Metalearning with Very Few Samples Per Task

Maryam Aliakbarpour
Department of Computer Science, Rice University

Konstantina Bairaktari
Khoury College of Computer Sciences, Northeastern University

Gavin Brown
Paul G. Allen School of Computer Science and Engineering, University of Washington.

Adam Smith
Department of Computer Science, Boston University.

Nathan Srebro
Toyota Technological Institute at Chicago

Jonathan Ullman
Khoury College of Computer Sciences, Northeastern University

Editors: Shipra Agrawal and Aaron Roth

Abstract

Metalearning and multitask learning are two frameworks for solving a group of related learning tasks more efficiently than we could hope to solve each of the individual tasks on their own. In multitask learning, we are given a fixed set of related learning tasks and need to output one accurate model per task, whereas in metalearning we are given tasks that are drawn i.i.d. from a metadistribution and need to output some common information that can be easily specialized to new, previously unseen tasks from the metadistribution.

In this work, we consider a binary classification setting where tasks are related by a *shared representation*, that is, every task P of interest can be solved by a classifier of the form $f_P \circ h$ where $h \in \mathcal{H}$ is a map from features to some representation space that is shared across tasks, and $f_P \in \mathcal{F}$ is a task-specific classifier from the representation space to labels. The main question we ask in this work is *how much data do we need to metalearn a good representation?* Here, the amount of data is measured in terms of both the number of tasks t that we need to see and the number of samples n per task. We focus on the regime where the number of samples per task is extremely small.

Our main result shows that, in a distribution-free setting where the feature vectors are in $\mathbb{R}^d$, the representation is a linear map from $\mathbb{R}^d \to \mathbb{R}^k$, and the task-specific classifiers are halfspaces in $\mathbb{R}^k$, we can metalearn a representation with error ε using just $n = k + 2$ samples per task, and $d \cdot (1/\varepsilon)^{O(k)}$ tasks. Learning with so few samples per task is remarkable because metalearning would be impossible with $k + 1$ samples per task, and because we cannot even hope to learn an accurate task-specific classifier with just $k + 2$ samples per task. To obtain this result, we develop a sample-and-task-complexity theory for distribution-free metalearning and multitask learning, which identifies what properties of $\mathcal{F}$ and $\mathcal{H}$ make metalearning possible with few samples per task. Our theory also yields a simple characterization of distribution-free multitask learning. Finally, we give sample-efficient reductions between metalearning and multitask learning, which, when combined with our characterization of multitask learning, give a characterization of metalearning in certain parameter regimes.

Keywords: metalearning, multitask learning

1. Introduction

Metalearning and multitask learning are frameworks for solving a group of related learning tasks more efficiently than we could hope to solve each of the individual tasks on their own. It is convenient to think of tasks as corresponding to people—for example, each task could involve completing sentences of one person's email or recognizing one person's friends' faces in photographs. These tasks are clearly different from person to person, since people have different writing styles and friends, but there is also a great deal of shared structure in both examples. Even though each person may not supply enough text messages or photographs to build an accurate model, one can leverage shared structure to solve all of the tasks more effectively.

In these frameworks (specialized to binary classification), we model each task as a distribution P over labeled examples $\mathcal{X} \times \{\pm 1\}$. A family of tasks can be either a finite list of tasks $P_1, \ldots, P_t$ (in multitask learning) or, more generally, a metadistribution $\mathfrak{Q}$ over tasks (in metalearning). There are several ways to model the relationships among a family of tasks (see Section 1.2). In this paper, we assume that the tasks are well labeled by classifiers that share a common *representation* $h : \mathcal{X} \to \mathcal{Z}$ for some space $\mathcal{Z}$, usually of lower dimension than $\mathcal{X}$. For each task P_j, there is a specialized classifier $f_j : \mathcal{Z} \to \{\pm 1\}$ such that $f_j \circ h$ has high accuracy on P_j. For example, when recognizing faces in photographs, a common approach uses one neural network to extract facial features followed by a final layer that differs for each person and maps facial features to names of that person's friends.

Given classes $\mathcal{H}$ of possible representations and $\mathcal{F}$ of possible specializations, we ask *how much data do we need to find a representation that performs well for most of the tasks in our family of interest?* Suppose the learner has access to t data sets, where the j-th dataset consists of n observations drawn i.i.d. from task P_j. Following the literature, we consider two basic objectives:

(Proper[1]) Multitask learning: Output a representation $h \in \mathcal{H}$ and specialized classifiers $f_1, \ldots, f_t \in \mathcal{F}$ such that the error of the classifier $f_j \circ h$ on its task P_j is low on average over the tasks. For example, this objective corresponds to learning the classifiers for a given set of face-recognition tasks.

(Proper[1]) Metalearning: Assuming that the tasks $P_1, ..., P_t$ are themselves drawn i.i.d. from an unknown metadistribution $\mathfrak{Q}$, output a representation $\hat{h} \in \mathcal{H}$ that can then be specialized to an unseen task P drawn from $\mathfrak{Q}$. The benchmark is then the expected error of the best representation h^* on a new task $P \sim \mathfrak{Q}$, measured using the best specialized classifier $f \in \mathcal{F}$ for that task. This objective corresponds to finding an embedding that can be used to quickly build face-recognition classifiers for new people, as opposed to for people whose photos were used for training.

For both variants, one can consider *realizable* families of tasks or a more general *agnostic setting*. In the realizable case, there is a representation $h \in \mathcal{H}$ such that, for every task P in the family, we can find a classifier $f_P \in \mathcal{F}$ that perfectly labels example from P. In the agnostic setting, we have no such promise; we aim to compete with the benchmark no matter how good or bad it is.

The sample and task complexity of meta and multitask learning. The goal of this paper is to understand the sample complexity of multitask and metalearning. Specifically

> *How many samples n do we need per task to solve metalearning and multitask learning and, given a number of samples n per task, how many tasks t do we need to see?*

1. For simplicity, and to highlight the relationship with metalearning, we describe the *proper* version of multitask/meta learning, but we also consider a more general notion of improper multitask/meta learning.

To get some intuition for this question, it is easy to see that in the multitask learning setting we need at least enough samples from each task to learn a good specialized classifier. For realizable learning, in the distribution-free setting that we study, the number of samples per task must be at least $n \geq \mathrm{VC}(\mathcal{F})/\varepsilon$ to learn with error ε. In contrast, metalearning's only goal is to output a representation that can be specialized to new tasks, and there is no need to output a good classifier for any specific task. Thus we will focus on a more specific question

> *What is the minimum number of samples per task n that enable us to solve metalearning with a finite number of tasks t?*

Our main result answers this question *exactly* for the case of *halfspace classifiers* with a shared *linear representation* in the *distribution-free, realizable setting*. That is, when examples are in $\mathbb{R}^d$, representations are linear functions from $\mathbb{R}^d \to \mathbb{R}^k$, and specialized classifiers are halfspaces on $\mathbb{R}^k$. Our result shows that we can metalearn this class of functions with $k+2$ samples per task. Metalearning with $k+1$ samples per task is impossible in the distribution-free setting. For constant error, we show that the number of tasks required to learn the representation using $k+2$ samples per task is at most linear in d and exponential in k.

To prove our main result, we develop a uniform-convergence theory for distribution-free metalearning and multitask learning. This theory complements prior work, which developed a theory of meta- and multitask learning under specific assumptions about the task distributions (see Section 1.2). The main novel feature of our uniform-convergence theory is that it enables us to prove bounds on the number of tasks required for metalearning *even in the regime where we lack the samples per task needed to accurately evaluate the quality of a representation.* That is, given a representation h and just $n = \mathrm{VC}(\mathcal{F}) + 1$ samples from a task, we can learn an arbitrarily good representation, even though we do not have enough data to find a specialized classifier $f \circ h$ with small error ε.

Our theory identifies two key features of the classes $\mathcal{H}$ and $\mathcal{F}$ that enable metalearning in the realizable case: (1) whether $\mathcal{F}$ has small *dual Helly number*, meaning that given an unrealizable sample of size n, there is a small subset of samples that are also unrealizable, and (2) whether the class of *realizability predicates* for $\mathcal{H}$ and $\mathcal{F}$ has small VC dimension, where the realizability predicates are functions that indicate whether the sample S is realizable by a function of the form $f \circ h$ for $f \in \mathcal{F}$.

Along the way, we also study the multitask setting. We give a simple characterization of the total number of samples nt that we need to learn and pin down the sample complexity of learning halfspaces with a shared linear representation up to constant factors. These results help to highlight the differences between metalearning and multitask learning. By showing that we can metalearn (improperly) if and only if we can multitask learn in the setting where we have enough samples per task to output a good classifier for a given representation, we obtain task and sample complexity bounds for improper metalearning.

Our results shed light on some of the intriguing empirical phenomena in modern machine learning. For example, the training data sets for foundation models are increasingly being augmented with data sources from a wide variety of subpopulations, even though those sources may be very small. That's normally motivated by the desire to improve accuracy on these subpopulations. However, our work highlights a different and complementary reason why that can be valuable: even a few examples from many subpopulations can vastly improve learning on all populations, even the "data rich" ones.

1.1. Our Results and Techniques

In this section we give a more detailed overview of our main results and techniques. We focus both on the general conditions for meta or multitask learnability that we establish as well as our specific sample- and task-complexity results for linear classifiers over linear representations. We begin by describing the multitask learning setting because it is simpler and serves as a contrast to highlight some of the unusual features of metalearning.

Multitask Learning: A General Characterization and a Tight Bound for Linear Classes. Our first result is a complete characterization of the conditions for distribution-free multitask learning. Given $\mathcal{H}$ and $\mathcal{F}$ and a number of tasks t, one may view the collection of functions $h, f_1, ..., f_t$ output by a proper learner as a single function from $[t] \times \mathcal{X}$ to $\{\pm 1\}$ that maps (j, x) to $f_j(h(x))$. We denote by $\mathcal{F}^{\otimes t} \circ \mathcal{H}$ the class of such composite functions:

$$\mathcal{F}^{\otimes t} \circ \mathcal{H} := \left\{ g : [t] \times \mathcal{X} \to \{\pm 1\} \;\middle|\; \exists h \in \mathcal{H}, f_1, \ldots, f_t \in \mathcal{F} \text{ s.t. } g(j, x) = f_j(h(x)) \right\}$$

The composite class $\mathcal{F}^{\otimes t} \circ \mathcal{H}$ has been studied in other multitask learning settings (see Section 1.2). We show that in our setting, the VC dimension of this class characterizes distribution-free multitask learnability in both the realisable and agnostic settings.

Theorem 1.1 (Informal version of Theorem C.1) *Distribution-free multitask learning to error ε with n samples per task and t tasks is possible if and only if $nt \gtrsim \mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H})/\varepsilon^2$. In the realizable setting $1/\varepsilon^2$ is replaced with $\log(1/\varepsilon)/\varepsilon$.*

See Definition B.1 for a formal definition of the setting and error parameter ε. Since VC dimension characterizes distribution-free PAC learning, this result shows that multitask learning is possible if and only if the larger composed function class is PAC learnable. See Appendix C.2 and Appendix F for basic upper and lower bounds on the VC dimension of the composed class.

Nevertheless, an interesting phenomenon emerges since t appears in both the definition of the function and the sample complexity bound. As a concrete example, consider the case of a linear classifier over a linear representation

$$\mathcal{H}_{d,k} = \left\{ h \mid h(\mathbf{x}) = \mathbf{B}\mathbf{x}, \mathbf{B} \in \mathbb{R}^{k \times d} \right\} \text{ and } \mathcal{F}_k = \left\{ f \mid f(\mathbf{z}) = \mathrm{sign}(\mathbf{a} \cdot \mathbf{z} - w), \mathbf{a} \in \mathbb{R}^k, w \in \mathbb{R} \right\},$$

which will be the main example we study in this paper. Using a bound on the number of possible sign patterns in a family of polynomials due to Warren (1968), which was also used by Baxter (2000) for meta-learning bounds on regression problems, we characterize the VC dimension of the associated composite class up to a constant.

Theorem 1.2 *For all $t, d, k \in \mathbb{N}$, we have*

$$\mathrm{VC}(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}) = \begin{cases} t(d+1), & \text{if } t \leq k \\ \Theta(dk + kt), & \text{if } t > k. \end{cases}$$

We see two distinct regimes: When $t \leq k$, the VC dimension is exactly $t \cdot (d+1)$ which is also the sample complexity of learning t separate d-dimensional linear classifiers, so there is no benefit to the existence of a common representation. When $t > k$, the sample complexity scales as

$n \gtrsim (\frac{dk}{t} + k)/\varepsilon^2$, so there is a fixed cost of dk/ε^2 samples that can be amortized over many tasks, with an unavoidable marginal cost of just k/ε^2 per task. See Appendix E.1 for more details.

Metalearning: The Case of Linear Representations. For metalearning, a different and more complex picture emerges. For example, one might expect that the conditions for multitask learning are also *necessary* for learning a good representation. However, this is not always the case. Consider again the linear classes $\mathcal{H}_{d,k}$ and $\mathcal{F}_k$. While multitask learning requires $\Omega(k/\varepsilon)$ samples per task, we show that one can accurately learn a good k-dimensional linear representation for any realizable family with just $n = k + 2$ samples per task:

Corollary 1.3 (Informal version of Corollary 4.3) *One can metalearn $(\mathcal{H}_{d,k}, \mathcal{F}_k)$ to error ε in the realizable case if $t = dk^2 \cdot O(\ln(1/\varepsilon)/\varepsilon)^{k+2}$ and $n = k + 2$.*

See Definition 3.1 for a formal definition of the setting and notion of error. The algorithm underlying this result selects a representation that minimizes the *fraction of nonrealizable data sets* among the training tasks—see the explanations after Theorem 1.1 for details and intuition.

When we restrict further to the case of specialized monotone thresholds $\mathcal{F}_{\text{mon}}$ applied to a 1-dimensional linear representation, we get a stronger bound than what would be obtained by applying the result above with $k = 1$, namely, just $n = 2$ samples per task suffice. (see Theorem D.4).

These results consider a regime of extreme data scarcity. For linear classifiers, any nondegenerate set of $k+1$ labeled examples can be perfectly fit by a halfspace and so, absent additional assumptions like a large margin, one gets no information from seeing just $k + 1$ points per task. Thus $n = k + 2$ is the exact minimum number of samples per task that suffice for metalearning. However, with so few examples we cannot even reliably estimate how well a candidate representation h performs on a task. Essentially the only signal one gets is whether the data set is realizable, so it is surprising that representation learning is possible at all in such a regime. Indeed, once we choose a representation and sample a new task P from the metadistribution $\mathfrak{Q}$, we need a larger number of samples $n_{\text{spec}} \approx k/\varepsilon^2$ to find a good specialized classifier for the new task. Although such a large number of samples isn't necessary for metalearning, when samples are plentiful we can show that one can learn a good representation with task complexity polynomial in d, k and $1/\varepsilon$, even in the agnostic setting.

Corollary 1.4 (Informal version of Corollaries 4.6 and E.2) *One can metalearn the class $(\mathcal{H}_{d,k}, \mathcal{F}_k)$ to error ε with $n = O(k \ln(1/\varepsilon)/\varepsilon^2)$ samples per task and* either *$t = \tilde{O}(dk^2/\varepsilon^4)$ tasks via a proper learner,* or *$t = O(d)$ tasks via an improper learner.*

Metalearning for realizable families with minimal sample complexity. To prove these results, we develop a general uniform-convergence theory for metalearning. Our results for realizable learning rely on two major ideas: first, we show that there is a sample size n which depends only on $\mathcal{F}$ (not on $\mathcal{H}$ or ε) that allows for training a representation given many tasks; second, we identify a set of predicates based on $\mathcal{H}$ and $\mathcal{F}$ that determine how many tasks are needed.

Definition 1.5 (The dual Helly number (Bousquet et al., 2020)) *Let $\mathcal{F}$ be a class of functions $f : \mathcal{Z} \to \{\pm 1\}$. We define the dual Helly number, denoted $\mathrm{DH}(\mathcal{F})$, to be the smallest integer m for which every set $S \subseteq \mathcal{Z} \times \{\pm 1\}$ that is not realizable by $\mathcal{F}$ has a nonrealizable subset of size at most m. If no such m exists, then we write $\mathrm{DH}(\mathcal{F}) = \infty$.*

The dual Helly number is essentially the smallest sample size at which we are guaranteed that realizability provides a meaningful signal. For example, the class of monotone threshold functions $\mathcal{F}_{\text{mon}}$ has $\text{DH}(\mathcal{F}_{\text{mon}}) = 2$, since every nonrealizable set has two points with different labels that are out of order. Halfspaces in $\mathbb{R}^k$ have dual Helly number $k+2$ by a classic duality argument (Kirchberger, 1903). In general, $\text{DH}(\mathcal{F})$ and $\text{VC}(\mathcal{F})$ dimension can differ arbitrarily in either direction, as shown in Kane et al. (2019), Bousquet et al. (2020) and Lemma F.1.

Given this sample size m, we consider a class of *realizability predicates*:

Definition 1.6 (Realizability predicate) *Let $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$ and $n \in \mathbb{N}$. Given a representation function $h : \mathcal{X} \to \mathcal{Z}$, the* $(n, \mathcal{F})$-realizability predicate *is $r_h((x_1, y_1), \ldots, (x_n, y_n)) \stackrel{def}{=} \mathbb{I}_{\pm}\{\exists f \in \mathcal{F} \text{ s.t. } \forall i \in [n], f(h(x_i)) = y_i\}$. The class of realizability predicates, $\mathcal{R}_{n,\mathcal{F},\mathcal{H}}$, contains all realizability predicates r_h for $h \in \mathcal{H}$.*

For example, for the class of monotone thresholds, we can view the representation h as a 1-dimensional function that orders the samples on the real line. Then, r_h returns $+1$ if and only if all the positively labeled samples have greater representation value than the negative ones.

Our key result is that realizable metalearning with just m samples per task is possible whenever the VC dimension of this predicate class is finite—in fact, it scales linearly with its VC dimension and at most exponentially in $\text{DH}(\mathcal{F})$—by minimizing the fraction of nonrealizable samples.

Theorem 1.7 (Informal version of Theorem 3.2) *We can metalearn $(\mathcal{H}, \mathcal{F})$ with error ε in the realizable case with $\text{DH}(\mathcal{F})$ samples per task and task complexity that, when $\text{VC}(\mathcal{F}) = O(\text{DH}(\mathcal{F}))$, scales as*

$$t = \text{VC}\big(\mathcal{R}_{\text{DH}(\mathcal{F}),\mathcal{F},\mathcal{H}}\big) \cdot O\left(\frac{\log(1/\varepsilon)}{\varepsilon}\right)^{\text{DH}(\mathcal{F})}.$$

For natural classes (such as halfspaces), we expect the VC dimension of $\mathcal{F}$ and the dual Helly number to be comparable, and so we get a complexity of roughly $\text{VC}(\mathcal{R}_{m,\mathcal{F},\mathcal{H}})/\varepsilon^m$ for $m = \text{DH}(\mathcal{F})$.

To get some intuition for Theorem 1.7, consider a fixed representation $\hat{h}$ and a task P on $\mathcal{X} \times \{\pm 1\}$. The heart of our argument is a relationship between two important quantities: the *population error* of $\mathcal{F} \circ \hat{h}$ on P, and the *probability of nonrealizability* of a dataset from $P^{\otimes m}$ by functions in $\mathcal{F} \circ \hat{h}$. Letting B denote an arbitrary distribution on $\mathcal{Z} \times \{\pm 1\}$ (in the rest of the paper, this distribution will generally be $(\hat{h}(X), Y)$ for $(X, Y) \sim P$ and some candidate representation $\hat{h}$), we define

$$\text{err}(B, \mathcal{F}) \stackrel{\text{def}}{=} \min_{f \in \mathcal{F}} \mathbf{Pr}_{(x,y)\sim B}\big[f(x) \neq y\big] \quad \text{and} \quad p_{\text{nr}}(B, \mathcal{F}, m) \stackrel{\text{def}}{=} \mathbf{Pr}_{S \sim B^m}\big[S \text{ is not realizable by } \mathcal{F}\big].$$

In Lemma 3.3, we give a general lower bound on the probability of nonrealizability $p_{\text{nr}}(B, \mathcal{F}, m)$ in terms of the error of the best classifier $\text{err}(B, \mathcal{F})$ and the complexity $\text{DH}(\mathcal{F})$: for any class of specialization functions $\mathcal{F}$ and any distribution B on the intermediate space $\mathcal{Z} \times \{\pm 1\}$,

$$p_{\text{nr}}(B, \mathcal{F}, m) \gtrsim \left(\tfrac{m}{m+\text{VC}(\mathcal{F})} \cdot \text{err}(B, \mathcal{F})\right)^m \quad \text{for } m = \text{DH}(\mathcal{F}). \tag{1}$$

Bounding p_{nr} is nontrivial since the sample size $m = \text{DH}(\mathcal{F})$ is typically much smaller than $N = \text{VC}(\mathcal{F})/\text{err}(B, \mathcal{F})$, which is what standard concentration arguments require. To get around this, we consider a thought experiment in which a larger data set size N is drawn and then a random

subsample of size m within it is observed. Since the larger set will almost certainly be unrealizable, by definition it must contain an unrealizable subset of size m. The random subsample will find such a witness with probability at least $1/\binom{N}{m}$, which is bounded by the expression in (1).

The lower bound on p_{nr} implies that a representation $\hat{h}$ which is poor for typical tasks from a meta distribution $\mathfrak{Q}$ will also lead to a typical sample of size m drawn from a typical task being unrealizable. When $\mathcal{R}_{m,\mathcal{F},\mathcal{H}}$ has low VC dimension, the number of unrealizable samples in the training data will concentrate uniformly across representations $\hat{h}$, and so the representation which minimizes the number of unrealizable training data sets will also, up some loss, minimize the expected population level loss $\mathrm{err}(B, \mathcal{F})$. This line of argument leads to Theorem 1.7.

Metalearning with More Data Per Task. Checking the realizability of very small data sets works when we are guaranteed to find a representation for which all the training data sets will be realizable. In the agnostic setting, this approach makes less sense. Additionally, the bounds we obtain scale poorly with the $\mathrm{DH}(\mathcal{F})$.

To obtain polynomial scaling and cope with the agnostic setting, we consider larger training data sets—large enough to assess the quality of a given representation on the task at hand—and a different set of function classes associated to the pair $(\mathcal{H}, \mathcal{F})$. Instead of realizability, we consider a function that returns the best achievable empirical error for a given representation on a dataset:

Definition 1.8 (Empirical error function) *Let $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$ and $n \in \mathbb{N}$. Given a representation function $h : \mathcal{X} \to \mathcal{Z}$, the* $(n, \mathcal{F})$-empirical error function *of h is $q_h(x_1, y_1, \ldots, x_n, y_n) \stackrel{def}{=} \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}\{f(h(x_i)) \neq y_i\}$. The class of empirical error functions $\mathcal{Q}_{n,\mathcal{F},\mathcal{H}}$ contains all empirical error functions q_h for $h \in \mathcal{H}$.*

We show that the task complexity of metalearning with data sets of size n can be bounded via the complexity of $\mathcal{Q}_{n,\mathcal{F},\mathcal{H}}$. Specifically, we consider the *pseudodimension* of $\mathcal{Q}_{n,\mathcal{F},\mathcal{H}}$, which is essentially the VC dimension of the class of all binary thresholdings of the class. The pseudodimension of $\mathcal{Q}_{n,\mathcal{F},\mathcal{H}}$ is thus at least VC dimension of the realizability class $\mathcal{R}_{m,\mathcal{F},\mathcal{H}}$, but can in general be larger.

Theorem 1.9 (see Theorem 3.6) *Let $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$ and $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$. Then, for every ε and $\delta \in (0, 1)$ we can (ε, δ)-properly metalearn $(\mathcal{H}, \mathcal{F})$ with t tasks and n samples per task when $t = O\Big(\frac{\mathit{PDim}(\mathcal{Q}_{n,\mathcal{F},\mathcal{H}}) \ln(1/\varepsilon) + \ln(1/\delta)}{\varepsilon^2}\Big)$ and $n = O\Big(\frac{\mathrm{VC}(\mathcal{F}) + \ln(1/\varepsilon)}{\varepsilon^2}\Big)$.*

There exist other ways of measuring the complexity of real-valued functions—fat-shattering dimension and Rademacher complexity are more general than pseudodimension—but we show that for the linear representations and halfspaces, the pseudodimension lends itself directly to bounds based on the signs of polynomials.

Generic reductions between metalearning and multitask learning. To complement our uniform convergence theory for metalearning, we also show a generic equivalence between metalearning and multitask learning in Appendix B.3. This equivalence leads to improved sample complexity for linear classes in the regime where we have enough samples per task to solve multitask learning.

In contrast to our earlier results, these reductions *do not* yield any bounds for metalearning when we have just $n = \mathrm{VC}(\mathcal{F}) + 1$ samples per task. Interestingly, the reduction gives metalearners that are *improper*, even when the original multitask learner is proper. Instead of a succinct representation,

the reduction produces a specialization algorithm whose description is no simpler than the entire training data set. We leave it as an open question to determine if one can match the sample complexity bounds we obtain from our reduction via proper metalearning.

Analyzing the classes associated to linear representations. To apply our general results to linear classes, we produce bounds on the VC dimension of the realizability predicates and, using similar tools, the pseudodimension of the empirical error function, yielding the following result:

Theorem 1.10 (see Theorem 4.5) *For all d, k, n with $n \geq k + 2$,* $\mathrm{PDim}(Q_{n,\mathcal{F}_k,\mathcal{H}_{d,k}}) = \tilde{O}(dkn)$.

At a high level, our approach shows that the predicate r_h can be expressed in low-complexity language. Concretely, we first derive a list of low-degree polynomials $p^{(1)}(h), \ldots, p^{(w)}(h)$ which are positive exactly when certain conditions on the data and representation are met. We then show that the realizability predicate can be written as a Boolean function over these conditions: the function's ℓ-th input is $\mathrm{sign}(p^{(\ell)}(h))$. A result of Warren (1968) allows us to bound the VC dimension of signs of low-degree polynomials, and we prove a novel extension of Warren's result applies to functions over these objects.

This approach involves a subtle conceptual shift. Let $D = (x_1, y_1, \ldots, x_n, y_n)$ be a data set. Our notation for the realizability predicate $r_h(D)$ suggests that it is "parameterized" by representations and receives as input a data set. In this section of analysis, these roles are reversed: we construct and analyze polynomials $p_D^{(1)}(h), \ldots, p_D^{(w)}(h)$. The *coefficients* of these polynomials depend on D, but we measure their complexity via their degree as polynomials in h.

1.2. Related Work

There is a large body of literature on both multitask learning and metalearning, including related concepts or alternative names such as *transfer learning*, *learning to learn*, and *few-shot learning*, going back at least as far as the 90s (see Caruana (1997); Thrun and Pratt (1998) for early surveys). Although this line of work is too broad to survey entirely, we summarize here some of the main themes and highlight points of difference with our work.

The most closely related work to ours is that of Baxter (2000), which proves sample complexity bounds for both multitask learning and metalearning in a framework that is equivalent to our shared representation framework. Our main advances compared to Baxter (2000) are: (1) we prove sample complexity bounds in the regime where the number of samples per task is too small to learn an accurate specialized classifier, whereas the generalization arguments in Baxter (2000) crucially rely on having enough samples per task to learn an accurate specialized classifier for a given representation and task, (2) our bounds hold in a distribution-free, classification setting and do not rely on the sort of margin assumptions that are essential for the covering arguments used in Baxter (2000) for the analogous setting. Other works differ from ours along one or more of the following axes.

Stronger assumptions on the tasks or meta-distribution. Several works (Srebro and Ben-David, 2006; Maurer, 2009; Maurer and Pontil, 2012; Pontil and Maurer, 2013; Maurer et al., 2016) prove dimension-free generalization bounds for various forms of linear classifiers over linear representations under stronger margin assumptions on the input. In contrast, our work proves distribution-free bounds, which are necessarily dimension-dependent. Another line of work (Tripuraneni et al., 2020; Du et al., 2020; Tripuraneni et al., 2021) considers multitask/meta learning under *squared loss* under *task diversity* assumptions that make it easier to identify the optimal representation. In contrast, our work gives bounds for the $0/1$-loss without diversity assumptions. We note that the previous works all rely

on covering arguments that are suitable for reasoning about Lipschitz loss functions and margin-based bounds. However, as in the case of single-task learning, these techniques are insufficient for reasoning about $0/1$-loss over worst-case distributions, and we need to use VC-dimension arguments instead.

Other ways to model task relatedness based on Kolmogorov/information complexity include Juba (2006); Ben-David and Borbely (2008); Hassan Mahmud (2009). Lastly Hanneke and Kpotufe (2022) consider a setting where the tasks share the same optimal classifier, not just a common representation that can be specialized to different optimal classifiers.

Computationally efficient algorithms. While our work focuses on the intrinsic information-theoretic sample complexity of multitask and metalearning, there is a long line of work (Balcan et al., 2019; Kong et al., 2020a,b; Saunshi et al., 2020; Fallah et al., 2020; Thekumparampil et al., 2021; Chen et al., 2021; Collins et al., 2022) that studies the sample complexity of specific practical heuristics such as *model-agnostic metalearning (MAML)* under more restrictive assumptions where these heuristics are provably effective. Bairaktari et al. (2023) also give efficient algorithms for multitask learning sparse low-weight halfspaces.

PAC-Bayes bounds for multitask and metalearning. Another important line of work (Pentina and Lampert, 2014; Amit and Meir, 2018; Lucas et al., 2020; Rezazadeh et al., 2021; Rothfuss et al., 2021; Chen et al., 2021; Rezazadeh, 2022) extends the PAC-Bayes and other information-theoretic generalization bounds to the multitask and metalearning problems. These works typically consider properties of both a data distribution and an algorithm that bound generalization error, but do not address the optimal sample complexity of learning specific families, which is the focus of our work.

2. Preliminaries

In this work we consider several different types of objects—representations and classifiers—and several different types of error—training and test error over different distributions. The following table summarizes these error measures. For further details about our notation see Appendix A.

$\mathrm{err}(S, f) = \frac{1}{\lvert S\rvert}\sum_{i=1}^{\lvert S\rvert} \mathbb{I}\{f(x_i) \neq y_i\}$	training error of classifier f
$\mathrm{err}(P, f) = \mathbf{Pr}_{(x,y)\sim P}[f(x) \neq y]$	test error of classifier f
$\mathrm{err}(P, \mathcal{F}) = \min_{f\in\mathcal{F}} \mathbf{Pr}_{(x,y)\sim P}[f(x) \neq y]$	test error of class $\mathcal{F}$
$\text{rep-err}(S, h, \mathcal{F}) = \min_{f\in\mathcal{F}} \mathrm{err}(S, f \circ h)$	training error of representation h
$\text{rep-err}(P, h, \mathcal{F}) = \min_{f\in\mathcal{F}} \mathrm{err}(P, f \circ h) = \mathrm{err}(h(P), \mathcal{F})$	test error of representation h
$\text{rep-err}(\mathfrak{Q}, h, \mathcal{F}) = \mathbb{E}_{P\sim\mathfrak{Q}}[\text{rep-err}(P, h, \mathcal{F})]$	meta-error of representation h
$p_{nr}(P, \mathcal{F}, m) = \mathbf{Pr}_{S\sim P^m}[S \text{ is not realizable by } \mathcal{F}]$	probability of non-realizability

Our sample complexity results are based on classical generalization bounds via VC dimension and pseudodimension. These definitions and results are included in Appendix A and in standard references (Shalev-Shwartz and Ben-David, 2014; Anthony and Bartlett, 1999). The VC dimension of a function class $\mathcal{F}$ is denoted $\mathrm{VC}(\mathcal{F})$ and the pseudodimension is $\mathrm{PDim}(\mathcal{F})$.

3. Metalearning

In this section, we define the metalearning model and outline our arguments bounding the number of tasks and samples per task needed to properly metalearn general classes of representations and specialized classifiers. For complete proofs and additional discussion see Appendix D.

3.1. The Proper Metalearning Model

Suppose there exists a distribution $\mathfrak{Q}$ of different but related binary classification tasks. For each task, specified by a distribution P, we can draw a dataset of labeled examples in $\mathcal{X} \times \{\pm 1\}$. In proper metalearning, we draw t tasks from distribution $\mathfrak{Q}$ and pool together the data from these tasks. We exploit the relatedness between the tasks to learn a common representation $h : \mathcal{X} \to \mathcal{Z}$ in $\mathcal{H}$ that maps the features to a new domain, in the hope that the learned representation facilitates learning new tasks from the same distribution $\mathfrak{Q}$ with fewer samples compared to the baseline where we had to learn those tasks from scratch.

The accuracy of the representation is measured based on how well we can use it to solve a new binary classification task drawn from the same distribution as the ones we have seen in the metalearning phase. The data efficiency of the metalearning algorithm is measured based on two parameters, the number of tasks and the number of samples per task. The proper metalearning setting is particularly interesting when the number of samples per task is too small to learn a good classifier for each task independently, so data must be pooled across tasks to find a representation that can be specialized to new tasks using less data than we would need to learn from scratch.

Definition 3.1 (Proper Metalearning) *Let $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$ and $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$. We say that $(\mathcal{H}, \mathcal{F})$ is* (ε, δ)-properly meta-learnable for t tasks and n samples per task*, if there exists an algorithm $\mathcal{A}$ that for all metadistributions $\mathfrak{Q}$ over distributions on $\mathcal{X} \times \{\pm 1\}$ has the following property: Given n i.i.d. samples per task distribution P_j, where $j \in [t]$, that was drawn indepedently from $\mathfrak{Q}$, $\mathcal{A}$ returns representation $\hat{h} \in \mathcal{H}$ such that with probability at least $1 - \delta$ over the randomness of the samples and the algorithm*

$$\textit{rep-err}\left(\mathfrak{Q}, \hat{h}, \mathcal{F}\right) \leq \min_{h \in \mathcal{H}} \textit{rep-err}(\mathfrak{Q}, h, \mathcal{F}) + \varepsilon. \tag{2}$$

If there exists an algorithm $\mathcal{A}$ such that the same guarantee holds only for all metadistributions $\mathfrak{Q}$ such that $\min_{h \in \mathcal{H}}$ rep-err$(\mathfrak{Q}, h, \mathcal{F}) = 0$ then we say that $(\mathcal{H}, \mathcal{F})$ is (ε, δ)-properly meta-learnable for t tasks and n samples per task in the realizable case.

Definition 3.1 only requires us to learn a good representation $\hat{h}$, and does not explicitly discuss the number of samples required to learn a good specialized classifier for a new task. However, the sample complexity for specialization is essentially determined by the complexity of $\mathcal{F}$. That is, given a new task P' drawn from $\mathfrak{Q}$ and a representation $\hat{h}$ that satisfies (2), $n_{\text{spec}} = O(\text{VC}(\mathcal{F})/\varepsilon^2)$ suffice in order to find an $f \in \mathcal{F}$ such that $f \circ \hat{h}$ has error rep-err$(\mathfrak{Q}, \hat{h}, \mathcal{F}) + 2\varepsilon$. The size n_{spec} of the data used for specialization could be much larger or smaller than the per-task training set size n.

3.2. Sample and Task Complexity Bounds

In this section we give generic sample complexity bounds in terms of two quantities, the VC dimension of the realizability predicate (see Definition 1.6) and the pseudodimension of the empirical

error predicate (see Definition 1.8). For the realizable case, Theorem 3.2 covers the case where we have very few samples per task, which we consider to be the most interesting case, and Theorem 3.5 covers the case where we have more samples per task. The agnostic case is covered in Theorem 3.6.

Theorem 3.2 (Complete statement of Theorem 1.7) *Let $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$ with dual Helly number $\mathrm{DH}(\mathcal{F}) = m$ and $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$. Then, for every ε and δ in $(0, 1)$, we can (ε, δ)-properly metalearn $(\mathcal{H}, \mathcal{F})$ in the realizable case for t tasks and n samples per task when*

$$t = \left(\mathrm{VC}(\mathcal{R}_{m,\mathcal{F},\mathcal{H}}) + \ln(1/\delta)\right) \cdot \left(\frac{O(\max(\mathrm{VC}(\mathcal{F}), m)\ln(1/\varepsilon))}{m\varepsilon}\right)^m \quad \text{and } n = m\,.$$

We use Lemmas 3.3 and 3.4 to prove Theorem 3.2. For a fixed distribution P and class $\mathcal{F}$, Lemma 3.3 allows us to relate the probability of drawing a nonrealizable dataset, $p_{nr}(P, \mathcal{F}, \mathrm{DH}(\mathcal{F}))$, to the test error of class $\mathcal{F}$.

Lemma 3.3 *Let $\mathcal{F}$ be the class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$ with $\mathrm{DH}(\mathcal{F}) = m$. Fix an arbitrary distribution B over $\mathcal{Z} \times \{\pm 1\}$. If $err(B, \mathcal{F}) > 0$, then*

$$p_{nr}(B, \mathcal{F}, m) \geq \frac{1}{2}\left(\left[\frac{m \cdot err(B, \mathcal{F})}{16e \cdot v \ln\left(16/err(B, \mathcal{F})\right)}\right]^m\right),$$

where $v = \max(\mathrm{VC}(\mathcal{F}), m)$.

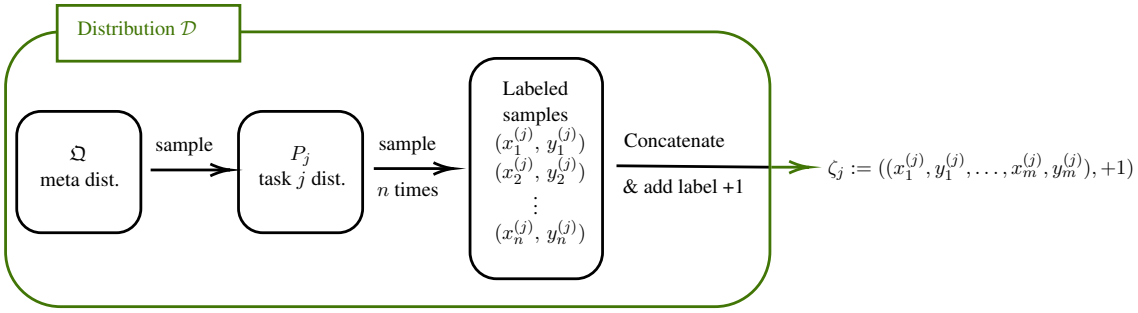

Figure 1: Constructing a datapoint drawn from distribution $\mathcal{D}$.

The prior lemma established a quantitative relationship between the test error of $\mathcal{F}$ and the probability of non-realizability. The following lemma shows how we can exploit this relationship for provable metalearning. For its proof we view the dataset of every task j as a single "data point" ζ_j constructed as depicted in Figure 1. Our goal is to learn a realizability predicate that labels these points correctly. We prove that such a predicate corresponds to a representation with bounded meta-error.

Lemma 3.4 *Let $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$, $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$ and m be a positive integer. Suppose there exists a strictly increasing convex function $\phi : (0, 1] \to [0, 1]$ such that for all metadistributions Ω satisfying $\min_{h\in\mathcal{H}}$ rep-err$(\Omega, h, \mathcal{F}) = 0$, all data distributions P in the support of Ω and all representations $h \in \mathcal{H}$, if rep-err$(P, h, \mathcal{F}) > 0$ the following holds:*

$$\phi(\textit{rep-err}(P, h, \mathcal{F})) \leq \mathbf{Pr}_{(x_1,y_1),\ldots,(x_m,y_m)\sim P^m}\left[(h(x_1), y_1), \ldots, (h(x_m), y_m) \text{ not realizable by } \mathcal{F}\right].$$

Then, we can (ε, δ)-properly metalearn $(\mathcal{H}, \mathcal{F})$ with $t = O\left(\frac{\mathrm{VC}(\mathcal{R}_{m,\mathcal{F},\mathcal{H}})+\ln(1/\delta)}{\phi(\varepsilon)}\right)$ tasks and m samples per task in the realizable case.

Theorem 3.5 *Let $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$ and $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$. For every $\varepsilon, \delta \in (0,1)$, we can (ε, δ)-properly metalearn $(\mathcal{H}, \mathcal{F})$ in the realizable case with t tasks and n samples per task when $t = O\left(\frac{\mathrm{VC}(\mathcal{R}_{n,\mathcal{F},\mathcal{H}})\ln(1/\varepsilon)+\ln(1/\delta)}{\varepsilon}\right)$ and $n = O\left(\frac{\mathrm{VC}(\mathcal{F})\ln(1/\varepsilon)}{\varepsilon}\right)$.*

Next, we consider metalearning when the underlying metadistribution is not realizable, and show that the pseudodimension of the empirical error predicate (Definition 1.8) bounds the number of tasks needed.

Theorem 3.6 (Restatement of Theorem 1.9) *Let $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$ and $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$. Then, for every ε and $\delta \in (0,1)$ we can (ε, δ)-properly metalearn $(\mathcal{H}, \mathcal{F})$ with t tasks and n samples per task when $t = O\left(\frac{\mathit{PDim}(\mathcal{Q}_{n,\mathcal{F},\mathcal{H}})\ln(1/\varepsilon)+\ln(1/\delta)}{\varepsilon^2}\right)$ and $n = O\left(\frac{\mathrm{VC}(\mathcal{F})+\ln(1/\varepsilon)}{\varepsilon^2}\right)$.*

4. Metalearning of Halfspaces over Linear Representations

In this section, we focus on metalearning of halfspaces with a shared linear representation:

$$\mathcal{H}_{d,k} = \left\{h \mid h(\mathbf{x}) = \mathbf{B}\mathbf{x}, \mathbf{B} \in \mathbb{R}^{k\times d}\right\}, \text{ and } \mathcal{F}_k = \left\{f \mid f(\mathbf{z}) = \mathrm{sign}(\mathbf{a}\cdot\mathbf{z} - w), \mathbf{a} \in \mathbb{R}^k, w \in \mathbb{R}\right\}.$$

More precisely, the representation is a linear projection that maps vectors in d dimensions to vectors in k dimensions. The specialized classifiers are halfspaces over the k-dimensional representation.

We use our results for general classes from Section 3 to bound the number of tasks and the number of samples per task we need to properly metalearn $(\mathcal{H}_{d,k}, \mathcal{F}_k)$. Our results (Corollary 4.3 for realizable learning and Corollary 4.6 for agnostic) rely on bounding the VC dimension of the class of realizability predicates and the pseudodimension of the class of empirical error functions. See Appendix E for proofs and extensions.

Before presenting our results for proper metalearning, we provide a lemma that generalizes Warren's theorem (Lemma E.1). We construct a parameterized class of hypotheses based on a given Boolean function g whose inputs are signs of polynomials. Appealing to Warren's theorem, we show that the VC dimension of this class cannot be too large.

Lemma 4.1 *Let $w, \ell, d \in \mathbb{N}$ with $\ell \geq 2$ and fix a domain of features $\mathcal{X}$ and a function $g : \{\pm 1\}^w \to \{\pm 1\}$. For each $x \in \mathcal{X}$, let $p_x^{(1)}, \ldots, p_x^{(w)}$ be degree-ℓ polynomials in d variables. Consider the hypothesis class $\mathcal{C}$ consisting of, for all $\mathbf{v} \in \mathbb{R}^d$, the functions $c_{\mathbf{v}} : \mathcal{X} \to \{\pm 1\}$ defined as*

$$c_{\mathbf{v}}(x) := g(\mathit{sign}(p_x^{(1)}(\mathbf{v})), \ldots, \mathit{sign}(p_x^{(w)}(\mathbf{v}))). \tag{3}$$

We have $\mathrm{VC}(\mathcal{C}) \leq 8d\log_2(\ell w)$.

Our central result on halfspaces over linear representations in the realizable setting is Theorem 4.2, which bounds the VC dimension of the realizability predicates (see Definition 1.6). This VC dimension bound, combined with our metalearning bounds for general classes in Section 3, immediately yields our statement about metalearning presented in Corollary 4.3.

Theorem 4.2 *For every d, k, n with $n \geq k+2$, $\mathrm{VC}(\mathcal{R}_{n,\mathcal{F}_k,\mathcal{H}_{d,k}}) \leq O(dkn + dk\log(kn))$.*

Corollary 4.3 (Detailed version of Corollary 1.3) *We can (ε,δ)-properly metalearn $(\mathcal{H}_{d,k},\mathcal{F}_k)$ in the realizable case with t tasks and n samples per task when $t = \big(dk^2 + \ln(1/\delta)\big)\cdot O\Big(\frac{\ln(1/\varepsilon)}{\varepsilon}\Big)^{k+2}$ and $n = k+2$, or $t = O\Big(\frac{dk^2\ln^2(1/\varepsilon)}{\varepsilon^2} + \frac{\ln(1/\delta)}{\varepsilon}\Big)$ and $n = O\Big(\frac{k\ln(1/\varepsilon)}{\varepsilon}\Big)$.*

Proof [of Corollary 4.3] For the class of halfspaces, the following result of Kirchberger gives the exact dual Helly number:

Fact 4.4 (Kirchberger (1903)) *Every dataset of size at least $k+2$ that is not realizable by k-dimensional halfspaces has a subset of size $k+2$ that is not realizable by k-dimensional halfspaces. Therefore, $\mathrm{DH}(\mathcal{F}_k) = k+2$.*

Theorem A.6 states that the VC dimension of the linear separators for k-dimensional points is $k+1$. Using these facts, Theorem 3.2, and Theorem 4.2, we obtain the first task and sample complexities in the statement of the corollary. Similarly, the second line of complexities is derived when we use Theorem 3.5 instead of Theorem 3.2. ■

We now turn to bounding the pseudodimension (Definition A.7) of the class of empirical error functions (Definition 1.8). This bound on the pseudodimension immediately gives the statement about the task complexity of metalearning linear representations.

Theorem 4.5 (Thm. 1.10, Restated) *For all d, k, n with $n \geq k+2$, $\mathrm{PDim}(Q_{n,\mathcal{F}_k,\mathcal{H}_{d,k}}) = \tilde{O}(dkn)$.*

Corollary 4.6 *We can (ε,δ)-properly metalearn $(\mathcal{H}_{d,k},\mathcal{F}_k)$ with t tasks and n samples per task when $t = O\Big(\frac{dk^2\ln(1/\varepsilon)+dk\ln(1/\varepsilon)^2}{\varepsilon^4} + \frac{\ln(1/\delta)}{\varepsilon^2}\Big)$ and $n = O\Big(\frac{k+\ln(1/\varepsilon)}{\varepsilon^2}\Big)$.*

Acknowledgments

Part of this work was performed while MA was at Northeastern University and Boston University, and while GB was at Boston University. MA, KB, and JU were supported by NSF awards CNS-2120603, CNS-2232692, and CNS-2247484. MA and GB, while at Boston University, and AS were supported in part by NSF awards CCF-1763786 and CNS-2120667 and Faculty Awards from Google and Apple. While at the University of Washington, GB was supported by NSF Award 2019844. While at Boston University, MA was supported by BU's Hariri Institute of Computing.

References

Ron Amit and Ron Meir. Meta-learning by adjusting priors based on extended pac-bayes theory. In *International Conference on Machine Learning*, pages 205–214. PMLR, 2018.

Martin Anthony and Peter L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999. doi: 10.1017/CBO9780511624216.

Konstantina Bairaktari, Guy Blanc, Li-Yang Tan, Jonathan Ullman, and Lydia Zakynthinou. Multitask learning via shared features: Algorithms and hardness. In *The Thirty Sixth Annual Conference on Learning Theory*, COLT '23. PMLR, 2023.

Maria-Florina Balcan, Mikhail Khodak, and Ameet Talwalkar. Provable guarantees for gradient-based meta-learning. In *International Conference on Machine Learning*, pages 424–433. PMLR, 2019.

Jonathan Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12: 149–198, 2000.

Shai Ben-David and Reba Schuller Borbely. A notion of task relatedness yielding provable multiple-task learning guarantees. *Machine learning*, 73:273–287, 2008.

Olivier Bousquet, Steve Hanneke, Shay Moran, and Nikita Zhivotovskiy. Proper learning, helly number, and an optimal svm bound. In Jacob Abernethy and Shivani Agarwal, editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 582–609. PMLR, 09–12 Jul 2020.

Rich Caruana. Multitask learning. *Machine Learning*, 28:41–75, 1997.

Qi Chen, Changjian Shui, and Mario Marchand. Generalization bounds for meta-learning: An information-theoretic analysis. *Advances in Neural Information Processing Systems*, 34:25878–25890, 2021.

Liam Collins, Aryan Mokhtari, Sewoong Oh, and Sanjay Shakkottai. Maml and anil provably learn representations. In *International Conference on Machine Learning*, pages 4238–4310. PMLR, 2022.

Simon S Du, Wei Hu, Sham M Kakade, Jason D Lee, and Qi Lei. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*, 2020.

Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. On the convergence theory of gradient-based model-agnostic meta-learning algorithms. In *International Conference on Artificial Intelligence and Statistics*, pages 1082–1092. PMLR, 2020.

Steve Hanneke and Samory Kpotufe. A no-free-lunch theorem for multitask learning. *The Annals of Statistics*, 50(6):3119–3143, 2022.

M.M. Hassan Mahmud. On universal transfer learning. *Theoretical Computer Science*, 410(19): 1826–1846, 2009. ISSN 0304-3975. doi: https://doi.org/10.1016/j.tcs.2009.01.013. Algorithmic Learning Theory.

Brendan Juba. Estimating relatedness via data compression. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 441–448, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933832.

Daniel Kane, Roi Livni, Shay Moran, and Amir Yehudayoff. On communication complexity of classification problems. In Alina Beygelzimer and Daniel Hsu, editors, *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 1903–1943. PMLR, 25–28 Jun 2019.

Paul Kirchberger. Über tchebychefsche annäherungsmethoden. *Mathematische Annalen*, 57(4): 509–540, 1903.

Weihao Kong, Raghav Somani, Sham Kakade, and Sewoong Oh. Robust meta-learning for mixed linear regression with small batches. *Advances in neural information processing systems*, 33: 4683–4696, 2020a.

Weihao Kong, Raghav Somani, Zhao Song, Sham Kakade, and Sewoong Oh. Meta-learning for mixed linear regression. In *International Conference on Machine Learning*, pages 5394–5404. PMLR, 2020b.

James Lucas, Mengye Ren, Irene Kameni, Toniann Pitassi, and Richard Zemel. Theoretical bounds on estimation error for meta-learning. In *International Conference on Learning Representations*, 2020.

Andreas Maurer. Transfer bounds for linear feature learning. *Machine learning*, 75(3):327–350, 2009.

Andreas Maurer and Massimiliano Pontil. Transfer learning in a heterogeneous environment. In *2012 3rd International Workshop on Cognitive Information Processing (CIP)*, pages 1–6, 2012. doi: 10.1109/CIP.2012.6232893.

Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. The benefit of multitask representation learning. *Journal of Machine Learning Research*, 17(81):1–32, 2016.

Anastasia Pentina and Christoph Lampert. A pac-bayesian bound for lifelong learning. In *International Conference on Machine Learning*, pages 991–999. PMLR, 2014.

Massimiliano Pontil and Andreas Maurer. Excess risk bounds for multitask learning with trace norm regularization. In *Conference on Learning Theory*, pages 55–76. PMLR, 2013.

Arezou Rezazadeh. A general framework for pac-bayes bounds for meta-learning. *arXiv preprint arXiv:2206.05454*, 2022.

Arezou Rezazadeh, Sharu Theresa Jose, Giuseppe Durisi, and Osvaldo Simeone. Conditional mutual information-based generalization bound for meta learning. In *2021 IEEE International Symposium on Information Theory (ISIT)*, pages 1176–1181. IEEE, 2021.

Jonas Rothfuss, Vincent Fortuin, Martin Josifoski, and Andreas Krause. Pacoh: Bayes-optimal meta-learning with pac-guarantees. In *International Conference on Machine Learning*, pages 9116–9126. PMLR, 2021.

Nikunj Saunshi, Yi Zhang, Mikhail Khodak, and Sanjeev Arora. A sample complexity separation between non-convex and convex meta-learning. In *International Conference on Machine Learning*, pages 8512–8521. PMLR, 2020.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Nathan Srebro and Shai Ben-David. Learning bounds for support vector machines with learned kernels. In Gábor Lugosi and Hans Ulrich Simon, editors, *19th Annual Conference on Learning Theory, COLT 2006*, pages 169–183. Springer, 2006.

Kiran Koshy Thekumparampil, Prateek Jain, Praneeth Netrapalli, and Sewoong Oh. Sample efficient linear meta-learning by alternating minimization. *arXiv preprint arXiv:2105.08306*, 2021.

Sebastian Thrun and Lorien Y. Pratt, editors. *Learning to Learn*. Springer, 1998.

Nilesh Tripuraneni, Michael Jordan, and Chi Jin. On the theory of transfer learning: The importance of task diversity. *Neural Information Processing Systems*, 33:7852–7862, 2020.

Nilesh Tripuraneni, Chi Jin, and Michael Jordan. Provable meta-learning of linear representations. In *International Conference on Machine Learning*, ICML '21, pages 10434–10443. PMLR, 2021.

Hugh E. Warren. Lower bounds for approximation by nonlinear manifolds. *Transactions of the American Mathematical Society*, 133(1):167–178, 1968. ISSN 00029947.

Appendix A. Notation and Background

In this paper, we use bold lowercase letters for vectors, e.g. $\mathbf{a}$, and bold uppercase letters for matrices, e.g. $\mathbf{A}$. Given a d-dimensional vector $\mathbf{a}$ and a value b, $(\mathbf{a}\|b)$ is a $d+1$-dimensional vector, where we have appended value b to vector $\mathbf{a}$. For distributions over labeled data points we use the regular math font, e.g. D, whereas for metadistributions (distributions over task data distributions) we use fraktur letters, e.g. $\mathfrak{D}$. We define the indicator function of a predicate p as

$$\mathbb{I}\{p\} = \begin{cases} 1 & \text{if } p \text{ is true} \\ 0 & \text{o.w.} \end{cases}.$$

Occasionally, we use the alternative indicator $\mathbb{I}_{\pm}\{p\}$, which takes value $+1$ if p is true and -1 otherwise. We also use the following sign function

$$\text{sign}(x) = \begin{cases} +1 & \text{if } x \geq 0 \\ -1 & \text{o.w.} \end{cases}.$$

Contrary to the convention in Boolean analysis, when necessary we interpret $+1$ as logical "true" and -1 as logical "false."

VC dimension. We start by recalling the standard definition of VC dimension and the Sauer-Shelah Lemma.

Definition A.1 *Let $\mathcal{F}$ be a set of functions mapping from a domain $\mathcal{X}$ to $\{\pm 1\}$ and suppose that $X = (x_1, \ldots, x_n) \subseteq \mathcal{X}$. We say that $\mathcal{F}$* shatters *X if, for all $b \in \{\pm 1\}^n$, there is a function $f_b \in \mathcal{F}$ with $f_b(x_i) = b_i$ for each $i \in [n]$.*

The VC dimension *of $\mathcal{F}$, denoted $\mathrm{VC}(\mathcal{F})$, is the size of the largest set X that is shattered by $\mathcal{F}$.*

VC dimension is closely related to the *growth function*, which bounds the number of distinct labelings a hypothesis class can produce on any fixed data set.

Definition A.2 *Let $\mathcal{F}$ be a class of functions $f : \mathcal{Z} \to \{\pm 1\}$ and $S = \{z_1, \ldots, z_n\}$ be a set of points in $\mathcal{Z}$. The* restriction *of $\mathcal{F}$ to S is the set of functions*

$$\mathcal{F}_S = \{(f(z_1), \ldots, f(z_n)) \mid f \in \mathcal{F}\}.$$

The growth function *of $\mathcal{F}$, denoted $\mathcal{G}_{\mathcal{F}}(n)$, is $\mathcal{G}_{\mathcal{F}}(n) := \sup_{S:|S|=n} |\mathcal{F}_S|$.*

The Sauer-Shelah Lemma, which bounds the growth function in terms of the VC dimension.

Lemma A.3 *If* $\mathrm{VC}(\mathcal{F}) = v$*, then for every* $n > v$*,* $\mathcal{G}_{\mathcal{F}}(n) \leq (en/v)^v$.

VC dimension and PAC learning. Now we recall the relationship between VC dimension and the sample complexity of distribution-free PAC learning. Here we refer to the textbook notion of PAC learning without giving a formal definition.

Theorem A.4 *Let* $\mathcal{C}$ *be a hypothesis class of functions* $f : \mathcal{X} \to \{\pm 1\}$ *with* $\mathrm{VC}(\mathcal{C}) = v < \infty$*, then for every* $\varepsilon, \delta \in (0, 1)$

1. $\mathcal{C}$ *has uniform convergence with sample complexity* $O(\frac{v+\ln(1/\delta)}{\varepsilon^2})$
2. $\mathcal{C}$ *is agnostic PAC learnable with sample complexity* $O(\frac{v+\ln(1/\delta)}{\varepsilon^2})$
3. $\mathcal{C}$ *is PAC learnable with sample complexity* $O(\frac{v\ln(1/\varepsilon)+\ln(1/\delta)}{\varepsilon})$.

While uniform convergence requires $O(1/\varepsilon^2)$ samples, with just $O(\ln(1/\varepsilon)/\varepsilon)$ samples we will have the property that every hypothesis that has error ε on the distribution has non-zero error on the sample.

Theorem A.5 *Let* $\mathcal{C}$ *be a hypothesis class with VC dimension* v*. Let* D *be a probability distribution over* $\mathcal{X} \times \{\pm 1\}$*. For any* $\varepsilon, \delta > 0$ *if we draw a sample* S *from* D *of size* n *satisfying*

$$n \geq \frac{8}{\varepsilon}\left(v \ln\left(\frac{16}{\varepsilon}\right) + \ln\left(\frac{2}{\delta}\right)\right)$$

then with probability at least $1 - \delta$*, all hypotheses* f *in* $\mathcal{F}$ *with err*$(D, f) > \varepsilon$ *have err*$(S, f) > 0$.

VC dimension of halfspaces. For a significant part of this paper, we work with linear classifiers of the form

$$\mathcal{F}_d = \left\{ f \middle| f(\mathbf{x}) = \mathrm{sign}(\mathbf{a} \cdot \mathbf{x} - w), \mathbf{a} \in \mathbb{R}^d, w \in \mathbb{R} \right\}.$$

We also consider the class of linear classifiers that pass through the origin

$$\tilde{\mathcal{F}}_d = \left\{ f \middle| f(\mathbf{x}) = \mathrm{sign}(\mathbf{a} \cdot \mathbf{x}), \mathbf{a} \in \mathbb{R}^d \right\}.$$

Theorem A.6 *We have* $\mathrm{VC}(\mathcal{F}_d) = d + 1$ *and* $\mathrm{VC}(\tilde{\mathcal{F}}_d) = d$.

Pseudodimension. For real-valued functions we use generalization bounds based on a generalization of VC dimension called the *pseudodimension.*

Definition A.7 *Let* $\mathcal{F}$ *be a set of functions mapping from a domain* $\mathcal{X}$ *to* $\mathbb{R}$ *and suppose that* $X = (x_1, \ldots, x_n) \subseteq \mathcal{X}$*. We say that* X *is* pseudoshattered *by* $\mathcal{F}$ *if there are real numbers* $r_1, \ldots, r_n$ *such that for each* $b \in \{\pm 1\}^n$ *there is a function* $f_b \in \mathcal{F}$ *with* $\mathrm{sign}(f_b(x_i) - r_i) = b_i$ *for each* $i \in [n]$.

The pseudodimension *of* $\mathcal{F}$*, denoted* $\mathrm{Pdim}(\mathcal{F})$*, is the size of the largest set* X *that is pseudoshattered by* $\mathcal{F}$.

Theorem A.8 *Let* $\mathcal{C}$ *be a hypothesis class of functions* $f : \mathcal{X} \to \mathbb{R}$ *with PDim*$(\mathcal{C}) = \psi < \infty$*, then* $\mathcal{C}$ *has uniform convergence with sample complexity* $O(\frac{\psi\ln(1/\varepsilon)+\ln(1/\delta)}{\varepsilon^2})$.

Appendix B. Multitask Learning and Metalearning Models

In this section we introduce two other learning models we also consider in this work, one for multitask learning and one for general (improper) metalearning.

B.1. The Multitask Learning Model

In multitask learning, we pool data together from t related tasks with the aim of finding one classifier per task so that the average test error per task is low. When these tasks are related, we may need fewer samples per task than if we learn them separately, because samples from one task inform us about the distribution of another task. In this work, we consider classifiers that use a shared representation to map features to an intermediate space in which the specialized classifiers are defined. We want to achieve low error on tasks that are related by a single shared representation that can be specialized to obtain a low-error classifier for most tasks.

Definition B.1 (Multitask learning) *Let $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$ and $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$. We say that $(\mathcal{H}, \mathcal{F})$ is* distribution-free (ε, δ)-multitask learnable for t tasks with n samples per task *if there exists an algorithm $\mathcal{A}$ such that for every t probability distributions $P_1, \ldots, P_t$ over $\mathcal{X} \times \{\pm 1\}$, for every $h \in \mathcal{H}$ and every $f_1, \ldots, f_t \in \mathcal{F}$, given n i.i.d. samples from each P_i, returns hypothesis $g : [t] \times \mathcal{X} \to \{\pm 1\}$ such that with probability at least $1 - \delta$ over the randomness of the samples and the algorithm*

$$\frac{1}{t} \sum_{j \in [t]} \mathit{err}(P_j, g(j, \cdot)) \leq \min_{h \in \mathcal{H}, f_1, \ldots, f_t \in \mathcal{F}} \frac{1}{t} \sum_{j \in [t]} \mathit{err}(P_j, f_j \circ h) + \varepsilon.$$

If there exists an algorithm $\mathcal{A}$ such that the same guarantee holds except that we only quantify over all distributions $P_1, \ldots, P_t$ such that

$$\min_{h \in \mathcal{H}, f_1, \ldots, f_t \in \mathcal{F}} \frac{1}{t} \sum_{j \in [t]} \mathit{err}(P_j, f_j \circ h) = 0$$

then we say that $(\mathcal{H}, \mathcal{F})$ is distribution-free (ε, δ)-multitask learnable for t tasks with n samples per task in the realizable case*.*

For brevity, we will typically omit the term "distribution-free," which applies to all of the results in this paper.

The natural approach to learning $(\mathcal{H}, \mathcal{F})$ is to learn a classifier that, given the index of a task and the features of a sample, computes the representation of the sample and then labels it using the task-specific specialized classifier. We call the class of these classifiers $\mathcal{F}^{\otimes t} \circ \mathcal{H}$.

Definition B.2 *Let $\mathcal{H}$ be a class of representations $h : \mathcal{X} \to \mathcal{Z}$ and f be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$. We define the class of composed classifiers for multitask learning with t tasks as*

$$\mathcal{F}^{\otimes t} \circ \mathcal{H} = \{g : [t] \times \mathcal{X} \to \{\pm 1\} \mid \exists h \in \mathcal{H}, f_1, \ldots, f_t \in \mathcal{F} \text{ s.t. } g(j, x) = f_j(h(x))\}.$$

B.2. The General (Improper) Metalearning Model

We can also think of metalearning as a more general process than the proper metalearning model we defined which finds a good representation. At a high level the metalearning process pools together datasets from multiple tasks and returns an algorithm that we can use in the future to obtain good classifiers for new tasks drawn from the same metadistribution. In proper metalearning we did not just specify the form of the output of the specialization algorithm, but also the specialization algorithm itself. In that case, the metalearning process finds a representation $\hat{h} \in \mathcal{H}$ and everytime the specialization algorithm is called it performs ERM using the samples from the new task to find a $f \in \mathcal{F}$ that minimizes the training error of $f \circ \hat{h}$.

Definition B.3 (General Metalearning) *Let $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$ and $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$. We say that $(\mathcal{H}, \mathcal{F})$ is* (ε, δ)-meta-learnable for t tasks, n samples per task and n_{spec} specialization samples*, if there exists an algorithm $\mathcal{A}$ that for all metadistributions $\mathfrak{Q}$ over distributions on $\mathcal{X} \times \{\pm 1\}$ has the following property: Given n i.i.d. samples from each data distribution P_j, where $j \in [t]$, that was drawn independently from $\mathfrak{Q}$, $\mathcal{A}$ returns a specialization algorithm $\mathcal{A}'$. Algorithm $\mathcal{A}'$, given a set S of n_{spec} samples from P which was drawn from $\mathfrak{Q}$, returns hypothesis $g : \mathcal{X} \to \{\pm 1\}$ such that with probability at least $1 - \delta$ over the randomness of the samples of the t tasks and algorithms $\mathcal{A}$ and $\mathcal{A}'$*

$$\mathop{\mathbb{E}}_{P,S}\Big[err(P, \mathcal{A}'(S))\Big] \leq \min_{h \in \mathcal{H}} \mathop{\mathbb{E}}_{P}\left[\min_{f \in \mathcal{F}} err(P, f \circ h)\right] + \varepsilon.$$

B.3. Reductions between improper metalearning and multitask learning

We show how we can reduce metalearning to multitask learning. The algorithm we construct for the reduction stores the given t datasets. For every new task it performs multitask learning on the $t + 1$ datasets and returns the hypothesis that corresponds to the new task. We need to highlight that this is not a proper metalearning algorithm, since it does not return one representation that is used for all new tasks, but the algorithm builds the classifier from scratch every time it is given a new task.

Theorem B.4 *Suppose $(\mathcal{H}, \mathcal{F})$ is $(\varepsilon, \varepsilon)$-multitask learnable for $t + 1$ tasks and n samples per task. Then, for all $c > 0$, it is (improperly) $(c\varepsilon, 2/c)$-meta-learnable, for t tasks, n samples per task and n specialization samples.*

Proof Given a multitask learning algorithm $\mathcal{A}_{\text{multi}}$, we construct a metalearning algorithm $\mathcal{A}_{\text{meta}}$ that stores the t data sets and, when given the new dataset, runs $\mathcal{A}_{\text{multi}}$ on the collected $t + 1$ datasets. Formally, recall that a multitask learning algorithm sees the datasets of $t + 1$ tasks and outputs one hypothesis per task minimizing the average test error, while a metalearning algorithm sees t datasets from tasks drawn from the same metadistribution and outputs an algorithm $\mathcal{A}_{\text{spec}}$ that can be used to train a model for a new task.

In our reduction, the metalearning algorithm $\mathcal{A}_{\text{meta}}$, on input data sets $S_1, ..., S_t$, returns an algorithm $\mathcal{A}_{\text{spec}}$ which, given a set S_{new} of n samples from distribution P_{new} for a new task, executes the following steps. First, it draws an index j_{new} uniformly at random from $[t + 1]$ and inserts the new task in this position by setting $S'_{j_{\text{new}}} = S_{\text{new}}$. For $j < j_{\text{new}}$ it sets $S'_j = S_j$ and for $j > j_{\text{new}}$ $S'_j = S_{j+1}$. For notation purposes, let P'_j be the distribution that S'_j was drawn from. Second, it runs the multitask algorithm $\mathcal{A}_{\text{multi}}$ for $t + 1$ tasks and n samples for each of these tasks. At the end, it outputs the classifier it has computed for the j_{new}-th task.

By our assumption, the guarantee that multitask learning gives is that $\mathcal{A}_{\text{multi}}$ outputs $g_1, \ldots, g_{t+1}$ such that with probability at least $1-\varepsilon$ over the randomness of $\mathcal{A}_{\text{multi}}$ and the $t+1$ datasets $S'_1, \ldots, S'_{t+1}$

$$\frac{1}{t+1}\sum_{j\in[t+1]} \text{err}(P'_j, g_j) \leq \min_{h\in\mathcal{H}; f_1,\ldots,f_{t+1}\in\mathcal{F}} \frac{1}{t+1}\sum_{j\in[t+1]} \text{err}(P'_j, f_j \circ h) + \varepsilon. \tag{4}$$

Therefore, for all distributions $P'_1, \ldots, P'_{t+1}$ in the support of $\mathfrak{Q}$

$$\mathop{\mathbb{E}}_{\mathcal{A}_{\text{multi}}, S_1,\ldots,S_{t+1}}\left[\frac{1}{t+1}\sum_{j\in[t+1]} \text{err}(P'_j, g_j)\right] \leq \min_{h\in\mathcal{H}; f_1,\ldots,f_{t+1}\in\mathcal{F}} \frac{1}{t+1}\sum_{j\in[t+1]} \text{err}(P'_j, f_j \circ h) + 2\varepsilon.$$

We now want to compute the expected error of the metalearning algorithm over the randomness of the algorithm and the stored datasets. Since we draw index j_{new} uniformly at random,

$$\mathop{\mathbb{E}}_{\mathcal{A}_{\text{spec}}, P_1,\ldots,P_t, S_1,\ldots,S_t,}\left[\mathop{\mathbb{E}}_{P,S}\left[\text{err}(P, \mathcal{A}_{\text{spec}}(S))\right]\right] =$$

$$\mathop{\mathbb{E}}_{\mathcal{A}_{\text{multi}}, j\sim\text{Unif}([t+1]), P_1,\ldots,P_t, S_1,\ldots,S_t}\left[\mathop{\mathbb{E}}_{P,S}\left[\text{err}(P, \mathcal{A}_{\text{multi}}(S'_1, \ldots, S'_{t+1})_j)\right]\right] =$$

$$\mathop{\mathbb{E}}_{\mathcal{A}_{\text{multi}}, P_1,\ldots,P_t, S_1,\ldots,S_t}\left[\frac{1}{t+1}\sum_{j\in[t+1]}\mathop{\mathbb{E}}_{P,S}\left[\text{err}(P, \mathcal{A}_{\text{multi}}(S'_1, \ldots, S'_{t+1})_j)\right]\right] =$$

$$\mathop{\mathbb{E}}_{\mathcal{A}_{\text{multi}}, P'_1,\ldots,P'_{t+1}, S'_1,\ldots,S'_{t+1}}\left[\frac{1}{t+1}\sum_{j\in[t+1]} \text{err}(P'_j, \mathcal{A}_{\text{multi}}(S'_1, \ldots, S'_{t+1})_j)\right].$$

Combining the above results we obtain the following inequality

$$\mathop{\mathbb{E}}_{\mathcal{A}_{\text{spec}}, P_1,\ldots,P_t, S_1,\ldots,S_t}\left[\mathop{\mathbb{E}}_{P,S}\left[\text{err}(P, \mathcal{A}_{\text{spec}}(S))\right]\right] \leq$$

$$\mathop{\mathbb{E}}_{P'_1,\ldots,P'_{t+1}}\left[\min_{h\in\mathcal{H}, f_1,\ldots,f_{t+1}\in\mathcal{F}} \frac{1}{t+1}\sum_{j\in[t+1]} \text{err}(P'_j, f_j \circ h)\right] + 2\varepsilon \leq$$

$$\min_{h\in\mathcal{H}} \mathop{\mathbb{E}}_{P'_1,\ldots,P'_{t+1}}\left[\frac{1}{t+1}\sum_{j\in[t+1]} \min_{f_j\in\mathcal{F}} \text{err}(P'_j, f_j \circ h)\right] + 2\varepsilon =$$

$$\min_{h\in\mathcal{H}} \mathop{\mathbb{E}}_{P}\left[\min_{f\in\mathcal{F}} \text{err}(P, f \circ h)\right] + 2\varepsilon.$$

Therefore, we have shown that

$$\mathop{\mathbb{E}}_{\mathcal{A}_{\text{spec}}, P_1,\ldots,P_t S_1,\ldots,S_t}\left[\mathop{\mathbb{E}}_{P,S}\left[\text{err}(P, \mathcal{A}_{\text{spec}}(S))\right] - \min_{h\in\mathcal{H}} \mathop{\mathbb{E}}_{P}\left[\min_{f\in\mathcal{F}} \text{err}(P, f \circ h)\right]\right] < 2\varepsilon.$$

We apply Markov's inequality to obtain the following probability

$$\mathbf{Pr}_{\mathcal{A}_{\text{spec}}, P_1,\ldots,P_t, S_1,\ldots,S_t}\left[\mathop{\mathbb{E}}_{P,S}\left[\text{err}(P, \mathcal{A}_{\text{spec}}(S))\right] - \min_{h\in\mathcal{H}} \mathop{\mathbb{E}}_{P}\left[\min_{f\in\mathcal{F}} \text{err}(P, f \circ h)\right] > c\varepsilon\right]$$
$$\leq \frac{\mathbb{E}_{\mathcal{A}_{\text{spec}}, P_1,\ldots,P_t, S_1,\ldots,S_t}\left[\mathbb{E}_{P,S}\left[\text{err}(P, \mathcal{A}_{\text{spec}}(S))\right] - \min_{h\in\mathcal{H}} \mathbb{E}_P[\min_{f\in\mathcal{F}} \text{err}(P, f \circ h)]\right]}{c\varepsilon} < \frac{2}{c}.$$

To conclude, we showed that with probability at least $1 - \frac{2}{c}$ over algorithm $\mathcal{A}_{\text{spec}}$

$$\mathop{\mathbb{E}}_{P,S}\left[\text{err}(P, \mathcal{A}_{\text{spec}}(S))\right] \leq \min_{h \in \mathcal{H}} \mathop{\mathbb{E}}_{P}\left[\min_{f \in \mathcal{F}} \text{err}(P, f \circ h)\right] + c\varepsilon.$$

■

Corollary 4.6 captures one application of this general reduction to the case of linear classes with n roughly k/ε^2 samples per task.

We can also show the opposite direction, how to reduce multitask learning to metalearning. We first use some of the samples from the tasks to learn a specialization algorithm and then we run this algorithm to learn one classifier per task. Since metalearning assumes the existence of a metadistribution, we set it to be the uniform distribution over the distributions of the seen tasks. As a result, when we draw tasks from the metadistribution we might draw the same task more than once, which means we need additional samples from that task. We use a standard balls-and-bins argument (Lemma B.5) to argue that we have enough samples per task to generate independent samples for metalearning.

Lemma B.5 *Suppose we draw t samples from a uniform distribution over $[t]$. Let F_i denote the frequency of each $i \in [t]$ in the sample set. With probability $1 - \delta$, $\max_i F_i$ is at most $\frac{\ln((t/\delta)^2)}{\ln\ln((t/\delta)^2)}$.*

Theorem B.6 *Suppose $(\mathcal{H}, \mathcal{F})$ is (improperly) $(\varepsilon, \varepsilon)$-metalearnable for t tasks, n samples per task and n_{spec} specialization samples. Then, for all $c > 0$, it is $(c\varepsilon, 3/c)$-multitask learnable for t tasks and $n\lceil 2\ln(ct)\rceil + n_{spec}$ samples per task.*

Proof Given a metalearning algorithm $\mathcal{A}_{\text{meta}}$ that returns a specialization algorithm $\mathcal{A}_{\text{spec}}$, we construct a multitask learning algorithm $\mathcal{A}_{\text{multi}}$.

In our reduction, $\mathcal{A}_{\text{multi}}$ gets as input t datasets $S_1, \ldots, S_t$, where each S_j has $n\lceil \ln(ct)\rceil + n_{\text{spec}}$ samples drawn i.i.d. from distribution P_j, and executes the following steps:

1. Draw t indices of tasks $i_1, \ldots, i_t$ uniformly at random from $[t]$ with replacement. A pair of tasks $i_j, i_{j'}$ might correspond to the same original task.
2. Build datasets $S'_1, \ldots, S'_t$ as follows. For each j, if there are no samples remaining in S_{i_j}, fail and output $\perp$, otherwise let S'_j be the next n samples from S_{i_j} and delete those samples from S_{i_j}.
3. Run $\mathcal{A}_{\text{meta}}(S'_1, \ldots, S'_t)$ which returns $\mathcal{A}_{\text{spec}}$.
4. For every $j \in [t]$ define dataset S_j^{spec} as a set of n_{spec} datapoints from S_j. As in Step 2, avoid using the same datapoint twice (and output $\perp$ if that's impossible).
5. For every task j, set $g_j = \mathcal{A}_{\text{spec}}(S_j^{\text{spec}})$.

The goal in multitask learning is to bound the probability

$$\mathbf{Pr}_{S_1,\ldots,S_t,\mathcal{A}_{\text{multi}}}\left[\frac{1}{t}\sum_{j\in[t]} \text{err}(P_j, g_j) \leq \min_{h\in\mathcal{H}, f_1,\ldots,f_t\in\mathcal{F}} \frac{1}{t}\sum_{j\in[t]} \text{err}(P_j, f \circ h) + c\varepsilon\right]$$

for all $P_1, \ldots, P_t$.

Fix t tasks $P_1, \ldots, P_t$. The multitask learning algorithm receives a dataset per task S_j of $n\lceil \ln(ct) \rceil + n_{\text{spec}}$ samples as input, and in steps 1 and 2 constructs an appropriate input for the metalearning algorithm. Let $\mathcal{B}$ be the event that the algorithm outputs $\perp$. In other words, it's the event it drew some task j more than $\lceil 2\ln(ct) \rceil$ times in step 1. By Lemma B.5 we know that $\mathbf{Pr}_{S_1,\ldots,S_t,\mathcal{A}_{\text{multi}}}[\mathcal{B}] \leq \frac{1}{c}$.

By our assumption, if $\mathcal{B}$ does not hold, then $\mathcal{A}_{\text{meta}}$ returns specialization algorithm $\mathcal{A}_{\text{spec}}$ with the guarantee that with probability at least $1-\varepsilon$ over the randomness of the samples and the algorithms

$$\mathop{\mathbb{E}}_{P,S}\left[\text{err}(P, (\mathcal{A}_{\text{meta}}(S'_1, \ldots, S'_t))(S))\right] \leq \min_{h \in \mathcal{H}} \mathop{\mathbb{E}}_{P}\left[\min_{f \in \mathcal{F}} \text{err}(P, f \circ h)\right] + \varepsilon,$$

where P is P_j for j drawn uniformly at random from $[t]$.

Thus,

$$\mathop{\mathbb{E}}_{i_1,\ldots,i_t,S'_1,\ldots,S'_t,\mathcal{A}_{\text{meta}},\mathcal{A}_{\text{spec}}}\left[\mathop{\mathbb{E}}_{j,S}\left[\text{err}(P_j, (\mathcal{A}_{\text{meta}}(S'_1, \ldots, S'_t))(S)) \middle| \text{ not } \mathcal{B}\right]\right] \leq \min_{h \in \mathcal{H}} \mathop{\mathbb{E}}_{j}\left[\min_{f \in \mathcal{F}} \text{err}(P_j, f \circ h)\right] + 2\varepsilon.$$

Conditioning on $\mathcal{B}$ not happening, we see that

$$\mathop{\mathbb{E}}_{\substack{S_1,\ldots,S_t,\\ \mathcal{A}_{\text{multi}}}}\left[\frac{1}{t}\sum_{j \in [t]} \text{err}(P_j, g_j) \middle| \text{ not } \mathcal{B}\right] =$$

$$\mathop{\mathbb{E}}_{\substack{S_1,\ldots,S_t,\\ i_1,\ldots,i_t,\\ \mathcal{A}_{\text{meta}},\mathcal{A}_{\text{spec}}}}\left[\frac{1}{t}\sum_{j \in [t]} \text{err}(P_j, \mathcal{A}_{\text{meta}}(S'_1, \ldots, S'_t)(S_j^{\text{spec}}) \middle| \text{ not } \mathcal{B}\right] =$$

$$\mathop{\mathbb{E}}_{\substack{S_1 \setminus S_1^{\text{spec}},\ldots,S_t \setminus S_t^{\text{spec}},\\ i_1,\ldots,i_t,\\ \mathcal{A}_{\text{meta}},\mathcal{A}_{\text{spec}}}}\left[\frac{1}{t}\sum_{j \in [t]} \mathop{\mathbb{E}}_{S_j^{\text{spec}}}\left[\text{err}(P_j, \mathcal{A}_{\text{meta}}(S'_1, \ldots, S'_t)(S_j^{\text{spec}})\right] \middle| \text{ not } \mathcal{B}\right] =$$

$$\mathop{\mathbb{E}}_{\substack{S_1 \setminus S_1^{\text{spec}},\ldots,S_t \setminus S_t^{\text{spec}},\\ i_1,\ldots,i_t,\\ \mathcal{A}_{\text{meta}},\mathcal{A}_{\text{spec}}}}\left[\mathop{\mathbb{E}}_{j,S}\left[\text{err}(P_j, \mathcal{A}_{\text{meta}}(S'_1, \ldots, S'_t)(S)\right] \middle| \text{ not } \mathcal{B}\right] =$$

$$\mathop{\mathbb{E}}_{\substack{i_1,\ldots,i_t,\\ S'_1,\ldots,S'_t,\\ \mathcal{A}_{\text{meta}},\mathcal{A}_{\text{spec}}}}\left[\mathop{\mathbb{E}}_{j,S}\left[\text{err}(P_j, \mathcal{A}_{\text{meta}}(S'_1, \ldots, S'_t)(S)\right] \middle| \text{ not } \mathcal{B}\right] \leq$$

$$\min_{h \in \mathcal{H}} \mathop{\mathbb{E}}_{j}\left[\min_{f \in \mathcal{F}} \text{err}(P_J, f \circ h)\right] + 2\varepsilon =$$

$$\min_{h \in \mathcal{H}, f_1, \ldots, f_t \in \mathcal{F}} \frac{1}{t}\sum_{j \in [t]} \text{err}(P_j, f_j \circ h) + 2\varepsilon.$$

By Markov's inequality we can bound the corresponding probability

$$\mathbf{Pr}_{S_1,\ldots,S_t,\mathcal{A}_{\text{multi}}}\left[\frac{1}{t}\sum_{j\in[t]}\text{err}(P_j,g_j)-\min_{h\in\mathcal{H},f_1,\ldots,f_t\in\mathcal{F}}\frac{1}{t}\sum_{j\in[t]}\text{err}(P_j,f\circ h)>c\varepsilon\middle|\text{ not }\mathcal{B}\right]\leq$$
$$\frac{1}{c\varepsilon}\cdot\mathop{\mathbb{E}}_{S_1,\ldots,S_t,\mathcal{A}_{\text{multi}}}\left[\frac{1}{t}\sum_{j\in[t]}\text{err}(P_j,g_j)-\min_{h\in\mathcal{H},f_1,\ldots,f_t\in\mathcal{F}}\frac{1}{t}\sum_{j\in[t]}\text{err}(P_j,f\circ h)>c\varepsilon\middle|\text{not }\mathcal{B}\right]\leq\frac{2}{c}.$$

Combining the results above we can show that for all $P_1,\ldots,P_t$

$$\mathbf{Pr}_{S_1,\ldots,S_t,\mathcal{A}_{\text{multi}}}\left[\frac{1}{t}\sum_{j\in[t]}\text{err}(P_j,g_j)>\min_{h\in\mathcal{H},f_1,\ldots,f_t\in\mathcal{F}}\frac{1}{t}\sum_{j\in[t]}\text{err}(P_j,f\circ h)+c\varepsilon\right]\leq$$
$$\mathbf{Pr}_{S_1,\ldots,S_t,\mathcal{A}_{\text{multi}}}\left[\frac{1}{t}\sum_{j\in[t]}\text{err}(P_j,g_j)-\min_{h\in\mathcal{H},f_1,\ldots,f_t\in\mathcal{F}}\frac{1}{t}\sum_{j\in[t]}\text{err}(P_j,f\circ h)>c\varepsilon\middle|\text{ not }\mathcal{B}\right]$$
$$+\,\mathbf{Pr}_{S_1,\ldots,S_t,\mathcal{A}_{\text{multi}}}[\mathcal{B}]\leq\frac{3}{c}.$$

This concludes the proof. ■

Appendix C. Multitask Learning

In this section, we bound the number of tasks and samples per task we need to multitask learn using the VC dimension. Moreover, we provide upper and lower bounds of this VC dimension for general classes of representations and specialized classifiers.

C.1. Sample and Task Complexity Bounds for General Classes

Here we prove that the VC dimension of the composite class $\mathcal{F}^{\otimes t}\circ\mathcal{H}$ determines the total number of samples required for multitask learning.

Theorem C.1 *For any $(\mathcal{H},\mathcal{F})$, any $\varepsilon,\delta>0$, and any t,n,*

1. *In the realizable case, $(\mathcal{H},\mathcal{F})$ is (ε,δ)-multitask learnable with t tasks and n samples per task when $nt=O(\frac{\text{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})\cdot\ln(1/\varepsilon)+\ln(1/\delta)}{\varepsilon})$.*
2. *In the agnostic case, $(\mathcal{H},\mathcal{F})$ is (ε,δ)-multitask learnable with t tasks and n samples per task when $nt=O(\frac{\text{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})\cdot\ln(1/\delta)}{\varepsilon^2})$.*
3. *If $nt\leq\frac{1}{4}\cdot\text{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})$ then $(\mathcal{H},\mathcal{F})$ is* not *$(1/8,1/8)$-multitask learnable with t tasks and n samples per task, even in the realizable case.*

The proof of this theorem closely follows the standard proofs on VC dimension which upper and lower bound on the sample complexity of distribution-free classification. The main difference is

that the samples we analyze, while independent, are not identically distributed. This is because each sample comes from a specific task distribution and these distributions are potentially different.

Proof We'll start by proving part 1, which covers the realizable case. We will omit the proof of part 2, which generalizes the realizable case in a standard way. Finally, we will prove part 3.

Proof of 1. For every task $j \in [t]$ we have n i.i.d. samples $S_j = \{(x_i^{(j)}, y_i^{(j)})\}_{i\in[n]}$ drawn from P_j. Our dataset is equivalent to dataset $S = \{(j, x_i^{(j)}, y_i^{(j)})\}_{i\in[n],j\in[t]}$, where the js have fixed values. In standard PAC learning we assume that all the datapoints are i.i.d. However, in our case the samples are independent but not identically distributed because different tasks have (potentially) different distributions.

As in standard PAC learning, our proof follows the "double-sampling trick". We want to bound the probability of bad event

$$B : \exists g \in \mathcal{F}^{\otimes} \circ \mathcal{H} \text{ s.t. } \mathrm{err}(S, g) = 0, \text{ but } \frac{1}{t}\sum_{j\in[t]} \mathrm{err}(P_j, g(j, \cdot)) > \varepsilon.$$

We consider an auxiliary dataset $\hat{S} = \{(j, \hat{x}_i^{(j)}, \hat{y}_i^{(j)})\}_{i\in[n],j\in[t]}$, where $(\hat{x}_i^{(j)}, \hat{y}_i^{(j)})$ are drawn independently from P_j. In our proof we will bound the probability of B by bounding the probability of event

$$B' : \exists g \in \mathcal{F}^{\otimes} \circ \mathcal{H} \text{ s.t. } \mathrm{err}(S, g) = 0, \text{ and } \mathrm{err}(\hat{S}, g) > \frac{\varepsilon}{2}.$$

We can show that if $nt > \frac{8}{\varepsilon}$, then $\mathbf{Pr}_S[B] \leq 2\mathbf{Pr}_{S,\hat{S}}[B']$. We can show this by applying a multiplicative Chernoff bound on $\mathrm{err}(\hat{S}, g)$, which is an average of independent random variables. Due to this step, it suffices to bound $\mathbf{Pr}_{S,\hat{S}}[B']$.

We also define a third event B'' as follows. We give S and $\hat{S}$ as inputs to randomized process *Swap* which iterates over $j \in [t]$ and $i \in [n]$ and at every step it swaps $(x_i^{(j)}, y_i^{(j)})$ with $(\hat{x}_i^{(j)}, \hat{y}_i^{(j)})$ with probability $1/2$. Let T and $\hat{T}$ be the two datasets this process outputs. We define event

$$B'' : \exists g \in \mathcal{F}^{\otimes t} \circ \mathcal{H} \text{ s.t. } \mathrm{err}(T, g) = 0 \text{ and } \mathrm{err}(\hat{T}, g) > \frac{\varepsilon}{2}.$$

We see that $\mathbf{Pr}_{S,\hat{S},\mathit{Swap}}[B''] = \mathbf{Pr}_{S,\hat{S}}[B']$. This happens because T, $\hat{T}$, S and $\hat{S}$ are identically distributed. Thus, what we need to do now is bound $\mathbf{Pr}_{S,\hat{S},\mathit{Swap}}[B'']$.

We start by showing that for a fixed g

$$\mathbf{Pr}_{\mathit{Swap}}\Big[\mathrm{err}(T, g) = 0 \text{ and } \mathrm{err}(\hat{T}, g) > \frac{\varepsilon}{2} \mid S, \hat{S}\Big] \leq 2^{-nt\varepsilon/2}.$$

Given S and $\hat{S}$, B'' happens if for every $j \in [t]$ and $i \in [n]$ g predicts the label of $x_i^{(j)}$ or $\hat{x}_i^{(j)}$ correctly and makes $m > \varepsilon nt/2$ mistakes overall. Additionally, all m mistakes g makes are in dataset $\hat{T}$. This means that *Swap* assigns all these points to $\hat{T}$, which happens with probability $1/2^m \leq 1/2^{\varepsilon nt/2}$.

Let $\big(\mathcal{F}^{\otimes t} \circ \mathcal{H}\big)(S \cup \hat{S}) \subset \mathcal{F}^{\otimes t} \circ \mathcal{H}$ be a set of hypotheses which contains one hypothesis for every labeling of $S \cup \hat{S}$. Then,

$$\begin{aligned}\mathbf{Pr}_{S,\hat{S},\mathit{Swap}}\big[B''\big] &= \mathop{\mathbb{E}}_{S,\hat{S}}\Big[\mathbf{Pr}_{\mathit{Swap}}\Big[\exists g \in \mathcal{F}^{\otimes t}\circ\mathcal{H} \text{ s.t. } \mathrm{err}(T,g)=0 \text{ and } \mathrm{err}(\hat{T},g) > \frac{\varepsilon}{2} \mid S,\hat{S}\Big]\Big] \\ &\le \mathop{\mathbb{E}}_{S,\hat{S}}\left[\sum_{g\in\mathcal{F}^{\otimes t}\circ\mathcal{H}(S\cup\hat{S})} \mathbf{Pr}_{\mathit{Swap}}\Big[\mathrm{err}(T,g)=0 \text{ and } \mathrm{err}(\hat{T},g) > \frac{\varepsilon}{2} \mid S,\hat{S}\Big]\right] \\ &\le \mathcal{G}_{\mathcal{F}^{\otimes t}\circ\mathcal{H}}(2nt)2^{-\varepsilon nt/2}.\end{aligned}$$

For the probability of bad event B happening to be at most δ, by the steps above we need $nt \ge 2\frac{\log_2 \mathcal{G}_{\mathcal{F}^{\otimes t}\circ\mathcal{H}}(2nt)+\log_2(2/\delta)}{\varepsilon}$ samples in total.

By Lemma A.3, for $nt > \mathrm{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})$,

$$\log_2\left(\mathcal{G}_{\mathcal{F}^{\otimes t}\circ\mathcal{H}}(2nt)\right) \le \mathrm{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})\log_2\left(\frac{e2nt}{\mathrm{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})}\right).$$

One can show that if $nt \ge \frac{\mathrm{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})}{\varepsilon}\log_2\left(\frac{2e}{\varepsilon}\right)$, then $nt \ge \log_2\left(\frac{e2nt}{\mathrm{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})}\right)\frac{\mathrm{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})}{\varepsilon}$. Therefore, we get that for $nt \ge \frac{\mathrm{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})\log_2(2e/\varepsilon)+\log_2(2/\delta)}{\varepsilon}$ samples in total the probability of bad event B is at most δ.

Proof of 3. Our goal is to construct distributions $P_1,\dots,P_t$ over $\mathcal{X}\times\{\pm1\}$. We will first build their support and then define the probability distributions.

Let $v = \mathrm{VC}(\mathcal{F}^{\otimes t}\circ\mathcal{H})$. Since $4nt \le v$, there exists a dataset $S = \{(j_i,x_i)\}_{i\in[4nt]}$ that can be shattered by $\mathcal{F}^{\otimes t}\circ\mathcal{H}$. Let $S_j = \{(j_i,x_i)\in S \mid j_i = j\}$. In general for every task j the size of S_j, i.e. $|S_j|$, will be different. We want to use S to get a new dataset S' that can be shattered by $\mathcal{F}^{\otimes t}\circ\mathcal{H}$, but also has at least $2n$ points for every task. For every $j\in[t]$ we throw away points from S_j until we have a multiple of $2n$. After doing this, we have thrown away at most $2n$ per task, which is at most $2nt$ points in total. We can redistribute points so that we have at least $2n$ points per task as follows. For every task $j\in[t]$, if there are no points in this task, there must be another task k with at least $4nt$ points. In this case, we move nt points from task k to task j by replacing k with j in (k,x_i). We denote this new dataset by S' and the subset for task j, by S'_j.

We claim that S' , which has at least $2n$ points per task, can also be shattered by $\mathcal{F}^{\otimes t}\circ\mathcal{H}$. To see this, suppose that S was shattered by $\tilde{g}$. Now, if task j did not lose all its points, the remaining points can be shattered by $\tilde{g}(j,\cdot)$. Otherwise, we are in the case where j's points were initially assigned to task k, so they can be shattered by $\tilde{g}(k,\cdot)$.

Let $\big(\mathcal{F}^{\otimes t}\circ\mathcal{H}\big)(S')$ be a set of hypotheses that contains one function for each labeling of S'. We choose a labeling function g uniformly at random from $\big(\mathcal{F}^{\otimes t}\circ\mathcal{H}\big)(S')$. For all tasks $j\in[t]$ we define P_j as the distribution of (x,y) obtained by sampling x uniformly from S'_j (ignoring the task index in the sample) and labeling it according to g.

Suppose that the (potentially randomized) learning algorithm $\mathcal{A}$ returns $\hat{g}$ after seeing datasets $\hat{S}_1,\dots,\hat{S}_t$, where every $\hat{S}_j$ has n points drawn i.i.d. from P_j. For a fixed task j, the probability that $\hat{g}$ makes a mistake on a new point (x,y) drawn from P_j is

$$\begin{aligned}\mathbf{Pr}_{g,\hat{S}_j,(x,y)\sim P_j}\big[\hat{g}(j,x)\ne y\big] &\ge \mathbf{Pr}_{g,\hat{S}_j,(x,y)\sim P_j}\Big[\hat{g}(j,x)\ne y \text{ and } x\notin\hat{S}_j\Big] \\ &= \mathbf{Pr}_{g,\hat{S}_j,(x,y)\sim P_j}\Big[x\notin\hat{S}\Big]\mathbf{Pr}_{g,\hat{S},(x,y)\sim P_j}\Big[\hat{g}(j,x)\ne y \mid x\notin\hat{S}_j\Big].\end{aligned}$$

We know that $\mathbf{Pr}_{\hat{S}_j,(x,y)\sim P_j}\left[x \notin \hat{S}\right] \geq 1/2$ because x is chosen uniformly at random out of more than $2n$ points that are in S'_j and $\hat{S}_j$ has only n points. When $x \notin \hat{S}_j$ the algorithm has not seen the label that corresponds to this x and, thus, $y = g(j,x)$ is independent of $\hat{g}(j,x)$. We have picked g uniformly at random from a class with exactly one function per labeling, which means that for each (j,x) we see $+1$ and -1 with equal probability. Therefore, $\mathbf{Pr}_{g,\hat{S}_j,(x,y)\sim P_j}\left[\hat{g}(j,x) \neq y \mid x \notin \hat{S}_j\right] = 1/2$. Thus, $\mathbf{Pr}_{g,\hat{S}_j,(x,y)\sim P_j}[\hat{g}(j,x) \neq y] \geq 1/4$.

The average error is

$$\frac{1}{t}\sum_{j\in[t]} \mathbf{Pr}_{g,\hat{S}_j,(x,y)\sim P_j}[\hat{g}(j,x) \neq y] \geq \frac{1}{4}.$$

In expectation over the labeling functions g and the randomness of algorithm $\mathcal{A}$, we have

$$\mathop{\mathbb{E}}_{g,\mathcal{A}}\left[\frac{1}{t}\sum_{j\in[t]} \mathbf{Pr}_{\hat{S}_j,(x,y)\sim P_j}[\hat{g}(j,x) \neq y]\right] \geq \frac{1}{4}$$

Hence, there exists a labeling function g in $\mathcal{F}^{\otimes t} \circ \mathcal{H}(S')$ such that

$$\mathop{\mathbb{E}}_{\mathcal{A}}\left[\frac{1}{t}\sum_{j\in[t]} \mathbf{Pr}_{\hat{S}_j,(x,y)\sim P_j}[\hat{g}(j,x) \neq y]\right] \geq \frac{1}{4}.$$

For this g we have that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathcal{A}}\left[\frac{1}{t}\sum_{j\in[t]} \mathbf{Pr}_{\hat{S}_j,(x,y)\sim P_j}[\hat{g}(j,x) \neq y]\right] \\
&= \mathop{\mathbb{E}}_{\mathcal{A}}\left[\frac{1}{t}\sum_{j\in[t]} \mathop{\mathbb{E}}_{\hat{S}_j}[\mathrm{err}(P_j,\hat{g}(j,\cdot))]\right] \\
&= \mathop{\mathbb{E}}_{\mathcal{A},\hat{S}_1,\ldots,\hat{S}_t}\left[\frac{1}{t}\sum_{j\in[t]} \mathrm{err}(P_j,\hat{g}(j,\cdot))\right] \\
&= \mathbf{Pr}_{\mathcal{A},\hat{S}_1,\ldots,\hat{S}_t}\left[\frac{1}{t}\sum_{j\in[t]} \mathrm{err}(P_j,\hat{g}(j,\cdot)) > \frac{1}{8}\right] \mathop{\mathbb{E}}_{\mathcal{A},\hat{S}_1,\ldots,\hat{S}_t}\left[\frac{1}{t}\sum_{j\in[t]} \mathrm{err}(P_j,\hat{g}(j,\cdot)) \middle| \sum_{j\in[t]} \mathrm{err}(P_j,\hat{g}(j,\cdot)) > \frac{1}{8}\right] \\
&+ \mathbf{Pr}_{\mathcal{A},\hat{S}_1,\ldots,\hat{S}_t}\left[\frac{1}{t}\sum_{j\in[t]} \mathrm{err}(P_j,\hat{g}(j,\cdot)) < \frac{1}{8}\right] \mathop{\mathbb{E}}_{\mathcal{A},\hat{S}_1,\ldots,\hat{S}_t}\left[\frac{1}{t}\sum_{j\in[t]} \mathrm{err}(P_j,\hat{g}(j,\cdot)) \middle| \sum_{j\in[t]} \mathrm{err}(P_j,\hat{g}(j,\cdot)) < \frac{1}{8}\right] \\
&\leq \mathbf{Pr}_{\mathcal{A},\hat{S}_1,\ldots,\hat{S}_t}\left[\frac{1}{t}\sum_{j\in[t]} \mathrm{err}(P_j,\hat{g}(j,\cdot)) > \frac{1}{8}\right] + \frac{1}{8}
\end{aligned}$$

Thus, we showed that there exist $P_1,\ldots,P_t$ such that $\mathbf{Pr}_{\mathcal{A},\hat{S}_1,\ldots,\hat{S}_t}\left[\frac{1}{t}\sum_{j\in[t]} \mathrm{err}(P_j,\hat{g}(j,\cdot)) > \frac{1}{8}\right] \geq \frac{1}{8}$. ■

C.2. Generic Bounds on $\mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H})$

Ideally, given $\mathcal{H}$ and $\mathcal{F}$ we would like to be able to determine the VC dimension of the composite class $\mathcal{F}^{\otimes t} \circ \mathcal{H}$. For general classes, we can state a few simple upper and lower bounds. For one lower bound, consider the *easier* problem where someone gives us the representation h and we only need to find the specialized classifiers $f_1, \ldots, f_t$ for each task. Even in this simpler setting we need $\mathrm{VC}(\mathcal{F})$ samples per task, and thus the total sample complexity is $t\mathrm{VC}(\mathcal{F})$. For another lower bound, consider the easier problem where all specialized classifiers are the same, so the data is labeled by a single concept $f \circ h \in \mathcal{F} \circ \mathcal{H}$, which requires total sample complexity $\mathrm{VC}(\mathcal{F} \circ \mathcal{H})$.

For an upper bound, there is a naïve strategy for multitask learning where we treat the data for each task in isolation and simply try to learn a classifier from $\mathcal{F} \circ \mathcal{H}$. This requires sample complexity $\mathrm{VC}(\mathcal{F} \circ \mathcal{H})$ per task.

Lemma C.2 formalizes these results.

Lemma C.2 *For any representation class $\mathcal{H}$ that contains a surjective function, any class of specialized classifiers $\mathcal{F}$, and any t,*

$$\max\{t \cdot \mathrm{VC}(\mathcal{F}), \mathrm{VC}(\mathcal{F} \circ \mathcal{H})\} \leq \mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H}) \leq t \cdot \mathrm{VC}(\mathcal{F} \circ \mathcal{H}).$$

The upper bound still holds even if $\mathcal{H}$ does not contain a surjective function.

Proof We will show the two parts of the statement separately.

Proof of the upper bound on VC dimension. Assume that we have a dataset $X = ((j_1, x_1), \ldots, (j_n, x_n))$ of size $n = \mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H})$ which can be shattered by $\mathcal{F}^{\otimes t} \circ \mathcal{H}$. We split it into t disjoint datasets $X_1, \ldots, X_t$, where $X_j = \{x_i : (j, x_i) \in X\}$. Each one of these datasets can be shattered by $\mathcal{F} \circ \mathcal{H}$.

Let n_j be the size of dataset X_j. Then, we have that $n_j \leq \mathrm{VC}(\mathcal{F} \circ \mathcal{H})$. As a result, we obtain $\mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H}) = \sum_{j \in [t]} n_j \leq t\mathrm{VC}(\mathcal{F} \circ \mathcal{H})$.

Proof of the lower bound on VC dimension. We will first show that $\mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H}) \geq \mathrm{VC}(\mathcal{F} \circ \mathcal{H})$. Suppose $X = (x_1, \ldots, x_n)$ is a dataset of size n that can be shattered by class $\mathcal{F} \circ \mathcal{H}$. Then, for any $j_1, \ldots, j_n \in [t]$, the dataset $((j_1, x_1), \ldots, (j_n, x_n))$ can be shattered by $\mathcal{F}^{\otimes t} \circ \mathcal{H}$. To see this, fix a labeling $(y_1, \ldots, y_n)$ of $((j_1, x_1), \ldots, (j_n, x_n))$. Since we assumed X could be shattered by $\mathcal{F} \circ \mathcal{H}$, there exists $f^* \in \mathcal{F}$ and $h^* \in \mathcal{H}$ such that $\forall i \in [n]$, $y_i = f^*(h^*(x_i))$. Thus, there is a function in $\mathcal{F}^{\otimes t} \circ \mathcal{H}$ (namely, the one with representation h^* and all t personalization functions equal to f^*) that realizes this labeling. Therefore, $\mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H}) \geq \mathrm{VC}(\mathcal{F} \circ \mathcal{H})$.

Next, we will prove that $\mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H}) \geq t\mathrm{VC}(\mathcal{F})$. Let $(z_1, \ldots, z_n) \in \mathcal{Z}^n$ be a dataset that $\mathcal{F}$ can shatter. Since there exists an $h \in \mathcal{H}$ whose image is $\mathcal{Z}$, there exist $(x_1, \ldots, x_n) \in \mathcal{X}^n$ such that $\forall i \in [n]$ $h(x_i) = z_i$. We now construct a new dataset $\cup_{j \in [t]}\{(j, x_1) \ldots, (j, x_n)\}$, which has nt datapoints. Our function class $\mathcal{F}^{\otimes t} \circ \mathcal{H}$ can shatter this dataset. To see this: for any labeling we split the dataset to t parts according to the value of j, use h to get $(z_1, \ldots, z_n)$ for each part. We then label each part using an $f \in \mathcal{F}$. This means that there exists a dataset of size $t\mathrm{VC}(\mathcal{F})$ that $\mathcal{F}^{\otimes t} \circ \mathcal{H}$ can shatter. Thus, $\mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H}) \geq t\mathrm{VC}(\mathcal{F})$. ■

See Appendix F for more precise bounds that hold for finite classes.

Appendix D. Metalearning

In this section, we illustrate the techniques used to achieve the results in Section 3 via the special case of monotone thresholds applied to 1-dimensional representations (Appendix D.1). This special case corresponds to a natural setting where the representation assigns a real-valued score to each example, but the threshold for converting that score into a binary label may vary from task to task. We then restate the main theorems and provide their complete proofs for the realizable (Appendix D.2) and the agnostic cases (Appendix D.3), respectively. Finally, we include a sample and task complexity bound for general metalearning (Appendix D.4).

D.1. Warm Up and Techniques Overview

In this subsection, we describe the technical tools we use to bound the number of samples and tasks needed to find a good representation $\hat{h}$. As a warm-up, we first prove Theorem D.4, which covers metalearning for the class of monotone thresholds over linear representations in realizable case with minimal sample complexity. For this example, we prove a stronger result than what is implied by our general bounds. This example considers the class of linear representations that map from d dimensions to 1 dimension, $\mathcal{H}_{d,1} = \left\{h \mid h(\mathbf{x}) = \mathbf{b} \cdot \mathbf{x}, \mathbf{b} \in \mathbb{R}^d\right\}$, and the class of specialized monotone thresholds $\mathcal{F}_{\text{mon}} = \{f \mid f(z) = \text{sign}(z - w), w \in \mathbb{R}\}$. Then, we sketch the techniques for metalearning with more samples per task. Finally, we discuss extending our techniques to the agnostic case.

Metalearning in the realizable case. A data set $S \in (\mathcal{Z} \times \{\pm 1\})^*$ is *realizable* by a concept class $\mathcal{F}$ if there is an $f \in \mathcal{F}$ such that $y_i = f(z_i)$ for every $(z_i, y_i) \in S$. A distribution B over $\mathcal{Z} \times \{\pm 1\}$ is realizable by $\mathcal{F}$ if there exists $f \in \mathcal{F}$ such that $y = f(z)$ with probability 1 over $(z, y) \sim B$. In our metalearning model, realizability has two "layers," one for the representation and one for the specialized classifiers. Let $\mathcal{P}$ be a family of data distributions P over $\mathcal{X} \times \{\pm 1\}$. We say that $\mathcal{P}$ is *meta-realizable* by $\mathcal{F} \circ \mathcal{H}$ if there exists a shared representation function h^* for which, for all $P \in \mathcal{P}$, the test error of h^*, rep-err$(P, h^*, \mathcal{F})$, is zero. In other words, there exists an h^* such that for all $P \in \mathcal{P}$ there exists a specialized classifiers $f \in \mathcal{F}$ such that err$(P, f \circ h^*) = 0$. We say a meta distribution is *meta-realizable* if its support is meta-realizable.

By these definitions we see that in the realizable case there exists a representation h^* in $\mathcal{H}$ such that all tasks drawn from distribution $\mathfrak{Q}$ are realizable by $f \circ h^*$ for an f in $\mathcal{F}$. Therefore, during the training process we want to find a representation that allows for perfect classification of all the points of all the seen tasks. Our technical approach to achieve this while getting generalization for unseen tasks depends on the number of samples per task we have.

The key component of our analysis is a class of binary functions that we call *realizability predicates* (recall Definition 1.6). Given a representation h, a dataset of n points and a set of specialized classifiers $\mathcal{F}$, the realizability predicate returns $+1$ if and only if the mapping of the dataset using h is realizable by a function in $\mathcal{F}$.

For instance, in class $\mathcal{R}_{2,\mathcal{F}_{\text{mon}},\mathcal{H}}$ the realizability predicate r_h takes two labeled datapoints $(\mathbf{x}_1, y_1)$ and $(\mathbf{x}_2, y_2)$ and returns $+1$ if h orders them correctly. More formally, given an $h \in \mathcal{H}$

$$r_h(\mathbf{x}_1, y_1, \mathbf{x}_2, y_2) = \begin{cases} -1, & y_1 = +1, y_2 = -1 \text{ and } h(\mathbf{x}_1) < h(\mathbf{x}_2) \\ -1, & y_1 = -1, y_2 = +1 \text{ and } h(\mathbf{x}_1) > h(\mathbf{x}_2) \\ +1, & \text{otherwise.} \end{cases}$$

We can think of r_h as a classifier on "data points" of the form $(\mathbf{x}_1, y_1, \mathbf{x}_2, y_2)$, indicating whether h is a good representation or not.

Since $\mathfrak{Q}$ is meta-realizable by $\mathcal{F} \circ \mathcal{H}$, there exists a representation $h^* \in \mathcal{H}$ for which all tasks have a specialized classifier that perfectly classifies all samples. This implies that the corresponding realizability predicate $r_{h^*} \in \mathcal{R}_{n,\mathcal{F},\mathcal{H}}$ outputs $+1$ for any n samples drawn from a distribution $P \sim \mathfrak{Q}$.

At a high level, we exploit the above fact about r_{h^*} and class $\mathcal{R}_{n,\mathcal{F},\mathcal{H}}$. We consider a new data distribution $\mathcal{D}$ that generates samples of the form $\zeta_j := ((x_1^{(j)}, y_1^{(j)}, \ldots, x_n^{(j)}, y_n^{(j)}), +1)$. The feature part of the sample is a task training set of size n generated by a data distribution $P_j \sim \mathfrak{Q}$: $\{(x_i^{(j)}, y_i^{(j)})\}_{i\in[n]}$. The label part of the sample is always $+1$. See Figure 1. Note that the realizability predicate r_{h^*} also labels the task training sets of size n with $+1$:

$$r_{h^*}((x_1^{(j)}, y_1^{(j)}), \ldots, (x_n^{(j)}, y_n^{(j)})) = +1.$$

Hence, we can say that the new data distribution $\mathcal{D}$ is realizable by the hypothesis class $\mathcal{R}_{n,\mathcal{F},\mathcal{H}}$. Now, if we have t samples from $\mathcal{D}$ for a sufficiently large t (that is t tasks $P_1, \ldots, P_t$ drawn from $\mathfrak{Q}$ and n samples from each), by the fundamental theorem of PAC learning, Theorem A.4, we can PAC learn the class $\mathcal{R}_{n,\mathcal{F},\mathcal{H}}$ with respect to $\mathcal{D}$. More precisely, we can show that any $r_{\hat{h}} \in \mathcal{R}_{n,\mathcal{F},\mathcal{H}}$ that labels samples ζ_j's correctly has low mislabeling error under $\mathcal{D}$. Recall that all the labels according to $\mathcal{D}$ were always $+1$. Thus, mislabeling a ζ_j implies that the realizability predicate $r_{\hat{h}}$ of the training set corresponding to ζ_j is false. Thus, the representation function we have found, which has a low error probability for mislabeling a ζ, allows the realizability of the dataset corresponding to ζ with high probability. Hence, $\hat{h}$ is an accurate shared representation function which we need to metalearn $\mathfrak{Q}$ by $\mathcal{F} \circ \mathcal{H}$.

Going back to our example of monotone thresholds over linear representations, suppose that for every task $j \in [t]$ we see two points $(\mathbf{x}_1^{(j)}, y_1^{(j)})$ and $(\mathbf{x}_2^{(j)}, y_2^{(j)})$ drawn from P_j, which itself is drawn from $\mathfrak{Q}$. Since we are in the realizable case, there is a representation $h^* \in \mathcal{H}_{d,1}$ that always orders points from the same task correctly on the real line. Therefore, for every task $j \in [t]$ we have $r_{h^*}(\mathbf{x}_1^{(j)}, y_1^{(j)}, \mathbf{x}_2^{(j)}, y_2^{(j)}) = +1$. Intuitively we want to learn an $\hat{h} \in \mathcal{H}_{d,1}$ which has low error:

$$\mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\left[\mathbf{Pr}_{(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2)\sim P^2}\left[r_{\hat{h}}(\mathbf{x}_1, y_1, \mathbf{x}_2, y_2) \neq +1\right]\right] = \mathbf{Pr}_{\zeta\sim\mathcal{D}}\left[r_{\hat{h}}(\zeta_{\text{feat}}) \neq +1\right].$$

In other words, we want to find a representation that will allow two points of a new task drawn from $\mathfrak{Q}$ to be classified correctly using a monotone threshold in $\mathcal{F}_{\text{mon}}$. By Theorem A.4, if we have $t = O\left(\frac{\text{VC}(\mathcal{R}_{2,\mathcal{F}_{\text{mon}},\mathcal{H}_{d,1}})\ln(1/\varepsilon)+\ln(1/\delta)}{\varepsilon^2}\right)$ tasks, each with a datapoint $\zeta_j = ((\mathbf{x}_1^{(j)}, y_1^{(j)}, \mathbf{x}_2^{(j)}, y_2^{(j)}), +1)$ drawn from $\mathcal{D}$, then by choosing $\hat{h} \in \mathcal{H}_{d,1}$ which minimizes $\frac{1}{t}\sum_{j\in[t]} \mathbb{I}\{r_h(\zeta_{j,\text{feat}}) \neq +1\}$ we get that with probability at least $1-\delta$ over the dataset

$$\mathbf{Pr}_{\zeta\sim\mathcal{D}}\left[r_{\hat{h}}(\zeta_{\text{feat}}) \neq +1\right] = \mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\left[\mathbf{Pr}_{(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2)\sim P^2}\left[r_{\hat{h}}(\mathbf{x}_1, y_1, \mathbf{x}_2, y_2) \neq +1\right]\right] < \varepsilon^2.$$

In Lemma D.1 we show that the VC dimension of $\mathcal{R}_{2,\mathcal{F}_{\text{mon}},\mathcal{H}_{d,1}}$ is at most d. Thus, we can learn with $t = O\left(\frac{d\ln(1/\varepsilon)+\ln(1/\delta)}{\varepsilon^2}\right)$ tasks.

Lemma D.1 *We have* $\text{VC}(\mathcal{R}_{2,\mathcal{F}_{mon},\mathcal{H}_{d,1}}) \leq d$.

Proof Fix a set of inputs to r_h: $((\mathbf{x}_1^{(1)}, y_1^{(1)}, \mathbf{x}_2^{(1)}, y_2^{(1)}), \ldots, (\mathbf{x}_1^{(n)}, y_1^{(n)}, \mathbf{x}_2^{(n)}, y_2^{(n)}))$. Assume for all i we have $y_1^{(i)} \neq y_2^{(i)}$, since otherwise we have $r_h(\mathbf{x}_1^{(i)}, y_1^{(i)}, \mathbf{x}_2^{(i)}, y_2^{(i)}) = +1$ for all h, which means the overall set cannot be shattered.

Because the thresholds are monotone, if $y_1^{(i)} > y_2^{(i)}$ then $r_h(\mathbf{x}_1^{(i)}, y_1^{(i)}, \mathbf{x}_2^{(i)}, y_2^{(i)}) = +1$ when h places $h(\mathbf{x}_1^{(i)})$ above $h(\mathbf{x}_2^{(i)})$. Equivalently, associating h with $\mathbf{b} \in \mathbb{R}^d$, this holds when $\operatorname{sign}(\mathbf{b} \cdot (\mathbf{x}_1^{(i)} - \mathbf{x}_2^{(i)})) = +1$. The opposite holds for $y_1^{(i)} < y_2^{(i)}$.

We can achieve a labeling $\mathbf{s} \in \{\pm 1\}^n$ if there exists $\mathbf{b}$ such that, for all i,

$$\operatorname{sign}(\mathbf{b} \cdot (\mathbf{x}_1^{(i)} - \mathbf{x}_2^{(i)})) = \mathbf{s}_i \cdot \operatorname{sign}(y_1^{(i)} - y_2^{(i)}).$$

From this, we see that the number of achievable labelings is determined by the number of possible signs of $\mathbf{b} \cdot (\mathbf{x}_1^{(i)} - \mathbf{x}_2^{(i)})$. This is at most 2^d by Theorem A.6, which bounds the capacity of halfspaces passing through the origin. Thus, $\mathcal{R}_{2,\mathcal{F}_{\text{mon}},\mathcal{H}_{d,1}}$ cannot shatter a set of size $d+1$. ■

So far, we have described how to find a representation $\hat{h}$ that, evaluated on data sets of size n drawn from new tasks, is likely to produce something which $\mathcal{F}$ can perfectly classify, i.e., the following expressions are small:

$$\mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\Big[\mathbf{Pr}_{(x_1,y_1),\ldots,(x_n,y_n)\sim P^n}\big[r_{\hat{h}}((x_1,y_1),\ldots,(x_n,y_n)) \neq +1\big]\Big] =$$
$$\mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\Big[\mathbf{Pr}_{(x_1,y_1),\ldots,(x_n,y_n)\sim P^n}\Big[(\hat{h}(x_1),y_1),\ldots,(\hat{h}(x_n),y_n) \text{ is not realizable by } \mathcal{F}\Big]\Big].$$

However, in metalearning we want to bound the following error

$$\text{rep-err}(\mathfrak{Q},\hat{h},\mathcal{F}) = \mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\Big[\min_{f\in\mathcal{F}} \mathbf{Pr}_{(x,y)\sim P}\Big[f(\hat{h}(x)) \neq y\Big]\Big].$$

The next step is to derive a bound on this error. We consider two cases based on the number of samples per task we have.

Metalearning with $\mathrm{DH}(\mathcal{F})$ samples per task. Often, for a given concept class $\mathcal{F}$ we can see that every dataset that is not realizable by $\mathcal{F}$ has a subset of size at most m that is still not realizable. The smallest m for which this condition holds is the dual Helly number, which we denote by $\mathrm{DH}(\mathcal{F})$.

For example, the dual Helly number of the class of monotone thresholds is 2. Every dataset with more than 2 points that is not realizable by $\mathcal{F}_{\text{mon}}$ has a non-realizability witness of size 2. To see this, suppose all subsets of two points are realizable by $\mathcal{F}_{\text{mon}}$. This means that any point with a positive label is greater than a point with a negative label. In this case, there exists a monotone threshold that labels these points correctly by placing the threshold between two consecutive points in the real line with opposite labels, which leads to a contradiction. Just seeing one point of a dataset does not suffice to show non-realizability cause every one-sample dataset is realizable.

For a fixed data distribution B over $\mathcal{Z} \times \{\pm 1\}$ that is realizable by $\mathcal{F}$, we define the *probability of nonrealizability* of a dataset from B by functions in $\mathcal{F}$:

$$p_{\text{nr}}(B,\mathcal{F},m) \stackrel{\text{def}}{=} \mathbf{Pr}_{(z_1,y_1),\ldots,(z_m,y_m)\sim B^m}[(z_1,y_1),\ldots,(z_m,y_m) \text{ is not realizable by } \mathcal{F}],$$

and the *population error* of $\mathcal{F}$ on B:

$$\text{err}(B,\mathcal{F}) \stackrel{\text{def}}{=} \min_{f\in\mathcal{F}}\big\{\mathbf{Pr}_{(z,y)\sim B}[f(z) \neq y]\big\}.$$

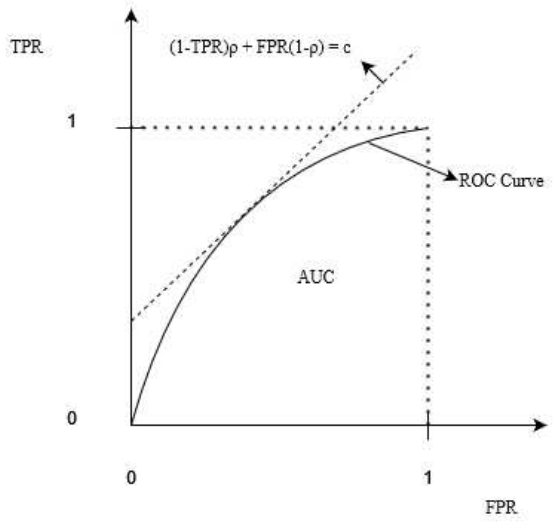

Figure 2: Example of an ROC curve and the line $(1-\text{TPR})\rho + \text{FPR}(1-\rho) = c$, which corresponds to points with error c. The line and curve intersect when $c = \text{err}(B, \mathcal{F}_{\text{mon}})$.

Notice that here distribution B is over labeled points in the intermediate space, the codomain of the representation function. Given $\text{DH}(\mathcal{F})$ points from B, we derive a bound of the form $\phi(\text{err}(B, \mathcal{F})) \leq p_{\text{nr}}(B, \mathcal{F}, \text{DH}(\mathcal{F}))$, for a strictly increasing and convex ϕ.

For the class of monotone thresholds $\mathcal{F}_{\text{mon}}$ with 2 samples drawn from B, we show in Lemma D.3 that $(\text{err}(B, \mathcal{F}_{\text{mon}}))^2 \leq p_{\text{nr}}(B, \mathcal{F}_{\text{mon}}, 2)$. The proof uses Fact D.2 and a geometric argument linking $\text{err}(B, \mathcal{F}_{\text{mon}})$ to $p_{\text{nr}}(B, \mathcal{F}_{\text{mon}}, 2)$.

For a fixed distribution P, we analyze the ROC curve of classifiers with threshold w: $\text{sign}(z - w)$. The ROC curve plots the true positive rate (TPR) against the false positive rate (FPR), which are defined as

$$\text{TPR}(w) := \mathbf{Pr}_{(z,y)\sim B}[z \geq w \mid y = +1]$$
$$\text{FPR}(w) := \mathbf{Pr}_{(z,y)\sim B}[z \geq w \mid y = -1].$$

We need the following standard fact about ROC curves.

Fact D.2 *For any distribution B, the area under the ROC curve is equal to*

$$\mathbf{Pr}[z_1 > z_2 \mid y_1 = +1, y_2 = -1].$$

Lemma D.3 *Fix a distribution B over $\mathcal{Z} \times \{\pm 1\}$. We have $(err(B, \mathcal{F}_{mon}))^2 \leq p_{nr}(B, \mathcal{F}_{mon}, 2)$.*

Proof Let $\rho = \mathbf{Pr}_{(z,y)\sim B}[y = 1]$ and S be a dataset $\{(z_1, y_1), (z_2, y_2)\}$, where (z_1, y_1) and (z_2, y_2) were drawn independently from distribution B. We start by showing that

$$\text{err}(B, \mathcal{F}_{\text{mon}})^2 \leq 2\rho(1-\rho)(1 - \mathbf{Pr}_{S\sim B^2}[z_1 > z_2 \mid y_1 = +1, y_2 = -1)]). \tag{5}$$

By the law of total probability, we have

$$\begin{aligned}\mathbf{Pr}_{(z,y)\sim B}[\text{sign}(z-w)\neq y] &= \mathbf{Pr}_{(z,y)\sim B}[z<w \mid y=+1]\rho + \mathbf{Pr}_{(z,y)\sim B}[z\geq w \mid y=-1](1-\rho)\\ &= (1-\text{TPR}(w))\rho + \text{FPR}(w)(1-\rho).\end{aligned}$$

Fact D.2 says that the area under the curve equals $\mathbf{Pr}_{S\sim B^2}[z_1 > z_2 \mid y_1 = +1, y_2 = -1]$. Therefore, we can geometrically upper bound this quantity by the area of the space which is (i) under the isoerror line for error $c = \text{err}(B, \mathcal{F}_{\text{mon}})$ and (ii) under the line $\text{TPR} = 1$. See Figure 2 for an illustration. Note that $\text{err}(B, \mathcal{F}_{\text{mon}})$ is exactly $\min_{w\in\mathbb{R}}\{\mathbf{Pr}_{(z,y)\sim B}[\text{sign}(z-w)\neq y]\}$. This approach gives us the following bound:

$$\mathbf{Pr}_{S\sim B^2}[z_1 > z_2 \mid y_1 = +1, y_2 = -1] \leq 1 - \frac{(\text{err}(B,\mathcal{F}_{\text{mon}}))^2}{2\rho(1-\rho)}.$$

Rearranging yields Equation (5). Next, we show that the right-hand side of Equation (5) is exactly $p_{\text{nr}}(B, \mathcal{F}_{\text{mon}}, 2)$, the probability of non-realizability.

Let E be the event that $(z_1, y_1), (z_2, y_2)$ is not realizable by $\mathcal{F}_{\text{mon}}$. By the law of total probability, we have that

$$\begin{aligned}\mathbf{Pr}_{S\sim B^2}[E] &= \sum_{i\in\{\pm1\}}\sum_{j\in\{\pm1\}} \mathbf{Pr}_{S\sim B^2}[E \mid y_1 = i, y_2 = j]\mathbf{Pr}_{S\sim B^2}[y_1 = i, y_2 = j]\\ &= 2\rho(1-\rho)\mathbf{Pr}_{S\sim B^2}[z_1 \leq z_2 \mid y_1 = +1, y_2 = -1]\\ &= 2\rho(1-\rho)(1-\mathbf{Pr}_{S\sim B^2}[z_1 > z_2 \mid y_1 = +1, y_2 = -1]).\end{aligned}$$

Therefore, $\text{err}(B, \mathcal{F}_{\text{mon}})^2 \leq p_{\text{nr}}(B, \mathcal{F}_{\text{mon}}, 2)$. ■

In our example, since both $\text{err}(B, \mathcal{F}_{\text{mon}})$ and $p_{\text{nr}}(B, \mathcal{F}_{\text{mon}}, 2)$ are non-negative, $\text{err}(B, \mathcal{F}_{\text{mon}}) \leq \sqrt{p_{\text{nr}}(B, \mathcal{F}_{\text{mon}}, 2)}$. Therefore, we conclude that with probability at least $1-\delta$ over the data (that we get by seeing $t = O\left(\frac{d\ln(1/\varepsilon)+\ln(1/\delta)}{\varepsilon^2}\right)$ tasks with data distributions P_j drawn from Ω and 2 samples from P_j for every task $j \in [t]$)

$$\mathop{\mathbb{E}}_{P\sim Q}\left[\min_{w\in\mathbb{R}}\left\{\mathbf{Pr}_{(\text{x},y)\sim P}\left[\text{sign}(\hat{h}(\mathbf{x}) - w) \neq y\right]\right\}\right] \leq$$
$$\sqrt{\mathop{\mathbb{E}}_{P\sim\Omega}\left[\mathbf{Pr}_{(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2)\sim P^2}\left[r_{\hat{h}}(\mathbf{x}_1, y_1, \mathbf{x}_2, y_2)\right] \neq +1\right]} \leq \varepsilon.$$

This proves Theorem D.4.

Theorem D.4 *We can metalearn $(\mathcal{H}_{d,1}, \mathcal{F}_{mon})$ to accuracy ε in the realizable case with t tasks and n samples per task when $t = O\big(d\ln(1/\varepsilon)/\varepsilon^2\big)$ and $n = 2$*

From Monotone Thresholds to Arbitrary Function Classes. Although the argument above relies on the particular structure of monotone function classes in several ways, the next few sections will show how the results can be generalized to arbitrary representations and specialized classifiers. First, we study the realizable case with very few samples per task (Section 3.2), and then we study the agnostic case (Appendix D.3) with more samples per task. The results for the agnostic case also apply to the realizable case, yielding incomparable statements. After establishing these general bounds in terms of properties of the representation and specialized classifiers, we use them to prove specific sample complexity bounds for linear classes in Section 4.

D.2. Sample and Task Complexity Bounds for the Realizable Case

When the metadistribution is meta-realizable, Theorems 3.2 and 3.5 bound the number of tasks and samples per task we need to metalearn. Their proofs follow the structure described in Appendix D.1.

Theorem 3.2 (Restated) *Let $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$ with dual Helly number $\mathrm{DH}(\mathcal{F}) = m$ and $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$. Then, for every ε and δ in $(0, 1)$, we can (ε, δ)-properly metalearn $(\mathcal{H}, \mathcal{F})$ in the realizable case for t tasks and n samples per task when*

$$t = \big(\mathrm{VC}(\mathcal{R}_{m,\mathcal{F},\mathcal{H}}) + \ln(1/\delta)\big) \cdot \left(\frac{O(\max(\mathrm{VC}(\mathcal{F}), m) \ln(1/\varepsilon))}{m\varepsilon}\right)^m \quad \text{and } n = m\,.$$

We first prove Lemma 3.3 and Lemma 3.4, which we use in the proof of Theorem 3.2.

Lemma 3.3 (Restated) *Let $\mathcal{F}$ be the class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$ with $\mathrm{DH}(\mathcal{F}) = m$. Fix an arbitrary distribution B over $\mathcal{Z} \times \{\pm 1\}$. If $err(B, \mathcal{F}) > 0$, then*

$$p_{nr}(B, \mathcal{F}, m) \geq \frac{1}{2}\left(\left[\frac{m \cdot err(B, \mathcal{F})}{16e \cdot v \ln\left(16/err(B, \mathcal{F})\right)}\right]^m\right),$$

where $v = \max(\mathrm{VC}(\mathcal{F}), m)$.

Proof Let $g(\varepsilon) = \frac{16}{\varepsilon}\mathrm{VC}(\mathcal{F}) \ln(16/\varepsilon)$. Function g is continuous and strictly decreasing in ε for $\varepsilon \in (0, 1]$. The proof analyzes two cases

In case one, if $m > g(\mathrm{err}(B, \mathcal{F}))$, then by Theorem A.5 the probability that a dataset of m points drawn from B is not realizable by $\mathcal{F}$ is

$$\begin{aligned} p_{nr}(B, \mathcal{F}, m) &= \mathbf{Pr}_{S_m \sim B^m}[S_m \text{ is not realizable by } \mathcal{F}] \\ &= \mathbf{Pr}_{S_m \sim B^m}\left[\min_{f \in \mathcal{F}} \mathrm{err}(S_m, f) > 0\right] \geq \frac{1}{2}, \end{aligned}$$

because $\mathrm{err}(B, \mathcal{F}) > 0$ and $m \geq \frac{8}{\varepsilon}[\mathrm{VC}(\mathcal{F}) \ln(16/\varepsilon) + \ln(2/\delta)]$ for $\varepsilon = \mathrm{err}(B, \mathcal{F})$ and $\delta = \frac{1}{2}$. This is stronger than the claimed lower bound. To see this, recall $v \geq m$ and observe

$$\begin{aligned} \frac{1}{2}\left(\frac{m \cdot \mathrm{err}(B, \mathcal{F})}{16e \cdot v \ln\left(16/\mathrm{err}(B, \mathcal{F})\right)}\right)^m &\leq \frac{1}{2}\left(\frac{\mathrm{err}(B, \mathcal{F})}{16e \cdot \ln\left(16/\mathrm{err}(B, \mathcal{F})\right)}\right)^m \\ &\leq \frac{1}{2}\left(\frac{1}{16e \cdot \ln(16)}\right)^m, \end{aligned}$$

which is less than $\frac{1}{2}$.

In case two, suppose $m \leq g(\mathrm{err}(B, \mathcal{F}))$. Drawing m i.i.d. samples from B is equivalent to drawing a larger dataset $S_n = \{(z_i, y_i)\}_{i \in [n]}$ from B of some size $n \geq m$, set later in the proof, and picking a uniformly random subset $S_m \subseteq S_n$ of size m. More formally,

$$\begin{aligned} p_{nr}(B, \mathcal{F}, m) &= \mathbf{Pr}_{S_m \sim B^m}[S_m \text{ is not realizable}] \\ &= \mathbf{Pr}_{\substack{S_n \sim B^n \\ S_m \sim \binom{S_n}{m}}}[S_m \text{ is not realizable}]. \end{aligned}$$

Furthermore, we notice that if S_n is labeled correctly by some $f \in \mathcal{F}$, then f labels S_m correctly, too. Hence,

$$\begin{aligned} p_{nr}(B, \mathcal{F}, m) &= \mathbf{Pr}[S_m \text{ is not realizable}] \\ &= \mathbf{Pr}[S_m \text{ is not realizable} \mid S_n \text{ is not realizable}] \cdot \mathbf{Pr}[S_n \text{ is not realizable}]. \end{aligned} \tag{6}$$

We know that if S_n is not realizable then there exists a non-realizable subset of size m. Since there are $\binom{n}{m}$ subsets, S_m is exactly this subset with probability at least $1/\binom{n}{m}$.

We now provide a lower bound on the probability that S_n is not realizable. We set $n := \frac{16}{\text{err}(B,\mathcal{F})}\text{VC}(\mathcal{F})\ln(16/\text{err}(B,\mathcal{F}))$, which satisfies $n \geq m$ by hypothesis. Then, by Theorem A.5 the probability that dataset S_n is not realizable by $\mathcal{F}$ is

$$\mathbf{Pr}_{S_n \sim B^n}[S_n \text{ is not realizable by } \mathcal{F}] = \mathbf{Pr}_{S_n \sim B^n}\left[\min_{f \in \mathcal{F}} \text{err}(S_n, f) > 0\right] \geq \frac{1}{2},$$

again because $\text{err}(B, \mathcal{F}) > 0$ and n is sufficiently large.

Thus, continuing from Equation (6) and using a bound on the binomial coefficient, we see that $p_{nr}(B, \mathcal{F}, m) \geq \frac{1}{2\binom{n}{m}} \geq \frac{1}{2(\frac{en}{m})^m}$. Plugging in $n = \frac{16}{\text{err}(B,\mathcal{F})}\text{VC}(\mathcal{F})\ln(16/\text{err}(B,\mathcal{F}))$, we get that

$$p_{\text{nr}}(B, \mathcal{F}, m) \geq \frac{1}{2}\left[\frac{m \cdot \text{err}(B, \mathcal{F})}{16e \cdot \text{VC}(\mathcal{F}) \ln\left(16/\text{err}(B, \mathcal{F})\right)}\right]^m. \tag{7}$$

As $\text{VC}(\mathcal{F}) \leq v$, this is stronger than the claim in the lemma. This concludes the proof. ■

Lemma 3.4 (Restated) *Let $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$, $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$ and m be a positive integer. Suppose there exists a strictly increasing convex function $\phi : (0, 1] \to [0, 1]$ such that for all metadistributions $\mathfrak{Q}$ satisfying $\min_{h \in \mathcal{H}}$ rep-err$(\mathfrak{Q}, h, \mathcal{F}) = 0$, all data distributions P in the support of $\mathfrak{Q}$ and all representations $h \in \mathcal{H}$, if rep-err$(P, h, \mathcal{F}) > 0$ the following holds:*

$$\phi(\textit{rep-err}(P, h, \mathcal{F})) \leq \mathbf{Pr}_{(x_1,y_1),\ldots,(x_m,y_m) \sim P^m}[(h(x_1), y_1), \ldots, (h(x_m), y_m) \textit{ not realizable by } \mathcal{F}].$$

Then, we can (ε, δ)-properly metalearn $(\mathcal{H}, \mathcal{F})$ with $t = O\left(\frac{\text{VC}(\mathcal{R}_{m,\mathcal{F},\mathcal{H}}) + \ln(1/\delta)}{\phi(\varepsilon)}\right)$ tasks and m samples per task in the realizable case.

Proof Our goal is to show that there exists a shared representation that achieves small representation error for $\mathfrak{Q}$. The proof proceeds in two stages. First, we use our bound on the VC dimension of $\mathcal{R}_{m,\mathcal{F},\mathcal{H}}$ to show that we can find a representation $\hat{h}$ that, when applied to data from a new task, admits a perfect specialized classifier with high probability. Second, we connect this to rep-err$(\mathfrak{Q}, \hat{h}, \mathcal{F})$, the error of $\hat{h}$ on the meta-distribution.

Recall our approach for monotone thresholds in Appendix D.1: for each task, we receive a data set $S^{(j)} = (x_1^{(j)}, y_1^{(j)}, \ldots, x_m^{(j)}, y_m^{(j)})$ and construct a single "data point" ζ_j:

$$\zeta_j \stackrel{\text{def}}{=} (S^{(j)}, +1).$$

We set the label to "+1" because, by the meta-realizability assumption, there exists an h^* such that $r_{h^*}(S^{(j)}) = +1$ for all j. Since each task distribution P_j is drawn independently from $\mathfrak{Q}$, these

ζ_j observations are drawn i.i.d. from some distribution $\mathcal{D}$. For an illustration of this process, see Figure 1.

We find an $\hat{h}$ such that $r_{\hat{h}}$ has zero error on the dataset $\zeta_1, \ldots, \zeta_t$. Because we set $t = \Theta\Big(\frac{\mathrm{VC}(\mathcal{R}_{m,\mathcal{F},\mathcal{H}})+\ln(1/\delta)}{\phi(\varepsilon)}\Big)$, by Theorem A.4 we know that $r_{\hat{h}}$ generalizes. That is, with probability at least $1-\delta$, we have

$$\mathbf{Pr}_{\zeta=(S,+1)\sim\mathcal{D}}\big[r_{\hat{h}}(S) \neq +1\big] \leq \phi(\varepsilon)\,. \tag{8}$$

For a data set $S = ((x_1, y_1), \ldots, (x_m, y_m))$ and representation h, define S_h to be $((h(x_1), y_1), \ldots, (h(x_m), y_m))$. Recall that by the assumption of this lemma, for $\hat{h}$ and all P in the support of $\mathfrak{Q}$, if rep-err$(P, \hat{h}, \mathcal{F}) > 0$ we have:

$$\phi\Big(\text{rep-err}(P, \hat{h}, \mathcal{F})\Big) \leq \mathbf{Pr}_{S\sim P^m}\big[S_{\hat{h}} \text{ is not realizable by } \mathcal{F}\big]. \tag{9}$$

Note that ϕ is a strictly increasing function and, thus, it has an inverse function ϕ^{-1} that is also strictly increasing. Therefore, the above bound implies that

$$\text{rep-err}(P, \hat{h}, \mathcal{F}) \leq \phi^{-1}\big(\mathbf{Pr}_{S\sim P^m}\big[S_{\hat{h}} \text{ is not realizable by } \mathcal{F}\big]\big). \tag{10}$$

Now we are ready to bound the meta-error of $\hat{h}$: rep-err$(\mathfrak{Q}, \hat{h}, \mathcal{F})$. We start by bounding $\phi\Big(\text{rep-err}(\mathfrak{Q}, \hat{h}, \mathcal{F})\Big)$. Since ϕ is convex, ϕ^{-1} is concave and we can apply Jensen's inequality. We get that:

$$\begin{aligned}
\text{rep-err}(\mathfrak{Q}, \hat{h}, \mathcal{F}) &= \mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\Big[\text{rep-err}(P, \hat{h}, \mathcal{F})\Big] \\
&\leq \mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\Big[\text{rep-err}(P, \hat{h}, \mathcal{F}) \mid \text{rep-err}(P, \hat{h}, \mathcal{F}) > 0\Big] \\
&\leq \mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\big[\phi^{-1}\big(\mathbf{Pr}_{S\sim P^m}\big[S_{\hat{h}} \text{ is not realizable by } \mathcal{F}\big]\big)\big] && \text{(by Eq. 10)} \\
&\leq \phi^{-1}\left(\mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\big[\mathbf{Pr}_{S\sim P^m}\big[S_{\hat{h}} \text{ is not realizable by } \mathcal{F}\big]\big]\right) && \text{(Jensen's inequality)} \\
&\leq \phi^{-1}(\phi(\varepsilon)) = \varepsilon && \text{(by Eq. 8)}
\end{aligned}$$

The above bound implies:

$$\text{rep-err}(\mathfrak{Q}, \hat{h}, \mathcal{F}) \leq \varepsilon = \min_{h\in\mathcal{H}} \text{rep-err}(\mathfrak{Q}, h, \mathcal{F}) + \varepsilon\,.$$

As a result, using $t = O\Big(\frac{\mathrm{VC}(\mathcal{R}_{m,\mathcal{F},\mathcal{H}})+\log(1/\delta)}{\phi(\varepsilon)}\Big)$ tasks and m samples from each task, we have found a representation function $\hat{h}$ that has the desired error bound for metalearning with probability $1-\delta$. Hence, the proof is complete. ■

We can now prove Theorem 3.2 by combining the results of Lemma 3.3 and Lemma 3.4.

Proof [Proof of Theorem 3.2] Let

$$\phi(\varepsilon) = \frac{1}{2}\left[\frac{m\varepsilon}{16e\cdot\max(\mathrm{VC}(\mathcal{F}), m)\ln(16/\varepsilon)}\right]^m.$$

By differentiating ϕ twice we see that it is a strictly increasing convex function in ε, for $\varepsilon \in (0,1)$ and $m \geq 1$.

Applying Lemma 3.3, we obtain that for every representation $\hat{h}$ and distribution P if $\min_{f\in\mathcal{F}} \mathbf{Pr}_{(x,y)\sim P}\Big[f(\hat{h}(x)) \neq y\Big] > 0$, then

$$\phi\Big(\min_{f\in\mathcal{F}} \mathbf{Pr}_{(x,y)\sim P}\Big[f(\hat{h}(x)) \neq y\Big]\Big) \leq \mathbf{Pr}_{(x_1,y_1),\ldots,(x_m,y_m)\sim P^m}\Big[(\hat{h}(x_1),y_1),\ldots,(\hat{h}(x_m),y_m) \text{ is not realizable by } \mathcal{F}\Big].$$

This satisfies the assumptions of Lemma 3.4 which says that we can metalearn $(\mathcal{H},\mathcal{F})$ with $t = O\Big(\frac{\mathrm{VC}(\mathcal{R}_{m,\mathcal{F},\mathcal{H}})+\ln(1/\delta)}{\phi(\varepsilon)}\Big)$ tasks and m samples per task. This concludes our proof. ■

Our next result analyzes metalearning with fewer tasks and more samples per task. Formally, Theorem 3.5 shows that we can metalearn $(\mathcal{H},\mathcal{F})$ for a meta-realizable $\mathfrak{Q}$ with $\tilde{O}(\mathrm{VC}(\mathcal{R}_{n,\mathcal{F},\mathcal{H}})/\varepsilon)$ tasks and $\tilde{O}(\mathrm{VC}(\mathcal{F})/\varepsilon)$ samples per task.

Theorem 3.5 (Restated) *Let $\mathcal{H}$ be a class of representation functions $h : \mathcal{X} \to \mathcal{Z}$ and $\mathcal{F}$ be a class of specialized classifiers $f : \mathcal{Z} \to \{\pm 1\}$. For every $\varepsilon,\delta \in (0,1)$, we can (ε,δ)-properly metalearn $(\mathcal{H},\mathcal{F})$ in the realizable case with t tasks and n samples per task when $t = O\Big(\frac{\mathrm{VC}(\mathcal{R}_{n,\mathcal{F},\mathcal{H}})\ln(1/\varepsilon)+\ln(1/\delta)}{\varepsilon}\Big)$ and $n = O\Big(\frac{\mathrm{VC}(\mathcal{F})\ln(1/\varepsilon)}{\varepsilon}\Big)$.*

Proof First, we set $\varepsilon_1 := \varepsilon/3$. Set the number of tasks

$$t := O\left(\frac{\mathrm{VC}(\mathcal{R}_{n,\mathcal{F},\mathcal{H}})\ln(1/\varepsilon_1)+\ln(1/\delta)}{\varepsilon_1}\right).$$

Each task j has n samples $S_j = \{(x_i^{(j)}, y_i^{(j)})\}_{i\in[n]}$, we construct a dataset Z of t points $\zeta_j = ((x_1^{(j)}, y_1^{(j)}, \ldots, x_n^{(j)}, y_n^{(j)}), +1)$, one for each task j, as in Figure 1. Each ζ_j is drawn i.i.d. from data distribution $\mathcal{D}$ (where we first draw P from $\mathfrak{Q}$ and then n points from P). Since we are in the realizable case, there exists a representation h such that r_h returns $+1$ for every sample drawn from $\mathcal{D}$. By Theorem A.8 we have that for $\hat{h} = \arg\min_{h\in\mathcal{H}} \mathrm{err}(Z, r_h)$ with probability at least $1-\delta$ over dataset Z

$$\mathrm{err}(\mathcal{D}, r_{\hat{h}}) \leq \frac{\varepsilon}{3}.$$

Fix a distribution P in the support of $\mathfrak{Q}$. We start by assuming that $\text{rep-err}(P,\hat{h},\mathcal{F}) > \varepsilon/3$. By Theorem A.5 for a dataset S of

$$n := \frac{24}{\varepsilon}(\mathrm{VC}(\mathcal{F})\ln(48/\varepsilon)+\ln(4))$$

samples drawn from P, with probability at least $1/2$ over S all specialized classifiers $f \in \mathcal{F}$ with $\mathrm{err}(P, f\circ\hat{h}) > \varepsilon/3$ have $\mathrm{err}(S, f\circ\hat{h}) > 0$. Therefore,

$$\mathbf{Pr}_{S\sim P^n}\Big[\min_{f\in\mathcal{F}} \mathrm{err}(S, f\circ\hat{h}) > 0\Big] = \mathbf{Pr}_{S\sim P^n}\big[r_{\hat{h}}(S) \neq -1\big] \geq \frac{1}{2}.$$

By Markov's inequality, we see that

$$\mathbf{Pr}_{P\sim\mathfrak{Q}}\big[\mathbf{Pr}_{S\sim P^n}\big[r_{\hat{h}}(S)\neq+1\big]\geq 1/2\big]\leq 2\mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\big[\mathbf{Pr}_{S\sim P^n}\big[r_{\hat{h}}(S)\neq+1\big]\big]=2\,\text{err}(\mathcal{D},r_{\hat{h}})\leq\frac{2\varepsilon}{3}.$$

So far we have shown that for a fixed P if rep-err$(P,\hat{h},\mathcal{F})>\varepsilon/3$, then $\mathbf{Pr}_{S\sim P^n}\big[r_{\hat{h}}(S)\neq+1\big]\geq 1/2$. As a result, we have that

$$\mathbf{Pr}_{P\sim\mathfrak{Q}}\Big[\text{rep-err}(P,\hat{h},\mathcal{F})>\varepsilon/3\Big]\leq\mathbf{Pr}_{P\sim\mathfrak{Q}}\big[\mathbf{Pr}_{S\sim P^n}\big[r_{\hat{h}}\neq+1\big]\geq 1/2\big]\leq\frac{2\varepsilon}{3}.$$

We can use this to bound the meta-error of representation $\hat{h}$ as follows. We see that with probability at least $1-\delta$ over the datasets of the t tasks

$$\begin{aligned}\text{rep-err}(\mathfrak{Q},\hat{h},\mathcal{F})&=\mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\Big[\min_{f\in\mathcal{F}}\mathbf{Pr}_{(x,y)\sim P}\Big[f(\hat{h}(x))\neq y\Big]\Big]\\&\leq\frac{\varepsilon}{3}+\mathbf{Pr}_{P\sim\mathfrak{Q}}\Big[\min_{f\in\mathcal{F}}\mathbf{Pr}_{(x,y)\sim P}\Big[f(\hat{h}(x))\neq y\Big]>\varepsilon/3\Big]\\&\leq\frac{\varepsilon}{3}+2\frac{\varepsilon}{3}=\varepsilon.\end{aligned}$$

This concludes our proof. ■

D.3. Sample and Task Complexity Bounds for the Agnostic Case

In this section, we consider metalearning in the agnostic case.

Theorem 3.6 (Restated) *Let $\mathcal{H}$ be a class of representation functions $h:\mathcal{X}\to\mathcal{Z}$ and $\mathcal{F}$ be a class of specialized classifiers $f:\mathcal{Z}\to\{\pm1\}$. Then, for every ε and $\delta\in(0,1)$ we can (ε,δ)-properly metalearn $(\mathcal{H},\mathcal{F})$ with t tasks and n samples per task when $t=O\Big(\frac{\mathit{PDim}(\mathcal{Q}_{n,\mathcal{F},\mathcal{H}})\ln(1/\varepsilon)+\ln(1/\delta)}{\varepsilon^2}\Big)$ and $n=O\Big(\frac{\mathrm{VC}(\mathcal{F})+\ln(1/\varepsilon)}{\varepsilon^2}\Big)$.*

Proof We begin by setting our parameters: Set $\varepsilon_1:=\varepsilon/3$ and $\varepsilon_2:=\varepsilon/3$. Let the number of tasks be the following for a sufficiently large constant in the O notation:

$$t:=O\left(\frac{\mathrm{PDim}(\mathcal{Q}_{n,\mathcal{F},\mathcal{H}})\ln(1/\varepsilon_1)+\ln(1/\delta))}{\varepsilon_1^2}\right).\tag{11}$$

Suppose for each task $j\in[t]$ has n samples $S_j=\{(x_i^{(j)},y_i^{(j)})\}_{i\in[n]}$, we construct a dataset Z of t points $\zeta_j=((x_1^{(j)},y_1^{(j)},\ldots,x_n^{(j)},y_n^{(j)}))$, one for each task j. Each ζ_j is drawn i.i.d. from the data distribution $\mathcal{D}$ for which we first draw P from $\mathfrak{Q}$ and then n points from P. By Theorem A.8, we have that for $\hat{h}=\arg\min_{h\in\mathcal{H}}\frac{1}{t}\sum_{j=1}^t q_h(\zeta_j)$ with probability at least $1-\delta$:

$$\mathop{\mathbb{E}}_{\zeta\sim\mathcal{D}}\big[q_{\hat{h}}(\zeta)\big]\leq\min_{h\in\mathcal{H}}\mathop{\mathbb{E}}_{\zeta\sim\mathcal{D}}[q_h(\zeta)]+\frac{\varepsilon}{3}.$$

Fix a task distribution P and a representation h. For any fixed specialized classifier $f' \in \mathcal{F}$, we have:

$$\mathop{\mathbb{E}}_{(x_1,y_1),\ldots,(x_n,y_n)\sim P^n}\left[\min_{f\in\mathcal{F}}\frac{1}{n}\sum_{i\in[n]}\mathbb{I}\{f(h(x_i))\neq y_i\}\right] \leq \mathop{\mathbb{E}}_{(x_1,y_1),\ldots,(x_n,y_n)\sim P^n}\left[\frac{1}{n}\sum_{i\in[n]}\mathbb{I}\{f'(h(x_i))\neq y_i\}\right].$$

Since the inequality holds for any f' in $\mathcal{F}$, it holds when we take minimum over all f'. Hence, we obtain:

$$\mathop{\mathbb{E}}_{(x_1,y_1),\ldots,(x_n,y_n)\sim P^n}\left[\min_{f\in\mathcal{F}}\frac{1}{n}\sum_{i\in[n]}\mathbb{I}\{f(h(x_i))\neq y_i\}\right] \leq \min_{f\in\mathcal{F}}\mathop{\mathbb{E}}_{(x_1,y_1),\ldots,(x_n,y_n)\sim P^n}\left[\frac{1}{n}\sum_{i\in[n]}\mathbb{I}\{f(h(x_i))\neq y_i\}\right]. \tag{12}$$

Next, we use the above inequality to continue bounding the expectation of $q_{\hat{h}}$:

$$\begin{aligned}
\mathop{\mathbb{E}}_{\zeta\sim\mathcal{D}}\left[q_{\hat{h}}(\zeta)\right] &\leq \min_{h\in\mathcal{H}}\mathop{\mathbb{E}}_{\zeta\sim\mathcal{D}}[q_h(\zeta)] + \frac{\varepsilon}{3} && \text{(Eq. (11))}\\
&= \min_{h\in\mathcal{H}}\mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\left[\mathop{\mathbb{E}}_{(x_1,y_1),\ldots,(x_n,y_n)\sim P^n}\left[\min_{f\in\mathcal{F}}\frac{1}{n}\sum_{i=1}^{n}\mathbb{I}\{f(h(x_i))\neq y_i\}\right]\right] + \frac{\varepsilon}{3}\\
&\leq \min_{h\in\mathcal{H}}\mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\left[\min_{f\in\mathcal{F}}\mathop{\mathbb{E}}_{(x_1,y_1),\ldots,(x_n,y_n)\sim P^n}\left[\frac{1}{n}\sum_{i=1}^{n}\mathbb{I}\{f(h(x_i))\neq y_i\}\right]\right] + \frac{\varepsilon}{3} && \text{(Eq. 12)}\\
&= \min_{h\in\mathcal{H}}\text{rep-err}(\mathfrak{Q},h,\mathcal{F}) + \frac{\varepsilon}{3}.
\end{aligned}$$

Consider an arbitrary task distribution P. Suppose we have a dataset S of n labeled sample from P where n is the following with a sufficiently large constant in the O notation:

$$n \coloneqq O\left(\frac{\text{VC}(\mathcal{F}) + \ln(1/\varepsilon_2)}{\varepsilon_2^2}\right).$$

Since n is sufficiently large, we have uniform convergence of the empirical error for all $f \in \mathcal{F}$ by Theorem A.4. Furthermore, the specialized classifier $\hat{f}$ minimizing the empirical error over S, (i.e. $\hat{f} = \arg\min_{f\in\mathcal{F}}\text{err}(S, f\circ\hat{h})$) must have low true error as well. Therefore, with probability $1-\frac{\varepsilon}{3}$ over the randomness in S, we get:

$$\text{rep-err}(P,\hat{h},\mathcal{F}) \leq \mathbf{Pr}_{(x,y)\sim P}\left[\hat{f}(\hat{h}(x))\neq y\right] \leq \text{rep-err}(S,\hat{h},\mathcal{F}) + \frac{\varepsilon}{3}.$$

Thus, if we take the expectation over the dataset S, we see that:

$$\begin{aligned}
\text{rep-err}(P,\hat{h},\mathcal{F}) &= \mathop{\mathbb{E}}_{S\sim P^n}\Big[\text{rep-err}(P,\hat{h},\mathcal{F})\Big] \\
&= \mathop{\mathbb{E}}_{S\sim P^n}\Big[\text{rep-err}(P,\hat{h},\mathcal{F})\Big|\text{rep-err}(P,\hat{h},\mathcal{F}) \le \text{rep-err}(S,\hat{h},\mathcal{F})+\frac{\varepsilon}{3}\Big] \\
&\quad\times \mathbf{Pr}_{S\sim P^n}\Big[\text{rep-err}(P,\hat{h},\mathcal{F}) \le \text{rep-err}(S,\hat{h},\mathcal{F})+\frac{\varepsilon}{3}\Big] \\
&\quad+ \mathop{\mathbb{E}}_{S\sim P^n}\Big[\text{rep-err}(P,\hat{h},\mathcal{F})\Big|\text{rep-err}(P,\hat{h},\mathcal{F}) > \text{rep-err}(S,\hat{h},\mathcal{F})+\frac{\varepsilon}{3}\Big] \\
&\quad\times \mathbf{Pr}_{S\sim P^n}\Big[\text{rep-err}(P,\hat{h},\mathcal{F}) > \text{rep-err}(S,\hat{h},\mathcal{F})+\frac{\varepsilon}{3}\Big] \\
&\le \left(\mathop{\mathbb{E}}_{S\sim P^n}\Big[\text{rep-err}(S,\hat{h},\mathcal{F})\Big]+\frac{\varepsilon}{3}\right)\cdot 1 + 1\cdot\frac{\varepsilon}{3} \\
&= \mathop{\mathbb{E}}_{S\sim P^n}\Big[\text{rep-err}(S,\hat{h},\mathcal{F})\}\Big]+\frac{2\varepsilon}{3}.
\end{aligned}$$

Now, we take the expectation over $P\sim\mathfrak{Q}$ and obtain that:

$$\begin{aligned}
\text{rep-err}(\mathfrak{Q},\hat{h},\mathcal{F}) &\le \mathop{\mathbb{E}}_{P\sim\mathfrak{Q}}\left[\mathop{\mathbb{E}}_{S\sim P^n}\Big[\text{rep-err}(S,\hat{h},\mathcal{F})\}\Big]\right]+\frac{2\varepsilon}{3} \\
&= \mathop{\mathbb{E}}_{\zeta\sim\mathcal{D}}\big[q_{\hat{h}}(\zeta)\big]+\frac{2\varepsilon}{3}\,.
\end{aligned}$$

Note that in the last line above, the S dataset drawn from a random P can be viewed as a random data set according to $\mathcal{D}$. Combining with the bound we have derived earlier for the expected value of $q_{\hat{h}}$, we get the following that holds with probability at least $1-\delta$:

$$\text{rep-err}(\mathfrak{Q},\hat{h},\mathcal{F}) \le \min_{h\in\mathcal{H}}\text{rep-err}(\mathfrak{Q},h,\mathcal{F})+\varepsilon\,.$$

Hence, the proof is complete. ■

D.4. Sample and Task Complexity Bounds for General Metalearning

For general metalearning we obtain the following corollary of Theorem B.4 by applying Theorem C.1 for agnostic multitask learning. It is important to note that this method does not return a representation $h\in\mathcal{H}$, but a more general specialization algorithm that uses the datasets of the already seen tasks to learn a classifier for the new task.

Corollary D.5 *For any $(\mathcal{H},\mathcal{F})$, any $\varepsilon>0$, constant $c>0$ and any t,n, $(\mathcal{H},\mathcal{F})$ is $(\varepsilon,2/c)$-meta-learnable in the agnostic case with t tasks, n samples per task and n specialization samples when*

$$nt = O\left(\frac{\text{VC}(\mathcal{F}^{\otimes t+1}\circ\mathcal{H})\ln(1/\varepsilon)}{\varepsilon^2}\right).$$

Proof By Theorem B.4 we can $(\varepsilon/c,\varepsilon/c)$-multitask learn $(\mathcal{H},\mathcal{F})$ for $t+1$ tasks and n per task when

$$nt+n = \frac{c'c^2\text{VC}(\mathcal{F}^{\otimes t+1}\circ\mathcal{H})\ln(c/\varepsilon)}{\varepsilon^2},$$

for a constant $c' > 0$. As a result, the reduction of Theorem B.4 implies that we can $(\varepsilon, 2/c)$-metalearn $(\mathcal{H}, \mathcal{F})$ for t tasks, n samples per task and n specialization samples when

$$nt = O\left(\frac{\mathrm{VC}(\mathcal{F}^{\otimes t+1} \circ \mathcal{H}) \ln(1/\varepsilon)}{\varepsilon^2}\right).$$

This concludes the proof. ■

Appendix E. Bounds for Halfspaces over Linear Representations

In this section, we provide results on multitask learning and metalearning of linear projections (as representations) and halfspaces (as specialized classifiers):

$$\mathcal{H}_{d,k} = \left\{h \mid h(\mathbf{x}) = \mathbf{B}\mathbf{x}, \mathbf{B} \in \mathbb{R}^{k \times d}\right\}, \text{ and } \mathcal{F}_k = \left\{f \mid f(\mathbf{z}) = \mathrm{sign}(\mathbf{a} \cdot \mathbf{z} - w), \mathbf{a} \in \mathbb{R}^k, w \in \mathbb{R}\right\}.$$

E.1. VC Dimension of $\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}$

The general bounds of Lemma C.2 give us that the VC dimension of $\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}$ is in the range

$$\max(kt + t, d + 1) \leq \mathrm{VC}(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}) \leq dt + t.$$

We know the VC dimension of the class of composite functions $\mathcal{F}_k \circ \mathcal{H}_{d,k}$ because $\mathcal{F}_k \circ \mathcal{H}_{d,k}$ and $\mathcal{F}_d$, the class of d-dimensional halfspaces, are the same class.

In Theorem 1.2 we characterize the VC dimension of class $\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}$ up to a constant. The bound we give matches the intuition from counting the number of parameters, which yields $dk+kt+t$ when $t > k$ and $dt + t$ when $t \leq k$.

Theorem 1.2 (Restated) *For all $t, d, k \in \mathbb{N}$, we have*

$$\mathrm{VC}(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}) = \begin{cases} t(d+1), & \text{if } t \leq k \\ \Theta(dk + kt), & \text{if } t > k. \end{cases}$$

In the proof for the upper bound we use Warren's Theorem, which we state here as a lemma:

Lemma E.1 (Warren's Theorem, Warren (1968)) *Let $p_1, \ldots, p_n$ be real polynomials in v variables, each of degree at most $\ell \geq 1$. If $n \geq v$, then the number of distinct sequences $\mathrm{sign}(p_1(x)), \ldots, \mathrm{sign}(p_n(x))$ for all x does not exceed $(4e\ell n/v)^v$. In particular, if $\ell \geq 2$ and $n \geq 8v \log_2 \ell$, then the number of distinct sequences of $+1, -1$ taken by $\mathrm{sign}(p_1(x)), \ldots, \mathrm{sign}(p_n(x))$ is less than 2^n.*

Proof We first show that for $t \leq k$ tasks $\mathrm{VC}\left(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}\right) = dt + t$. To lower-bound the VC dimension, we will show that there exists a dataset of size $dt + t$ which can be shattered by $\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}$. We know that the VC dimension of the class of d-dimensional thresholds $\mathcal{F}_d$ is $d + 1$, which means that there exists a dataset $(\mathbf{x}_1, \ldots, \mathbf{x}_{d+1})$ which can be shattered by $\mathcal{F}_d$. Consider the dataset $\cup_{j \in [t]}\{(j, \mathbf{x}_1), \ldots, (j, \mathbf{x}_{d+1})\}$. This dataset can be shattered by $\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}$. To see this, fix a labeling $\mathbf{y} \in \{\pm 1\}^{t \times (d+1)}$, where $y_{j,i}$ is the label of datapoint $(j, \mathbf{x}_i)$. For a $j \in [t]$ we know that there exist $\mathbf{b}_j \in \mathbb{R}^d$ and $w_j \in \mathbb{R}$ such that for all $i \in [d + 1]$ we have $\mathrm{sign}(\mathbf{b}_j \mathbf{x}_i - w_j) = y_{j,i}$. Since

$t \leq k$, we set $\mathbf{B}$ to be the matrix with rows b_j^T for the first t rows and all zeros everywhere else and $\mathbf{a}_j \in \mathbb{R}^k$ to be the one-hot encoding of j. We pick $h(\mathbf{x}) = \mathbf{B}\mathbf{x}$ and $f_j(\mathbf{z}) = \text{sign}(\mathbf{a}_j\mathbf{z} - w_j)$. Therefore, we have that for all $j \in [t]$ and $i \in [d+1]$, $y_{j,i} = f_j(h(\mathbf{x}_i))$. As a result,

$$\text{VC}(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}) \geq dt + t.$$

For the upper bound Lemma C.2 says that $\text{VC}(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}) \leq t\text{VC}(\mathcal{F}_k \circ \mathcal{H}_{d,k})$. Furthermore, since the composition of linear functions is linear the class of composite functions $\mathcal{F}_k \circ \mathcal{H}_{d,k}$ and $\mathcal{F}_d$ are the same. Hence,

$$\text{VC}(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}) \leq t\text{VC}(\mathcal{F}_d) = dt + t.$$

We now look at the case where we have more than k tasks. For the upper bound we rewrite functions $g \in \mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}$ as $g(j, \mathbf{x}) = \text{sign}(\mathbf{a}_j\mathbf{B}\mathbf{x} - w_j)$. Observe that this is equivalent to

$$g(j, \mathbf{x}) = \text{sign}(\mathbf{e_j}^T\mathbf{A}\mathbf{B}\mathbf{x} - \mathbf{e_j}^T\mathbf{w}),$$

where $\mathbf{e}_j \in \{0,1\}^t$ is the one-hot encoding of j and

$$\mathbf{A} = \begin{pmatrix} \mathbf{a}_1^T \\ \vdots \\ \mathbf{a}_t^T \end{pmatrix} \text{ and } \mathbf{w} = \begin{pmatrix} w_1 \\ \vdots \\ w_t \end{pmatrix}.$$

Every combination of $\mathbf{A} \in \mathbb{R}^{t\times k}, \mathbf{B} \in R^{k\times d}$ and $\mathbf{w} \in \mathbb{R}^t$ gives us a specific labeling function g. Let $n \geq 8(tk + kd + t)$. Take a dataset $((j_1, \mathbf{x}_1), \ldots, (j_n, \mathbf{x}_n)) \in ([t] \times \mathbb{R}^d)^n$. We will show that $\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}$ does not shatter this data set. For each $i \in [n]$, we define a polynomial $p_i(\mathbf{A}, \mathbf{B}, \mathbf{w}) = \mathbf{e}_{j_i}^T\mathbf{A}\mathbf{B}\mathbf{x}_i - \mathbf{e}_{j_i}^T\mathbf{w}$. Each of these is a degree-2 polynomial in $tk + kd + t$ variables. By construction, $g(j_i, \mathbf{x}_i) = \text{sign}(p_i(\mathbf{A}, \mathbf{B}, \mathbf{w}))$. By Lemma E.1, $\text{VC}\left(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}\right) \leq 8(dk+kt+t)$.

By Lemma C.2 we have that $\text{VC}(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}) \geq kt + t$. Additionally, for $t > k$ any dataset that is shattered by $\mathcal{F}_k^{\otimes k} \circ \mathcal{H}_{d,k}$ can be shattered by $\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}$. As a result, $\text{VC}(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}) \geq \text{VC}(\mathcal{F}_k^{\otimes k} \circ \mathcal{H}_{d,k})$. We proved above that $\text{VC}(\mathcal{F}_k^{\otimes k} \circ \mathcal{H}_{d,k}) = dk + k$. Therefore,

$$\text{VC}(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}) \geq \Omega(dk + kt).$$

■

Recall that we can reduce metalearning to multitask learning and obtain the task and sample complexity in Corrolary D.5. Therefore, the characterization of the VC dimension of class $\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}$ implies the following bound for the task and sample complexity of metalearning.

Corollary E.2 *For $\varepsilon > 0$ and constant $c > 0$, we can $(\varepsilon, c/2)$-metalearn $(\mathcal{H}_{d,k}, \mathcal{F}_k)$ with t tasks and n samples per task when*

$$n = O\left(\frac{k\ln(1/\varepsilon)}{\varepsilon^2}\right) \text{ and } t \geq d.$$

Proof Based on Theorem 1.2 when $t \geq d$, we know that $\text{VC}(\mathcal{F}_k^{\otimes t} \circ \mathcal{H}_{d,k}) = O(kt)$. Hence, by Corollary D.5 we can metalearn $(\mathcal{H}_{d,k}, \mathcal{F}_k)$ up to error ε when $n = O\left(\frac{k\ln(1/\varepsilon)}{\varepsilon^2}\right)$. ■

E.2. Bounding the VC Dimension of Boolean Functions of Polynomials

Here we restate and prove Lemma 4.1.

Lemma 4.1 (Restated) *Let $w, \ell, d \in \mathbb{N}$ with $\ell \geq 2$ and fix a domain of features $\mathcal{X}$ and a function $g : \{\pm 1\}^w \to \{\pm 1\}$. For each $x \in \mathcal{X}$, let $p_x^{(1)}, \ldots, p_x^{(w)}$ be degree-ℓ polynomials in d variables. Consider the hypothesis class $\mathcal{C}$ consisting of, for all $\mathbf{v} \in \mathbb{R}^d$, the functions $c_{\mathbf{v}} : \mathcal{X} \to \{\pm 1\}$ defined as*

$$c_{\mathbf{v}}(x) := g(sign(p_x^{(1)}(\mathbf{v})), \ldots, sign(p_x^{(w)}(\mathbf{v}))). \tag{13}$$

We have $\mathrm{VC}(\mathcal{C}) \leq 8d \log_2(\ell w)$.

Proof We bound the growth function of $\mathcal{C}$. Fix n datapoints $x_1, \ldots, x_n \in \mathcal{X}$. By the definition of hypothesis class $\mathcal{C}$, the sequence $c_{\mathbf{v}}(x_1), \ldots, c_{\mathbf{v}}(x_n)$ is exactly the sequence

$$g\big(\mathrm{sign}(p_{x_1}^{(1)}(\mathbf{v})), \ldots, \mathrm{sign}(p_{x_1}^{(w)}(\mathbf{v}))\big), \ldots, g\big(\mathrm{sign}(p_{x_n}^{(1)}(\mathbf{v})), \ldots, \mathrm{sign}(p_{x_n}^{(w)}(\mathbf{v}))\big)$$

By Lemma E.1, we know that the number of distinct sequences

$$\mathrm{sign}(p_{x_1}^{(1)}(\mathbf{v})), \ldots, \mathrm{sign}(p_{x_1}^{(w)}(\mathbf{v})), \ldots, \mathrm{sign}(p_{x_n}^{(1)}(\mathbf{v})), \ldots, \mathrm{sign}(p_{x_n}^{(w)}(\mathbf{v}))$$

for all $\mathbf{v} \in \mathbb{R}^d$ does not exceed $(4e\ell wn/d)^d$. The number of distinct outputs of a function is upper bounded by the number of distinct inputs it gets. As we saw above, the number of distinct inputs is at most $(4e\ell wn/d)^d$ and, thus, $|\{(c_{\mathbf{v}}(x_1), \ldots, c_{\mathbf{v}}(x_n)) \mid \mathbf{v} \in \mathbb{R}^d\}| \leq (4e\ell wn/d)^d$. Since this holds for fixed $x_1, \ldots, x_n \in \mathcal{X}$, the growth function of $\mathcal{C}$ is at most $(4e\ell wn/d)^d$.

To bound the VC dimension of $\mathcal{C}$ it suffices to show that for $n > 8d \log_2(\ell w)$ points, the growth function $\mathcal{G}_{\mathcal{C}}(n)$ is less than 2^n. If $n > 8d \log_2(\ell w)$, then $n > 8d$ because $\ell \geq 2$ and $w \geq 1$. We know that when $n/d > 8$, $\log_2(4e) < n/(2d)$ and $\log_2(n/d) < 3n/(8d)$. Therefore, we see that

$$\log_2\left(\frac{4e\ell wn}{d}\right) = \log_2(4e) + \log_2(\ell w) + \log_2\left(\frac{n}{d}\right) < \frac{n}{2d} + \frac{n}{8d} + \frac{3n}{8d} = \frac{n}{d}.$$

Combining the above steps, we have proven that for $n > 8 \log_2(\ell w)$, $\mathcal{G}_{\mathcal{C}}(n) \leq (4e\ell wn/d)^d < 2^n$. ■

E.3. VC Dimension of the Realizability Predicate Class

In this section we provide the full proof of Theorem 4.2.

Theorem 4.2 (Restated) *For every d, k, n with $n \geq k+2$,* $\mathrm{VC}(\mathcal{R}_{n,\mathcal{F}_k,\mathcal{H}_{d,k}}) \leq O(dkn + dk\log(kn))$.

Given a dataset $D = \{(\mathbf{z}_i, y_i)\}_{i\in[n]}$ of n points in $\mathbb{R}^k \times \{\pm 1\}$, we define $\mathbf{Z}$ as the matrix whose i-th row is the $k+1$-dimensional vector $\mathbf{z}_i' = y_i(\mathbf{z}_i \| 1)$, for $i \in [n]$. For $I \subseteq [n]$ a set of indices, we define $\mathbf{Z}_I$ to be the matrix whose rows are the vectors $\mathbf{z}_i'$ for i in I.

To prove Theorem 4.2, we establish a set of conditions on a data set which allow us to check for separability. These conditions can be expressed via a limited number of low-degree polynomials, which allows us to apply Warren's Theorem.

Our first lemma in this section equates separability of a data set with the existence of a special subset of points. If these points are "on the margin" of a linear separator defined by vector $\mathbf{a}$, then we know that $\mathbf{a}$ is a strict separator.

Lemma E.3 *Dataset $D = \{(\mathbf{z}_i, y_i)\}_{i\in[n]}$ is strictly separable if and only if there exists a subset I of linearly independent rows of $\mathbf{Z}$ such that for all $\mathbf{a} \in \mathbb{R}^{k+1}$ if $\mathbf{z}'_i \cdot \mathbf{a} = 1$ for all $i \in I$, then $\mathbf{z}'_i \cdot \mathbf{a} \geq 1$ for all $i \in [n]$.*

Proof By definition, D is strictly separable iff there exists a linear separator $\mathbf{a} \in \mathbb{R}^{k+1}$ such that for all $i \in [n]$ we have $y_i(\mathbf{z}_i \| 1) \cdot \mathbf{a} = \mathbf{z}'_i \cdot \mathbf{a} \geq 1$. We define $A = \{\mathbf{a} \mid \mathbf{z}'_i \cdot \mathbf{a} \geq 1, \forall i \in [n]\}$ as the set of all linear separators of D and, for any $\mathbf{a}$, $I_\mathbf{a}$ as the set of tight constraints for $\mathbf{a}$, i.e., $I_\mathbf{a} = \{i \mid \mathbf{z}'_i \cdot \mathbf{a} = 1\}$. Note that A and $I_\mathbf{a}$ might be empty.

$\Leftarrow$) Assume that $I \subseteq [n]$ defines a subset of linearly independent rows such that for all $\mathbf{a} \in \mathbb{R}^{k+1}$ if $\mathbf{z}'_i \cdot \mathbf{a} = 1$ for all $i \in I$, then $\mathbf{z}'_i \cdot \mathbf{a} \geq 1$ for all $i \in [n]$. We find a vector $\mathbf{a}$ that makes the constraints in I tight, that is, solve the linear system $\mathbf{Z}_I a = \mathbf{1}$, where $\mathbf{1}$ is the $|I|$-dimensional all-ones vector. Since $\mathbf{Z}_I$ has linearly independent rows, this system has at least one solution $\hat{\mathbf{a}} = \mathbf{Z}_I^+ \mathbf{1}$, where $\mathbf{Z}_I^+ = \mathbf{Z}_I^T (\mathbf{Z}_I \mathbf{Z}_I^T)^{-1}$ is the pseudoinverse. Thus, for all $i \in I$ we have $\mathbf{z}'_i \cdot \hat{\mathbf{a}} = 1$, which implies $\mathbf{z}'_i \cdot \hat{\mathbf{a}} \geq 1$ for all $i \in [n]$ by our assumption on I. Therefore, the points in D are strictly separable.

$\Rightarrow$) Assume that D is strictly separable. Then, by Lemma E.4, there exists a strict separator $\mathbf{a}$ such that $\mathrm{rank}(\mathbf{Z}_{I_\mathbf{a}}) = \mathrm{rank}(\mathbf{Z})$ and $\mathbf{Z} \cdot \mathbf{a} \geq \mathbf{1}$, where the inequality holds entry-wise. If needed, we can remove indices from $I_\mathbf{a}$ to produce I, a subset with the same rank but linearly independent rows.

Now suppose that $\mathbf{a}'$ satisfies $\mathbf{z}'_i \cdot \mathbf{a}' = 1$ for all $i \in I$. In other words, $\mathbf{a}'$ satisfies $\mathbf{Z}_I \cdot \mathbf{a}' = \mathbf{1}$. This means we can write $\mathbf{a}' = \mathbf{a} + \mathbf{u}$, where $\mathbf{u}$ is in the right nullspace of $\mathbf{Z}_I$. Since $\mathbf{Z}$ and $\mathbf{Z}_I$ share a rowspace, they also share a right nullspace. Thus $\mathbf{Z} \cdot \mathbf{a}' = \mathbf{Z} \cdot \mathbf{a} \geq \mathbf{1}$, where the inequality holds entry-wise. This completes the proof. ■

The proof of lemma E.3 uses the following lemma, which says that every strictly separable data set admits a separator whose set of tight constraints is full rank.

Lemma E.4 *For a data set $D = \{(\mathbf{z}_i, y_i)\}_{i\in[n]}$, let $\mathbf{Z}$ be its associated matrix and, for a vector $\mathbf{a}$, let $I_\mathbf{a} \subseteq [n]$ be the set of tight constraints (i.e., the largest set I such that $\mathbf{Z}_I \cdot \mathbf{a} = \mathbf{1}$). If D is strictly separable, then there exists a vector $\mathbf{a}^*$ such that $\mathbf{z}_i \cdot \mathbf{a}^* \geq 1$ for all $i \in [n]$ and $\mathrm{rank}(\mathbf{Z}) = \mathrm{rank}(\mathbf{Z}_{I_{\mathbf{a}^*}})$.*

Proof

D is strictly separable, so by definition there exists a vector $\mathbf{a}_0$ such that $\mathbf{z}_i \cdot \mathbf{a}_0 \geq 1$ for all $i \in [n]$. Suppose $\mathrm{rank}(\mathbf{Z}) > \mathrm{rank}(\mathbf{Z}_{I_{\mathbf{a}_0}})$, since otherwise we are done. We will construct another separator $\mathbf{a}_1$ that satisfies $\mathrm{rank}(\mathbf{Z}_{I_{\mathbf{a}_0}}) < \mathrm{rank}(\mathbf{Z}_{I_{\mathbf{a}_1}})$. Since $\mathrm{rank}(\mathbf{Z})$ is finite, repeating this process will yield a separator $\mathbf{a}^*$ with $\mathrm{rank}(\mathbf{Z}_{I_{\mathbf{a}^*}}) = \mathrm{rank}(\mathbf{Z})$.

If $\mathbf{a}$ is a solution to $\mathbf{Z}_{I_{\mathbf{a}_0}} \cdot \mathbf{a} = \mathbf{1}$, we can write it as $\mathbf{a}' = \mathbf{a}_0 + \mathbf{u}$, where $\mathbf{u}$ is in the right nullspace of $\mathbf{Z}_{I_{\mathbf{a}_0}}$. Since the rank of $\mathbf{Z}_{I_{\mathbf{a}_0}}$ is strictly less than the rank of $\mathbf{Z}$, there exists a vector $\mathbf{v}$ that lies in the right nullspace of $\mathbf{Z}_{I_{\mathbf{a}_0}}$ but *not* in the right nullspace of $\mathbf{Z}$. Our separator $\mathbf{a}_1$ will be of the form $\mathbf{a}_0 + c \cdot \mathbf{v}$ for some real value c. Note that halfspaces of this form keep the constraints in $I_{\mathbf{a}_0}$ tight: by construction we have $\mathbf{z}'_i \cdot (\mathbf{a}_0 + c \cdot \mathbf{v}) = 1$ for all $i \in I_{\mathbf{a}_0}$.

Let $I^\perp \subseteq \bar{I}_{\mathbf{a}_0}$ be the subset of rows which are not in the right rowspace of $\mathbf{Z}_{I_{\mathbf{a}_0}}$. This set is nonempty, since $\mathrm{rank}(\mathbf{Z}_{I_{\mathbf{a}_0}}) < \mathrm{rank}(\mathbf{Z})$. For all $i \in I^\perp$, let $m_i(c) = \langle \mathbf{z}'_i, \mathbf{a}_0 \rangle + c \cdot \langle \mathbf{z}'_i, \mathbf{u} \rangle$. Each of these is a linear function in c and, by the definition of $I_{\mathbf{a}_0}$ and the fact that $\mathbf{a}_0$ is a linear separator, we see that $m_i(0) > 1$ for all $i \in I^\perp$.

For each $i \in I^\perp$, there exists an interval $[L_i, R_i]$ such that, if $c \in [L_i, R_i]$, then $m_i(c) \geq 1$, i.e., point i lies on the correct side of $\mathbf{a}_0 + c \cdot \mathbf{v}$. Because $m_i(0) > 1$ for all $i \notin I_{\mathbf{a}_0}$, we see that

$L_i < 0 < R_i$. The intersection of these intervals, $[\max L_i, \min R_i]$, is nonempty. For any c in the intersection, $\mathbf{a}_0 + c \cdot \mathbf{v}$ is a strict separator. (This holds for $i \in I^\perp$ by construction; if $i \notin I^\perp$ then i is in the rowspace of $\mathbf{Z}_{I_{\mathbf{a}_0}}$ and we have $\mathbf{z}_i' \cdot (\mathbf{a}_0 + c \cdot \mathbf{v}) = \mathbf{z}_i' \cdot \mathbf{a}_0 \geq 1$, as $\mathbf{v}$ is in the nullspace of $\mathbf{Z}_{I_{\mathbf{a}_0}}$.)

It remains to select c so that at least one additional constraint is tight. This is easy: if $\max L_i$ is finite then we have $m_i(\max L_i) = 1$. The same holds for $\min R_i$. Since we know at least one of $\max L_i$ and $\min R_i$ are finite, we can select a finite one as our value of c^*.

We have constructed a vector $\mathbf{a}_1 = \mathbf{a}_0 + c^* \cdot \mathbf{v}$ that strictly separates D and makes constraint i^* tight for some $i^* \in I^\perp$, where $I^\perp$ is the set of constraints not in the rowspace of $\mathbf{Z}_{I_{\mathbf{a}_0}}$. Thus, $\operatorname{rank}(\mathbf{Z}_{I_{\mathbf{a}_0}}) < \operatorname{rank}(\mathbf{Z}_{I_{\mathbf{a}_1}})$. This concludes the proof. ■

A simple corollary to Lemma E.3 says that, on separable data sets, the special subset of points allows us to identify a specific separator.

Corollary E.5 *Dataset $D = \{(\mathbf{z}_i, y_i)\}_{i \in [n]}$ is strictly separable if and only if there exists a subset of points I for which*

1. *$\mathbf{Z}_I$ is full rank, and*
2. *for all $i \in [n]$, we have that $\mathbf{z}_i' \cdot \hat{\mathbf{a}} \geq 1$, where $\hat{\mathbf{a}} = \mathbf{Z}_I^+ \mathbf{1}_{|I|}$.*

Proof $\Rightarrow$) Assume D is strictly separable. By Lemma E.3 we know that there exists a subset I of linearly independent rows such that for all $\mathbf{a} \in \mathbb{R}^{k+1}$ if $\mathbf{z}_i' \cdot \mathbf{a} = 1$ for all $i \in I$ then $\mathbf{z}_i' \cdot \mathbf{a} \geq 1$ for all $i \in [n]$. Since the rows in subset I are linearly independent, Z_I is full rank. By construction, $\hat{\mathbf{a}}$ satisfies $Z_I \cdot \hat{\mathbf{a}} = \mathbf{1}_{|I|}$. Thus $\mathbf{z}_i' \cdot \hat{\mathbf{a}} \geq 1$ for all $i \in [n]$.

$\Leftarrow$) Assume that such a subset I exists. We see that $\hat{\mathbf{a}}$ strictly separates D. ■

We now show how to express this characterization of separability in the language of polynomials. With this lemma in hand, the proof of Theorem 4.2 will be a direct application of Lemma 4.1, our extension of Warren's Theorem.

Lemma E.6 *Let $w = (n+1) \cdot 2^n$. For a data set $D = \{(x_i, y_i)\}_{i \in [n]}$, there exists a Boolean function $g : \{\pm 1\}^w \to \{\pm 1\}$ and a list of polynomials $p_D^1(h), \ldots, p_D^w(h)$, each of degree at most $4(k+1)$, such that we can express the $(n, \mathcal{F}_k)$-realizability predicate r_h as*

$$r_h(D) = g\Big(\operatorname{sign}\Big(p_D^{(1)}(h)\Big), \ldots, \operatorname{sign}\Big(p_D^{(w)}(h)\Big)\Big).$$

Proof A representation h induces a labeled dataset in the representation space $D_h = \{(h(x_i), y_i)\}_{i \in [n]}$. By Corollary E.5, we can check whether D_h is linearly separable by checking whether any of the $\sum_{i=1}^{k+1} \binom{n}{i} \leq 2^n$ subsets of D_h of size at most $k+1$ satisfies the two conditions of the corollary. In the remainder of the proof, we fix a subset I and construct a Boolean function g_I over signs of polynomials that checks if I satisfies these conditions. This function will use at most $n+1$ polynomials, each of degree at most $4(k+1)$. The proof is finished by taking g to be the OR of these functions for each subset I.

We write D_h as a matrix $\mathbf{Z}$ whose i-th row is the $k+1$ dimensional vector $\mathbf{z}_i' = y_i(h(x_i) \| 1)$. We construct a polynomial $p_I^{(0)}(h)$ that is *negative* iff Z_I is full rank (so that $\operatorname{sign} p_I^{(0)} = +1$ indicates

rank deficiency). For each $i \in [n]$, we construct a polynomial $p_I^{(i)}(h)$ that, when Z_I is full rank, is nonnegative iff $z_i' \cdot \hat{\mathbf{a}} \geq 1$, where $\hat{\mathbf{a}} = \mathbf{Z}_I^+ \mathbf{1}_{|I|}$. We then take g_I to be

$$g_I(h) := \left(\neg \operatorname{sign} p_I^{(0)}(h)\right) \wedge \left(\bigwedge_{i \in [n]} \operatorname{sign} p_I^{(i)}(h)\right).$$

Recall that we interpret $+1$ as logical "true."

Matrix $\mathbf{Z}_I$ is full rank if and only if $\det(\mathbf{Z}_I \mathbf{Z}_I^T) \neq 0$. Therefore, we set $p_I^{(0)}(h) = -\det(\mathbf{Z}_I \mathbf{Z}_I^T)^2$, which is a polynomial in $k \times d$ variables of degree $4(k+1)$. To see this, observe that each entry in $\mathbf{Z}_I \mathbf{Z}_I^T$ is a polynomial of degree 2 in the variables of h. Then, $\det(\mathbf{Z}_I \mathbf{Z}_I^T)$ is a polynomial of degree $2I$. Finally, $\det(\mathbf{Z}_I \mathbf{Z}_I^T)^2$ is a polynomial of degree $4I \leq 4(k+1)$. Taking the negation gives us the desired polynomial $p_I^{(0)}$.

We now check that $z_i' \cdot \hat{\mathbf{a}} - 1 \geq 0$ for a given i. Let $\Delta = \det(\mathbf{Z}_I \mathbf{Z}_I^T)$, a polynomial of degree $2(k+1)$. Then $\hat{\mathbf{a}} = \mathbf{Z}_I^T (\mathbf{Z}_I \mathbf{Z}_I^T)^{-1} \mathbf{1}_{|I|} = \frac{\mathbf{Z}_I^T \operatorname{adj}(\mathbf{Z}_I \mathbf{Z}_I^T) \mathbf{1}_{|I|}}{\Delta}$, where adj denotes the adjugate matrix. We have $z_i' \cdot \hat{\mathbf{a}} - 1 \geq 0$ when $\Delta \neq 0$ and $z_i' \cdot \Delta \hat{\mathbf{a}} - \Delta \geq 0$. Each entry in $\Delta \hat{\mathbf{a}}$ is a polynomial of degree $2|I| - 1 \leq 2k + 1$, so setting $p_I^{(i)}(h) = z_i' \cdot \Delta \hat{\mathbf{a}} - \Delta$ we have a polynomial of degree at most $2(k+1)$ that is, when $\mathbf{Z}_I$ is full rank, is nonnegative iff the constraint is satisfied. ■

Proof [Proof of Theorem 4.2] By Lemma E.6, we have that the $(n, \mathcal{F}_k)$-realizability predicate is a Boolean function of $(n+1) \cdot 2^n$ signs of polynomials in dk variables of degree $4(k+1)$. Lemma 4.1 gives us that the VC dimension of the class of realizability predicates $\mathcal{R}_{n,\mathcal{F}_k,\mathcal{H}_{d,k}}$ is upper-bounded by $8dk \log_2(4(k+1)n2^n) = O(dkn + dk \log(kn))$. ■

E.4. Pseudodimension of the Empirical Error Function Class

We can now also bound the pseudodimension (Definition A.7) of the class of empirical error functions (Definition 1.8).

Theorem 4.5 (Restated) *For all d, k, n with $n \geq k + 2$, $\operatorname{PDim}(Q_{n,\mathcal{F}_k,\mathcal{H}_{d,k}}) = \tilde{O}(dkn)$.*

Our first step toward proving these results is the following lemma, which follows almost immediately from Corollary E.5. The proof uses the fact that a data set which can be classified with accuracy at least α can be classified perfectly if we change $1 - \alpha$ labels.

Lemma E.7 *Dataset $D = \{(\mathbf{z}_i, y_i)\}_{i \in [n]}$ can be linearly separated with accuracy α if and only if there exists a subset of points I and a vector $\boldsymbol{\sigma} \in \{\pm 1\}^n$ with $\sum_{i=1}^n \frac{\mathbb{I}\{\sigma_i = +1\}}{n} = \alpha$ such that*

1. *$\mathbf{Z}_I$ is full rank*
2. *for all $i \in [n]$, we have that $(\mathbf{z}_i' \cdot \hat{\mathbf{a}}_{\boldsymbol{\sigma}})\sigma_i \geq 1$, where $\hat{\mathbf{a}}_{\boldsymbol{\sigma}} = Z_I^+ \boldsymbol{\sigma}_I$.*

Proof By definition, dataset D can be linearly separated with accuracy α if and only if there exists a vector $\boldsymbol{\sigma} \in \{\pm 1\}^n$ with $\sum_{i=1}^n \frac{\mathbb{I}\{\sigma_i = 1\}}{n} = \alpha$ such that dataset $D_\sigma = \{(\mathbf{z}_i, \sigma_i y_i)\}_{i \in [n]}$ is strictly separable. Now, we can apply Corollary E.5 to dataset D_σ. Let $\hat{\mathbf{Z}}$ be the matrix whose i-th row is

the vector $\hat{\mathbf{z}}'_i = \sigma_i y_i(\mathbf{z}_i \| 1)$. The rank of $\hat{\mathbf{Z}}_I$ remains the same as the rank of $\mathbf{Z}_I$ since every row is a scalar multiple of the corresponding row in Z_I. Additionally, observe that $Z_I^+ \boldsymbol{\sigma}_I = \hat{\mathbf{Z}}_I^+ \mathbf{1}_{|I|}$. ■

The next lemma shows how to express statements about the empirical error function in the language of low-degree polynomials.

Lemma E.8 *Given a dataset $D = \{(x_i, y_i)\}_{i\in[n]}$ and an $\alpha \in [0,1]$, the predicate $\mathbb{I}_{\pm}\{q_h(x_1, y_1, \ldots, x_n, y_n) = \alpha\}$ is a Boolean function of $(n+1)\cdot 2^{2n}$ signs of polynomials in the variables of h of degree $4(k+1)$.*

Proof By Lemma E.7 we have seen that $\mathbb{I}_{\pm}\{q_h(x_1, y_1, \ldots, x_n, y_n){=}\alpha\} = \mathbb{I}_{\pm}\{\exists \boldsymbol{\sigma} \in \{\pm 1\}^n$ with $\sum_{i=1}^n \mathbb{I}\{\sigma_i = +1\} = \alpha n$ s.t. $r_h(x_1, \sigma_1 y_1, \ldots, x_n, \sigma_n y_n) = +1\}$. Therefore, by Lemma E.6 for every value of $\boldsymbol{\sigma}$ that has α fraction of ones we can check a Boolean function of $(n+1)\cdot 2^n$ signs of polynomials in the variables of h of degree $4(k+1)$. There are no more than 2^n such vectors σ. In total, we need to evaluate $(n+1)\cdot 2^{2n}$ signs of polynomials. ■

We are now ready to prove the final result in this section.

Proof [Proof of Theorem 4.5]

A well-known fact about pseudodimension (see, e.g., Anthony and Bartlett (1999)) is that it equals the VC dimension of the class of subgraphs. As a function $q_h \in Q_{n,\mathcal{F},\mathcal{H}}$ maps data sets $D = \{(x_i, y_i)\}_{i\in[n]}$ to the interval $[0,1]$ (corresponding to error), the object (D, τ) lies in the subgraph of q_h if $q_h(D) \geq \tau$. In our case,

$$\mathrm{PDim}(Q_{n,\mathcal{F},\mathcal{H}}) = \mathrm{VC}(\{\mathbb{I}_{\pm}\{q_h(D) \geq \tau\} \mid q_h \in Q_{n,\mathcal{F},\mathcal{H}}\}). \tag{14}$$

To use the tools we have previously established, we will show that, for a fixed (D, τ), we can write the indicator $\mathbb{I}_{\pm}\{q_h(D) \geq \tau\}$ as a Boolean function of signs of polynomials.

But this is easy: the indicator is +1 exactly when there exists an $\alpha \geq \tau$ such that $q_h(D) = \alpha$, which we analyzed in Lemma E.8. We take an OR over the $n+1$ possible values of $\alpha \in \{0, 1/n, \ldots, 1\}$:

$$\mathbb{I}_{\pm}\{q_h(D) \geq \tau\} = \bigwedge_{\alpha} \big(\mathbb{I}_{\pm}\{q_h(D) = \alpha \text{ and } \alpha \geq \tau\}\big).$$

(Recall that we interpret $+1$ as logical "true.") When $\alpha < \tau$, we can represent this as $\mathrm{sign}(-1)$, a degree-0 polynomial. When $\alpha \geq \tau$, we apply Lemma E.8 to see that each term can be written as a Boolean function of $(n+1)\cdot 2^{2n}$ signs of polynomials in the variables of h of degree $4(k+1)$. Together, we see that $\mathbb{I}_{\pm}\{q_h(D) \geq \tau\}$ can be written as a Boolean function of at most $(n+1)^2 2^{2n}$ signs of polynomials, each of (at most) the same degree. By Lemma 4.1, the VC dimension (and thus the pseudodimension of $Q_{n,\mathcal{F},\mathcal{H}}$) is at most $8dk \log_2(4(k+1)\cdot(n+1)^2 2^{2n}) = O(dkn + dk\log(kn))$. ■

Appendix F. Bounds for Finite Specialized Classifiers and Representations

In this section, we study multitask learning and metalearning for a finite class of representations $\mathcal{H}$ and a finite class of specialized classifiers $\mathcal{F}$. Specifically, we use our results from Appendix C and Section 3 to provide bounds for the number of tasks and the number of samples per task needed to

multitask learn and metalearn $(\mathcal{H}, \mathcal{F})$. Recall that the VC dimension of a finite class $\mathcal{G}$ is at most $\log_2 |\mathcal{G}|$.

By Theorem C.1, we can (ε, δ)-multitask learn $(\mathcal{H}, \mathcal{F})$ when the number of tasks t and the number of samples per task n satisfy $nt = O(\frac{\mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H}) \cdot \ln(1/\varepsilon) + \ln(1/\delta)}{\varepsilon})$ in the realizable case and $nt = O(\frac{\mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H}) \cdot \ln(1/\delta)}{\varepsilon^2})$ in the agnostic case. For finite classes, we can bound the VC dimension of $\mathcal{F}^{\otimes t} \circ \mathcal{H}$ directly by counting:

$$\mathrm{VC}(\mathcal{F}^{\otimes t} \circ \mathcal{H}) \leq \log_2 |\mathcal{H}| + t \log_2 |\mathcal{F}|.$$

(For specific classes, the VC dimension can be significantly smaller.)

To apply our general metalearning theorems, we need to bound the VC dimension of the realizability predicate class $\mathcal{R}_{n,\mathcal{F},\mathcal{H}}$, the pseudodimension of the empirical error function class $\mathcal{Q}_{n,\mathcal{F},\mathcal{H}}$, and the dual Helly number $\mathrm{DH}(\mathcal{F})$. All of these bounds are direct since the underlying classes are finite.

Corollaries F.2 and F.3 present our results for metalearning in the realizable and agnostic settings, respectively. Note that for finite classes of specialized classifiers $\mathcal{F}$, the best possible bound on $\mathrm{DH}(\mathcal{F})$ that depends only on $|\mathcal{F}|$ is simply $\mathrm{DH}(\mathcal{F}) \leq |\mathcal{F}|$. (Lemma F.1 shows that this is tight, but for many classes we have $\mathrm{DH}(\mathcal{F}) \ll |\mathcal{F}|$.) Since the number of tasks depends exponentially on $\mathrm{DH}(\mathcal{F})$, the first result in Corollary F.2 is only useful for ε small enough so that $|\mathcal{F}|$ is smaller than $O(\frac{\log_2 |\mathcal{F}| \ln(1/\varepsilon)}{\varepsilon})$.

Lemma F.1 (VC dimension and dual Helly number) *For every integer $\ell \geq 2$,*

1. *There exists a class $\mathcal{F}_{\ell,\mathrm{VC}}$ with VC dimension ℓ and dual Helly number 2.*
2. *There exists a class $\mathcal{F}_{\ell,NR}$ with dual Helly number ℓ and VC dimension 1.*

Proof Fix $\ell \geq 2$. Let $\mathcal{F}_{\ell,\mathrm{VC}}$ be the class of all functions $f : \prime, \infty, ..., \ell \to \{\pm 1\}$ such that $f(0) = 1$. This class has VC dimension ℓ since it shatters the set $\{1, ..., \ell\}$ but not the entire domain. However, it has very small nonrealizable subsets: if a set cannot be realized, it must either contain the same value labeled with different points, or the example $(0, -1)$. Either way, there is a subset of 1 or 2 points that is not realizable.

Let $\mathcal{F}_{\ell,NR}$ denote the class of point functions on $[\ell]$; that is, the functions f_x that take the value 1 at *exactly* one point $x \in [\ell]$ (and -1 elsewhere). This class has VC dimension 1. However, the set $S = \{(x, -1) : x \in [\ell]\}$ is unrealizable (since no value is labeled with 1), but all of its subsets of size $\ell - 1$ are realizable (by f_x, where $(x, -1)$ is the point that was removed from S). ■

Corollary F.2 *If $\mathcal{H}, \mathcal{F}$ are finite, we can (ε, δ)-metalearn $(\mathcal{H}, \mathcal{F})$ in the realizable case with t tasks and n samples per task when*

$$t = (\log_2 |\mathcal{H}| + \ln(1/\delta)) \cdot O\left(\frac{\ln(1/\varepsilon)}{\varepsilon}\right)^{|\mathcal{F}|} \quad \textit{and} \quad n = |\mathcal{F}|$$

or

$$t = O\left(\frac{\log_2 |\mathcal{H}| \cdot \ln(1/\varepsilon) + \ln(1/\delta)}{\varepsilon}\right) \quad \textit{and} \quad n = O\left(\frac{\log_2 |\mathcal{F}| \cdot \ln(1/\varepsilon)}{\varepsilon}\right).$$

Proof For finite $\mathcal{F}$, we have $\mathrm{VC}(\mathcal{F}) \leq \log_2 |\mathcal{F}|$ and $\mathrm{DH}(\mathcal{F}) \leq |\mathcal{F}|$. The second inequality holds because, if a dataset is not realizable by $\mathcal{F}$, then for every $f \in \mathcal{F}$ there exists a data point that is not labeled correctly by f. The union of these at most $|\mathcal{F}|$ points is also not realizable by $\mathcal{F}$. For finite $\mathcal{H}$, the class of realizability predicates $\mathcal{R}_{n,\mathcal{F},\mathcal{H}}$ has at most one realizability predicate per representation. Thus, it has VC dimension $\mathrm{VC}(\mathcal{R}_{n,\mathcal{F},\mathcal{H}}) \leq \log_2 |\mathcal{H}|$. The corollary follows by plugging these measures into the results of Theorems 3.2 and 3.5. ■

Corollary F.3 *If $\mathcal{H}, \mathcal{F}$ are finite, we can (ε, δ)-metalearn $(\mathcal{H}, \mathcal{F})$ with t tasks and n samples per task when*

$$t = O\left(\frac{\log_2 |\mathcal{H}| \cdot \ln(1/\varepsilon) + \ln(1/\delta)}{\varepsilon^2}\right) \quad \textit{and} \quad n = O\left(\frac{\log_2 |\mathcal{F}| + \ln(1/\varepsilon)}{\varepsilon^2}\right).$$

Proof We have that $\mathrm{VC}(\mathcal{F}) \leq \log_2 |\mathcal{F}|$. Additionally, we see that $\mathrm{PDim}(\mathcal{Q}_{n,\mathcal{F},\mathcal{H}}) \leq \log_2 |\mathcal{H}|$ because $\mathrm{PDim}(\mathcal{Q}_{n,\mathcal{F},\mathcal{H}}) = \mathrm{VC}(\{\mathbb{I}_{\pm}\{q_h(D) \geq \tau\} \mid q_h \in Q_{n,\mathcal{F},\mathcal{H}}\})$ and the size of $\mathcal{Q}_{n,\mathcal{F},\mathcal{H}}$ is at most $|\mathcal{H}|$. By plugging these two bounds into Theorem 3.6 we obtain the result. ■