Teaching with Limited Information on the Learner's Behaviour

Ferdinando Cicalese [1] **Sergio Filho** [2] **Eduardo Laber** [2] **Marco Molinaro** [2]

Abstract

Machine Teaching studies how efficiently a Teacher can guide a Learner to a target hypothesis. We focus on the model of Machine Teaching with a black box learner introduced in [Dasgupta et al., ICML 2019], where the teaching is done interactively without having any knowledge of the Learner's algorithm and class of hypotheses, apart from the fact that it contains the target hypothesis h^*. We first refine some existing results for this model and, then, we study new variants of it. Motivated by the realistic possibility that h^* is not available to the learner, we consider the case where the teacher can only aim at having the learner converge to a best available approximation of h^*. We also consider weaker black box learners, where, in each round, the choice of the consistent hypothesis returned to the Teacher is not adversarial, and in particular, we show that better provable bounds can be obtained for a type of Learner that moves to the next hypothesis smoothly, preferring hypotheses that are close to the current one; and for another type of Learner that can provide to the Teacher hypotheses chosen at random among those consistent with the examples received so far. Finally, we present an empirical evaluation of our basic interactive teacher on real datasets.

1. Introduction

Machine Teaching studies how efficiently a Teacher can teach a target hypothesis to a Learner. The classic works (Shinohara, 1991; Goldman & Kearns, 1995) consider the setting where the Teacher sends in one shot a set of labeled examples to the Learner, which then has to output the correct target hypothesis. In more recent works, the focus has been on the interactive setting (Liu et al., 2017; Chen et al., 2018; Liu et al., 2018; Dasgupta et al., 2019) — where the Teacher and Leaner interact over multiple rounds. In each round, the Teacher sends examples to the Learner, which returns [1] some feedback; this process continues until the Learner reaches the target hypothesis (or a good approximation of it).

Machine teaching models have proved useful in several contexts, e.g., crowd sourcing (Johns et al., 2015; Zhou et al., 2018), intelligent tutoring systems (Rafferty et al., 2016; Zhu et al., 2018), analysis of training set attacks (Mei & Zhu, 2015). Moreover, commercial tools are under development by the Microsoft Machine Teaching Group, as detailed on their web page, which are based on, or employ, the paradigm of machine teaching, e.g., PICL, which leverages the selection of examples that maximize the training value of the interaction with the teacher; LUIS for natural language understanding; and other projects on building models for autonomous systems.

Most of the above works assume that the Teacher has significant knowledge about the Learner, e.g., its hypothesis class and the specific procedure employed for learning a hypothesis from labeled examples. However, this assumption excludes many important situations as human teaching and automatic learners with black box behaviour (e.g. Deep Nets). Thus, recent work in the field has focused on analysing scenarios in which the Teacher's knowledge about the Learner is limited (Liu et al., 2018; Dasgupta et al., 2019).

In particular, (Dasgupta et al., 2019) addressed machine teaching with a *black box* learner, where the only knowledge of the Teacher about the Learner is that its hypothesis class contains the target. They considered a model of interaction where at each round the Teacher sends labelled examples to the Learner and it provides a hypothesis that is consistent with all the examples received so far. The authors provide bounds on the number of examples required to teach the target hypothesis to *worst-case* learners.

Here, we refine some existing results for the model of machine teaching with a black box learners considered in

*Equal contribution [1]Department of Computer Science, University of Verona, Italy [2]Department of Computer Science, PUC-Rio, Brazil. Correspondence to: Eduardo Laber <eduardo.laber1@gmail.com>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

[1]Unless specified, we will tacitly assume that Learner and Teacher are machines, hence we use neutral pronouns.

(Dasgupta et al., 2019) and also introduce and analyse new variants of it. We are motivated, on the one hand, by the realistic scenario in which 'exact teaching' is not possible (the target does not belong to Learner's hypothesis class) and, on the other hand, by the fact that the Learner may not be adversarial to the Teacher.

1.1. Notation and Model

Before precisely stating our contributions we set some notation and explain the teaching model in more detail. There is a set $\mathcal{X}$ of examples, and a finite set $\mathcal{Y}$ of possible labels for each example. By a hypothesis we mean a function that maps each example in $\mathcal{X}$ to a label in $\mathcal{Y}$. We assume that:

Teacher: has a target hypothesis $h^* : \mathcal{X} \to \mathcal{Y}$, unknown to Learner ($h^*(e)$ is the correct label for e).

Learner: has a hypothesis class $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$, unknown to Teacher

In each round the Teacher sends the Learner a set of labeled examples $(e, h^*(e))$, then the Learner returns a hypothesis from its class with minimum number of errors in the examples received thus far. The goal of the Teacher is to send a minimum number of examples so as to make the Learner return a hypothesis from $\mathcal{H}$ with the smallest total number of errors in the whole dataset $\mathcal{X}$ (in the realizable case $h^* \in \mathcal{H}$, this means returning the correct hypothesis h^*). In the basic setting the Teacher does not have additional information on the learning algorithm used by the Learner to select among its minimum error hypotheses.

A fundamental notion in machine teaching is that of a *teaching set* for h^* (Goldman & Kearns, 1995), which is a set of examples $X \subseteq \mathcal{X}$ that distinguishes h^* from every other hypothesis in $\mathcal{H}$, that is, for every $h \neq h^*$ there is an example $e \in X$ for which $h(e) \neq h^*(e)$. We use $\mathcal{TS}(\mathcal{H}, h^*)$ to denote the size of the smallest teaching set for h^*. When $(\mathcal{H}, h^*)$ is clear from the context we use $\mathcal{TS}$ as a shorthand. We use $m = |\mathcal{X}|$ and $n = |\mathcal{H}|$ to denote the size of the sets of examples and hypotheses, respectively, and use $wrong(h)$ to denote the set of examples in which h fails (differs from h^*).

1.2. Contributions and Related Work

Our first contribution is a teaching algorithm $\mathcal{A}_{\text{base}}$ that *with high probability* guarantees convergence to the target hypothesis h^* using $O(\mathcal{TS} \log m \log n)$ examples (Theorems 1 and 2). Its correctness relies on a novel analysis of the on-line set cover algorithm proposed by (Alon et al., 2009). The main obstacle to derive this analysis is handling the non-trivial dependence between the Teacher's and the Learner's actions over time. We rely on martingale techniques and arguments reminiscent of decoupling (de la Peña & Giné, 1999) to overcome this difficulty.

(Dasgupta et al., 2019) present interesting results for black box learners and one of them is also a teaching algorithm based on a (different) adaptation of the on line set cover algorithm from (Alon et al., 2009). Their analysis guarantees bounds similar to ours, although it is based on the knowledge of an upper bound on n. However, some relevant subtleties arising from the interdependence between Teacher and Learner were not addressed in the proofs from (Dasgupta et al., 2019). In this respect, we understand that an additional contribution of our analysis is to clarify and formalize (via an application of our Lemma 2) the validity of a key statement in their argument (Lemma 5 of (Dasgupta et al., 2019)). Details are presented in Suppl. Material, Appendix A. That said, we would like to emphasize that the algorithm and the statements from (Dasgupta et al., 2019) are correct.

We also use our algorithm $\mathcal{A}_{\text{base}}$ as a basis for both improved results and extension to other variants of the problem. In Section 2.2, we propose a modified algorithm that obtains a stronger bound that depends on the (unknown) distribution of the number of errors among the hypotheses in $\mathcal{H}$ (Theorem 3). In Section 2.3, we generalize the above bound to the *non-realizable case*, where the Teacher cannot assume that the Learner's class contains the target hypothesis. Our algorithm guarantees that the Learner converges to the hypothesis $\bar{h}$ that is the closest to h^* in its hypothesis class, after receiving $O(\mathcal{TS}_k \log m \log(m+n))$ examples, where $\mathcal{TS}_k$ is a lower bound on the number of examples needed for this task.

These results are valid for the *worst-case* Learner model (Shinohara, 1991; Goldman & Kearns, 1995), where no assumption is made on which hypothesis the Learner selects among those of minimum error in the examples received thus far. Different models for the Learner's behavior have been recently considered (Zilles et al., 2011; Gao et al., 2017; Chen et al., 2018; Mansouri et al., 2019; Kirkpatrick et al., 2019). The assumption that the Learner smoothly navigates over its hypothesis class, always updating its current hypothesis to one that is 'close' to it, was used to motivate the *local preference* model introduced in (Chen et al., 2018) and extended in (Mansouri et al., 2019). For this type of Learner, when the closeness is measured in terms of Hamming distance, we present a teaching algorithm that with high probability sends $O(\mathcal{TS} \log n(\log err_1 + \log\log n))$ examples, where err_1 is the number of errors of the first hypothesis provided by the Learner (Theorems 5 and 6)[2] . Thus, this teaching algorithm benefits from a Learner that starts close to the target hypothesis. It is possible to show that this bound is not achievable by efficient

[2]We remark that this result holds if $n = |\mathcal{H}|$ is redefined as the number of non-equivalent hypotheses in $\mathcal{H}$ w.r.t. h^*, where two hypotheses are equivalent if they agree with h^* in exactly the same examples of $\mathcal{X}$; hence, $n \leq 2^m$ and $\log\log n \leq \log m$.

algorithms, in the worst-case model, through a simple modification of a lower bound presented in (Korman, 2004).

We also obtain improved bounds for the model where the Teacher can ask the Learner for a batch of random hypotheses consistent with the examples presented thus far. We propose a teaching algorithm that, with high probability, teaches h^* by sending in total $O(\mathcal{TS}\log(n+m))$ examples and requesting $O(\mathcal{TS}\log(m+n))$ random hypotheses per round (Theorem 7). For the relevant case where the number of hypotheses n is larger than the number of examples m, the bound on the number of examples is the best possible for poly-time algorithms under the assumption $\mathcal{P} \neq \mathcal{NP}$, even when the Teacher knows the class of hypotheses $\mathcal{H}$ (Raz & Safra, 1997).

We note that our non-adversarial models for learners are related to models that have already been considered (Chen et al., 2018; Balbach & Zeugmann, 2011; Singla et al., 2014; Angluin & Dohrn, 2020). In fact, the model of smooth transitions can be seen as an instance of the local preference model from (Chen et al., 2018), where hypotheses close to the current one, in terms of the Hamming distance, are preferred. Moreover, learners that return a random hypothesis have been considered in (Balbach & Zeugmann, 2011; Singla et al., 2014). However, in these works, in contrast to ours, the Teacher is aware of the Learner's hypothesis class. An analogous result, in a different context of teaching, where separation is proved between worst case and random adversary is (Angluin & Dohrn, 2020).

Although the teaching algorithms mentioned so far aim at obtaining a small teaching set, they may end up producing a non-minimal one (w.r.t. example deletion). In Section 4 we show that with a limited amount of extra interactions the Teacher is able to construct a minimal teaching set. This result may be useful when the main goal is to obtain a compressed training set.

Finally, to complement our theoretical results, we present in Section 5 experiments with 12 real datasets that show that our basic Teacher (the $\mathcal{A}_{\text{agno}}$ from Section 2) sends significantly fewer examples, to reach a given level of accuracy, than a Teacher that does not interact with the Learner.

2. Teaching with Worst-case Learner

Recall the teaching model from Section 1.1. In this section we consider worst-case learners that can return any hypothesis $h \in \mathcal{H}$ that has smallest number of errors on the examples received thus far.

2.1. Realizable Hypothesis Case

We first consider the realizable case when the target hypothesis h^* belongs to the learner's class $\mathcal{H}$. Notice that in this case the Learner always sends a hypothesis that is correct on all examples seen thus far.

As in (Dasgupta et al., 2019), we leverage the connection between teaching and *set cover*. We say that an example $e \in \mathcal{X}$ *covers* hypothesis $h \in \mathcal{H}$ if the latter makes a mistake in this example, namely $h(e) \neq h^*(e)$. Notice that covered hypotheses are out of consideration from the Learner, namely it never sends a hypothesis that is covered by the examples it has received. Thus, after the examples sent by the Teacher cover all hypotheses other than h^* the Learner must send back the correct hypothesis h^*, achieving the learning goal. This means that one can reduce the problem of teaching to that of *online set cover*: in the beginning of each round, the Teacher receives a hypothesis from Learner and sends examples that cover this hypothesis (and hopefully other unknown hypotheses), in a way that the total number of examples sent is small.

Our proposed teaching algorithm $\mathcal{A}_{\text{base}}$ uses the online set cover algorithm of (Alon et al., 2009), and can be described as follows (see Figure 1). It maintains weights W_e^t over the examples $e \in \mathcal{X}$ for each round t. When a new hypothesis h comes from the Learner (so it is not covered by the examples thus far), the Teacher first verifies whether $h = h^*$. If so, it accepts h. Otherwise, it increases in exponential fashion the weights of the examples where h is wrong until the sum of these weights becomes at least 1; then it randomly sends examples to Learner with probability proportional to the increase of the weights of the examples in this round. If h is neither accepted nor covered (i.e., no example is sent) by the end of the round, the algorithm returns FAIL.

While our algorithm is based on (Alon et al., 2009) the main novelty is in its analysis: the hypotheses ("elements" to be covered) depend on the examples ("sets") sent, and not only the analysis in (Alon et al., 2009) does not allow such dependencies but it also known that m examples are required for more general dependencies [(Korman, 2004), Theorem 2.1.3]. However, the dependencies that arise in this context of teaching are just so that we can handle them using martingale techniques.

Theorem 1. *Consider teaching a worst-case learner. In the realizable case $h^* \in \mathcal{H}$, algorithm $\mathcal{A}_{base}$ (in Fig. 1) initialized with $N \geq n$ and $\omega = m$ always sends $O(\mathcal{TS}\log N \log m)$ examples, and returns the correct hypothesis h^* with probability at least $1 - \frac{1}{N}$.*

Proof of Theorem 1. The first important observation is that the algorithm terminates in at most $O(\mathcal{TS}\log\omega)$

Algorithm $\mathcal{A}_{\text{base}}$

Input: Examples $\mathcal{X}$, (guess of) the number of Learner's hypotheses N, initial weight_ parameter $\omega \geq 0$

1. Initialize weights $W_e^0 = \frac{1}{2\omega}$ for all examples $e \in \mathcal{X}$
2. For each round $t = 1, 2, \ldots$:
   - Receive hypothesis $H_t \in \mathcal{H}$ from the Learner
   - If H_t is correct in all examples (i.e., $H_t = h^*$), stop and return H_t
   - **(Weight update)** Double the weights of all wrong examples until their weight adds up to at least 1. That is, define

$$W_e^t = \begin{cases} 2^\ell \cdot W_e^{t-1} & \text{, if } e \in \text{wrong}(H_t) \\ W_e^{t-1} & \text{, if } e \notin \text{wrong}(H_t), \end{cases}$$

     where ℓ is the smallest non-negative integer such that $W^t(H_t) := \sum_{e \in wrong(H_t)} W_e^t \geq 1$
   - **(Sending examples)** For every example e, let $D_e^t := W_e^t - W_e^{t-1}$ be the weight increase of example e (note $D_e^t = 0$ if H_t is not wrong on e)

     Repeat $4 \log N$ times: sample at most one example so that e is sampled with probability D_e^t, and send it to Learner together with its correct label (note that H_t is wrong on this example)
   - If no examples were sent, return FAIL

Figure 1. Teacher's algorithm for teaching a realizable hypothesis

rounds [(Alon et al., 2009), Lemma 1]. In addition, since in each round the Teacher sends at most $O(\log N)$ examples we get the following.

Lemma 1. *$\mathcal{A}_{base}$ sends $O(\mathcal{TS} \log \omega \log N)$ examples.*

So we only need to upper bound the probability that the algorithm returns FAIL. Let $W^t(h) := \sum_{e \in \text{wrong}(h)} W_e^t$ be the weight of h at the end of round t, and let $D^t(h) := W^t(h) - W^{t-1}(h)$ be the increase of this weight at round t. The intuition why the failure probability should be low is the following: If the algorithm fails on a hypothesis h it means that its weight is at least 1 by the end of the interaction and no example that covers h was sent. Let X_t be an indicator variable that is equal to 0 if no examples that cover h are sent at round t. We have $Pr[X_t = 0] = (1 - D^t(h))^{4 \log N}$, so the failure probability *should be* about $\prod_t (1 - D^t(h))^{4 \log N} \approx e^{-(4 \log N) \sum_t D^t(h)} \leq e^{-2 \log N} = 1/N^2$, where the last inequality holds because, at the beginning, the weight of h is at most $1/2$ and by the end of the algorithm is at least 1. By taking the union bound over all hypotheses $h \in \mathcal{H}$ we would conclude that the failure probability is at most $1/N$.

The problem is that this argument ignores crucial stochastic dependencies: the actual examples sent affect (through the Learner's response) the evolution of the weight of h, so that the set of random variables $\{X_t\}$ are not independent and, hence, we cannot take the product of probabilities as above. To handle this situation we abstract it as a sequence of dependent Bernoulli random variables X_t's whose biases (corresponding to $1 - (1 - D^t(h))^{4 \log n}$) depend on the history. Our main technical lemma shows that, regardless of the correlations, the probability that none of the indicators X_t is active is what we expect.

Lemma 2 (Adaptive Bernoullis)**.** *Consider a finite probability space with filtration $\mathcal{F}_1 \subseteq \mathcal{F}_2 \subseteq \ldots$ and let $X^1, \ldots, X^n \in \{0, 1\}$ be an adapted sequence of Bernoulli random variables, possibly correlated. Let $Z^t := \Pr(X^t = 0 \mid \mathcal{F}_{t-1})$ be the conditional probability that X_t is 0. Then for any stopping time τ w.r.t. $\mathcal{F}$ and $\alpha \geq 0$*

$$\Pr\left(X^1 = \ldots = X^\tau = 0 \text{ and } \prod_{t \leq \tau} Z^t \leq \alpha\right) \leq \alpha.$$

Proof. We can assume without loss of generality that there is no stopping time (i.e., τ always equals n): we can apply the result to the variables $\tilde{X}^t := \mathbf{1}(\tau \geq t) \cdot X^t$ and $\tilde{B}^t := (1 - \mathbf{1}(\tau \geq t)) \cdot Z^t = \Pr(\tilde{X}^t = 0 \mid \mathcal{F}_{t-1})$ to obtain the result in the stopped case.

In this specific proof we use bold for vectors and capital letters for random variables, respectively. Moreover, for a vector $\mathbf{v} = (v^1, \ldots, v^n)$ and $t \leq n$, we use $\mathbf{v}^{\leq t}$ (resp. $\mathbf{v}^{<t}$) to denote the vector $(v^1, \ldots, v^t)$ (resp. $(v^1, \ldots, v^{t-1})$). The same notation is employed to restrict a sequence of random variables to its t first elements.

Let $X = (X^1, X^2, \ldots, X^n)$ and let $Z = (Z^1, Z^2, \ldots, Z^n)$. Peeling off the variables in order $Z^1, X^1, Z^2, X^2, \ldots$ we have that for any fixing $\mathbf{z} = (z^0, z^1, \ldots, z^n)$ (without any independence assumption)

$$\Pr\left(X = 0 \text{ and } Z = \mathbf{z}\right) = \prod_{t=1}^{n} f(t, \mathbf{z}) \cdot \prod_{t=1}^{n} g(t, \mathbf{z})$$

where

$$\begin{aligned} f(t,\mathbf{z}) &= \Pr(X^t = 0 \mid Z^{\leq t} = \mathbf{z}^{\leq t}, X^{<t} = 0), \\ g(t,\mathbf{z}) &= \Pr(Z^t = z^t \mid Z^{<t} = \mathbf{z}^{<t}, X^{<t} = 0). \end{aligned}$$

For any history $\sigma \in \mathcal{F}_{t-1}$ up to time $t-1$ where $Z^t = z^t$ we have $\Pr(X^t = 0 \mid \sigma) = z^t$, which implies $\Pr(X^t = 0 \mid Z^{\leq t} = \mathbf{z}^{\leq t}, X^{<t} = 0) = z^t$. So we obtain $\Pr(X = 0 \text{ and } Z = \mathbf{z}) = prod(\mathbf{z}) \cdot \prod_{t=1}^{n} z^t$, where

$$prod(\mathbf{z}) = \prod_{t=1}^{n} g(t,\mathbf{z}).$$

Letting Ω be the set of all $\mathbf{z}$'s such that $\prod_{t=1}^{n} z^t \leq \alpha$ we have

$$\begin{aligned} \Pr\left(X = 0 \text{ and } \prod_{t=1}^{n} Z^t \leq \alpha\right) &\leq \alpha \sum_{\mathbf{z}\in\Omega} prod(\mathbf{z}) \\ &\leq \alpha \sum_{z^1} \cdots \sum_{z^n} prod(\mathbf{z}), \end{aligned}$$

where the sum $\sum_{z^t}$ ranges over all possible values of Z^t (recall that we assumed the probability space to be finite). Finally, we claim that the sum in the RHS equals 1: by using the definition of $prod(\mathbf{z})$ we get that

$$\begin{aligned} &\sum_{z^1} \cdots \sum_{z^n} prod(\mathbf{z}) \\ &\quad = \sum_{z^1} \cdots \sum_{z^n} prod(\mathbf{z}^{<n}) \cdot g(n,\mathbf{z}) \\ &\quad = \sum_{z^1} \cdots \sum_{z^{n-1}} prod(\mathbf{z}^{<n}) \left(\sum_{z^n} g(n,\mathbf{z})\right) \\ &\quad = \sum_{z^1} \cdots \sum_{z^{n-1}} prod(\mathbf{z}^{<n}), \end{aligned}$$

Iterating this argument $n-1$ times gives that

$$\sum_{z^1} \cdots \sum_{z^n} prod(\mathbf{z}) = \sum_{z^1} \Pr(Z^1 = z^1) = 1,$$

which concludes the proof. □

With this we can bound the failure probability of the algorithm, concluding its analysis.

Lemma 3. *$\mathcal{A}_{base}$ with $\omega \geq m$ and $N \geq n$ returns* FAIL *with probability at most $\frac{1}{N}$.*

Proof. Fix a hypothesis $h \in \mathcal{H}$. By taking a union bound over all hypotheses, it suffices to show that the probability that $\mathcal{A}_{\text{base}}$ fails upon receiving hypothesis h is at most $\frac{1}{N^2}$.

Notice that h is received at most once by $\mathcal{A}_{\text{base}}$: after receiving h for the first time, the algorithm either sends an example that covers h (so the Learner never resends h) or returns FAIL. So let τ be the time when h is received, i.e., $H_\tau = h$ (let τ be the last round if h is never received), and let X^t be the indicator that at time t an example covering h was sent to Learner by the algorithm. As mentioned before, by the weight update of the algorithm, at round τ we have weight $W^\tau(h) = W^\tau(H_\tau) \geq 1$. Since $\omega \geq m$, the initial weights satisfies $W^0(h) \leq \frac{1}{2}$ and hence the weight increments up to round τ satisfy $\sum_{t\leq\tau} D^t(h) \geq \frac{1}{2}$. Moreover, if the algorithm fails on h we have $X^1 = \ldots = X^\tau = 0$, thus

$$\Pr(\text{fails on } h) \leq \Pr\left(X^t = 0, \forall t \leq \tau, \text{ and } \sum_{t\leq\tau} D^t(h) \geq \frac{1}{2}\right). \tag{1}$$

Let $\mathcal{F}_{t-1}$ be the σ-algebra generated by the history up to round $t-1$ plus the hypothesis at round t. By the sampling procedure, the conditional probability $Z^t := \Pr(X^t = 0 \mid \mathcal{F}_{t-1})$ that no example covering h was added in round t is

$$Z^t = (1 - D^t(h))^{4\log N} \leq e^{-4(\log N) D^t(h)}, \tag{2}$$

recalling that $(1-x) \leq e^{-x}$ for all x. Then $\sum_{t\leq\tau} D^t(h) \geq \frac{1}{2}$ implies $\prod_{t\leq\tau} Z^t \leq e^{-2\log N} = \frac{1}{N^2}$, and the RHS of (1) can be upper bounded

$$\Pr(\text{fails on } h) \leq \Pr\left(X^t = 0, \forall t \leq \tau \text{ and } \prod_{t\leq\tau} Z^t \leq \frac{1}{N^2}\right).$$

From Lemma 2 this can be further upper bounded by just $\frac{1}{N^2}$, and hence $\Pr(\text{fails on } h) \leq \frac{1}{N^2}$. This concludes the proof. □

Making the algorithm agnostic to the size of $\mathcal{H}$. In Theorem 1 we assume that the Teacher knows the number of hypotheses $n = |\mathcal{H}|$ of Learner. However, a guess and double strategy can be used to overcome this limitation. More concretely, let $\mathcal{A}_{\text{agno}}$ be a teaching algorithm that implements a sequence of calls to $\mathcal{A}_{\text{base}}$, where in the i-th call the parameter N is set to 2^{2^i} while ω is set to m. Moreover, for $i > 1$, the initial hypothesis for the i-th call is the one that failed in the call $i-1$. The procedure ends as soon as some call to $\mathcal{A}_{\text{base}}$ accepts h^*.

Let $t = \lceil \log\log n \rceil$. At the t-th call of $\mathcal{A}_{\text{base}}$ the parameter N is not smaller than n. Thus, it follows from Theorem 1 that this call sends $O(\mathcal{TS} \cdot \log m \cdot 2^t)$ examples and returns h^* with probability at least $1 - 1/n$. Moreover, by Lemma 1 the previous calls to $\mathcal{A}_{\text{base}}$ send $\sum_{i=1}^{t-1} O(\mathcal{TS} \log m \cdot 2^i) = O(\mathcal{TS} \log m \log n)$ examples. Therefore, we have the following result.

Theorem 2. *Consider teaching a worst-case learner in the realizable case $h^* \in \mathcal{H}$. The algorithm $\mathcal{A}_{agno}$, with probability at least $1 - \frac{1}{n}$, returns the correct hypothesis h^* and sends at most $O(\mathcal{TS} \log m \log n)$ examples.*

2.2. Improved Guarantee Based on the Quality of the Hypotheses

The next theorem shows that it is possible to obtain an improved bound when the distribution of the number of errors of the hypotheses in $\mathcal{H}$ is taken into account. For instance if only $O(1)$ hypotheses make a non-constant number of errors, then the bound on the number of examples sent is improved from $O(\mathcal{TS} \log m \log n)$ to $O(\mathcal{TS}(\log n + \log m))$.

Theorem 3. *Consider teaching a worst-case learner in the realizable case $h^* \in \mathcal{H}$. Let n_i be the number of hypotheses in $\mathcal{H}$ whose number of errors is between $[2^{2^i}, 2^{2^{i+1}})$ for $i \geq 1$, and let n_0 be the number of hypotheses with error in $[1, 4)$. Then there is an algorithm for the Teacher that with probability at least $\frac{4}{5}$ returns a correct hypothesis h^* and the number of examples sent is*

$$O(\mathcal{TS}(\mathcal{H})) \cdot \left(\log m + \sum_{i=0}^{\log \log m} 2^i \log(n_i + 1) \right)$$

In particular, this is $O(\mathcal{TS}(\mathcal{H}) \log m \log(\max_i n_i + 1))$.

The starting point is to notice that if we run $\mathcal{A}_{\text{base}}$ initialized with ω being the maximum number of errors of a hypothesis in the class, say err, then it sends at most $O(\mathcal{TS} \log err \cdot \log n)$ examples. Then the idea of the algorithm of Theorem 3 is the following: Instantiate copies $\mathcal{A}_1, \ldots, \mathcal{A}_{\log \log m}$ of the algorithm $\mathcal{A}_{\text{base}}$, where $\mathcal{A}_i$ is initialized with $\omega = 2^{2^i}$. Then when a hypothesis h comes from Learner, we see in which bucket $[2^{2^i}, 2^{2^{i+1}})$ its number of errors falls into, and send the hypothesis to algorithm $\mathcal{A}_i$.

Since the size of smallest teaching set for the hypotheses that fall in the same bucket is no larger than $\mathcal{TS}(\mathcal{H}, h^*)$, we can compose the guarantees from the algorithms $\mathcal{A}_i$'s to get the above guarantee (Details in Appendix B).

2.3. Non-realizable Case

We now consider the non-realizable case where the correct hypothesis h^* may not be in the Learner's class $\mathcal{H}$. Recall that in this case in each round the Learner sends a hypothesis in $\mathcal{H}$ with the smallest number of errors in the examples received so far, and the Teacher's goal is to make the Learner return a hypothesis in $\mathcal{H}$ with the smallest number of total errors over the whole set of examples $\mathcal{X}$.

We first need a generalization of the notion of teaching set. Informally, if the best hypothesis in $\mathcal{H}$ has k errors in $\mathcal{X}$, to isolate it the Teacher should send examples that certify that the other hypotheses have at least $k+1$ errors. We say that a set of examples $\mathcal{X}' \subseteq \mathcal{X}$ is a *k-extended teaching set* with respect to h^* if for each hypothesis $h \in \mathcal{H}$ with more than k errors there are at least $k+1$ examples in $\mathcal{X}'$ where h is wrong (differs from h^*). We let $\mathcal{TS}_k = \mathcal{TS}_k(\mathcal{H}, h^*)$ denote the size of the smallest k-extended teaching set w.r.t. h^*.

Notice that after the Learner receives a set of labeled examples that contain a k-extended teaching set, it returns a hypothesis with at most k total errors since such hypotheses have at most k errors in the examples received, while all other hypotheses have at least $k+1$ errors in them. If k is set to be the number of errors of the best hypothesis in $\mathcal{H}$, the Learner then returns an optimal hypothesis.

Theorem 4. *Consider teaching a worst-case learner where h^* may not belong to $\mathcal{H}$. Let k be the smallest number of errors of a hypothesis in $\mathcal{H}$. Then there is a Teacher's algorithm that with probability at least $1 - \frac{1}{m}$ returns a hypothesis that makes k errors and sends at most $O(\mathcal{TS}_k \log m \log(m+n))$ examples.*

The high-level idea of the algorithm is the same as in the realizable case: it tries to compute in an online fashion a k-extended teaching set of small size, but now based on an *online generalized set cover* algorithm (Buchbinder & Naor, 2009), where elements may need to be covered multiple times. However, since the minimum number of errors k is unknown, the algorithm also needs to keep a lower bound on k that is given by the number of errors of the last received hypothesis over the examples already sent. The algorithm stops when it receives from the Learner a hypothesis whose total number of errors matches this lower bound. Details are provided in Appendix C.

3. Other Learner Models

In this section we show that better bounds are possible under reasonable assumptions on the way the Learner can choose the consistent hypotheses to return. We address the realizable case where $h^* \in \mathcal{H}$ and, here, we use $\mathcal{H}_t \subseteq \mathcal{H}$ to denote the set of hypotheses consistent with all the examples sent by the Teacher in the rounds $1, \ldots, t-1$ (so $\mathcal{H}_t$ is the set of possible hypotheses that Learner can send in this round t).

3.1. Smooth Transition Learners

We first consider the *smooth transition model* where we further assume that the Learner sends a hypothesis "close" to the one sent in the previous round. Concretely, we use the number of disagreements between hypotheses $d(h, h') = |\{x \in \mathcal{X} \mid h(x) \neq h'(x)\}|$ as measure of closeness, and assume the hypothesis h_t that Learner sends at round t is one in $\mathcal{H}_t$ with $(1+\alpha)$-approximate minimum distance to the hypothesis h_{t-1} sent in the previous round, namely

$$d(h_t, h_{t-1}) \leq (1+\alpha) \min_{h \in \mathcal{H}_t} d(h, h_{t-1}).$$

We provide an algorithm $\mathcal{A}^{\alpha}_{\text{close}}$ for this model whose guar-

antee depends on the number of errors err_1 of the first hypothesis sent by the Learner. That means that if the Learner has a good guess for the right hypothesis, fewer examples are needed to complete the teaching. Algorithm $\mathcal{A}^{\alpha}_{\text{close}}$ is obtained from $\mathcal{A}_{\text{base}}$ via two simple modifications: The starting weights of the examples W^0_e are set to $1/2err_1$ instead of $1/2m$, and the number of examples sampled per round is $\frac{8}{1-2\alpha}\log N$ instead of $4\log N$. This algorithm has the following guarantee.

Theorem 5. *Consider the smooth transition model with $\alpha \in [0, \frac{1}{2})$ in the realizable case $h^* \in \mathcal{H}$. Let err_1 be the number of errors over $\mathcal{X}$ of the initial hypothesis sent by Learner. Then algorithm $\mathcal{A}^{\alpha}_{close}$ set with $N \geq n$ sends $O(\mathcal{TS}\frac{1}{1-2\alpha}\log err_1 \log N)$ examples and with probability at least $1 - \frac{1}{N}$ returns the correct hypothesis h^*.*

The main observation for obtaining a guarantee that depends on err_1 is that in the smooth transition model the number of errors of the hypotheses sent by the Learner cannot increase rapidly. We have the following.

Lemma 4. *Let h, h' be the hypotheses returned by Learner at rounds $t-1$ and t respectively. Then:*

a) $|wrong(h')| \leq 2\,|wrong(h') \cap wrong(h)| + \alpha\,|wrong(h)$

b) $|wrong(h')| \leq (4 + 2\alpha)\,err_1$.

Proof. We first prove item (a). Let $\text{wrong}(h \setminus h')$ (resp. $\text{wrong}(h' \setminus h)$) be the set of examples where only h (resp. h') is wrong. In addition, let $DIFF$ (resp. EQ) be the number of examples that both h and h' are wrong but give give different (resp. equal) classification. In formulae,

$$DIFF = \{e \in \text{wrong}(h) \cap \text{wrong}(h') \mid h(e) \neq h'(e)\}$$
$$EQ = \{e \in \text{wrong}(h) \cap \text{wrong}(h') \mid h(e) = h'(e)\}.$$

The number of disagreements between these hypotheses is

$$d(h, h') = |\text{wrong}(h \setminus h')| + |\text{wrong}(h' \setminus h)| + DIFF$$
$$d(h, h^*) = |\text{wrong}(h \setminus h')| + DIFF + EQ$$

The smooth transition model guarantees that $d(h, h') \leq (1 + \alpha)\,d(h, h^*)$ so that $|\text{wrong}(h' \setminus h)| \leq \alpha\,|\text{wrong}(h' \setminus h)| + (1 + \alpha)\,|\text{wrong}(h) \cap \text{wrong}(h')|$, and hence

$$\begin{aligned}|\text{wrong}(h')| &= |\text{wrong}(h' \setminus h)| + |\text{wrong}(h) \cap \text{wrong}(h')| \\ &\leq \alpha\,|\text{wrong}(h \setminus h')| \\ &\quad + (2 + \alpha)\,|\text{wrong}(h) \cap \text{wrong}(h')| \\ &= \alpha\,|\text{wrong}(h)| + 2|\text{wrong}(h) \cap \text{wrong}(h')|,\end{aligned}$$

which establishes item (a).

Proof of item [b]. If h' is the first hypothesis, the result clearly holds because the first hypothesis makes err_1 mistakes.

Thus, let $t > 1$ be the round in which h' is received and let h be hypothesis received at the round $t - 1$. We have that $|\text{wrong}(h)| < 2err_1$, for otherwise h would have weight at least 1 by the beginning of round $t - 1$ and, hence, $t - 1$ would be the last round of the interaction. Thus, it follows from item (a) and from $|\text{wrong}(h) \cap \text{wrong}(h') \leq |\text{wrong}(h)|$ that $|\text{wrong}(h')| \leq (4 + 2\alpha)err_1$. □

Proof of Theorem 5. The bound on the number of examples follows directly from Lemma 1, since $\mathcal{A}^{\alpha}_{\text{close}}$ behaves as $\mathcal{A}_{\text{base}}$ initialized with $\omega = err_1$, but sending $\frac{2}{1-\alpha}$ times as many examples per round.

The proof that the probability of returning the correct hypothesis is at least $1 - \frac{1}{N}$ is similar to that of Lemma 3: we need to show that if the algorithm receives hypothesis h' on round τ then $\sum_{t \leq \tau} D^t(h')$, the total increase of the weights of the wrong examples of a hypothesis h' is "large". That is enough since the concentration arguments following inequality (1), then, guarantee that the probability of failing at this point (i.e., no examples covering h' were sent) is small. More precisely, it suffices to show

$$\sum_{t \leq \tau} D^t(h') \geq \frac{1 - 2\alpha}{4}. \tag{3}$$

We note that in Lemma 3 we have the stronger lower bound with RHS $\frac{1}{2}$; the difference is compensated by the extra number of examples sent in each round by $\mathcal{A}^{\alpha}_{\text{close}}$. We prove inequality (3) by considering two cases:

Case 1. $|\text{wrong}(h')| \leq err_1$. In this case $\sum_{t \leq \tau} D^t(h')$ is at least $\frac{1}{2} > \frac{1-2\alpha}{4}$ since the initial weight of h' is at most $\frac{1}{2}$ and its final weight is at least 1 by the weight update step of the algorithm.

Case 2. $|\text{wrong}(h')| > err_1$. It follows from item (a) of Lemma 4 that the hypothesis received at round $\tau - 1$, say h, shares at least $(|\text{wrong}(h')| - \alpha\,|\text{wrong}(h)|)/2$ wrong examples with h'. Moreover, we have $|\text{wrong}(h)| \leq 2err_1$: otherwise the starting weight of this hypothesis $W^0(h)$ is already at least 1, so the algorithm fails on round $\tau - 1$, contradicting that it fails on round τ. Together with the assumption from being in Case 2, this gives $|\text{wrong}(h)| \leq 2|\text{wrong}(h')|$ and hence the number of common wrong examples between h and h' is at least $\frac{1-2\alpha}{2}\,|\text{wrong}(h')|$. Since the weight of each of these common examples was increased by at least $1/2err_1$ in round $\tau - 1$ (weights are doubled and start at $1/2err_1$), we have that

$$\sum_{t \leq \tau} D^t(h') \geq D^{\tau-1}(h') \geq \frac{1 - 2\alpha}{2} \cdot err_1 \cdot \frac{1}{2err_1} = \frac{1 - 2\alpha}{4}.$$

This proves (3) and concludes the proof of Theorem 5.

Making the algorithm agnostic to the size of $\mathcal{H}$. To obtain an algorithm agnostic to the size of $\mathcal{H}$ we proceed as in Section 2.1 but performing a sequence of calls to $\mathcal{A}^{\alpha}_{\text{close}}$, rather than to $\mathcal{A}_{\text{base}}$.

The only different issue that arises in the analysis of this algorithm is how to bound the number of errors $err_{1,i}$ made by the first hypothesis of ith call of $\mathcal{A}^{\alpha}_{\text{close}}$. By using the fact that this hypothesis is exactly the last one returned by the previous call together with item (b) of Lemma 4, we get that $err_{1,i} \leq (4+2\alpha)\, err_{1,i-1}$ and, hence, $err_{1,i} \leq (4+2\alpha)^{i-1}\, err_{1,1}$. This observation together with the same arguments employed in the analysis of $\mathcal{A}_{\text{agno}}$ allows us to establish the following theorem:

Theorem 6. *Under the same assumptions as in Theorem 5, there is a teacher's algorithm agnostic to the number of hypothesis n that sends $O(\mathcal{TS}\ \log n\,(\log err_1 + \log\log n))$ examples and with probability at least $1-\frac{1}{n}$ returns the correct hypothesis h^*.*

Note that in the worst-case learner model this bound is not achievable by poly-time algorithms unless $\mathcal{NP} \subset \mathcal{BPP}$. This is shown through a simple modification of a lower bound from (Korman, 2004) [See Appendix D]

3.2. The Random Learner Model

We assume that in each round the Learner sends a batch of *random* i.i.d. hypotheses from the ones that are consistent with the examples received thus far.

We show that in this situation the Teacher can exploit the randomness of the Learner's choice in order to estimate the example that covers (i.e., falsifies) the highest number of hypotheses (which are consistent with the examples seen so far). With this knowledge, the Teacher can resort to algorithms for the *offline* set cover problem and significantly improve the number of examples used: we show that Teacher sends with high probability $O(\mathcal{TS}\ \log(n+m))$ examples, which is the best bound achievable in polytime, under the assumption that $\mathcal{P} \neq \mathcal{NP}$, for the relevant case where the number of hypotheses n is larger than the number of examples m (Raz & Safra, 1997).

Algorithm $\mathcal{A}_{\text{rand}}$ (Figure 2) runs the greedy approximation algorithm for offline set cover over the empirical process: At each round t the Teacher requests a batch $\widetilde{\mathcal{H}}_t$ of T random hypotheses from the set $\mathcal{H}_t$ of hypotheses consistent with the examples sent thus far, and sends to Learner an example $\tilde{e}$ that covers the largest number of hypotheses from $\widetilde{\mathcal{H}}_t$. The size T of the requested batch ideally depends on the size of the smallest teaching set $\mathcal{TS}(\mathcal{H}, h^*)$, but since this quantity is unknown the algorithm also employs a guess-and-double approach: In phase i it uses $T = 2^i$ as a guess for $\mathcal{TS}$ and runs the greedy procedure for $2T$ rounds. If within these rounds the algorithm does not terminate, the phase is concluded and phase $i+1$ is started.

Algorithm $\mathcal{A}_{\text{rand}}$
Input: Examples $\mathcal{X}$

1. Initialize round counter $t = 0$
2. For each phase $i = 1, 2, \ldots$:
   - Update the teaching set size guess $T = 2^i$
   - For $2T$ rounds
     - Update round counter $t = t + 1$
     - Receive batch $\widetilde{\mathcal{H}}_t$ of T random hypotheses from $\mathcal{H}_t$
     - If all hypotheses in $\widetilde{\mathcal{H}}_t$ equal h^*, then **Return** **(*)**
     - Send $\tilde{e} = \text{argmax}_e\, |\{h \in \widetilde{\mathcal{H}}_t \mid e \in \text{wrong}(h)\}|$

Figure 2. Teacher $\mathcal{A}_{\text{rand}}$

The following theorem is the main result of this section.

Theorem 7. *Consider the random learner model in the realizable case $h^* \in \mathcal{H}$. With probability at least $1 - O(1/m)$, $\mathcal{A}_{rand}$ satisfies the following: (i) it accepts the target hypothesis h^* (line **(*)** of $\mathcal{A}_{rand}$); (ii) it sends $O(\mathcal{TS} \cdot \log(n+m))$ examples and (iii) it receives $O(\mathcal{TS} \cdot \log(n+m))$ hypotheses per round.*

We note that without the bound on the number of hypotheses received per round the result would be straightforward since the Teacher could request infinitely many hypotheses to acquire a very accurate knowledge of the class $\mathcal{H}$ and then resort to the off-line greedy set cover algorithm.

To prove Theorem 7, given a set of hypotheses $\mathcal{H}' \subseteq \mathcal{H}$ let $c^*(\mathcal{H}')$ be the maximum number of hypotheses from $\mathcal{H}'$ that can be covered by a single example in $\mathcal{X}$. We rely on two lemmas whose proofs can be found in Appendix E. The first shows that with high probability the example that covers the largest number of hypotheses in the empirical set $\widetilde{\mathcal{H}}_t$ also covers a large number of hypotheses from $\mathcal{H}_t$.

Lemma 5. *Let $\tilde{e}$ be an example that covers the largest number of hypotheses in $\widetilde{\mathcal{H}}_t$ and let $c_{\tilde{e}}$ be the number of hypotheses* covered by $\tilde{e}$ *in $\mathcal{H}_t$. If $|\widetilde{\mathcal{H}}_t| \geq 40\mathcal{TS}\ln m$ then,*

$$\Pr\left(c_{\tilde{e}} \leq \frac{1}{12} c^*(\mathcal{H}_t)\right) \leq \frac{2}{m^4}.$$

The second lemma gives an upper bound on the number of rounds executed by an approximate version of the greedy offline set cover algorithm.

Lemma 6. *Consider $0 < \alpha < 1$ and $\mathcal{A}_\alpha$ be any teacher's algorithm that at each round t sends to Learner an example that covers at least $\alpha \cdot c^*(\mathcal{H}_t)$ hypotheses from $\mathcal{H}_t$. Then $\mathcal{A}_\alpha$ executes $O(\frac{1}{\alpha}\mathcal{TS}\ln n) = O(\mathcal{TS}\ln n)$ rounds before the only consistent hypothesis is h^*, i.e., $\mathcal{H}_t = \{h^*\}$.*

Proof of Theorem 7. Let $\alpha = 1/12$ and let $\hat{i}$ be the first phase where $T = 2^{\hat{i}}$ is at least $\mathcal{TS} \cdot \max\{\frac{1}{\alpha}\ln n, 40 \ln m\}$.

Since the previous $\hat{i}-1$ phases send $\sum_{j=1}^{\hat{i}-1} 2^j = O(\mathcal{TS}\ln(m+n))$ examples and each of them requests $O(\mathcal{TS}\ln(n+m))$ hypotheses per round, it suffices to analyse phase $\hat{i}$ onwards.

We say that an example is *bad* for round t if it does not cover $\alpha c^*(\mathcal{H}_t)$ hypothesis of $\mathcal{H}_t$. Due to Lemma 5 every round t that occurs after the beginning of phase $\hat{i}$ sends a bad example with probability at most $2/m^4$. Thus, it follows from the union bound that a bad example is sent during the $\frac{1}{\alpha}\mathcal{TS}\log n$ first rounds of phase $\hat{i}$ with probability at most $(2\frac{1}{\alpha}\mathcal{TS}\log n)/m^4 \leq 24/m$, where the inequality holds because $\mathcal{TS} \leq m$ and $n \leq |\mathcal{Y}|^m \leq m^m$. Hence, it follows by Lemma 6 that, with probability at least $1-O(1/m)$, the only consistent hypothesis that remains after these round is h^*. Since each of these rounds requests $O(\mathcal{TS}\log(n+m))$ hypotheses, the theorem is proved. □

4. Non-redundant Teaching Sets

We say that a teaching set X is *redundant* if it contains a redundant example, that is, an example e such that $X \setminus \{e\}$ is still a teaching set. The algorithms discussed so far may construct teaching sets that are redundant. We show that $|X| \cdot (|\mathcal{Y}|-1)$ additional rounds suffice to obtain a non-redundant teaching set from a teaching set X w.r.t. $(h^*, \mathcal{H})$, where $\mathcal{Y}$ is the set of possible labels for the examples.

For that we consider a more general interaction model where at each round Teacher sends a set of labelled examples to Learner and the latter returns a hypothesis that makes the smallest number of errors in this set (ignoring the examples received at previous rounds). Differently from the previous sections, here the Teacher may send an example e with label different from $h^*(e)$.

The following proposition gives a simple condition for deciding whether e is redundant for teaching set X.

Proposition 1. *Let X be a teaching set for $(\mathcal{H}, h^*)$. If there exists a hypothesis $h \in \mathcal{H}$ with $h(e') = h^*(e')$ for every $e' \in X \setminus \{e\}$ and $h(e) \neq h^*(e)$, then the example e is non-redundant for the set X, and also for any teaching set contained in X. Otherwise, e is redundant for X.*

Given this observation, the algorithm for obtaining a non-redundant teaching set from X is straightforward: It scans the examples in X and for each $e \in X$ verifies whether e is redundant (w.r.t. the set of examples that have not been removed from X) or not; if it is, the example is removed from X and the scan continues over the examples that have not been tested yet. To verify whether e is redundant the Teacher interacts with the Learner in $|\mathcal{Y}|-1$ rounds testing the existence of a label $y \neq h^*(e)$ for which the Learner returns a hypothesis consistent with the labelled set of examples $\mathcal{D}_y = \{(e', h^*(e')) \mid e' \in X \setminus e\} \cup \{(e, y)\}$.

Table 1. Percentage of the size of the full dataset required by each Teacher-Learner to achieve an accuracy larger than $z\%$ (with $z \in \{90, 95, 99\}$) of that achieved by the Learner when it is trained and tested in the full dataset.

Teacher-Learner	90%	95%	99%
$\mathcal{A}_{\text{agno}}$-Random Forest	10.1%	14.7%	20.6 %
NIT-Random Forest	14.7%	30.4%	59.9%
$\mathcal{A}_{\text{agno}}$-LGBM	2.8 %	5.7 %	8.9 %
NIT-LGBM	3.4 %	7.8 %	28.0 %

If such label does not exist then the algorithm concludes that e is redundant. It is clear that the overall number of rounds is at most $|X| \cdot (|\mathcal{Y}|-1)$.

In simulations on synthetic data (Appendix F) this method significantly reduced the teaching sets found by $\mathcal{A}_{\text{agno}}$.

5. Computational Experiments

Although our work is mainly theoretical we performed experiments to understand how our Teacher $\mathcal{A}_{\text{agno}}$ compares with a non interactive one over real datasets.

The non interactive teacher, denoted by NIT, receives an integer ℓ and then sends to the Learner ℓ randomly selected examples. We compared the number of examples that both $\mathcal{A}_{\text{agno}}$ and NIT need to send in order to attain a certain level of accuracy. For our evaluation we used Random Forest and Light Gradient Boosting Machine (LGBM) as learners, and conducted experiments on 12 datasets: mnist and 11 others from the UCI repository (mushroom, avila, bank_marketing, car, Credit Card, Firm_Teacher, crowdsourced, Electrical_grid, HTRU, nursery and Sensorless_drive).

Table 1 shows the number of examples (relative to the size of the full dataset) required by each pair Teacher-Learner to attain an accuracy over the full dataset larger than $z\%$ of that obtained when the Learner is trained/tested over the full dataset. Each numeric entry of this table is an average of 12 values, where each of them corresponds to a distinct dataset. More details are presented in Appendix G.

The results provide evidence that $\mathcal{A}_{\text{agno}}$ requires significantly fewer examples than NIT (e.g. a factor of 3 for the target 99%). Furthermore, an interesting observation is that the advantage of $\mathcal{A}_{\text{agno}}$ increases as the level of accuracy requested gets higher. A reasonable explanation is that when little is known (low accuracy in our setting), most of the examples are useful (hence, random sampling also works well); however, when a certain level of knowledge has already been reached, more specific examples, as those provided by $\mathcal{A}_{\text{agno}}$, are needed to further increase it.

Acknowledgements

We would like to thank the anonymous reviewers for their useful comments including also some pointers to very recent related literature. We would also like to thank Daniel dos Santos Marques for the fruitful discussions about applications of machine teaching and to Sanjoy Dasgupta for clarifications about the paper "Teaching a black-box learner".

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

The work of the third author is partially supported by CNPq under grant 307572/2017- 0 and by FAPERJ, grant Cientista do Nosso Estado E- 26/202.823/2018.

The work of the of the fourth author is supported by CNPq grants Universal #431480/2016-8 and Bolsa de Produtividade 35 em Pesquisa #4310516/2017-0, FAPERJ grant Jovem Cientista do Nosso Estado.

References

Alon, N., Awerbuch, B., Azar, Y., Buchbinder, N., and Naor, J. The online set cover problem. *SIAM J. Comput*, 39(2):361–370, 2009.

Angluin, D. and Dohrn, T. The power of random counterexamples. *Theor. Comput. Sci.*, 808:2–13, 2020.

Balbach, F. J. and Zeugmann, T. Teaching randomized learners with feedback. *Inf. Comput*, 209(3):296–319, 2011.

Buchbinder, N. and Naor, J. S. Online primal-dual algorithms for covering and packing. *Math. Oper. Res.*, 34(2):270–286, May 2009. ISSN 0364-765X. doi: 10.1287/moor.1080.0363. URL https://doi.org/10.1287/moor.1080.0363.

Chen, Y., Singla, A., Mac Aodha, O., Perona, P., and Yue, Y. Understanding the role of adaptivity in machine teaching: The case of version space learners. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 1476–1486. 2018.

Dasgupta, S., Hsu, D., Poulis, S., and Zhu, X. Teaching a black-box learner. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1547–1555. PMLR, 2019. URL http://proceedings.mlr.press/v97/.

de la Peña, V. H. and Giné, E. *Decoupling: From Dependence to Independence*. Probability and Its Applications. Springer New York, 1999. ISBN 9780387986166. URL https://books.google.hu/books?id=DHVfJrvTRvcC.

Gao, Z., Ries, C., Simon, H. U., and Zilles, S. Preference-based teaching. *J. Mach. Learn. Res*, 18:31:1–31:32, 2017. URL http://jmlr.org/papers/v18/16-460.html.

Goldman, S. A. and Kearns, M. J. On the complexity of teaching. *J. Comput. Syst. Sci*, 50(1):20–31, 1995.

Johns, E., Aodha, O. M., and Brostow, G. J. Becoming the expert - interactive multi-class machine teaching. In *CVPR*, pp. 2616–2624. IEEE Computer Society, 2015. ISBN 978-1-4673-6964-0. URL http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=7293313.

Kirkpatrick, D., Simon, H. U., and Zilles, S. Optimal collusion-free teaching. In Garivier, A. and Kale, S. (eds.), *Proceedings of the 30th International Conference on Algorithmic Learning Theory*, volume 98 of *Proceedings of Machine Learning Research*, pp. 506–528, Chicago, Illinois, 22–24 Mar 2019. PMLR. URL http://proceedings.mlr.press/v98/kirkpatrick19a.html.

Korman, S. *On the Use of Randomization in the Online Set Cover Problem*. PhD thesis, Department of Computer Science and Applied Mathematics, The Weizmann Institute of Science, Israel, 2004.

Liu, W., Dai, B., Humayun, A., Tay, C., Yu, C., Smith, L. B., Rehg, J. M., and Song, L. Iterative machine teaching. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2149–2158. PMLR, 2017. URL http://proceedings.mlr.press/v70/.

Liu, W., Dai, B., Li, X., Liu, Z., Rehg, J. M., and Song, L. Towards black-box iterative machine teaching. In Dy, J. G. and 0001, A. K. (eds.), *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3147–3155. PMLR, 2018. URL http://proceedings.mlr.press/v80/.

Mansouri, F., Chen, Y., Vartanian, A., Zhu, J., and Singla, A. Preference-based batch and sequential teaching: Towards a unified view of models. In *Advances in Neural Information Processing Systems 32*, pp. 9195–9205. Curran Associates, Inc., 2019.

Mei, S. and Zhu, X. Using machine teaching to identify optimal training-set attacks on machine learners. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI?15, pp. 2871?2877. AAAI Press, 2015. ISBN 0262511290.

Rafferty, A. N., Brunskill, E., Griffiths, T. L., and Shafto, P. Faster teaching via pomdp planning. *Cognitive Science*, 40(6):1290–1332, 2016.

Raz and Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP. In *STOC: ACM Symposium on Theory of Computing (STOC)*, 1997.

Shinohara, A. Teachability in computational learning. *New Generation Comput*, 8(4):337–347, 1991.

Singla, A., Bogunovic, I., Bartók, G., Karbasi, A., and Krause, A. Near-optimally teaching the crowd to classify. In *ICML*, volume 32 of *JMLR Workshop and Conference Proceedings*, pp. 154–162. JMLR.org, 2014. URL http://proceedings.mlr.press/v32/.

Zhou, Y., Nelakurthi, A. R., and He, J. Unlearn what you have learned: Adaptive crowd teaching with exponentially decayed memory learners. In Guo, Y. and Farooq, F. (eds.), *KDD*, pp. 2817–2826. ACM, 2018.

Zhu, X., Singla, A., Zilles, S., and Rafferty, A. N. An overview of machine teaching. *CoRR*, abs/1801.05927, 2018. URL http://arxiv.org/abs/1801.05927.

Zilles, S., Lange, S., Holte, R., and Zinkevich, M. Models of cooperative teaching and learning. *J. Mach. Learn. Res*, 12:349–384, 2011. URL http://portal.acm.org/citation.cfm?id=1953059.