteq \mathcal{H}$ finite, $\exists A' \subseteq A, B' \subseteq B$, such that $|A'| \geq c_1|A|$, $|B'| \geq c_2|B|$, and (A', B') forms a symmetric ϵ good pair.* The antigood set bound stands for the following: *For any $B \subseteq \mathcal{H}$, there is $B' \subseteq B$ with small ϵ' -antigood set: concretely, $|B'| \geq c'_2 \cdot |B|$, $|\text{Anti}(B', \epsilon)| \leq c'_1|A|$.*

values obtained in each of these results, and in what applications are the bounds of each these results more telling? Another is whether any of the alluded proofs could be improved, giving better bounds.

REFERENCES

- [1] Hasan Abasi. Error-tolerant non-adaptive learning of a hidden hypergraph. In *43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.
- [2] Hasan Abasi and Nader Bshouty. On learning graphs with edge-detecting queries. In *Algorithmic Learning Theory*, pages 3–30. PMLR, 2019.
- [3] Hasan Abasi, Nader Bshouty, and Hanna Mazzawi. On exact learning monotone DNF from membership queries. In *Algorithmic Learning Theory: 25th International Conference, ALT 2014, Bled, Slovenia, October 8-10, 2014. Proceedings 25*, pages 111–124. Springer, 2014.
- [4] Hasan Abasi, Nader Bshouty, and Hanna Mazzawi. Non-adaptive learning of a hidden hypergraph. *Theoretical Computer Science*, 716:15–27, 2018.
- [5] Raghavendra Addanki, Andrew McGregor, and Cameron Musco. Non-adaptive edge counting and sampling via bipartite independent set queries. *arXiv preprint arXiv:2207.02817*, 2022.
- [6] Nir Ailon, Anup Bhattacharya, and Ragesh Jaiswal. Approximate correlation clustering using same-cluster queries. In *LATIN 2018: Theoretical Informatics: 13th Latin American Symposium, Buenos Aires, Argentina, April 16-19, 2018, Proceedings*, pages 14–27. Springer, 2018.
- [7] Nir Ailon, Moses Charikar, and Alanthan Newman. Aggregating inconsistent information: ranking and clustering. *Journal of the ACM (JACM)*, 55(5):1–27, 2008.
- [8] Noga Alon and Vera Asodi. Learning a hidden subgraph. *SIAM Journal on Discrete Mathematics*, 18(4):697–712, 2005.
- [9] Noga Alon, Richard Beigel, Simon Kasif, Steven Rudich, and Benny Sudakov. Learning a hidden matching. *SIAM Journal on Computing*, 33(2):487–501, 2004.
- [10] Noga Alon, Amos Beimel, Shay Moran, and Uri Stemmer. Closure properties for private classification and online prediction. *Proceedings of Machine Learning Research (PMLR)*, 125:119–152, 2020.
- [11] Noga Alon, Omri Ben-Eliezer, Yuval Dagan, Shay Moran, Moni Naor, and Eylon Yogev. Adversarial laws of large numbers and optimal regret in online classification. *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 447–455, 2021.
- [12] Noga Alon, Mark Bun, Roi Livni, Maryanthe Malliaris, and Shay Moran. Private and online learnability are equivalent. In *Journal of the ACM (Association for Computing Machinery)*, volume 69, No. 4, Article 28. ACM, 2022.

- [13] Noga Alon, Richard A. Duke, Hanno Leffman, Vojtěch Rödl, and Raphael Yuster. The algorithmic aspects of the regularity lemma. *FOCS 33; Journal of Algorithms 16 (1994)*, 80-109., pages 479–481, 1992.
- [14] Noga Alon, Richard A. Duke, Hanno Leffman, V. Rödl, and R. Yuster. The algorithmic aspects of the regularity lemma. *Journal of Algorithms 16 (1994) 80-109; FOCS 33(1992) 479-481*, 1994.
- [15] Noga Alon, Richard A. Duke, Hanno Leffman, Vojtech Rödl, and Raphael Yuster. The algorithmic aspects of the regularity lemma. *Journal of Algorithms, FOCS*, 33:80–109, 1994.
- [16] Noga Alon, Eldar Fischer, and Ilan Newman. Efficient testing of bipartite graphs for forbidden induced subgraphs. *SIAM Journal on Computing Vol. 37, Issue. 3*, 2007.
- [17] Noga Alon, Roi Livni, Maryanthe Malliaris, and Shay Moran. Private PAC learning implies finite littlestone dimension. *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC’19*:852 – 860, 2019.
- [18] Noga Alon and Joel H. Spencer. *The probabilistic method*. John Wiley & Sons, 2016.
- [19] Spyros Angelopoulos and Shahin Kamali. Rényi–Ulam games and online computation with imperfect advice. *arXiv preprint arXiv:2301.01631*, 2023.
- [20] Dana Angluin and Jiang Chen. Learning a hidden graph using $o(\log n)$ queries per edge. *Journal of Computer and System Sciences*, 74(4):546–556, 2008.
- [21] L. Aronsson and M. Chehreghani. Correlation clustering with active learning of pairwise similarities. *arXiv preprint arXiv:2302.10295*, 2023.
- [22] Hassan Ashtiani, Shrinu Kushagra, and Shai Ben-David. Clustering with same-cluster queries. *Advances in neural information processing systems*, 29, 2016.
- [23] Sepehr Assadi, Deeparnab Chakrabarty, and Sanjeev Khanna. Graph connectivity and single element recovery via linear and or queries. *arXiv preprint arXiv:2007.06098*, 2020.
- [24] D.J. Balding, W.J. Bruno, D.C. Torney, and E. Knill. A comparative survey of non-adaptive pooling designs. In *Genetic mapping and DNA sequencing*, pages 133–154. Springer, 1996.
- [25] N. Bansal, A. Blum, and S. Chawla. Correlation clustering. In *The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings.*, pages 238–247. IEEE, 2002.
- [26] Shai Ben-David, Dávid Pál, and Shai Shalev-Shwartz. Agnostic online learning. In *COLT 2009 - The 22nd Conference on Learning Theory, Montreal, Quebec, Canada, June 18-21, 2009*.

- [27] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred Warmuth. Learnability and the vapnik-chervonenkis dimension. *Journal of the Association of Computing Machinery*, 36.4:929–965, 1989.
- [28] Sebastian Böcker, Sebastian Briesemeister, Quang BA Bui, and Anke Truss. Going weighted: Parameterized algorithms for cluster editing. In *Combinatorial Optimization and Applications: Second International Conference, COCOA 2008, St. John’s, NL, Canada, August 21-24, 2008. Proceedings 2*, pages 1–12. Springer, 2008.
- [29] Olivier Bousquet, Steve Hanneke, Shay Moran, Ramon van Handel, and Amir Yehudayoff. A theory of universal learning. *STOC*, 2021.
- [30] Mathilde Bouvel, Vladimir Grebinski, and Gregory Kucherov. Combinatorial search on graphs motivated by bioinformatics applications: A brief survey. In *Graph-Theoretic Concepts in Computer Science: 31st International Workshop, WG 2005, Metz, France, June 23-25, 2005, Revised Selected Papers 31*, pages 16–27. Springer, 2005.
- [31] Marco Bressan, Nicolò Cesa-Bianchi, Silvio Lattanzi, and Andrea Paudice. Exact recovery of mangled clusters with same-cluster queries. *Advances in Neural Information Processing Systems*, 33:9324–9334, 2020.
- [32] Marco Bressan, Nicolò Cesa-Bianchi, Silvio Lattanzi, and Andrea Paudice. Exact recovery of clusters in finite metric spaces using oracle queries. In *Conference on Learning Theory*, pages 775–803. PMLR, 2021.
- [33] Marco Bressan, Nicolò Cesa-Bianchi, Silvio Lattanzi, and Andrea Paudice. On margin-based cluster recovery with oracle queries. *Advances in Neural Information Processing Systems*, 34:25231–25243, 2021.
- [34] Marco Bressan, Nicolò Cesa-Bianchi, Andrea Paudice, and Fabio Vitale. Correlation clustering with adaptive similarity queries. *Advances in Neural Information Processing Systems*, 32, 2019.
- [35] Mark Bun, Roi Livni, and Shay Moran. An equivalence between private classification and online prediction. *61th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020*, November 16-19, 2020.
- [36] Huilan Chang, Hung-Lin Fu, and Chih-Huai Shih. Learning a hidden graph. *Optimization Letters*, 8:2341–2348, 2014.
- [37] Hunter Chase and James Freitag. Bounds in query learning. *Conference on Learning Theory (PMLR)*, pages 1142–1160, 2020.
- [38] Hunter Chase and James Freitag. Model theory and machine learning. *The Bulletin of Symbolic Logic*, 25(03):319–332, Feb 2019. ArXiv preprint arXiv:1801.06566, 2018.
- [39] Hong-Bin Chen and Frank K. Hwang. A survey on nonadaptive group testing algorithms through the angle of decoding. *Journal of Combinatorial Optimization*, 15:49–59, 2008.

- [40] Yi Chen, Ramya Korlakai Vinayak, and Babak Hassibi. Crowdsourced clustering via active querying: Practical algorithm with theoretical guarantees. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 11(1), pages 27–37, 2023.
- [41] I Eli Chien, Chao Pan, and Olgica Milenkovic. Query k-means clustering and the double dixie cup problem. *Advances in Neural Information Processing Systems*, 31, 2018.
- [42] I Eli Chien, Huozhi Zhou, and Pan Li. Hs2: Active learning over hypergraphs with pointwise and pairwise queries. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2466–2475. PMLR, 2019.
- [43] Ferdinando Cicalese. Variations on a theme of Ulam and Rényi: More types of questions and lies. *Fault-Tolerant Search Algorithms: Reliable Computation with Unreliable Information*, pages 77–98, 2013.
- [44] Ferdinando Cicalese. Perfect strategies for the ulam-rényi game with multi-interval questions. *Theory of Computing Systems*, 54:578–594, 2014.
- [45] Ferdinando Cicalese and Christian Deppe. Q-ary Ulam–Rényi game with constrained lies. *General Theory of Information Transfer and Combinatorics*, pages 678–694, 2006.
- [46] Ferdinando Cicalese, Christian Deppe, and Daniele Mundici. Q-ary Ulam–Rényi game with weighted constrained lies. In *Computing and Combinatorics: 10th Annual International Conference, COCOON 2004, Jeju Island, Korea, August 17-20, 2004. Proceedings 10*, pages 82–91. Springer, 2004.
- [47] Ferdinando Cicalese and Massimiliano Rossi. On the multi-interval Ulam–Rényi game: For 3 lies 4 intervals suffice. *Theoretical Computer Science*, 809:339–356, 2020.
- [48] David Conlon and Jacob Fox. Bounds for graph regularity and removal lemmas. *Geom. Funct. Anal.* 22:5, page 1191–1256. MR Zbl, 2012.
- [49] Leonardo Nagami Coregliano and Maryanthe Malliaris. High-arity PAC learning via exchangeability. *Computer Research Repository 2024*, 2024.
- [50] Jurek Czyzowicz, Daniele Mundici, and Andrzej Pelc. Ulam’s searching game with lies. *Journal of Combinatorial Theory, Series A*, 52(1):62–76, 1989.
- [51] Susan Davidson, Sanjeev Khanna, Tova Milo, and Sudeepa Roy. Top-k and clustering with noisy comparisons. *ACM Transactions on Database Systems (TODS)*, 39(4):1–39, 2014.
- [52] Nicolaas Govert De Bruijn. *Asymptotic methods in analysis*, volume 4. Courier Corporation, 1981.
- [53] Alberto Del Pia, Mingchen Ma, and Christos Tzamos. Clustering with queries under semi-random noise. In *Conference on Learning Theory*, pages 5278–5313. PMLR, 2022.

- [54] D. Dellamonica, S. Kalyanasundaram, D. Martin, V. Rödl, and A. Shapira. A deterministic algorithm for the Frieze–Kannan regularity lemma. *SIAM J. Discrete Math.* 26, pages 15–29, 2012.
- [55] D. Dellamonica, S. Kalyanasundaram, D. Martin, V. Rödl, and A. Shapira. An optimal algorithm for finding Frieze–Kannan regular partitions. *Combin. Probab. Comput.* 24, page 407–437, 2015.
- [56] Erik D Demaine, Dotan Emanuel, Amos Fiat, and Nicole Immorlica. Correlation clustering in general weighted graphs. *Theoretical Computer Science*, 361(2-3):172–187, 2006.
- [57] Adela Frances DePavia, Olga Medrano Martín del Campo, and Erasmo Tani. Optimal algorithms for learning partitions with faulty oracles. *Advances Neural Information Processing Systems* 38, 2024. Arxiv preprint with different title: [2305.13402](#).
- [58] Christian Deppe. Solution of Ulam’s searching game with three lies or an optimal adaptive strategy for binary three-error-correcting codes. *Discrete Mathematics*, 224(1-3):79–98, 2000.
- [59] Christian Deppe. Strategies for the Rényi–Ulam game with fixed number of lies. *Theoretical computer science*, 314(1-2):45–55, 2004.
- [60] Dingzhu Du, Frank K Hwang, and Frank Hwang. *Combinatorial group testing and its applications*, volume 12. World Scientific, 2000.
- [61] P. Erdős and A. Hajnal. Ramsey-type theorems. *Discrete Applied Mathematics*, 1989.
- [62] Simone Fioravanti, Steve Hanneke, Shay Moran, Hilla Scheffer, and Iska Tsubari. Ramsey theorems for trees and a general ‘private learning implies online learning’ theorem. *2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1983–2009, 2024.
- [63] E. Fischer, A. Matsliah, and A. Shapira. Approximate hypergraph partitioning and applications. *SIAM Journal on Computing* 39, pages 3155–3185, 2010.
- [64] Sally Floyd and Manfred Warmuth. Sample compression, learnability, and the Vapnik–Chervonenkis dimension. *Machine learning*, page 269–304, 1995.
- [65] Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Springer, 2006.
- [66] Jacob Fox and László Lovász. A tight lower bound for Szemerédi’s Regularity Lemma”. *Combinatorica* 37:5, pages 911–951. MR Zbl, 2017.
- [67] Jacob Fox, László Lovász, and Yufei Zhao. A fast neq algorithm for weak graph regularity. *Cambridge University Press*, 2018.
- [68] Jacob Fox, János Pach, and Andrew Suk. Erdős-hajnal conjecture for graphs with bounded VC-dimension. *Discrete Computational Geometry*, 61(4):809–829, 2019.

- [69] A. Frieze and R. Kannan. The regularity lemma and approximation schemes for dense problems. *37th Annual Symposium on Foundations of Computer Science (Burlington, VT, 1996)*, *IEEE Comput. Soc. Press, Los Alamitos, CA, 1996*, pp. 12–20, 1996.
- [70] A. Frieze and R. Kannan. Quick approximation to matrices and applications. *Combinatorica* 19, pages 175–220, 1999.
- [71] David García-Soriano, Konstantin Kutzkov, Francesco Bonchi, and Charalampos Tsourakakis. Query-efficient correlation clustering. In *Proceedings of The Web Conference 2020*, pages 1468–1478, 2020.
- [72] Badih Ghazi, Noah Golowich, Ravi Kumar, and Pasin Manurangsi. Near-tight closure bounds for the Littlestone and threshold dimensions. *Proceedings of Machine Learning Research*, 132:686–696, 2021.
- [73] Ioannis Giotis and Venkatesan Guruswami. Correlation clustering with a fixed number of clusters. *Theory OF Computing*, 2:249–266, 2006.
- [74] W.T. Gowers. Lower bounds of tower type for Szémerédi’s uniformity lemma. *GAF*, vol. 7, pages 322–337, 1997.
- [75] W.T. Gowers. Lower bounds of tower type for Szemerédi’s Uniformity Lemma”. *Geom. Funct. Anal.*, 7:2, pages 322–337, MR Zbl, 1997.
- [76] Vladimir Grebinski and Gregory Kucherov. Reconstructing a Hamiltonian cycle by querying the graph: Application to dna physical mapping. *Discrete Applied Mathematics*, 88(1-3):147–165, 1998.
- [77] Vladimir Grebinski and Gregory Kucherov. Optimal reconstruction of graphs under the additive model. *Algorithmica*, 28:104–124, 2000.
- [78] Kasper Green Larsen, Michael Mitzenmacher, and Charalampos Tsourakakis. Clustering with a faulty oracle. In *Proceedings of The Web Conference 2020*, pages 2831–2834, 2020.
- [79] Meghal Gupta, Venkatesan Guruswami, and Rachel Yun Zhang. Binary error-correcting codes with minimal noiseless feedback. *arXiv preprint arXiv:2212.05673*, 2022.
- [80] Meghal Gupta and Rachel Yun Zhang. The optimal error resilience of interactive communication over binary channels. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 948–961, 2022.
- [81] Shubham Gupta, Peter WJ Staar, and Christian de Sainte Marie. Clustering items from adaptively collected inconsistent feedback. In *International Conference on Artificial Intelligence and Statistics*, pages 604–612. PMLR, 2024.
- [82] Venkatesan Guruswami, Atri Rudra, and Madhu Sudan. *Essential Coding Theory*. 2022. Draft available at <https://cse.buffalo.edu/faculty/atri/courses/coding-theory/book/>.

- [83] Bernhard Haeupler, Pritish Kamath, and Ameya Velingker. Communication with partial noiseless feedback. In *Approximation, randomization, and combinatorial optimization. Algorithms and techniques (APPROX/RANDOM 2015)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2015.
- [84] Richard W. Hamming. Error detecting and error correcting codes. *The Bell system technical journal*, 29(2):147–160, 1950.
- [85] Steve Hannecke, Roy Livni, and Shay Moran. Online learning with simple predictors and a combinatorial characterization of minimax in 0/1 games. *34th Annual Conference on Learning Theory*, 2021.
- [86] David Haussler. Sphere packing numbers for subsets of the boolean n-cube with bounded Vapnik–Chervonenkis dimension. *J. Comb. Theory, Ser. A*, 69 (2):217–232, 1995.
- [87] Ray Hill, Jehangir Karim, and E Berlekamp. The solution of a problem of ulam on searching with lies. In *Proceedings. 1998 IEEE International Symposium on Information Theory (Cat. No. 98CH36252)*, page 244. IEEE, 1998.
- [88] Wilfrid Hodges. Encoding orders and trees in binary relations. *Mathematika*, 28:67–81, 1981.
- [89] Wilfrid Hodges. *Model Theory - Encyclopedia of Mathematics*. Cambridge University Press, 1993.
- [90] Wilfrid Hodges. *A Shorter Model Theory*. Cambridge University Press, 1997.
- [91] Wasim Huleihel, Arya Mazumdar, Muriel Médard, and Soumyabrata Pal. Same-cluster querying for overlapping clusters. *Advances in Neural Information Processing Systems*, 32, 2019.
- [92] Zoran Kadelburg, Dusan Dukic, Milivoje Lukic, and Ivan Matic. Inequalities of Karata, Schur and Muirhead, and some applications. *The Teaching of Mathematics*, 8(1):31–45, 2005.
- [93] Michael Kearns and Umesh Vazirani. *An introduction to Computational Learning Theory*. The MIT Press, 1994.
- [94] János Komlós and Miklós Simonovits. Szemerédi’s regularity lemma and its applications in graph theory. In *Combinatorics: Paul Erdős is Eighty and DIMACS Technical Report 96-10*, 2:295–352, 1996.
- [95] Yuko Kuroki, Atsushi Miyauchi, Francesco Bonchi, and Wei Chen. Query-efficient correlation clustering with noisy oracle. *arXiv preprint arXiv:2402.01400*, 2024.
- [96] Michael C. Laskowski. Vapnik–Chervonenkis classes of definable sets. *J. London Math Soc*, 2(2):377–284, 1992.

- [97] Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2 (4):285–318, 1988.
- [98] Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. *Inf. Comput.*, 108, 2:212–261, 1994.
- [99] Xizhi Liu and Sayan Mukherjee. Tight query complexity bounds for learning graph partitions. In *Conference on Learning Theory*, pages 167–181. PMLR, 2022.
- [100] Bo Long, Xiaoyun Wu, Zhongfei Zhang, and Philip S Yu. Unsupervised learning on k-partite graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 317–326, 2006.
- [101] László Lovász and Balázs Szegedy. Regularity partitions and the topology of graphons. *Bárány, I., Solymosi, J., Sági, G. (eds) An Irregular Mind.*, 21:415–446, 2010.
- [102] Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Bilu–linial stable instances of max cut and minimum multiway cut. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 890–906. SIAM, 2014.
- [103] Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Correlation clustering with noisy partial information. In *Conference on Learning Theory*, pages 1321–1342. PMLR, 2015.
- [104] Maryanthe Malliaris, Olga Medrano Martín del Campo, and Shay Moran. ϵ -saturation for stable graphs and Littlestone classes. arxiv: [2508.21807](https://arxiv.org/abs/2508.21807), 2025.
- [105] Maryanthe Malliaris and Shay Moran. Agnostic online learning and excellent sets. *Transactions of the American Mathematical Society*, 2024.
- [106] Maryanthe Malliaris and Shay Moran. The unstable formula theorem revisited via algorithms. *Annals of Pure and Applied Logic*, 176, December 2025.
- [107] Maryanthe Malliaris and Saharon Shelah. Regularity lemmas for stable graphs. *Transactions of the American Mathematical Society*, 366 (3):1551–1585, 2014.
- [108] Maryanthe Malliaris and Saharon Shelah. Notes on the stable regularity lemma. *The Bulletin of Symbolic Logic*, 27, No. 4:415–425, 2021.
- [109] Adam Marcus, Eugene Wu, David Karger, Samuel Madden, and Robert Miller. Human-powered sorts and joins. *arXiv preprint arXiv:1109.6881*, 2011.
- [110] Claudio Marini and Franco Montagna. Probabilistic variants of Rényi-Ulam game and many-valued logic. *Task Quarterly*, 9(3):317–335, 2005.
- [111] Claire Mathieu and Warren Schudy. Correlation clustering with noisy input. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 712–728. SIAM, 2010.

- [112] Jiří Matoušek. *Geometric Discrepancy. An Illustrated Guide*, volume 18 of Algorithms and Combinatorics. Springer-Verlag, Berlin, 1999.
- [113] Jiří Matoušek. *Lectures on Discrete Geometry.*, volume 212 from Graduate Texts in Mathematics. New York: Springer, 2002.
- [114] Arya Mazumdar and Barna Saha. Clustering with noisy queries. *Advances in Neural Information Processing Systems*, 30, 2017.
- [115] Arya Mazumdar and Barna Saha. Query complexity of clustering with side information. *Advances in Neural Information Processing Systems*, 30, 2017.
- [116] Guy Moshkovitz and Asaf Shapira. A short proof of Gowers’ lower bound for the Regularity Lemma”. *Combinatorica* 36:2, page 187–194. MR Zbl, 2016.
- [117] Chandra Sekhar Mukherjee, Pan Peng, and Jiapeng Zhang. Recovering unbalanced communities in the stochastic block model with application to clustering with a faulty oracle. *Advances in Neural Information Processing Systems*, 36, 2024.
- [118] Tung Nguyen, Alex Scott, and Paul Seymour. Induced subgraph density. vi. bounded VC-dimension. *Advances in Mathematics*, 2025.
- [119] Andrzej Pelc. Solution of Ulam’s problem on searching with a lie. *Journal of Combinatorial Theory, Series A*, 44(1):129–140, 1987.
- [120] Andrzej Pelc. Searching with known error probability. *Theoretical Computer Science*, 63(2):185–202, 1989.
- [121] Andrzej Pelc. Searching games with errors—fifty years of coping with liars. *Theoretical Computer Science*, 270(1-2):71–109, 2002.
- [122] M. Pelillo, I. Elezi, and M. Fiorucci. Revealing structure in large graphs: Szémerédi’s regularity lemma and its use in pattern recognition. *Pattern Recognition Letters*, 2016.
- [123] Pan Peng and Jiapeng Zhang. Towards a query-optimal and time-efficient algorithm for clustering with a faulty oracle. In *Conference on Learning Theory*, pages 3662–3680. PMLR, 2021.
- [124] Basil Cameron Rennie and Annette Jane Dobson. On Stirling numbers of the second kind. *Journal of Combinatorial Theory*, 7(2):116–121, 1969.
- [125] Lev Reyzin and Nikhil Srivastava. Learning and verifying graphs using queries with a focus on edge counting. *International Conference on Algorithmic Learning Theory*, page 285–297, 2007.
- [126] Ronald L Rivest, Albert R Meyer, and Daniel J Kleitman. Coping with errors in binary search procedures (preliminary report). In *Proceedings of the tenth annual ACM symposium on Theory of computing*, pages 227–232, 1978.

- [127] I. Z. Ruzsa and E. Szemerédi. Triple systems with no six points carrying three triangles. *Combinatorics (Proc. Fifth Hungarian Colloq., Keszthely, 1976), Vol. II, Colloq. Math. Soc. János Bolyai, vol. 18, North-Holland, Amsterdam-New York, 1978, pp. 939–945.*, 1978.
- [128] Barna Saha and Sanjay Subramanian. Correlation clustering with same-cluster queries bounded by optimal cost. *arXiv preprint arXiv:1908.04976*, 2019.
- [129] Norbert Sauer. On the density of families of sets. *J. Comb. Theory, Ser. A*, 13(1):145–147, 1972.
- [130] Shai Shalev-Schwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014. Available on the first author’s webpage.
- [131] Claude E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.
- [132] Xu Shaoji. The size of uniquely colorable graphs. *Journal of Combinatorial Theory, Series B*, 50(2):319–320, 1990.
- [133] Saharon Shelah. A combinatorial problem; stability and order for models and theories in infinitary languages. *Pacific Journal of Mathematics*, 41:247–261, 1972.
- [134] Saharon Shelah. Classification theory and the number of non-isomorphic models. *Studies in Logic and the Foundations of Mathematics*, 92:705, 1978.
- [135] Saharon Shelah. *Classification Theory*. North-Holland, 1978. Revised edition, 1990.
- [136] Alexei Sorokin, Alla Lapidus, Veronique Capuano, Nathalie Galleron, Petar Pujic, and S. Dusko Ehrlich. A new approach using multiplex long accurate pcr and yeast artificial chromosomes for bacterial chromosome mapping and sequencing. *Genome research*, 6(5):448–453, 1996.
- [137] Joel Spencer. Guess a number—with lying. *Mathematics Magazine*, 57(2):105–108, 1984.
- [138] Joel Spencer. Ulam’s searching game with a fixed number of lies. *Theoretical Computer Science*, 95(2):307–321, 1992.
- [139] A. Sperotto and M. Pelillo. Szemerédi’s regularity lemma and its applications to pairwise clustering and segmentation. *6th International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR 2007)*, 2007.
- [140] Endre Szemerédi. Regular partitions of graphs. *Colloques Internationaux C.N.R.S., No. 260, Problèmes Combinatoires et Théorie des Graphes, Orsay*, pages 399–401, 1976.

- [141] Terrence Tao. *An epsilon of room, II*. American Mathematical Society, Providence, RI, 2010.
- [142] Caroline Terry. Growth of regular partitions 2: Weak regularity. arxiv: [2404.01293](https://arxiv.org/abs/2404.01293), 2024.
- [143] Caroline Terry and Julia Wolf. Stable arithmetic regularity in the finite-field model. *Bull. London Math. Soc.*, 51(1):70–88, 2019.
- [144] Caroline Terry and Julia Wolf. Irregular triads in 3-uniform hypergraphs. *Memoirs of the American Mathematical Society*, 2022.
- [145] Caroline Terry and Julia Wolf. Higher-order generalizations of stability and arithmetic regularity, 2023.
- [146] Charalampos E. Tsourakakis, Preetum Nakkiran, Ben Lawson, Vasileios Nakos, Kasper Green Larsen, Michael Mitzenmacher, and Jarosław Błasiok. Predicting positive and negative links with noisy queries: Theory & practice. *Internet Mathematics*, 2020.
- [147] Stanisław M. Ulam. *Adventures of a Mathematician*. Univ of California Press, 1991.
- [148] T. Van Craenendonck, S. Dumancic, and H. Blockeel. Cobra: A fast and simple method for active clustering with pairwise constraints. *arXiv preprint arXiv:1801.09955*, 2018.
- [149] Vladimir Vapnik and Alexey Červonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, XVI, no.2:264–280, 1971.
- [150] Vladimir Vapnik and Alexey Červonenkis. The uniform convergence of frequencies of the appearance of events to their probabilities. *Teor. Verojatnost. i Primenen.*, 16:264–279, 1971.
- [151] Ramya Korlakai Vinayak and Babak Hassibi. Crowdsourced clustering: Querying edges vs triangles. *Advances in Neural Information Processing Systems*, 29, 2016.
- [152] Jiannan Wang, Tim Kraska, Michael J. Franklin, and Jianhua Feng. Crowder: Crowdsourcing entity resolution. *arXiv preprint arXiv:1208.1927*, 2012.
- [153] Feng Xia, Ke Sun, Shuo Yu, Abdul Aziz, Liangtian Wan, Shirui Pan, and Huan Liu. Graph learning: A survey. *IEEE Transactions on Artificial Intelligence*, 2(2):109–127, apr 2021.
- [154] Jinghui Xia and Zengfeng Huang. Optimal clustering with noisy queries via multi-armed bandit. In *International Conference on Machine Learning*, pages 24315–24331. PMLR, 2022.
- [155] Shicheng Xiong, Javad Azimi, and Xiaoli Z. Fern. Active learning of constraints for semi-supervised clustering. *IEEE Transactions on Knowledge and Data Engineering*, 26(1):43–54, 2013.