# Understanding the Effects of Batching in Online Active Learning

**Kareem Amin**
Google, New York

**Corinna Cortes**
Google, New York

**Giulia DeSalvo**
Google, New York

**Afshin Rostamizadeh**
Google, New York

## Abstract

Online active learning (AL) algorithms often assume immediate access to a label once a query has been made. However, due to practical constraints, the labels of these queried examples are generally only available in “batches”. In this work, we present an analysis for a generic class of batch online AL algorithms, which reveals that the effects of batching are in fact mild and only result in an additional label complexity term that is quasilinear in the batch size. To our knowledge, this provides the first theoretical justification for such algorithms and we show how they can be applied to batch variants of three canonical online AL algorithms: IWAL, ORIWAL, and DHM. Finally, we also present empirical results across several benchmark datasets that corroborate these theoretical insights.

## 1 Introduction

Large labeled datasets are often used to train models in supervised learning. However, in some domains, such as those that require domain experts, labeling is a very costly. Active learning directly tackles the important task of training accurate models while at the same time minimizing the number of labeled points.

Previous work in active learning often analyzes the online, or streaming, setting where a learner observes a single unlabeled example at a time and decides whether or not to request the label of the example, receiving the label immediately if queried, and typically updating the learner with the additional label before receiving the next sample point. Generally, obtaining the label of just a single point is entirely impractical due to, for example, the overhead of assembling a pool of qualified raters and assigning enough work to each rater in order for their time to be well spent. Additionally, there is a considerable overhead in updating an active learner, e.g. computing a new version space, with one additional instance at a time. Thus, in practice, labels are requested only once a large enough batch of requests has been queued. As an example, consider remote sensors with a finite buffer that process a stream of unlabeled data, a subset of which may be useful for training a machine learning model. Due to the practical reasons discussed above, as well as potential communication costs, the sensor only sends batches of points to a labeling service once the buffer is full.

Motivated by these practical constraints, we analyze the *batch online active learning* setting. A common approach is to convert off-the-shelf online active learning algorithms to operate in the batch setting. This can be accomplished by delaying label feedback to the algorithm until a sufficiently large number of label requests are made. However, the effect of batching on the active learning algorithm’s generalization and label complexity guarantees is not well understood. Since the batch framework is a substantially more restrictive setting, the main questions we seek to answer are: in what ways will batching impact known theoretical guarantees? How strongly do label complexity and generalization guarantees depend on the batch size?

To that end, we present a label complexity analysis, that is a bound on the number of requested labels, for a generic batch online active meta-algorithm that is assumed to satisfy a mild *time-decreasing labeling rate* condition. This condition states that the probability of requesting the label of point decreases as a function of time, which is a natural property for any active learning algorithm that admits non-trivial bounds on label complexity. Crucially, our theoretical analysis shows that the label complexity of such batch active learning algorithms, ignoring logarithmic terms, admit a linear dependence on the batch size. This reveals that the effects of batching are minimal as long as the batch size is a constant independent of the total number of observations. We show that this result can be applied to batch variants of three well studied online

Proceedings of the 23$^{\text{rd}}$International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108. Copyright 2020 by the author(s).

active learning algorithms: IWAL, ORIWAL, and DHM [Beygelzimer et al., 2009, Cortes et al., 2019a, Dasgupta et al., 2008]. Moreover, we prove that the theoretical guarantees of these algorithms are not affected by batch size and empirically verify the insights provided by both the label complexity and generalization bounds. To our knowledge, this is the first work proving theoretical guarantees for batch online active learning.

Below, we review related work. In Section 2, we present our generic batch online active learning framework and in Section 3, we provide a novel theoretical analysis of its label complexity. Then, we show applications of this generic batch online AL framework along with the derivations of the generalization guarantees in Sections 4 and an empirical verification in Section 5. Most of the proofs of our analysis are found in the appendix.

**Related Work:** Most theoretical work in (non-batch) active learning considers the online setting with a focus on proving generalization guarantees for the hypothesis returned by an active learning algorithm and bounds on the active learner's label complexity. In the case of separable data, Cohn et al. [1994] derived an algorithm that exhibits an exponential decrease in label complexity when compared to passive learning. The main idea of this algorithm is to trim the hypothesis set of all classifiers that are inconsistent with the currently labeled data and to only ask for the labels of points the hypotheses in this set disagree on. The amount of disagreement among a set of hypothesis can be characterized by the *disagreement coefficient*, which was first introduced by Hanneke [2007]. These ideas on disagreement are the core of many active learning algorithms and the label complexity guarantees of these algorithms are often in terms of the disagreement coefficient [Balcan et al., 2006, Dasgupta et al., 2008, Beygelzimer et al., 2009, 2010, Cortes et al., 2019a,b]. Similar quantities will appear in the bounds we present, although we arrive at them in a significantly different fashion. Another line of work has focused on algorithms based on requesting labels along the margin of a linear separator, which only under certain distributional assumptions admit theoretical guarantees [Dasgupta et al., 2005, Balcan et al., 2007, Balcan and Long, 2013, Awasthi et al., 2014, 2015, Zhang, 2018].

Active learning has also been analyzed in pool-based setting where the entire pool of unlabeled data is available and the algorithm must choose subsets of this pool to be sent to raters for labeling. Unlike the batch online active learning setting, the algorithms in the pool based setting do not have limited memory and thus are qualitatively very different than the algorithms analyzed in this paper. Several authors have studied the pool-based setting, but the focus was primarily on finding solutions for specific tasks. For example, Kurihara and Sugiyama [2012] derives sampling objectives tailored to linear regression for choosing a single batch of examples to be labeled, Bach [2007] presents an asymptotic analysis for misspecified generalized linear models, and McCallum and Nigam [1998], Hoi et al. [2006a,b, 2008] focus on text and image classification tasks. Other work has focused on incorporating different definitions of diversity and informativeness, but without deriving any generalization and label complexity guarantees [Brinker, 2003, Xu et al., 2007, Guo and Schuurmans, 2008]. Dasgupta and Hsu [2008] develops an active learning algorithm with theoretical guarantees under specific assumptions on the ability to cluster the data.

Chen and Krause [2013] analyze the batch active learning problem, albeit in the pool based setting, and show that a greedy batch construction strategy is competitive with an optimal batch selection when the problem exhibits an adaptive submodularity condition. Our work is significantly different, in that we consider the online setting and make no submodularity assumption.

Online learning with delayed feedback has been studied in the general partial-monitoring setting. Joulani et al. [2013] demonstrate that in non-adversarial settings the price of delayed feedback is an additive regret term that is linear (ignoring log factors) in the length of the feedback delay. This strongly mirrors the label-complexity result we achieve in our setting as we bound the number of additional label requests made by an amount that is also linear (ignoring log factors) in the length of feedback delay. However, we emphasize that one setting does not subsume the other. In online learning with delayed feedback, the learner eventually receives feedback for every decision, while in active learning, there are some rounds where the learner never receives feedback at all (i.e., when a label is not requested). Moreover, in contrast to online learning, where there is a single objective (minimizing regret), active learning admits a bicriterion of bounding label complexity and generalization error.

## 2 Batch Active Learning Framework

Let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ denote an example space and $\mathcal{Z}_\perp = \mathcal{X} \times (\mathcal{Y} \cup \{\perp\})$ denote the same example space except where examples can be label-free (denoted by $\perp$). We assume that the data is drawn stochastically, that is the data points are drawn i.i.d. from an unknown distribution $\mathcal{D}$ over $\mathcal{Z}$ and define a hypothesis set $H$ where each function $h \in H$ maps from $\mathcal{X}$ to $\mathcal{W} \subseteq \mathbb{R}$. The quality of a hypothesis function is measured by a loss function, $\ell \colon \mathcal{W} \times \mathcal{Y} \to [0, 1]$, and we denote by $L(h) = \mathbb{E}[\ell(h(x), y)]$ the expected loss of hypothesis $h$.

Let $A$ denote an online active learning algorithm. At each time step $t \in [T] := \{1, \ldots, T\}$, algorithm $A$

maintains an internal state $\omega_t \in \Omega$, where $\Omega$ denotes a universe of possible internal states. Given an example $x_t \in \mathcal{X}$, the algorithm first decides whether or not to receive the true label of $x_t$. We denote by $\bar{y}_t \in \mathcal{Y} \cup \{\perp\}$ either the true revealed label $y_t$ or the decision not to request the label, $\perp$, at time $t$. Given this new information, the algorithm then updates its state to $\omega_{t+1}$. Both the decision of requesting the label and the update of its state may be a probabilistic process.

An online active learning algorithm can be fully characterized by its state, a function that decides whether to request for a label, and a function that updates the state of the algorithm. More formally, we define two functions $\texttt{Labeler}\colon \Omega \times \mathscr{X} \to [0, 1]$ which maps a state $\omega_t$ and an example $x_t$ to the probability of requesting a label and $\texttt{Updater}\colon \Omega \times \mathscr{Z}_\perp \to \Delta(\Omega)$ which maps the feedback received during a timestep to a distribution over next states, where $\Delta(\Omega)$ denotes the probability simplex over $\Omega$. Given an initial state $\omega_1$, an online active learning algorithm is then defined by the following triplet: $A = (\omega_1, \texttt{Labeler}, \texttt{Updater})$.

In order to convert such an online active learning algorithm $A$ into an algorithm $A_B$ for the batch setting, we "freeze" the state of $A$ until $B$ labels have been requested. That is, the algorithm makes decisions on a sequence of points without updating its state until $B$ labels have been requested. We call a sequence of timesteps where the state remains unchanged a *round*. At the end of the round, the algorithm receives the $B$ labels of the requested points and it updates its state accordingly.[1] This continues until the time horizon $T$ is met, at which point the algorithm selects a hypothesis using information from the final state. Note that in the sequel, during the execution of the algorithm, we denote the current round index using the variable $r$. Algorithm 1 defines the batch algorithm in detail.

**Algorithm 1** Batch Online AL Algorithm $A_B$.

```
Inputs : $A = (\omega_1, \texttt{Labeler}, \texttt{Updater})$, batch size $B \geq 1$, time horizon $T \geq 1$.
Set : current round $r = 1$, number labels requested in round $C_1 = 0$, previous round boundary $\tau_0 = 0$.
for $t = 1, 2, \ldots, T$ do
  Receive $x_t$.
  Draw $Q_t \sim \text{Bernoulli}(\texttt{Labeler}(\omega_r, x_t))$.
  Update $C_r = C_r + Q_t$.
  #$B$ requests or final timestep, end round.
  if $C_r = B$ or $t = T$ then
    #Receive batched labels.
    for $t' = \tau_{r-1} + 1, \ldots, t$ do
      if $Q_{t'} = 1$ then
        Receive $y_{t'}$.
        Set $\hat{y}_{t'} = y_{t'}$.
      else
        Set $\hat{y}_{t'} = \perp$.
    #Perform batched state updates.
    Initialize $\omega' = \omega_r$.
    for $t' = \tau_{r-1} + 1, \ldots, t$ do
      Draw $\omega' \sim \texttt{Updater}(\omega', x_{t'}, \hat{y}_{t'})$
    Update : round $r = r + 1$, previous round boundary $\tau_{r-1} = t$, frozen state $\omega_r = \omega'$.
    Reset label count $C_r = 0$.
return $\widehat{h}_T$ hypothesis learned with state $\omega_r$.
```

In Section 3 we will provide a sufficient condition in order to analyze the label complexity of Algorithm 1. Then in Section 4, we argue that this condition holds for batch variants of many existing online active learning algorithms and therefore our label complexity bounds can be applied directly. In Section 4, we also show that for these same algorithms, generalization guarantees are completely unaffected by batching.

# 3 Label Complexity

In this section, we show that converting an online active algorithm to a batch online active algorithm results in a label complexity bound that contains only a mild dependence on batch size. More specifically, the additional label complexity cost over the online active learning algorithm is an *additive* $\tilde{O}(B)$ term in the batch size $B$. Ignoring log factors, this is the best one can hope for, since a batch active algorithm must request $\Omega(B)$ labels in order to receive any labels at all. In order for this general result to hold, the active learning algorithm must satisfy a natural condition, which we called *time-decreasing labeling rate*.

## 3.1 Time-Decreasing Labeling Rate

Let $r(t)$ be a random variable denoting the round corresponding to timestep $t$, that is $r(t) = \min\{s \mid \tau_s \geq t\}$, and let $\mathcal{F}_t$ denote the sigma algebra containing all random variables up to time $t$. A batch active learning algorithm has time-decreasing labeling rate if the probability of requesting a label can be upper bounded by a decreasing function $p^+_\delta$ that depends only on the timesteps required for the previous round to elapse, $\tau_{r(t)-1}$ and a failure parameter $\delta$. Note that $\tau_{r(t)-1}$ is known at time $t - 1$, meaning it is $\mathcal{F}_{t-1}$-measurable (see the appendix for proof of this statement).

**Definition 1 (Time-Decreasing Labeling Rate)** *For any $\delta > 0$, we say $A_B$ has time-decreasing labeling rate if there exists a non-negative strictly decreasing function $p^+_\delta(t)\colon \mathbb{N} \to [0, 1]$ such that for all $t \geq 1$ with*

[1] The only exception is the last round, which allows for a batch of fewer than $B$ labels. One could instead ignore the requests made in the last round. This does not materially change the results, but would complicate the presentation.

*probability* $1-\delta$,

$$\mathbb{P}(Q_t = 1 \mid \mathcal{F}_{t-1}) \le p^+_\delta(\tau_{r(t)-1}).$$

In the following, we will elide the dependence on $\delta$ and write $p^+$, unless we wish to make the dependence explicit. The requesting probability of the active learning algorithm could be non-monotonic at each iteration, but the condition implies there is a monotonic decreasing upper bound, which is natural for any active learning algorithm with nontrivial label complexity.

## 3.2 Theoretical Analysis

In this section, we conduct a label complexity analysis for algorithm $A_B$ that admits the time-decreasing labeling rate property and which was derived from an arbitrary online active algorithm $A$ via Algorithm 1. As we will see, the label complexity analysis of batch algorithms departs significantly from the standard analysis for online active learning algorithms since the length of a round is itself a random variable with a particularly intricate dependence on all previous rounds.

Given a horizon $T \ge 1$, the number of labels requested by algorithm $A_B$ will be bounded by $r(T)B$ since $A_B$ requests exactly $B$ labels every time a round elapses and the algorithms halts after round $r(T)$ (except the last round, which may request fewer than $B$ labels). Thus, the goal of this analysis is to prove a high probability upper bound on $r(T)$.

Understanding $r(T)$, depends on analyzing the boundaries $\tau_0, \tau_1, \tau_2, \dots$ of the subsequent rounds. If this sequence grows quickly, it takes relatively few rounds (and therefore few labels) to reach $T$ timesteps. We henceforth focus on the length-of-round, $W_r = \tau_r - \tau_{r-1}$, where $W_r$ is a random variable denoting the waiting time for the $r$th round to elapse. The crux of the proof focuses on showing a lower bound on $W_r$ that holds with high probability. Equivalently, we argue that the lengths of rounds become increasingly longer.

Concretely, the argument proceeds in three stages. We first relate the length-of-round, $W_r$, to a simpler conservative random process, $\tilde{W}_r$, that can be directly defined as a sequence of dependent negative binomial variables. The process is conservative in the sense that the round lengths will, with high probability, be smaller compared to the $W_r$ necessitating additional rounds and therefore more labels before seeing $T$ examples. We then relate this conservative process to an idealized deterministic sequence, which loosely corresponds to the mean of the stochastic conservative process. Finally, we analyze the behavior of this deterministic sequence. At a high level, this analysis reduces the problem of lower bounding $W_r$ to the problem of lower bounding the growth of a deterministic process.

**Step 1: Relating $W_r$ to conservative process $\tilde{W}_r$** First recall $N/q$ is the expected value of a negative binomial distribution $\mathrm{NB}(q, N)$ which counts the number of independent Bernoulli trials with success parameter $p$ until exactly $B$ successes occur. Now, note that for any round $r$, if we condition on the value of $\tau_{r-1}$, then Definition 1 implies with high probability that for any $t$ occurring in round $r$, $\mathbb{P}(Q_t = 1 \mid \tau_{r-1}) \le p^+_\delta(\tau_{r-1})$. Since the length of the round $W_r$ is equal to the number of Bernoulli trials (with success parameter at most $p^+_\delta(\tau_{r-1})$) before $B$ labels are queried, its conditional expectation is lower bounded as follows $\mathbb{E}[W_r \mid \tau_{r-1}] \ge B/p^+_\delta(\tau_{r-1})$.

We construct a new process $\tilde{W}_r$ defined explicitly in terms of a negative binomial distribution with parameter $p^+_\delta(\tilde{\tau}_r)$, specifically

$$\tilde{W}_1 = B \qquad \tilde{\tau}_r = \sum_{s=1}^{r} \tilde{W}_s \qquad \tilde{W}_{r+1} \sim \mathrm{NB}(p^+_\delta(\tilde{\tau}_r), B)\,,$$

and relate it to $W_r$. The subsequent lemma proves that the conservative process $\tilde{W}_r$ is upper bounded by the length-of-round $W_r$.

**Lemma 1** *Fix $\delta > 0$, and suppose that $A_B$ has time-decreasing labeling rate. Then with probability at least $1-\delta$, for all $r$, it holds that $\tilde{W}_r \le W_r$, where $\tilde{W}_r$ has a dependence on $\delta$ via the parameter $p^+_\delta(\tilde{\tau}_r)$.*

The lemma is proven via a coupling argument along with the fact that $A_B$ has time-decreasing labeling rate.

**Step 2: Relating $\tilde{W}_r$ to deterministic seq. $w_r$**

Although the distribution of $\tilde{W}_r$ conditioned on a fixed $\tilde{\tau}_{r-1}$ can be directly related to a negative binomial distribution, the unconditioned distribution does not have this direct relationship. To help with this, we consider the trajectory of the process $\tilde{W}_1, \tilde{W}_2, \dots$ that occurs if every variable $\tilde{W}_r$ is equal to its mean conditioned on the past. In particular define the following deterministic sequence,

$$w_1 = \hat{B} \qquad w_1^r = \sum_{s=1}^{r} w_s \qquad w_{r+1} = \frac{\hat{B}}{p^+(w_1^r)}\,, \tag{1}$$

then letting $\hat{B} = B$ recovers the trajectory just described. We stress that even the expectation of the stochastic process $\tilde{W}_r$ does not follow the particular deterministic trajectory of $w_r$, i.e. $\mathbb{E}[\tilde{W}_r] \ne w_r$ in general.[2] However, if we let $\hat{B} = B/4$, we can show that with high probability the deterministic process $w_r$ lower bounds the stochastic process $\tilde{W}_r$.

First, we define a collection of independent non-identical negative binomial random variables

[2] In particular, even after two rounds, we have $\mathbb{E}[\tilde{W}_3] = \mathbb{E}[\mathbb{E}[\tilde{W}_3 \mid \tilde{\tau}_2]] \ne \mathbb{E}[\tilde{W}_3 \mid \tilde{\tau}_2 = w_1 + w_2] = w_3$.

$N(1), \ldots, N(T)$, where $N(t) \sim \mathrm{NB}(p^+(t), B)$ and where $\mu(t) = \mathbb{E}[N(t)] = B/p^+(t)$ is the expected value of $N(t)$ and also define $\tilde{r}(T) = \min\{r \mid \tilde{\tau}_r \geq T\}$. Then by definition we have $\tilde{W}_1 = B$, and $\tilde{W}_r = N(\tilde{\tau}_{r-1})$ for up to round $\tilde{r}(T)$.

Given that $\{\tilde{W}_r\}$ can be defined in terms of this collection of independent negative binomials $N(\cdot)$, we argue that the growth of $\tilde{W}_r$ is well-behaved as long as $N(\cdot)$ is well-behaved, specifically each $N(t)$ is not much smaller than its mean. To this end, let $T_{\mathrm{bad}}$ count the number of negative binomials that are significantly smaller than their means, that is, $T_{\mathrm{bad}} = \sum_{t=1}^{T} \mathbf{1}[N(t) < \frac{1}{4}\mu(t)]$. Moreover, consider the deterministic sequence $w_r$ defined in equation (1) with $\hat{B} = B/4$. As long as the process $N(\cdot)$ is well-behaved, $\tilde{W}_r$ grows faster than $w_r$. The next lemma states for a horizon $R$, in the worst case, $\tilde{W}_r$ grows like $w_r$ for the first $R - T_{\mathrm{bad}}$ rounds, and then like $B$ for the final $T_{\mathrm{bad}}$ rounds (since the outcome of any $N(\cdot)$ is at least $B$).

**Lemma 2** *Fix a horizon $T \geq 1$. Let $w_r$ be the sequence defined in equation 1 with $\hat{B} = B/4$. On any outcome of $N(1), \ldots, N(T)$, and any $R$ satisfying $T_{\mathrm{bad}} \leq R \leq \tilde{r}(T)$, $w_1^{R-T_{\mathrm{bad}}} + T_{\mathrm{bad}}B \leq \tilde{\tau}_R$, which implies $w_1^{R-T_{\mathrm{bad}}} \leq \tilde{\tau}_R$.*

Next, we bound $T_{\mathrm{bad}}$ with high probability. Specifically, we apply a Chernoff argument to bound the probability than an individual $N(t)$ takes a value of less than $1/4$ of its mean and then use Bernstein's inequality to bound the number of times that this can occur.

**Lemma 3** *For any $\delta < \sqrt{1/e}$, and $B \geq 2\log(T)$, it follows that $P(T_{\mathrm{bad}} > 1 + 3\log(1/\delta)) < \delta$.*

We now state a main theorem, which relates the total number of labels requested by a batch active learner $A_B$ with time decreasing labeling rate $p^+$ to the deterministic process $w_r$ generated by $p^+$. In particular, we relate the label complexity to $R^*$, the number of rounds sufficient for the deterministic process to satisfy $w_1^{R^*} \geq T$, which we analyze in the final step.

**Theorem 1** *Fix $\delta > 0$, and time horizon $T \geq 1$. Let $A_B$ be a batch active sampling algorithm with time decreasing labeling rate $p^+$, and batch size $B \geq 2\log(T)$. Let $w_r$ be the deterministic sequence defined in equation 1 with $\hat{B} = B/4$. Let $R^*$ be a number large enough such that $w_1^{R^*} \geq T$, then with probability at least $1 - 2\delta$, the total labels requested by $A_B$ is bounded by:*

$$Br(T) \leq BR^* + 3B\log(1/\delta) + 2B.$$

*Proof.* We want to show an upper bound on $r(T)$ that depends on $R^*$ and that holds with high probability. We first relate $\tilde{r}(T)$ to $r(T)$ by considering the event $\mathcal{E}'$ that $\tilde{W}_r \leq W_r$ for all $r$. This event implies that $\tilde{\tau}_r \leq \tau_r$ since $\tilde{\tau}_r$ and $\tau_r$ equal the sum of the length-of-rounds $\tilde{W}_{r'}$ and $W_{r'}$ for $r' \in [r]$, respectively. This in turn implies that $r(T) \leq \tilde{r}(T)$ by definition. Next, we focus on proving an upper bound on $\tilde{r}(T)$.

Define the event $\mathcal{E}$ as $T_{\mathrm{bad}} \leq Z$ and consider outcomes where it holds. For the sake of contradiction, suppose that $R^* + Z < \tilde{r}(T) - 1$ where $Z = 1 + 3\log(1/\delta)$. By definition of $R^*$, we have $T \leq w_1^{R^*}$. Combining this with the fact that $w_1^r$ is monotonic and that $T_{\mathrm{bad}} \leq Z$, it then follows that $T \leq w_1^{R^*} \leq w_1^{R^*+(Z-T_{\mathrm{bad}})}$ (call this Fact-1). Again by the event $\mathcal{E}$ and the contradiction assumption, it holds $T_{\mathrm{bad}} \leq Z \leq R^* + Z < \tilde{r}(T) - 1 \leq \tilde{r}(T)$. Then, we can apply Lemma 2 by taking $r = R^* + Z$ to conclude that $w_1^{R^*+Z-T_{\mathrm{bad}}} \leq \tilde{\tau}_{R^*+Z}$ (call this Fact-2). Combining Fact-1, Fact-2 and the contradiction assumption, respectively, we have $T \leq w_1^{R^*+Z-T_{\mathrm{bad}}} \leq \tilde{\tau}_{R^*+Z} \leq \tau_{\tilde{r}(T)-1}$. The inequality $T \leq \tau_{\tilde{r}(T)-1}$ contradicts the definition of $\tilde{r}(T) = \min\{r \mid \tilde{\tau}_r \geq T\}$ and thus, on the event $\mathcal{E}$, it holds that $\tilde{r}(T) \leq R^* + Z + 1$.

Taking a union bound and using Lemmas 1 and 3, with probability at least $1 - 2\delta$, both $\mathcal{E}$ and $\mathcal{E}'$ hold, and therefore $r(T) \leq \tilde{r}(T) \leq R^* + Z + 1 = R^* + 3\log(1/\delta) + 2$. Observing that the label complexity of $A_B$ is at most $Br(T)$ completes the proof. □

Next, we show that $BR^*$ is on the same order as standard active learning bounds; therefore, the above theorem shows that the effects of batching only costs an additional $3B\log(1/\delta) + 2B$ in label complexity.

**Step 3: Behavior of deterministic seq.** $w_r$

First, let us recall the label complexity bounds of standard online active learning algorithms, which can be written in the form of $a^*T + f(T)$. Here, $f(T)/T$ is $o(1)$ and $a^* \in [0, 1]$ is a problem-specific constant that typically contains quantities such as the disagreement coefficient and/or the loss $L^*$ of the best hypothesis. Most active learning algorithms admit a time-decreasing labeling rate either of $p^+(t) = O(1/\sqrt{T})$ or $p^+(t) = O(1/T)$ resulting in label complexity bounds of $a^*T + O(\sqrt{T})$ and $a^*T + O(\log T)$, respectively. In our analysis, we study these two labeling rates and show that the cost of batching on labeling is at most a single additive $\tilde{O}(B)$ on top of the standard rate.

Specifically, we consider $p^+(t) = a + bt^{-\alpha}$, where $\alpha \in \{1/2, 1\}$, $a \in [0, 1]$ and $b \geq 0$. Returning to the deterministic sequence defined in equation (1), recall that $w_1$ begins at $B$ and then converges to $B/a$ as $t \to \infty$. In the appendix, we give a general theorem for studying deterministic sequences that asymptote, but exhibit non-trivial growth before convergence. Utilizing this theorem, we can bound the number of rounds $R^*$ before $w_1^{R^*} \geq T$, thus allowing us to apply Theorem

1 when the functional form of $p^+$ is known.

**Theorem 2** *Fix $\delta < \sqrt{1/e}$, time horizon $T \geq 1$, and $B > \max\{16b, 16, 2\log(T)\}$. Let $w_t$ be the deterministic sequence defined in equation (1) by taking $\hat{B} = B/4$, and $p^+_\delta(t) = a + bt^{-\alpha}$ for values $a \in [0,1]$, $b \geq 0$ and $\alpha > 0$, where $a, b$ may depend on $\delta, B, T$ (but not $t$). Then, for $\alpha = 1$ and*

$$R_1^* = \frac{8aT}{B} + \log_2(T),$$

*it holds that $w_1^{R_1^*} \geq T$. Furthermore, for $\alpha = \frac{1}{2}$, and $b' = \max\{b, 1\}$:*

$$R_{1/2}^* = \frac{1}{B}\Big(8aT + 32b'\sqrt{T} + 4b^2\Big) + \log\log(B/4) + 2,$$

*it holds that $w_1^{R_{1/2}^*} \geq T$.*

The theorem shows that $BR_1^*$ and $BR_{1/2}^*$ are, indeed, of the same order as standard active learning bounds. This implies that in the batch setting we pay only an additive $\tilde{O}(B)$ cost over the standard non-batch label complexity. Using this theorem in combination with Theorem 1, we attain the label complexity for several canonical algorithms in the following section.

## 4 Applications

We analyze the IWAL, ORIWAL, and DHM algorithms where for each algorithm, we show how to extend it to the batch setting via Algorithm 1. For these batch variants, we prove that their generalization guarantees are of the same order as the original non-batch versions and that the theorems of the previous section can be leveraged to bound the label complexity with only a modest dependence on the batch size. In the next subsection, we provide a full description of the batch variant of the IWAL algorithm, generalization guarantee, bound on time-decreasing labeling rate, and resulting label complexity bound. Due to space constraints, we only provide the label complexity for the batch variants of the ORIWAL, and DHM algorithms in the body of the paper, but still provide the full algorithm description and corresponding guarantees in the appendix.

### 4.1 The IWAL algorithm

We start by recalling the IWAL algorithm of [Beygelzimer et al., 2009]. At each time $t \in [T]$, the IWAL algorithm observes a single point $x_t$ and to determine whether to request its label, the algorithm flips a coin $Q_t \in \{0,1\}$ with bias $p_t = \mathbb{P}(Q_t = 1)$. If $Q_t = 1$, then the algorithm requests the label of the point $x_t$ while, if $Q_t = 0$, it passes on this request. The bias probability is defined as $p_t = \max_{f,g \in H_t} \max_{y \in \mathcal{Y}} |\ell(f(x_t), y) - \ell(g(x_t), y)|$ where $H_t \subseteq H$ is the version space at time $t$ maintained by the algorithm. At each time $t$, the algorithm reduces the version space by removing any hypothesis far from the empirical best-in-class: $H_t = \big\{h \in H_{t-1} \colon L_{t-1}(h) \leq \min_{h' \in H_{t-1}} L_{t-1}(h') + \Delta_{t-1}\big\}$, where $L_t(f) = \frac{1}{t}\sum_{s=1}^t \frac{Q_s}{p_s}\ell(f(x_s), \bar{y}_s)$ is the weighted empirical loss and $\Delta_t$ is a slack term.

**Algorithm 2** `Labeler`$(H_r, x_t)$ for B-IWAL

```
$p_r(x_t) \leftarrow \max_{f,g \in H_r} \max_{y \in \mathcal{Y}} |\ell(f(x_t), y) - \ell(g(x_t), y)|$
return $p_r(x_t)$
```

**Algorithm 3** `Updater`$(H_r, \mathcal{Z}_\perp^r)$ for B-IWAL

```
$H_{r+1} \leftarrow \big\{h \in H_r \colon L_{\tau_r}(h) \leq \min_{h' \in H_r} L_{\tau_r}(h') + \Delta_{\tau_r}\big\}$
return $H_{r+1}$
```

The B-IWAL algorithm extends the IWAL algorithm to our setting by freezing the version space for the length-of-round. More concretely, by recalling Algorithm 1, the state for the B-IWAL algorithm is defined in terms of the version space, that is $\omega_r = H_r$ for round $r$ and initially, we set $H_1 = H$. The `Labeler`, which returns the probability of requesting a point, and `Updater`, which updates the state, are defined in Pseudocode 2 and Pseudocode 3. Due to a technicality, the slack term used in the version space is defined as $\Delta_t = \sqrt{8\log(2T^2(T+1)|H|^2/\delta)/t}$, which deviates slightly from slack term of IWAL as it contains $T^2$ instead of $T$. The following theorem provides generalization guarantees for the B-IWAL algorithm.

**Theorem 3** *Let $\widehat{h}_T$ denote the hypothesis returned by B-IWAL after $T$ time steps and let $h^* = \operatorname{argmin}_{h \in H} L(h)$. For any $\delta > 0$, with probability at least $1 - \delta$, $L(\widehat{h}_T) \leq L(h^*) + O\Big(\sqrt{\frac{\log(T|H|/\delta)}{T}}\Big)$.*

The theorem states that the expected loss of the best-in-class $h^*$ is close to that of the hypothesis returned by the algorithm. As $T$ grows, the difference in the expected loss of these two hypotheses decreases at the typical rate of $O(1/\sqrt{T})$. Thus, despite the fact that the version space is updated less frequently, the generalization bound of the B-IWAL algorithm is of the same order as that of the IWAL algorithm. At a high level, this follows from the fact that, although the version space is updated less frequently, when it is updated it will still "catch up" to the analogous hypothesis class that is updated immediately after each time step. However, the algorithm may request more labels due to maintaining a frozen state.

The label complexity of the active learning algorithm will depend on the disagreement coefficient $\theta(\mathcal{D}_\mathcal{X}, H)$, which is defined as the infimum value of $\theta > 0$ such that for all $\Lambda \geq 0$:

$$\mathop{\mathbb{E}}_{x \in \mathcal{D}_\mathcal{X}} \left[ \max_{h \in \mathcal{B}(h^*, \Lambda)} \max_{y \in \mathcal{Y}} |\ell(h(x), y) - \ell(h^*(x), y)| \right] \leq \theta \Lambda \,,$$

where $\mathcal{B}(h', \Lambda) = \{h \in H \colon \rho(h, h') \leq \Lambda\}$ is the ball of radius $\Lambda \geq 0$ and $\rho(h, h') = \mathbb{E}[|\ell(h(x), y) - \ell(h'(x), y)|]$ is the distance between two functions in $h, h' \in H$. Note, this definition of $\rho$ is taken from Cortes et al. [2019b] which allows for a tighter label complexity. For simplicity, we use $\theta$ instead of $\theta(\mathcal{D}_\mathcal{X}, H)$ in this section. The next lemma bounds the probability of the B-IWAL algorithm requesting a point and thereby implies that the time-decreasing labeling rate property is satisfied.

**Lemma 4** *For $\delta > 0$, with probability at least $1 - \delta$, at any round $r$, $\mathbb{E}_x[p_r(x)|\tau_{r-1}] \leq 4\theta(L(h^*) + \Delta_{\tau_{r-1}})$.*

Now, to attain the label complexity bound, we apply the general theory from Section 3. Lemma 4 implies an upper bound on the sampling probability of the form $p_\delta^+(t) = a + bt^{-\alpha}$ with $a = 4\theta L(h^*)$, $b = 4\theta\sqrt{8\log(2T^2(T+1)|H|^2/\delta)}$, and $\alpha = 1/2$. We then apply Theorem 2 with $R^*_{1/2}$ and Theorem 1 along with simplifying terms to prove the following corollary.

**Corollary 1** *Fix $\delta < \sqrt{1/e}$, time horizon $T \geq 1$, and batch size $B > \max\{16b, 16, 2\log(T)\}$. Then with probability at least $1 - 2\delta$, the total labels requested by* B-IWAL *is bounded by: $\widetilde{O}\big(\theta L(h^*)T + \theta\sqrt{T} + B\big)$ , where $\widetilde{O}(\cdot)$ is hiding absolute constants, $\log(T|H|)$ and $\log(1/\delta)$.*

Notice that an additive $\Omega(B)$ term is necessary since the algorithm must request at least $B$ points to see any labels. Thus, for practical label batch sizes, the bound is nearly optimal except for constants and log terms.

### 4.2 The ORIWAL algorithm

At a high level, the ORIWAL algorithm of Cortes et al. [2019a] works by partitioning the space into regions and running a separate active learning algorithm in each region while carefully allocating the labeling resources across regions. Specifically, in each region, the ORIWAL runs the algorithm EIWAL, which is an enhanced version of IWAL with stronger theoretical guarantees.

We first present some needed notation and recall the algorithm. We denote by $\mathcal{X}_k$ for $k \in [n]$ the regions that partition in the input $\mathcal{X}$ and by $H_k$ the hypothesis used in each region. The ORIWAL algorithm returns a hypothesis from the following region-based hypothesis set: $H_{[n]} = \{\sum_{k=1}^n 1_{x \in \mathcal{X}_k} h_k(x) \colon h_k \in H_k\}$. We define $L_k^* = \min_{h \in H_k} \mathbb{E}[\ell(h(x), y)|x \in \mathcal{X}_k]$ be the regional best-in-class and $\theta_k = \theta(\mathcal{D}_{\mathcal{X}_k}, H_k)$ to be the regional disagreement coefficient where $\mathcal{D}_{\mathcal{X}_k}$ is the conditional distribution of $x$ given region $k$.

At each time $t \in [T]$, ORIWAL receives the points $x_t$, finds the region $k_t$ it belongs to, and decides whether to pass this point to the sub-routine EIWAL in region $k_t$ by flipping a coin $A_t \in \{0, 1\}$ with bias $\alpha_{k_t}$. This bias probability carefully chosen to minimize the label complexity across the regions: $\alpha_k = \frac{(c_k/\mathtt{p}_k)^{1/3}}{\max_{k \in [n]}(c_k/\mathtt{p}_k)^{1/3}}$ where $c_k = \log[\frac{16T^2|H_k|^2\log(T)n}{\delta}]$ and where $\mathtt{p}_k = \mathbb{P}[\mathcal{X}_k]$. If $A_t = 1$, then the point $x_t$ is passed to the EIWAL instance in region $k_t$, which decides whether to request the label $y_t$, and updates its internal state.

In the appendix, we present the pseudo-code used to convert ORIWAL to its batch version, B-ORIWAL, and show that it exhibits a time-decreasing labeling rate of the order $O(1/T)$ in the case that $L_k^* = 0$ for all $k$ and $O(1/\sqrt{T})$ otherwise. This results in the following label complexity guarantee for B-ORIWAL.

**Corollary 2** *Fix $\delta < \sqrt{1/e}$, time horizon $T \geq 1$, and batch size $B > \max\{16b, 16, 2\log(T), \frac{4\log(2n/\delta)}{\min_{k \in [n]} \mathtt{q}_k}\}$. Then with probability at least $1 - 2\delta$, the total labels requested by* B-ORIWAL *is bounded as follows:*

*when $L_k^* = 0$ for all $k \in [n]$,*

$$O\big(B\log_2(T) + B\log(1/\delta) + B\big)$$

*and when $\exists k \in [n]$ such that $L_k^* > 0$,*

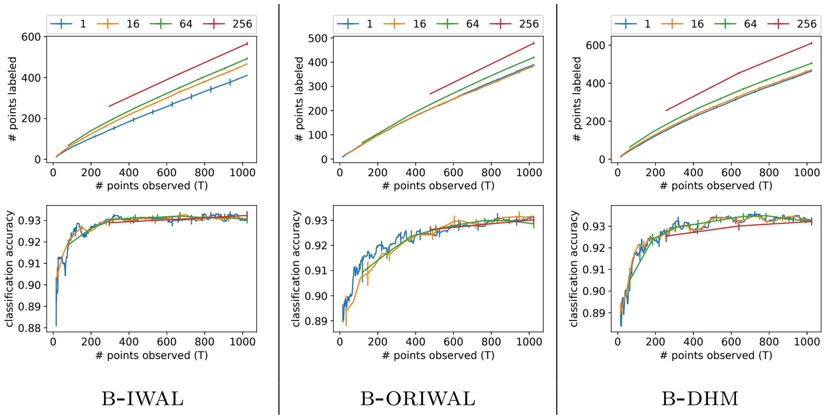
$$\widetilde{O}\big(\textstyle\sum_{k=1}^n \mathtt{q}_k\theta_k L_k^* T + \theta_k\sqrt{T} + B\big) \,,$$

*where $\mathtt{q}_k = \mathtt{p}_k\alpha_k / \sum_{k'=1}^n \mathtt{p}_{k'}\alpha_{k'}$.*

The above label complexity matches that of ORIWAL modulo additive $\tilde{O}(B)$ terms. Crucially, as in the case of B-IWAL, the generalization guarantee is unaffected by batching, which is proven in the appendix.

### 4.3 The DHM algorithm

We extend the DHM algorithm [Dasgupta et al., 2008] to the batch setting. Given a point $x_t$, the DHM algorithm decides to either request the label or assigns it a carefully chosen pseudolabel. Specifically, it constructs two sets, $\widehat{S}_t$ and $T_t$, such that $\widehat{S}_t$ contains examples with pseudolabels consistent with the best-in-class $h^*$ and $T_t$ contains examples with requested labels. The union of these two sets is thus an i.i.d. sample from the underlying marginal distribution on the input space. To decide whether pseudo-label or request the label for given a point $x_t$, the algorithm checks if the difference of the empirical error on $(\widehat{S}_{t-1}, T_{t-1})$ of two hypothesis learned via $h_{\widehat{y}} = \text{LEARN}_H(\widehat{S}_{t-1} \cup \{x_t, \widehat{y}\}, T_{t-1})$ for $\widehat{y} \in \{\pm 1\}$ is large enough. The $\text{LEARN}_H(A, B)$ denotes a learning algorithm that either returns hypothesis $h \in H$ consistent with $A$ and with minumum error on $B$.



B-IWAL

B-ORIWAL

B-DHM

Figure 1: Each column displays the behavior of a different online active learning algorithm on the `phishing` dataset under various batch size constraints. A batch size of 1 (blue curve) corresponds to the setting where the active learner receives a label as soon as it is requested. The top row shows the mean number of points labeled by the respective algorithm, while the bottom row measures the mean test accuracy of the selected model, and error bars indicate the standard error.

The pseudocode for the batch version of DHM, called B-DHM, and the proof that this algorithm satisfies the time-decreasing labeling rate of $O(1/T)$ is in the appendix. Then, the following label complexity bound holds where $\theta'$ is given by Definition 2 in Dasgupta et al. [2008].

**Corollary 3** *Fix $\delta < \sqrt{1/e}$, constant $c > 0$, time horizon $T \geq 1$, and batch size $B > \max\{16b, 16, 2\log(T)\}$. Then with probability at least $1 - 2\delta$, the total labels requested by* B-DHM *is bounded by:* $O\big(c\theta' L(h^*)T + B\log_2(T) + B\log(1/\delta) + B\big)$.

Once again, the label complexity of this batch algorithm admits, ignoring logarithmic terms, an additive linear dependence on the batch size $B$. In the appendix, we prove that the generalization guarantee of B-DHM is of the same order as DHM.

## 5 Empirical Verification

We empirically measure the effect of batching on online active learning algorithms, in the particular case of IWAL, ORIWAL, and DHM algorithms discussed in the previous section. We conduct the evaluation using six different publicly available benchmark datasets: `a9a`, `cod-rna`, `covtype`, `HIGGS`, `mnist`, and `phishing`.[3] For each dataset, the features are normalized to have zero mean and unit variance and subsequently scaled to ensure the maximum feature vector has unit norm. Furthermore, for each dataset, a finite hypothesis set of logistic regression models is generated to serve as the hypothesis set $H$. Each evaluation averages 10 trials, each with a random unlabeled pool and test set split. For details on the size of unlabeled pool and test fold, the number of features, the numbers of hypotheses, and hyperparameter settings used for each dataset, please refer to Appendix C. Details regarding plotting methodology are also found in the same appendix.

Figure 1 displays performance on the `phishing` dataset. Ignoring lower-order dependencies on $T$ and logarithmic factors, Corollaries 1, 2 and 3 predict that the cumulative number of labels requested should appear as $aT + B$ for some problem-dependant constant $a$. Indeed, for each algorithm, the number of labels requested increases with at most an additive dependence on the label query batch size, as is suggested by the additive dependence found in their respective label complexity bounds. Furthermore, we also observe that the test accuracy is essentially unaffected by the batch size, as suggested by the corresponding generalization guarantees. Due to space constraints, results for the remainder of the datasets for all three algorithms, which show similar behavior, are presented in Appendix C. Overall, we find that the empirical results corroborate the theoretical results of the previous sections.

## 6 Conclusion

We presented an analysis of the batch online active learning setting, which is directly motivated by practical constraints. We bound the label complexity of a generic batch online active learning algorithm, showed that the result can be applied to several well-studied online active learning algorithms, and verified the findings empirically. Future directions include analyzing batching effects in pool-based settings. In such settings, additional requirements, such as enforcing diversity of examples within a batch, may be necessary.

[3] https://www.csie.ntu.edu.tw/˜cjlin/libsvmtools/datasets/

## References

Pranjal Awasthi, Maria-Florina Balcan, and Philip M. Long. The power of localization for efficiently learning linear separators with noise. In *Proceedings of the forty-sixth annual ACM symposium on theory of computing*, pages 449–458. ACM, 2014.

Pranjal Awasthi, Maria-Florina Balcan, Nika Haghtalab, and Ruth Urner. Efficient learning of linear separators under bounded noise. In *Conference on Learning Theory*, pages 167–190, 2015.

Francis Bach. Asymtotic analysis of generalized linear models. *Advances in Neural Information Processing Systems*, 2007.

Maria-Florina Balcan and Phil M. Long. Active and passive learning of linear separators under log-concave distributions. In *Conference on Learning Theory*, pages 288–316, 2013.

Maria-Florina Balcan, Alina Beygelzimer, and John Langford. Agnostic active learning. In *Proceedings of the 23th International Conference on Machine Learning*, 2006.

Maria-Florina Balcan, Andrei Broder, and Tong Zhang. Margin based active learning. In *Conference on Learning Theory*, pages 35–50. Springer, 2007.

Alina Beygelzimer, Sanjoy Dasgupta, and John Langford. Importance weighted active learning. In *Proceedings of the 26th International Conference on Machine Learning*, pages 49–56. ACM, 2009.

Alina Beygelzimer, Daniel J. Hsu, John Langford, and Tong Zhang. Agnostic active learning without constraints. In *Advances in Neural Information Processing Systems*, pages 199–207, 2010.

Klaus Brinker. Incorporating diversity in active learning with support vector machines. In *Proceedings of the 20th International Conference on Machine Learning*, 2003.

Yuxin Chen and Andreas Krause. Near-optimal batch mode active learning and adaptive submodular optimization. *International Conference on Machine Learning*, 2013.

David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine Learning*, 15(2):201–221, 1994.

Corinna Cortes, Giulia DeSalvo, Claudio Gentile, Mehryar Mohri, and Ningshan Zhang. Region-based active learning. In *International Conference on Artificial Intelligence and Statistics*, 2019a.

Corinna Cortes, Giulia DeSalvo, Claudio Gentile, Mehryar Mohri, and Ningshan Zhang. Active learning with disagreement graphs. In *Proceedings of the 26th International Conference on Machine Learning*, 2019b.

Sanjoy Dasgupta and Daniel J. Hsu. Hierarchical sampling for active learning. In *Proceedings of the 25th International Conference on Machine Learning*, pages 208–215. ACM, 2008.

Sanjoy Dasgupta, Adam Tauman Kalai, and Claire Monteleoni. Analysis of perceptron-based active learning. In *Conference on Learning Theory*, pages 249–263. Springer Berlin Heidelberg, 2005.

Sanjoy Dasgupta, Daniel J. Hsu, and Claire Monteleoni. A general agnostic active learning algorithm. In *Advances in Neural Information Processing Systems*, pages 353–360, 2008.

Yuhong Guo and Dale Schuurmans. Discriminative batch mode active learning. In *Advances in Neural Information Processing Systems*, 2008.

Steve Hanneke. A bound on the label complexity of agnostic active learning. In *Proceedings of the 24th International Conference on Machine learning*, pages 353–360. ACM, 2007.

Steven Hoi, Rong Jin, and Michael Lyu. Large-scale text categorization by batch mode active learning. In *International Conference on World Wide Web*, 2006a.

Steven Hoi, Rong Jin, Jianke Zhu, and Michael Lyu. Batch mode active learning and its application to medical image classification. In *Proceedings of the 23rd International Conference on Machine Learning*, 2006b.

Steven Hoi, Rong Jin, Jianke Zhu, and Michael Lyu. Semi-supervised svm batch mode active learning for image retrieval. In *Computer Vision and Pattern Recognition*, 2008.

Svante Janson. Tail bounds for sums of geometric and exponential variables. *Statistics & Probability Letters*, 135:1–6, 2018.

Pooria Joulani, Andras Gyorgy, and Csaba Szepesvári. Online learning under delayed feedback. In *International Conference on Machine Learning*, pages 1453–1461, 2013.

Nozomi Kurihara and Masashi Sugiyama. Improving importance estimation in pool-based batch active learning for approximate linear regression. *Neural Networks*, 36:73–82, 2012.

Andrew McCallum and Kamal Nigam. Employing em in pool-based active learning for text classification. *Proceedings of the Fifteenth International Conference on Machine Learning*, 1998.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python.

*Journal of Machine Learning Research*, 12:2825–2830, 2011.

Zuobing Xu, Ram Akella, and Yi Zhang. Incorporating diversity and density in active learning for relevance feedback. In *European Conference on Information Retrieval*, 2007.

Chicheng Zhang. Efficient active learning of sparse halfspaces. In *Conference on Learning Theory*, 2018.

# A Label Complexity of Batch Active Learning Algorithms

## A.1 Time-Decreasing Labeling Rate

**Lemma 5** *$\tau_{r(t)-1}$ is $\mathcal{F}_{t-1}$-measurable (i.e., known at time $t-1$).*

*Proof.* By definition of $r(t)$, $\tau_{r(t)-1} < t$, and therefore $\tau_{r(t)-1}$ is known at time $t-1$ as long as $r(t)$ is known at time $t-1$. Note that the event $\{\tau_s \leq t-1\}$ is $\mathcal{F}_{t-1}$-measurable for any $s$, and therefore so is its complement $\{\tau_s \geq t\}$. This implies that $r(t)$ is $\mathcal{F}_{t-1}$-measurable by its definition. □

## A.2 A Negative Binomial Coupling

In this section we relate the random variable $W_r$ to a different random variable that is defined concisely in terms of the negative binomial distribution and, thus, will be more easily analyzed. More concretely, we analyze the process $W_1, W_2, \ldots$ by constructing a new process $\tilde{W}_1, \tilde{W}_2, \ldots$ where $\tilde{W}_r \leq W_r$ for all $r$ with high probability, and each $\tilde{W}_r$ has the negative binomial distribution when conditioned on the past. Define the analog of $\tau_r$, $\tilde{\tau}_r = \sum_{s=1}^{r} \tilde{W}_s$. Since there are many conventions for defining the negative binomial, $\mathrm{NB}(p, B)$, we emphasize that here we count the total number of trials (not failures), each with success probability $p$, until $B$ successes are observed. Under this definition, $E[N] = B/p$ and $P(N < B) = 0$ if $N \sim \mathrm{NB}(p, B)$.

We now describe the construction. Let $\tilde{W}_1 = B$. Fix an $r > 1$, and suppose we have defined $\tilde{W}_1, \ldots, \tilde{W}_{r-1}$. We define $\tilde{W}_r$ as a function of the execution of round $r$ of $A_B$, and $\tilde{\tau}_{r-1}$. Let $\bar{p} = \mathbb{P}[Q = 1 \mid \omega_r]$, where the probability is taken over the example $x$ and $Q \sim \mathrm{Bernoulli}(\texttt{Labeler}(\omega_r, x))$. This denotes the probability that $A_B$ requests a label on round $r$ (before seeing the unlabeled example). Let $z = p^+_\delta(\tilde{\tau}_{r-1}) - \bar{p}$ denote the difference between this probability and $p^+_\delta$ (as defined in 1) applied to the new process. Note, unless specifically referring to the dependence on $\delta$, we write $p^+$ in order to simplify notation throughout the remainder of this section.

Reindex the label requests made by $A_B$ in round $r$, $Q_{t'}$ for $t' \in \{\tau_{r-1}+1, \ldots, \tau_r\}$ as $Q_i$ for $i \in \{1, \ldots, W_r\}$. We now construct $\tilde{Q}_i$ as follows:

1. If $\bar{p} > p^+(\tilde{\tau}_{r-1})$, we draw $\tilde{Q}_i \sim \mathrm{Bernoulli}(p^+(\tilde{\tau}_{r-1}))$.
2. If $\bar{p} \leq p^+(\tilde{\tau}_{r-1})$ and $Q_i = 1$, we set $\tilde{Q}_i = 1$.
3. If $\bar{p} \leq p^+(\tilde{\tau}_{r-1})$ and $Q_i = 0$, we draw $\tilde{Q}_i \sim \mathrm{Bernoulli}(z/(1-\bar{p}))$.

Given the above, we can now define $\tilde{W}_r = \min\{j : \sum_{i=1}^{j} \tilde{Q}_i = B\}$.

**Lemma 1** *Fix $\delta > 0$, and suppose that $A_B$ has time-decreasing labeling rate. Then with probability at least $1-\delta$, for all $r$, it holds that $\tilde{W}_r \leq W_r$, where $\tilde{W}_r$ has a dependence on $\delta$ via the parameter $p^+_\delta(\tilde{\tau}_r)$.*

*Proof.* Let $\mathcal{E}$ be the event that $\mathbb{P}(Q_t = 1 \mid \omega_{r(t)}) = \mathbb{P}(Q_t = 1 \mid \mathcal{F}_{t-1}) \leq p^+(\tau_{r(t)-1})$ for all timesteps $t$. We know from definition 1 that $\mathcal{E}$ occurs with probability at least $1-\delta$.

We will argue that on the event $\mathcal{E}$, $\tilde{W}_r \leq W_r$ for all $r \geq 1$. The base case is immediate since $\tilde{W}_1 = B$, and $W_1 \geq B$. Assume $\tilde{W}_{r'} \leq W_{r'}$ for all $r' < r$. This implies that $\tilde{\tau}_{r-1} \leq \tau_{r-1}$. Fix an arbitrary timestep $i$ during the execution of round $r$ of $A_B$. On the event $\mathcal{E}$, $\mathbb{P}(Q_i = 1 \mid \omega_r) \leq p^+(\tau_{r-1}) \leq p^+(\tilde{\tau}_{r-1})$, where the second inequality follows since $p^+$ is a decreasing function. By construction, when $\bar{p} = \mathbb{P}(Q_i = 1 \mid \omega_r) \leq p^+(\tilde{\tau}_{r-1})$, we know that $Q_i \leq \tilde{Q}_i$, and therefore $\tilde{W}_r \leq W_r$ by definition of $\tilde{W}_r$, completing the inductive step. □

**Lemma 6** *Fix any $r > 1$, and condition on $\tilde{\tau}_{r-1}$, $\tilde{W}_r$ has the negative binomial distribution $N(p^+(\tilde{\tau}_{r-1}), B)$.*

*Proof.* Fix $r > 1$, Let $i$ be an arbitrary timestep during the execution of round $r$ of $A_B$. On the event $\bar{p} > p^+(\tilde{\tau}_{r-1})$, $P(\tilde{Q}_i = 1 \mid \omega_r, \tilde{\tau}_{r-1}) = p^+(\tilde{\tau}_{r-1})$. On the other hand, if $\bar{p} \leq p^+(\tilde{\tau}_{r-1})$, $P(\tilde{Q}_i = 1 \mid \omega_r, \tilde{\tau}_{r-1}) = P(Q_i = 1 \mid \omega_r) + P(Q_i = 0 \mid \omega_r) z/(1 - \bar{p}(\tau_{r-1})) = \bar{p} + z = p^+(\tilde{\tau}_{r-1})$. Thus, $P(\tilde{Q}_i = 1 \mid \omega_r, \tilde{\tau}_{r-1}) = p(\tilde{\tau}_{r-1})$, as desired. □

## A.3 Step 2: Relating process $\tilde{W}_r$ to deterministic sequence $w_r$

The first lemma proven below relies on a technical lemma (Lemma 12) which is provided in Section A.7.1.

**Lemma 2** *Fix a horizon $T \geq 1$. Let $w_r$ be the sequence defined in equation 1 with $\hat{B} = B/4$. On any outcome of $N(1), \ldots, N(T)$, and any $R$ satisfying $T_{\text{bad}} \leq R \leq \tilde{r}(T)$, $w_1^{R-T_{\text{bad}}} + T_{\text{bad}}B \leq \tilde{\tau}_R$, which implies $w_1^{R-T_{\text{bad}}} \leq \tilde{\tau}_R$.*

*Proof.* For any $s \leq r(T)$, $\tilde{W}_s = N(\tilde{\tau}_{s-1})$. By definition of $T_{\text{bad}}$ there are at $T_{\text{bad}}$ of the negative binomimals $N(1), \ldots, N(T)$ that take values much smaller than their mean, and therefore at most $T_{\text{bad}}$ rounds where $\tilde{W}_s < \frac{1}{4}\mu(\tilde{\tau}_{s-1})$. For the remaining rounds $s$, $\tilde{\tau}_s \geq \frac{1}{4}\mu(\tilde{\tau}_{s-1})$. Since $\mu$ is monotone increasing, Lemma 12 completes the proof as long as $T_{\text{bad}} \leq R$. □

The key lemma of this section bounds $T_{\text{bad}}$ with high probability. We will rely on a tail bound (Lemma 7) which can be derived using a Chernoff bound.

**Lemma 7 (Janson [2018])** *If $N \sim \text{NB}(p, B)$, then for any $\lambda \leq 1$, $P(N \leq \lambda \mathbb{E}[N]) \leq \exp(-B(\lambda - 1 - \ln \lambda))$.*

**Lemma 3** *For any $\delta < \sqrt{1/e}$, and $B \geq 2\log(T)$, it follows that $P(T_{\text{bad}} > 1 + 3\log(1/\delta)) < \delta$.*

*Proof.* When $\lambda = 1/4$, $\lambda - 1 - \ln \lambda \approx 0.636$, and so Lemma 7 guarantees that for any $N(t)$ we have that $P(N(t) < \frac{1}{4}\mu(t)) \leq \exp(-B/2)$. Suppose $B \geq 2\log(T)$. $\mathbb{E}[T_{\text{bad}}] = \sum_{t=0}^{T-1} P(N(t) < \frac{1}{4}\mu(t)) \leq T\exp(-B/2) \leq 1$, which in turn implies that $P(T_{\text{bad}} - 1 > \gamma) \leq P(T_{\text{bad}} - \mathbb{E}[T_{\text{bad}}] > \gamma)$ for any $\gamma$. Since the random variables $X_t = \mathbf{1}[N(t) < \frac{1}{4}\mu(t)]$ are in $\{0, 1\}$ and independent, Bernstein's inequality guarantees that

$$\begin{aligned} P(T_{\text{bad}} - \mathbb{E}[T_{\text{bad}}] > \gamma) &\leq \exp\Big( - \frac{\gamma^2/2}{\sum_{t=0}^{T-1} \mathbb{E}[(X_t - \mathbb{E}[X_t])^2] + \gamma/3}\Big) \\ &\leq \exp\Big( - \frac{\gamma^2/2}{\sum_{t=0}^{T-1} \mathbb{E}[X_t^2] + \gamma/3}\Big) \\ &= \exp\Big( - \frac{\gamma^2/2}{\sum_{t=0}^{T-1} \mathbb{E}[X_t] + \gamma/3}\Big) \\ &\leq \exp\Big( - \frac{\gamma^2/2}{T\exp(-B/2) + \gamma/3}\Big) \end{aligned}$$

If $B \geq 2\log(T)$, and $\gamma \geq 1$, then $T\exp(-B/2) \leq \gamma$, and:

$$P(T_{\text{bad}} - \mathbb{E}[T_{\text{bad}}] > \gamma) \leq \exp\left( - \frac{\gamma^2/2}{\frac{4}{3}\gamma} \right) \leq \exp\left( - \frac{3}{8}\gamma \right).$$

Taking $\gamma = \frac{8}{3}\log(\frac{1}{\delta})$, and recalling $\mathbb{E}[T_{\text{bad}}] \leq 1$ proves the lemma under the condition that $B \geq 2\log(T)$, and $\gamma \geq 1$. Note that $\gamma \geq 1$ if $\delta \leq \exp(-3/8)$, which is satisfied if $\delta \leq \sqrt{1/e}$. □

### A.4 General Deterministic Bound

Given an increasing sequence of positive numbers $w_1, w_2, \ldots$, where $w_1^r = \sum_{s=1}^{r} w_s$, and a fixed $T \geq 0$, we are interested in characterizing the number of rounds $r$ before $w_1^r \geq T$. In particular, we will be interested in cases where the $w_s$ asymptote to some constant, but have some non-trivial growth before reaching this asymptote. Let us suppose that we have lower bounded this growth. So there exists a strictly increasing function $h$ and constant $Z$, such that $w_r \geq \min\{Z, h(r)\}$. We think of $Z$ as being a constant factor of the sequence's asymptote, and $h$ as characterizing the growth of $w_r$ otherwise. We state the following theorem:

**Theorem 4** *Let $w_1, w_2, \ldots$ be an increasing sequence of positive numbers, $h$ be an strictly increasing function with $h(0) = 0$, and $Z > 0$ a constant such that $w_r \geq \min\{Z, h(r)\}$. Let $H(r) \leq \sum_{s=1}^{r} h(s)$, and define $H^{-1}$ as its inverse. Then for any $T \geq 0$:*

$$\textit{If } r \geq Z^{-1}T + H^{-1}(T), \textit{ then } w_1^r \geq T.$$

*Proof.* Since $h$ is strictly increasing and $Z$ is constant, there is some finite $r_0 + 1$ where $h(r_0 + 1) \geq Z$. Define $r_0 = \max\{s \geq 0 \mid h(s) < Z\} \wedge r + 1$ as the last round where $h(s) < Z$ (or 0 or $r + 1$ if this occurs outside of $[1, r]$). We can then express:

$$\begin{aligned} w_1^r \geq \sum_{s=1}^{r} \min\{Z, h(r)\} &= \sum_{s=1}^{r_0} h(s) + \sum_{s=r_0+1}^{r} Z & (2) \\ &\geq H(r_0) + (r - r_0)Z & (3) \end{aligned}$$

If for the chosen $r$, $r_0 \geq H^{-1}(T)$ then the first term in the above inequality lets us conclude that $w_1^r \geq T$ and the proof is complete. It therefore suffices to prove the theorem under the condition that $r_0 < H^{-1}(T)$.

In that case, by taking $r \geq Z^{-1}T + H^{-1}(T)$, we can conclude that $r \geq Z^{-1}T + r_0$ or equivalently $r - r_0 \geq Z^{-1}T$. Equation (3) implies that $w_1^r \geq (r - r_0)Z$. Combining these facts yields $w_1^r \geq T$ as desired. □

In the remainder of the analysis, we will bound sequences of specific interest to us.

### A.5 Bounds for Specific Sequences

We now apply the general theorem from the previous section to specific sequences $w_r$ that occur in the active learning literature. Fix a constant $\hat{B} > 0$. When applying these results to batch active learning, $\hat{B}$ will be a constant fraction of the batch-size. For the purposes of the analysis here, it is just a number. We will be interested in the growth of the sequence $w_1 = \hat{B}$, $w_r = \frac{\hat{B}}{p^+(\sum_{s=1}^{r-1} w_s)}$ for some strictly decreasing function $p^+$. $p^+$ corresponds to a bound on the probability of requesting a label. For algorithms in the literature $p^+(t)$ is typically either $O(\frac{1}{\sqrt{t}})$ or $O(\frac{1}{t})$. In the non-batch setting, the first type of bound yields algorithms whose total label complexity over $T$ rounds is $O(\eta T + \sqrt{T})$, and the second type of bound yields algorithms whose total label complexity over $T$ rounds is $O(\eta T + \log(T))$, where $\eta > 0$ is a problem-specific constant.

Thus, we will be interested in $p^+(t) = \min\{1, a + bt^{-\alpha}\}$, where $a \in [0, 1]$, $b \geq 0$, and $\alpha \in \{1/2, 1\}$. Note the role that $a$ plays in this definition. If $a = 0$, the sequence $w_r$ generated by $p^+$ grows unboundedly. When $a > 0$, the terms $w_r$ in this sequence eventually asymptote to $B/a$.

Our proofs will reduce the $a > 0$ case to the $a = 0$ case. Therefore, it is useful to also define $p_0(t) = bt^{-\alpha}$. And we will study the sequences:

$$w_1 = \hat{B}, \qquad w_1^{r-1} = \sum_{s=1}^{r-1} w_s, \qquad w_r = \frac{\hat{B}}{\min\{1, a + p_0(w_1^{r-1})\}}$$

$$v_1 = \hat{B}, \qquad v_1^{r-1} = \sum_{s=1}^{r-1} v_s, \qquad v_r = \frac{\hat{B}}{\min\{1, p_0(v_1^{r-1})\}}$$

### A.6 $O(\frac{1}{t})$ Time Decreasing Labeling Rate

In this section we analyze the case when $\alpha = 1$.

**Lemma 8** *When $\alpha = 1$, $w_r \geq \min\{\frac{\hat{B}}{2a}, v_r/2^r\}$.*

*Proof.* Define $s_0 = \min\{r > 0 \mid v_r \neq \hat{B} \vee w_r \neq \hat{B}\}$, to be the first round when either $v_s$ or $w_s$ is not equal to $\hat{B}$. We will also be interested in another round $s_1 = \min\{r > 0 \mid p_0(w_1^{r-1}) < a\}$, the first round where $p_0(w_1^{r-1})$ drops below $a$.

By definition of $s_0$, for any $r < s_0$, we have $v_r = w_r = \hat{B}$, and therefore $w_r \geq v_r/2^0$, and the lemma is satisfied on these rounds.

By definition of $s_1$, for any $r \geq s_1$,

$$\frac{\hat{B}}{2a} < \frac{\hat{B}}{a + p_0(w_1^{r-1})} \leq \frac{\hat{B}}{\min\{1, a + p_0(w_1^{r-1})\}} = w_r.$$

We next show the bound holds for rounds $s_0 \leq r < s_1$, if any such rounds exist.

First we show that for rounds $r \geq s_0$, $p_0(v_1^{r-1}) < 1$, and therefore $v_r = \frac{\hat{B}}{p_0(v_1^{r-1})}$. Since $v_1^r$ is strictly increasing in $r$ and $p_0(\cdot)$ is strictly decreasing, it suffices to show that $p_0(v_1^{s_0-1}) < 1$. To prove this note that by definition of $s_0$, either $v_{s_0} \neq \hat{B}$ or $w_{s_0} \neq \hat{B}$, which means either $p_0(v_1^{s_0-1}) < 1$ or $a + p_0(w_1^{s_0-1}) < 1$. Since on rounds $r < s_0$, $v_r = w_r = \hat{B}$, $v_1^{s_0-1} = w_1^{s_0-1} = (s_0 - 1)\hat{B}$, and the claim $a + p_0(w_1^{s_0-1}) < 1$ is equivalent to $a + p_0(v_1^{s_0-1}) < 1$, which implies $p_0(v_1^{s_0-1}) < 1$, since $a \geq 0$.

For $s_0 \leq r < s_1$, we will now argue that for all $s < r$, $w_s \geq v_s/2^s$ by induction. This holds at $r = s_0$, since $w_s = v_s/2^0 \geq v_s/2^s$ for preceding rounds $s$. Assume that $w_s \geq v_s/2^s$ for all $s < r$. Now we can conclude that:

$$v_r = \frac{\hat{B}}{p_0(v_1^{r-1})} \leq \frac{\hat{B}}{p_0(2^{r-1}w_1^{r-1})} = \frac{2^r\hat{B}}{2p_0(w_1^{r-1})} \leq \frac{2^rB}{a + p_0(w_1^{r-1})} \leq 2^r w_r.$$

The first equality uses the fact that $p_0(v_1^{r-1}) < 1$. The next inequality uses the induction hypothesis and the monotonicity of $p_0$. The next equality uses the definition of $p_0$ when $\alpha = 1$. The final inequality uses the fact that $r < s_1$ and so $p_0(w_1^{r-1}) \geq a$. This completes the induction and the proof. □

Lemma 8 seems incredibly loose, exponentially decaying with $r$. However, as we will see, the sequence $v_r$ grows exponentially as a function of $r$, and so this does not harm our ultimate bounds. We have reduced the analysis of the sequence $w_r$ to the (simpler) sequence $v_r$. The following theorems derive lower bounds for the growth of $v_r$ in specific instances that correspond to active learning labeling rates.

The next lemma characterizes the growth of $v_r$ when $\alpha = 1$.

**Lemma 9** *Suppose $\hat{B} \geq \max\{4b, 4\}$. When $\alpha = 1$, for any $r \geq 0$, $v_r \geq 4^r$.*

*Proof.* When $r = 1$, $v_r = \hat{B} \geq 4$, as desired. Moreover for any $r > 1$, $p_0(v_1^r) \leq p_0(v_1) = b/\hat{B} \leq 1/4$. And so $\min\{1, p_0(v_1^r)\} = p_0(v_1^r)$.

Proceeding by induction, we suppose that $v_r \geq 4^r$. And conclude:

$$4^{r+1} \leq 4^r(\hat{B}/b) = \frac{\hat{B}}{p_0(4^r)} \leq \frac{\hat{B}}{p_0(v_r)} \leq \frac{\hat{B}}{p_0(v_1^r)} \leq \frac{\hat{B}}{\min\{1, p_0(v_1^r)\}} = v_{r+1}.$$

□

We now have a bound of the form required by Theorem 4. And can state the following corollary.

**Corollary 4** *Let $\alpha = 1$. Suppose $\hat{B} \geq \max\{4b, 4\}$. For any $0 \leq a \leq 1$, and $T \geq 0$,*

$$\text{If } r \geq \frac{2a}{\hat{B}}T + \log_2(T), \text{ then } w_1^r \geq T.$$

*Proof.* Combining Lemma 8 and Lemma 9 we can bound $w_s \geq \min\{\frac{\hat{B}}{2a}, 2^r\}$ (allowing $\frac{\hat{B}}{2a} = \infty$ if $a = 0$). Applying Theorem 4 with $h(r) = 2^r$, gives us $H(r) = 2^{r+1} - 1$, and $H^{-1}(y) = \log_2(\frac{y+1}{2})$. Since $w_1^r$ is strictly increasing in $r$, the corollary follows by observing that $\log_2(\frac{y+1}{2}) \leq \log_2(y)$, when $y \geq 1$. □

### A.7 $O(\frac{1}{\sqrt{t}})$ Time Decreasing Labeling Rate

We next characterize the growth of $v_r$ when $\alpha = 1/2$. As in the previous analysis, we first relate the growth of $w_r$ to $v_r$,.

**Lemma 10** *When $\alpha = 1/2$, for all $r \geq 1$, $w_r \geq \min\{\frac{\hat{B}}{2a}, \frac{1}{4}v_r\}$.*

*Proof.* Define $s_0$ and $s_1$ as in the proof of Lemma 8. For rounds $r < s_0$, $w_r = v_r$, and for rounds $r \geq s_1$, $w_r \geq \frac{\hat{B}}{2a}$. This follows by identical arguments to Lemma 8, and so we omit them here. Moreover, just as in that lemma we have that $v_r = \frac{\hat{B}}{p_0(v_1^{r-1})}$ for all $r \geq s_0$.

We therefore proceed to prove that $\frac{1}{4}v_r \leq w_r$ for $s_0 \leq r < s_1$ by induction, where the base case ($s < s_0$) is established in the preceding. For the inductive step, we assume $\frac{1}{4}v_s \leq w_s$ for all $s < r$. The inductive hypothesis implies that $\frac{1}{4}v_1^{r-1} \leq w_1^{r-1}$. Therefore: $p_0(w_1^{r-1}) \leq p_0(\frac{1}{4}v_1^{r-1}) = 2p_0(v_1^{r-1})$ when $\alpha = 1/2$, which implies:

$$\frac{1}{4}v_r = \frac{1}{4}\frac{\hat{B}}{p_0(v_1^{r-1})} \leq \frac{\hat{B}}{2p_0(w_1^{r-1})} \leq \frac{\hat{B}}{a + p_0(w_1^{r-1})} \leq w_r$$

The first equality follows $v_r = \frac{\hat{B}}{p_0(v_1^{r-1})}$ when $r \geq s_0$. The next inequality follows because $\frac{1}{2}p_0(w_1^{r-1}) \leq p_0(v_1^{r-1})$. The next inequality follows from the definition of $s_1$ (so $p(w_1^{r-1}) \geq a$). And the final inequality follows by taking the min in the definition of $w_r$. This completes the induction and the proof. □

**Lemma 11** *Suppose $\hat{B} > b$. For any $\epsilon > 0$, define $r_\epsilon = \lceil \log_2(1/\epsilon) \rceil$, and $r_0 = \lceil \frac{b^2}{\hat{B}} \rceil$. Then for any $r \geq 1$, $\frac{r}{4}\left(\frac{\hat{B}}{b}\right)^{2-\epsilon} \leq v_{r_0+r_\epsilon+r}$.*

*Proof.* We first characterize a round $r_0$ after which the min in the definition of $v_r$ no longer takes effect. Taking $r_0 = \lceil \frac{b^2}{\hat{B}} \rceil$, we know that $v_1^{r_0} \geq \hat{B}r_0 \geq b^2$. Since $v_1^r$ is increasing in $r$ and $p_0$ is decreasing in its argument, for any $r > r_0$, we have the $p_0(v_1^{r-1}) \leq p_0(v_1^{r_0}) \leq p_0(b^2) = 1$, by definition of $p_0$ when $\alpha = 1/2$.

Next we claim that for any integer $x > 0$:

$$\left(\frac{\hat{B}}{b}\right)^{2-1/2^{x-1}} \leq v_{r_0+x} \, . \tag{4}$$

We proceed by induction. If $x = 1$, then $\frac{\hat{B}}{b} \leq \frac{\hat{B}}{b}\sqrt{v_1^{r_0}} = \frac{\hat{B}}{p_0(v_1^{r_0})} = v_{r_0+1}$, where the first inequality follows because $r_0 \geq 1$ by definition. Now assume the inequality $\left(\frac{\hat{B}}{b}\right)^{2-1/2^{x-1}} \leq v_{r_0+x}$ holds for some $x$. Then:

$$\begin{aligned}\left(\frac{\hat{B}}{b}\right)^{2-1/2^{x}} &= \left(\frac{\hat{B}}{b}\right)\left(\frac{\hat{B}}{b}\right)^{1-1/2^{x}} = \left(\frac{\hat{B}}{b}\right)\sqrt{\left(\frac{\hat{B}}{b}\right)^{2-1/2^{x-1}}} \\ &\leq \left(\frac{\hat{B}}{b}\right)\sqrt{v_{r_0+x}} = \frac{\hat{B}}{p_0(v_{r_0+x})} \leq \frac{\hat{B}}{p_0(v_1^{r_0+x})} = v_{r_0+x+1} \, ,\end{aligned}$$

which completes the inductive step for (4).

We now prove the lemma by using an additional induction. For an arbitrary $\epsilon > 0$, let $r_\epsilon = \lceil \log_2(1/\epsilon) \rceil$. Since $\hat{B} > b$, (4) implies that $\left(\frac{\hat{B}}{b}\right)^{2-\epsilon} \leq v_{r_0+r_\epsilon+r}$ for all $r \geq 1$. This also establishes the base case for the proof of the Lemma. Finally assume that the lemma holds for any $s \leq r$. Note that $\frac{r+1}{4} = \left(\frac{(r+1)^2}{16}\right)^{1/2} =$

$\left(\frac{r(r+1)}{16} + \frac{(r+1)}{16}\right)^{1/2} \leq \left(\frac{r(r+1)}{8}\right)^{1/2}$ when $r > 0$. Therefore:

$$\begin{aligned}
\frac{r+1}{4}\left(\frac{\hat{B}}{b}\right)^{2-\epsilon} &\leq \left(\frac{r(r+1)}{8}\right)^{1/2}\left(\frac{\hat{B}}{b}\right)^{2-\epsilon} \\
&\leq \left(\frac{r(r+1)}{8}\right)^{1/2}\left(\frac{\hat{B}}{b}\right)^{2-\epsilon/2} \\
&= \left(\sum_{s=1}^{r} s/4\right)^{1/2}\left(\frac{\hat{B}}{b}\right)^{2-\epsilon/2} \\
&= \left(\frac{\hat{B}}{b}\right)\left(\sum_{s=1}^{r} s/4\left(\frac{\hat{B}}{b}\right)^{2-\epsilon}\right)^{1/2} \\
&\leq \left(\frac{\hat{B}}{b}\right)\left(\sum_{s=1}^{r} v_{r_0+r_\epsilon+s}\right)^{1/2} \\
&\leq \left(\frac{\hat{B}}{b}\right)\sqrt{v_1^{r_0+r_\epsilon+r}} = v_{r_0+r_\epsilon+r+1}.
\end{aligned}$$

The second inequality follows because $\hat{B}/b > 1$. The third inequality follows from the induction hypothesis. And the final inequality follows because $v_1^{r_0+r_\epsilon+r}$ is a positive sum with more terms than $\sum_{s=1}^{r} v_{r_0+r_\epsilon+s}$. □

**Corollary 5** *Suppose $\alpha = 1/2$, $a \in [0,1]$ and $\hat{B} \geq b$ in the definition of $w_r$. For any $\epsilon > 0$ and $T \geq 0$, if*

$$r \geq \frac{2a}{\hat{B}}T + 4\sqrt{2}\left(\frac{b}{\hat{B}}\right)^{1-\epsilon/2}\sqrt{T} + \log_2(1/\epsilon) + \frac{b^2}{\hat{B}} + 2$$

*then $w_1^r \geq T$.*

*Proof.* Fix $\epsilon > 0$. Lemmas 10 and 11 let us conclude that $w_{r_0+r_\epsilon+r} \geq \min\{\frac{\hat{B}}{2a}, \frac{r}{16}(\frac{\hat{B}}{b})^{2-\epsilon}\}$ for any $r \geq 1$. Define a new sequence $\hat{w}_r = w_{r_0+r_\epsilon+r}$ consisting of just this suffix of the sequence $\{w_r\}$. We can apply Theorem 4 to this sequence with $Z = \frac{\hat{B}}{2a}$; $h(r) = cr$ with $c = \frac{1}{16}(\frac{\hat{B}}{b})^{2-\epsilon}$; $H(r) = \frac{c}{2}r^2 \leq cr(r+1)/2 = \sum_{s=1}^{r} h(s)$; and $H^{-1}(y) = \sqrt{\frac{2y}{c}}$. This implies that $\hat{w}_1^r \geq T$ when $r \geq Z^{-1}T + H^{-1}(T)$. The corollary follows from the fact that $w_1^{r+r_0+r_\epsilon} \geq \hat{w}_1^r$ and by removing the ceilings in the definitions of $r_0$ and $r_\epsilon$. □

**Corollary 6** *Suppose $\alpha = 1/2$, $a \in [0,1]$ and $\hat{B} \geq b$ in the definition of $w_r$. Define $b' = \max\{b, 1\}$. For any $T \geq 0$, if*

$$r \geq \frac{2a}{\hat{B}}T + 8\left(\frac{b'}{\hat{B}}\right)\sqrt{T} + \log_2\log_2(\hat{B}) + \frac{b^2}{\hat{B}} + 1$$

*then $w_1^r \geq T$.*

*Proof.* Corollary 5 has a free parameter $\epsilon > 0$. Taking $b' = \max\{b, 1\}$, we will now select an $\epsilon$ which makes the following two equations simultaneously true:

$$\left(\frac{b}{\hat{B}}\right)^{1-\epsilon/2} \leq \frac{\sqrt{2}b'}{\hat{B}} \qquad \text{and} \qquad \log_2(1/\epsilon) \leq \log_2\log_2(\hat{B}).$$

Substituting these back into Corollary 5 completes the proof. Selecting $\epsilon = \log_{\hat{B}}(2)$ implies:

$\left(\frac{b}{\hat{B}}\right)^{1-\epsilon/2} = \frac{1}{\hat{B}}\hat{B}^{\epsilon/2}b^{1-\epsilon/2} \leq \frac{1}{\hat{B}}\hat{B}^{\epsilon/2}{b'}^{1-\epsilon/2} \leq \frac{1}{\hat{B}}\hat{B}^{\epsilon/2}b' = \frac{\sqrt{2}b'}{\hat{B}}$. The first inequality follows because $b \leq b'$, and the second because $b' \geq 1$.

At the same time $\log_{\hat{B}}(2) = \frac{\log_2(2)}{\log_2(\hat{B})} = 1/\log_2(\hat{B})$, and so $\log_2(1/\epsilon) = \log_2 \log_2(\hat{B})$ as desired. □

Combining Corollaries 4 and 6 gives us Theorem 2.

### A.7.1 Technical Tools

We state a useful rearrangement Lemma needed in Lemma 2. In the context of our label complexity bounds, it states that we are worst-off when the $T_{\text{bad}}$ rounds which are significantly smaller than their mean occur at the end of time.

Let $g$ be a monotone increasing function on the reals. Consider the following optimization problem where $K \leq R$.

$$\begin{aligned}
&\underset{x}{\text{minimize}} && \sum_{r=1}^{R} x_r \\
&\text{s.t.} && \forall r, x_r \geq 0, \qquad |\{r \mid x_r = 0\}| = K \\
& && \forall r \in \{r \mid x_r > 0\},\ x_r \geq g(\sum_{s=1}^{r-1} x_s)
\end{aligned} \tag{5}$$

**Lemma 12** *Setting $x_{R-K+s} = 0$, for $1 \leq s \leq K$, $x_1 = g(0)$ and $x_r = g(\sum_{s=1}^{r-1} x_s)$ for $2 \leq r \leq R-K$ is an optimal solution to (5).*

*Proof.* Any solution where $|\{r \mid x_r = 0\}| < K$ is suboptimal. Modifying an arbitrary $x_r$ to 0 improves the objective and is still a feasible solution since $g$ is monotone increasing. Now consider an arbitrary solution with $|\{r \mid x_r = 0\}| = K$. Reindex the variables as $x'_1, \ldots, x'_{R-K}$, so that they appear in the same order as $x_1, \ldots, x_R$, but zero variables are skipped. In other words, $x'_r$ is the $r$th non-zero variable in the solution. Notice that the feasibility constraints on each of the non-zero variables can be written as $x'_r \geq g(\sum_{s=1}^{r-1} x'_s)$.

Any solution that does not set $x'_r = g(\sum_{s=1}^{r-1} x'_s)$ is suboptimal. Modifying $x'_r$ to $g(\sum_{s=1}^{r-1} x'_s)$ improves the objective and is still a feasible solution. Again, because $g$ is monotone increasing, modifying $x'_r$ to $g(\sum_{s=1}^{r-1} x'_s)$ only relaxes the constraint $x'_q \geq g(\sum_{s=1}^{q-1} x'_s)$ for $q > r$. Thus any optimal solution must set $x'_1 = g(0)$, and $x'_r = g(\sum_{s=1}^{r-1} x'_s)$ for $r > 1$. This is equivalent to the solution in the statement of the Lemma. □

# B Applications

## B.1 B-IWAL Supplement

We write the proofs of B-IWAL algorithm's generalization guarantee and the bound on its requesting probability.

**Lemma 13** *For all $\delta > 0$ with probability at least $1 - \delta$, for all $r > 1$, and for all $f, g \in H_r$,*

$$|L_{\tau_r}(f) - L_{\tau_r}(g) - L(f) + L(g)| \leq \Delta_{\tau_r}.$$

*Proof.* For any $f, g \in H_r$, define

$$Z_i = \frac{Q_i}{p_i}(\ell(f(x_i), y_i) - \ell(g(x_i), y_i) - (L(f) - L(g)),$$

for $i \in [1, \tau_r]$. The sequence $Z_i$ is a martingale difference since $\mathbb{E}[Z_i|Z_1, \ldots Z_{i-1}] = \mathbb{E}[\frac{Q_i}{p_i}(\ell(f(x_i), y_i) - \ell(g(x_i), y_i) - (L(f) - L(g))|Z_1, \ldots Z_{i-1}] = 0$ and since $|Z_i| \leq \frac{1}{p_i}|\ell(f(x_i)) - \ell(g(x_i))| + |L(f) - L(g)| \leq 2$ as $p_i \geq |\ell(f(x_i), y_i) - \ell(g(x_i), y_i)|$ for $f, g \in H_r$.

We then take a union bound over $\tau_r$ and apply Azuma's inequality.

$$\begin{aligned}
&\mathbb{P}[|L_{\tau_r}(f) - L_{\tau_r}(g) - L(f) + L(g)| \geq \Delta_{\tau_r}] \\
&\leq \mathbb{P}[\exists n \in [T]\colon |L_n(f) - L_n(g) - L(f) + L(g)| \geq \Delta_n] \leq \sum_{n=1}^{T} \frac{\delta}{T^2(T+1)|H|^2} = \frac{\delta}{T(T+1)|H|^2}.
\end{aligned}$$

Since $H_r$ is a random subset of $H$, we take the union bound over $f, g \in H$ and $r$. We then take another union bound over $T$ to conclude the proof. □

**Theorem 3** *Let $\widehat{h}_T$ denote the hypothesis returned by* B-IWAL *after $T$ time steps and let $h^* = \operatorname{argmin}_{h \in H} L(h)$. For any $\delta > 0$, with probability at least $1 - \delta$, $L(\widehat{h}_T) \leq L(h^*) + O\Big(\sqrt{\frac{\log(T|H|/\delta)}{T}}\Big)$.*

*Proof.* We show that $h^* \in H_r$ by induction. The base case holds as $h^* \in H_1 = H$. Now assuming that $h^* \in H_{r-1}$ holds, we show that $h^* \in H_r$. Let $h' = \operatorname{argmin}_{f \in H_{r-1}} L_{\tau_{r-1}}(f)$. By Lemma 13,

$$L_{\tau_{r-1}}(h^*) - L_{\tau_{r-1}}(h^{'}) \leq L(h^*) - L(h') + \Delta_{\tau_{r-1}} \leq \Delta_{\tau_{r-1}}$$

since $L(h^*) - L(h') \leq 0$. Thus, $L_{\tau_{r-1}}(h^*) \leq L_{\tau_{r-1}}(h^{'}) + \Delta_{\tau_{r-1}}$ which means that $h^* \in H_r$ by definition of $H_r$.

Since $H_r \subseteq H_{r-1}$, Lemma 13 implies that for any $f, g \in H_r$,

$$\begin{aligned} L(f) - L(g) &\leq L_{\tau_{r-1}}(f) - L_{\tau_{r-1}}(g) + \Delta_{\tau_{r-1}} \\ &\leq L_{\tau_{r-1}}(h') + \Delta_{\tau_{r-1}} - L_{\tau_{r-1}}(h') + \Delta_{\tau_{r-1}} \leq 2\Delta_{\tau_{r-1}}. \end{aligned}$$

Thus, for any $r \geq 1$,

1. $h^* \in H_r$
2. $L(f) - L(g) \leq 2\Delta_{\tau_{r-1}}$ for any $f, g \in H_r$

We can then complete the proof by using the fact that at time $T$, $\tau_r = T$ and that $\widehat{h}_T$ is defined with respect to $H_{R+1}$. □

**Lemma 4** *For $\delta > 0$, with probability at least $1 - \delta$, at any round $r$, $\mathbb{E}_x[p_r(x) | \tau_{r-1}] \leq 4\theta(L(h^*) + \Delta_{\tau_{r-1}})$.*

*Proof.* For all $h \in H$,

$$\rho(h, h^*) = \mathbb{E}[|\ell(h(x), y) - \ell(h^*(x), y)|] \leq L(h) + L(h^*).$$

From the proof of Theorem 3, with probability at least $1 - \delta$, $H_r \subset \{h \in H \colon L(h) \leq L(h^*) + 2\Delta_{\tau_{r-1}}\}$ and thus, for all $h \in H_r$,

$$\rho(h, h^*) \leq 2L(h^*) + 2\Delta_{\tau_{r-1}}.$$

Then, we can chose $\Lambda = (2L(h^*) + 2\Delta_{\tau_{r-1}})$ such that $H_r \subset \mathcal{B}(h^*, \Lambda)$. Thus,

$$\begin{aligned} \mathbb{E}[p_r \mid \tau_{r-1}] &= \mathbb{E}[\sup_{f,g \in H_r} \sup_y |\ell(f(x), y) - \ell(g(x), y)| \mid \tau_{r-1}] \\ &\leq 2\,\mathbb{E}[\sup_{h \in H_r} \sup_y |\ell(h(x), y) - \ell(h^*(x), y)| \mid \tau_{r-1}] \\ &\leq 2\,\mathbb{E}[\sup_{h \in B(h^*, \Lambda)} \sup_y |\ell(h(x), y) - \ell(h^*(x), y)| \mid \tau_{r-1}] \\ &\leq 2\,\mathbb{E}[\sup_{h \in B(h^*, \Lambda)} \sup_y |\ell(h(x), y) - \ell(h^*(x), y)| \mid \tau_{r-1}] \\ &\leq 2\theta\Lambda = 4\theta(L^* + \Delta_{\tau_{r-1}}), \end{aligned}$$

which completes the proof. □

### B.2 B-ORIWAL Supplement

Here, we reproduce the full argument for the derivation and analysis of B-ORIWAL.

At a high level, the ORIWAL algorithm of Cortes et al. [2019a] works by partitioning the space into regions and running a separate active learning algorithm in each region while carefully allocating the labeling resources across regions. Specifically, in each region, the ORIWAL runs the algorithm EIWAL, which is an enhanced version of IWAL with stronger theoretical guarantees. [4]

We first present some needed notation and recall the algorithm. We denote by $\mathcal{X}_k$ for $k \in [n]$ the regions that partition in the input $\mathcal{X}$ and by $H_k$ the hypothesis used in each region. The ORIWAL algorithm returns a hypothesis from the following region-based hypothesis set: $H_{[n]} = \{\sum_{k=1}^n 1_{x \in \mathcal{X}_k} h_k(x) \colon h_k \in H_k\}$. We define $L_k^* = \min_{h \in H_k} \mathbb{E}[\ell(h(x), y) | x \in \mathcal{X}_k]$ be the regional best-in-class and $\theta_k = \theta(\mathcal{D}_{\mathcal{X}_k}, H_k)$ to be the regional disagreement coefficient where $\mathcal{D}_{\mathcal{X}_k}$ is the conditional distribution of $x$ given region $k$.

At each time $t \in [T]$, ORIWAL receives the points $x_t$, finds the region $k_t$ it belongs to, and decides whether to pass this point to the sub-routine EIWAL in region $k_t$ by flipping a coin $A_t \in \{0, 1\}$ with bias $\alpha_{k_t}$. This bias probability is carefully chosen to minimize the label complexity across the regions: $\alpha_k = \frac{(c_k/\mathtt{p}_k)^{1/3}}{\max_{k \in [n]} (c_k/\mathtt{p}_k)^{1/3}}$ where $c_k = \log[\frac{16T^2 |H_k|^2 \log(T) n}{\delta}]$ and where $\mathtt{p}_k = \mathbb{P}[\mathcal{X}_k]$. If $A_t = 1$, then the point $x_t$ is passed to the EIWAL instance in region $k_t$, which decides whether to request the label $y_t$, and updates its internal state.

The B-ORIWAL extends the ORIWAL algorithm to the fixed batch setting by freezing the version spaces in each region for the length-of-round in the same way as was done in B-IWAL. Each regional EIWAL will request a number of points to get labeled and only when the total number of points requested by all regional algorithms equals $B$, we unfreeze the version in each region. We define B-ORIWAL in terms of Algorithm 1, where the state of the algorithm is defined by the set $\omega_r = \{H_{1,r}, \ldots H_{n,r}\}$ which is initialized as $\omega_1 = \{H_{1,1}, \ldots, H_{n,1}\}$ where $H_{k,1} = H, \forall k \in [n]$. The `Labeler` and `Updater` are defined in Pseudocode 4 and Pseudocode 5. In these pseudocodes, the empirical weighted loss and slack term[5] in region $k$ are defined with respect to the points passed to EIWAL$_k$. More concretely, they are defined with respect to the sequence of points, $\{(x_s, \bar{y}_s, Q_s, p_s) \colon x_s \in \mathcal{X}_k, A_s = 1\}$ for $s \in [\tau_r]$, as follows: $L_{\tau_{k,r}}(f) = \frac{1}{\tau_{k,r}} \sum_{s=1}^{\tau_{k,r}} \frac{Q_s}{p_s} L(f(x_s, \bar{y}_s))$ and $\Delta_{\tau_{k,r}} = \frac{2}{\tau_{k,r}} \Big( \sqrt{\sum_{s=1}^{\tau_{k,r}} p_s} + 6 \sqrt{\log \big( \frac{(3+T)T^3}{\delta} \big)} \Big) \times \sqrt{\log \big( \frac{8T^3 |H|^2 \log(T)}{\delta} \big)}$ where $\tau_{k,r} = |\{s \in [\tau_r] \colon x_s \in \mathcal{X}_k, A_s = 1\}|$.

The next theorem shows the generalization guarantee of B-ORIWAL.

**Theorem 5** *For any $B \geq 3$, let $\widehat{h}_T$ denote the hypothesis returned by* B-ORIWAL *after $T$ time steps and let $h^* = \operatorname{argmin}_{h \in H_{[n]}} L(h)$. For any $\delta > 0$, with probability at least $1 - \delta$,*

$$L(\widehat{h}_T) \leq L(h^*) + O\Bigg( \sum_{k=1}^n \mathtt{p}_k \sqrt{\frac{\theta_k L_k^*}{T_k} \log \left[ \frac{T |H_k| \log(T) n}{\delta} \right]} + \sum_{k=1}^n \frac{\mathtt{p}_k}{T_k} \log^2 \Big( \max_{k \in [n]} T |H_k| n / \delta \Big) \Bigg).$$

*where $T_k$ is the number of queries made to* EIWAL$_k$.

*Proof Sketch.* At each round $r > 0$ and $B \geq 3$, we prove that in each region $k$, for all $\delta > 0$, and for all $f, g, \in H_{k,r}$, with probability at least $1 - 2\delta$, the following holds.

$$|L_{\tau_{k,r}}(h) - L_{\tau_{k,r}}(g) - L^k(h) + L^k(g)| \leq \Delta_{\tau_{k,r}} \tag{6}$$

where $L^k(h) = \mathbb{E}[\ell(h(x), y) | x \in \mathcal{X}_k]$ is the expected loss of $h$ given region $k$ and recalling that $L_{\tau_{k,r}}(h)$ is the empirical loss estimate of $h$.

To prove the above, we first use Lemma 4 [Cortes et al. [2019a]] and apply a union bound over $T_k$, $\tau_{k,r}$, pair $(f, g)$. Then, via the same reasoning as in Lemma 5 [Cortes et al. [2019a]], the Inequality 6 holds and specifically,

$$L^k(\widehat{h}_{k,T}) \leq L_k^* + \frac{2}{T_k} \left( \sqrt{\sum_{s=1}^{T_k} p_s} + 6 \sqrt{\log \Big( \frac{(3+T)T^3}{\delta} \Big)} \right) \sqrt{\log \Big( \frac{8T^3 |H_k|^2 \log(T)}{\delta} \Big)} \tag{7}$$

where $p_s$ is the probability of requesting $x'_s$, the $s$th point passed on EIWAL$_k$.

The bound in Lemma 6 [Cortes et al. [2019a]] implies that for the sequence of $x'_s$ passed to EIWAL$_k$ in region $k$,

[4] The main difference between IWAL and ORIWAL is the definition of the slack term, $\Delta_t$.

[5] We need a slightly different slack term than the original ORIWAL algorithm in that there is an extra $T$ in the log terms.

**Algorithm 4** `Labeler`$(H_r, x_t)$ for B-ORIWAL

```
$k_t \leftarrow k$ such that $x_t \in \mathcal{X}_k$
$A_t \sim \text{Bernoulli}(\alpha_{k_t})$
**if** $A_t = 1$ **then**
    $p_r(x_t) \leftarrow \max_{f,g \in H_{k_t,r}} \max_{y \in \mathcal{Y}} |\ell(f(x_t), y) - \ell(g(x_t), y)|$
**else**
    $p_r(x_t) \leftarrow 0$
**return** $p_r(x_t)$
```

**Algorithm 5** `Updater`$(H_r, \mathcal{Z}_\perp^r)$ for B-ORIWAL

```
**for** $k \in [n]$ **do**
    $H_{k,r+1} \leftarrow \{h \in H_{k,r}\colon L_{\tau_{k,r}}(h) \le \min_{h' \in H_{k,r}} L_{\tau_{k,r}}(h') + \Delta_{\tau_{k,r}}\}$
**return** $\{H_{1,r+1}, \ldots, H_{n,r+1}\}$
```

the following holds:

$$\begin{aligned}\sqrt{\sum_{s=1}^{T_k} p_s} &\le \sqrt{4\theta_k \left(L_k^* T_k + O(\sqrt{L_k^* T_k \log(T|H_k|/\delta)})\right)} + O(\log^{\frac{3}{2}}(T|H_k|/\delta)) \\ &\le \sqrt{4\theta_k L_k^* T_k} + O(\log^{\frac{1}{2}}(T|H_k|/\delta)) + O(\log^{\frac{3}{2}}(T|H_k|/\delta)) \\ &\le \sqrt{4\theta_k L_k^* T_k} + O(\log^{\frac{3}{2}}(T|H_k|/\delta)).\end{aligned}$$

By plugging in the above in Equation 7, taking a union bound over $n$, and multiplying by $\mathtt{p}_k$, we attain the bound of the theorem since $L(\widehat{h}_T) = \sum_{k=1}^n \mathtt{p}_k L^k(\widehat{h}_{k,T})$ and $L(h^*) = \sum_{k=1}^n \mathtt{p}_k L_k^*$. □

The above theorem bounds the expected loss of the algorithm's hypothesis by the expected loss of the best in class and by a term that depends on the probability of each region, $\mathtt{p}_k$. Despite the fact that the regional versions spaces are updated less frequently, the generalization of B-ORIWAL is of the same order as that of ORIWAL algorithm.

Next, in order to apply the general theory analyzed in Section 3, we first prove that B-ORIWAL satisfies the time-decreasing labeling rate property.

**Lemma 14** *For any $B \ge \frac{4\log(2n/\delta)}{\min_{k \in [n]} \mathtt{q}_k}$, with probability at least $1-\delta$ on any round $r$, the following bound holds for* B-ORIWAL *algorithm*

$$\mathop{\mathbb{E}}_x[p_r(x)|\tau_{r-1}] \le \sum_{k=1}^n 4\mathtt{q}_k \theta_k L_k^* + O\left(\frac{\theta_k \log^2(T|H_k|n/\delta)}{\tau_{r-1}} + \theta_k \sqrt{\frac{\mathtt{q}_k L_k^* \log(T|H_k|n/\delta)}{\tau_{r-1}}}\right),$$

*where* $\mathtt{q}_k = \frac{\mathtt{p}_k \alpha_k}{\sum_{k'=1}^n \mathtt{p}_{k'} \alpha_{k'}}$.

*Proof Sketch.* By same reasoning as in Lemma 6 [Cortes et al. [2019a]], the probability of requesting $x_t$ at round $r$ in region $k$ is bounded by $\mathbb{E}[p_r(x_t)|x_t \in \mathcal{X}_k, A_t = 1, \tau_{r-1}] \le 4\theta_k L_k^* + O(\theta_k \sqrt{\log(T_k|H_k|n/\delta)})\sqrt{\frac{L_k^*}{\tau_{k,r-1}}} + \frac{O(\theta_k \log^2(T_k|H_k|n/\delta))}{\tau_{k,r-1}}$ after taking union bound over $n$.

We then multiply by $\mathtt{q}_k$ and sum over the regions since $\mathtt{q}_k$ is the probability that a point falls in region $k$ and is passed to EIWAL$_k$:

$$\mathop{\mathbb{E}}_x[p_r(x)|\tau_{r-1}] \le \sum_{k=1}^n 4\mathtt{q}_k \theta_k L_k^* + \mathtt{q}_k O(\theta_k \sqrt{\log(T_k|H_k|n/\delta)})\sqrt{\frac{L_k^*}{\tau_{k,r-1}}} + \mathtt{q}_k \frac{O(\theta_k \log^2(T_k|H_k|n/\delta))}{\tau_{k,r-1}}. \tag{8}$$

Then by applying a standard Chernoff bound, with probability at least $1 - \frac{\delta}{2}$, for all $k \in [n]$,

$$\frac{\tau_{k,r-1}}{\tau_{r-1}} \ge \mathtt{q}_k \Big(1 - \sqrt{\frac{2\log(2n/\delta)}{\tau_{r-1}\mathtt{q}_k}}\Big).$$

Thus, with probability at least $1 - \frac{\delta}{2}$, for all $k \in [n]$,

$$\frac{\mathsf{q}_k}{\sqrt{\tau_{k,r-1}}} = \sqrt{\frac{\mathsf{q}_k}{\tau_{r-1}}}\sqrt{\frac{\mathsf{q}_k}{(\tau_{k,r-1}/\tau_{r-1})}} \leq \sqrt{\frac{\mathsf{q}_k}{\tau_{r-1}}}\frac{1}{\sqrt{1-\sqrt{\frac{2\log(2n/\delta)}{\tau_{r-1}\mathsf{q}_k}}}}.$$

When $\tau_{r-1} \geq \frac{4\log(2n/\delta)}{\min_{k\in[n]}\mathsf{q}_k}$ (which holds since $B \geq \frac{4\log(2n/\delta)}{\min_{k\in[n]}\mathsf{q}_k}$), then $\frac{2\log(2n/\delta)}{\tau_{r-1}\mathsf{q}_k} < \frac{1}{2}$. Since $\frac{1}{\sqrt{1-\sqrt{x}}} \leq 1 + 2\sqrt{x}$ for any $x \leq \frac{1}{2}$, it follows that

$$\begin{aligned}\frac{\mathsf{q}_k}{\sqrt{\tau_{k,r-1}}} &\leq \sqrt{\frac{\mathsf{q}_k}{\tau_{r-1}}}\Big(1 + 2\sqrt{\frac{2\log(2n/\delta)}{\mathsf{q}_k}}\Big)\\ &= \sqrt{\frac{\mathsf{q}_k}{\tau_{r-1}}} + \frac{2\sqrt{2\log(2n/\delta)}}{\tau_{r-1}}.\end{aligned}$$

Then, by plugging the above inequality into Equality 8, we attain the bound of the theorem. □

To give a bound on the total number of labels requested, we split into two cases when $L_k^* = 0$ for all $k > 0$ and when $\exists k \in [n]$ such that $L_k^* > 0$. For the first case, we let $a = 0$, $b = \sum_{k=1}^n \theta_k \log^2(T|H_k|n/\delta)$ and $\alpha = 1$ while for the second case, we let $a = \sum_{k=1}^n 4\mathsf{q}_k\theta_k L_k^*$, $b = O(\sum_{k=1}^n \theta_k\sqrt{\mathsf{q}_k L_k^*}\log(T|H_k|n/\delta))$ and $\alpha = 1/2$. We then apply Theorem 2 and Theorem 1 in each case to conclude the following bound.

**Corollary 2** *Fix $\delta < \sqrt{1/e}$, time horizon $T \geq 1$, and batch size $B > \max\{16b, 16, 2\log(T), \frac{4\log(2n/\delta)}{\min_{k\in[n]}\mathsf{q}_k}\}$. Then with probability at least $1 - 2\delta$, the total labels requested by B-ORIWAL is bounded as follows:*

*when $L_k^* = 0$ for all $k \in [n]$,*

$$O\big(B\log_2(T) + B\log(1/\delta) + B\big)$$

*and when $\exists k \in [n]$ such that $L_k^* > 0$,*

$$\widetilde{O}\big(\textstyle\sum_{k=1}^n \mathsf{q}_k\theta_k L_k^* T + \theta_k\sqrt{T} + B\big),$$

*where $\mathsf{q}_k = \mathsf{p}_k\alpha_k / \sum_{k'=1}^n \mathsf{p}_{k'}\alpha_{k'}$.*

In the case when $L_k^* = 0$ for all $k \in [n]$, the bound is a very favorable logarithmic bound in the time horizon $T$. In the second case, the bound admits an additive term in the batch size $B$ and a term that depends on $\sqrt{T}$ scaled by the regional disagreement coefficient $\theta_k$.

### B.3 B-DHM Supplement

Here, we reproduce the full argument for the derivation and analysis of B-DHM.

We extend the DHM algorithm [Dasgupta et al., 2008] to the batch setting where we again start by recalling the algorithm. Given a point $x_t$, the DHM algorithm decides to either request the label or assigns it a carefully chosen pseudolabel. Specifically, it constructs two sets, $\widehat{S}_t$ and $T_t$, such that $\widehat{S}_t$ contains examples with pseudolabels consistent with the best-in-class $h^*$ and $T_t$ contains examples with requested labels. The union of these two sets is thus an i.i.d. sample from the underlying marginal distribution on the input space. To decide whether pseudo-label or request the label for given a point $x_t$, the algorithm checks if the difference of the empirical error on $(\widehat{S}_{t-1}, T_{t-1})$ of two hypothesis learned via $h_{\widehat{y}} = \text{LEARN}_H(\widehat{S}_{t-1} \cup \{x_t, \widehat{y}\}, T_{t-1})$ for $\widehat{y} \in \{\pm 1\}$ is large enough. The $\text{LEARN}_H(A, B)$ denotes a learning algorithm that either returns hypothesis $h \in H$ consistent with $A$ and with minumum error on $B$ or raises a flag whenever there is no hypothesis consistent on $A$. Notabtly, [Dasgupta et al., 2008] prove that whenever the gap between the error differences is large enough when using a zero-one loss, the algorithm can infer how $h^*$ will label this point and thus, it is safe to use a pseudo-label that matches the prediction of $h^*$ with high probability.

The B-DHM algorithm extends DHM algorithm by freezing the set $(\widehat{S}_r, T_r)$ at each round $r$ used by $\text{LEARN}_H$. That is, the decision whether to label a point at time $t$ depends on learning hypothesis over the points in the previous round. In terms of the general batch Algorithm 1, the state of the algorithm is given by $\omega_r = (\widehat{S}_r, T_r)$

**Algorithm 6** `Labeler`$((\widehat{S}_r, T_r), x_t)$ for B-DHM

```
for all $\widehat{y} \in \{\pm 1\}$ do
    $h_{\widehat{y}} \leftarrow \text{LEARN}_H(\widehat{S}_r \cup \{(x_t, \widehat{y})\}, T_r)$
if $(\exists \widehat{y} \in \{\pm 1\}, h_{-\widehat{y}} \leftarrow \emptyset$ or
 $L_{\tau_r}(h_{-\widehat{y}}) - L_{\tau_r}(h_{\widehat{y}}) > \Delta_{\tau_r})$ then
    $\widehat{y}_t \leftarrow \widehat{y}$
    $p_r(x_t) \leftarrow 1$
else
    $p_r(x_t) \leftarrow 0$
return $p_r(x_t)$
```

**Algorithm 7** `Updater`$((\widehat{S}_r, T_r), \mathcal{Z}_\perp^r)$ for B-DHM

```
$\widehat{S}_{r+1} \leftarrow \widehat{S}_r$
$T_{r+1} \leftarrow T_r$
for $s \in [\tau_r, \tau_{r+1}]$ do
    if $\bar{y}_s = \perp$ then
        $\widehat{S}_{r+1} \leftarrow \widehat{S}_{r+1} \cup (x_s, \widehat{y}_s)$        ▷ $\widehat{y}_s$ defined in Labeler
    else
        $T_{r+1} \leftarrow T_{r+1} \cup (x_s, y_s)$
return $(\widehat{S}_{r+1}, T_{r+1})$
```

where $\omega_1 = (\emptyset, \emptyset)$. In this section, since DHM holds only for the zero-one loss, we let $L_{\tau_r}(h) = \frac{1}{\tau_r} \sum_{i \in Z_r} 1_{h(x_i) \neq y_i}$ where $Z_r = \widehat{S}_r \cup T_r$ and $\tau_r = |Z_r|$. The `Labeler` and `Updater` are given in Pseudocode 6 and Pseudocode 7, respectively.

The following theorems prove the generalization guarantees and label complexity bound for B-DHM.

**Theorem 6** *Let $\widehat{h}_T$ denote the hypothesis returned by* B-DHM *after $T$ time steps, $h^* = \operatorname{argmin}_{h \in H} L(h)$ and let $d = VCDim(H)$. For a fixed constant $c > 0$, with probability at least $1 - \delta$, $L(\widehat{h}_T) \leq L(h^*) + c\big(\frac{1}{T}(d \log(T) + \log(1/\delta)) + \sqrt{\frac{L(h^*)}{T}(d \log(T) + \log(1/\delta))}\big)$.*

*Proof Sketch.* First, we note that Lemma 2 of Dasgupta et al. [2008] is a general result that holds for i.i.d. samples. Let $\mathcal{S}(H, t)$ is the $t$th shattering coefficient defined as the maximum number of ways a set of $t$ points can be labeled by $H$. Then, in order to apply Corollary 1 of that same paper to $(\widehat{S}_r, T_r)$ at each round $r$, we need to apply a union bound over $\tau_r$ and thus also need a slightly different definition of $\beta_t = \sqrt{(4/t) \log(8t(t^2 + t)\mathcal{S}(H, 2t)^2/\delta)}$ where we added an extra $t$ in the log term.

Next, we show that for any general iteration where a batch of points $\widetilde{S} = \widehat{S}_r - \widehat{S}_{r+1}$ is added to $\widehat{S}_r$, $h^*$ still consistent with $\widehat{S}_r \cup \widetilde{S} = \widehat{S}_{r+1}$. To see this, note that we can use Corollary 1 of Dasgupta et al. [2008] in conjunction with $\Delta_{\tau_r}$ as before, to show that with high probability, $h^*$ is consistent with $\widehat{S}_r \cup \{(x_t, \widehat{y})\}$ for all $(x_t, \widehat{y}) \in \widetilde{S}$. This is essentially reusing the argument from the non-batch setting, imagining that we are adding only a single point from $\widetilde{S}$ to the set $\widehat{S}_t$. However, if $h^*$ is consistent for each individual point in $\widetilde{S}$, it is also consistent with the union of the points and, thus, consistent for $\widehat{S}_r \cup \widetilde{S}$. Repeating this argument for every batch $\widetilde{S}$ that is added to $\widehat{S}_r$, we see that with high probability for all $t$, $h^*$ is consistent with $\widehat{S}_{r+1}$.

The remainder of the proof, follows exactly as in the proof of Theorem 1 of Dasgupta et al. [2008]. □

The following lemma bounds the probability of requesting a label during a round. Note that, unlike in the IWAL-based algorithms, the decision to draw a sample does not depend on a Bernoulli coin flip, but rather the decision is made deterministically based on the value $x_t$ and the data observed to that point. Nonetheless, in order to unify the presentation, we can still conceptually define a Bernoulli parameter $p_r(x)$ as in the IWAL-based algorithms, but that only take values in $\{0, 1\}$. Given a slightly different definition of disagreement coefficient $\theta'$ given by Definition 2 in Dasgupta et al. [2008], the following bound on requesting probability holds.

**Lemma 15** *For a constant $c > 0$, with probability at least $1 - \delta$ on any round $r$, the following bound holds for*

B-DHM *algorithm*

$$\mathop{\mathbb{E}}_{x}[p_r(x)|\tau_{r-1}] \leq c\theta'\Big(L(h^*) + \tilde{O}\Big(\frac{d\log(\tau_{r-1})}{\tau_{r-1}}\Big)\Big).$$

*Proof Sketch.* First note that

$$\mathop{\mathbb{E}}_{x}[p_r(x)|\tau_{r-1}] = \mathop{\mathbb{P}}_{x}[p_r(x)=1|\tau_{r-1}]$$

since $p_r(x) \in \{0,1\}$ for DHMand where the last expression is the probability that a label is requested by the DHM algorithm in round $r$ given a fixed history. This is precisely the quantity that is bounded with high probability in Lemma 5 of Dasgupta et al. [2008] for the standard (non-batch) DHM algorithm. Using the same arguments, which require with high probability concentration of the empirical error and that $h^*$ is consistent with $\widehat{S}_r$, we can show the bound of the theorem. □

Then, by letting $a = c\theta' L(h^*)$, $b = \tilde{O}(b\log(T))$, and $\alpha = 1$, we can apply Theorem 2 and Theorem 1 to attain a bound on the total number of labels requested.

**Corollary 3** *Fix $\delta < \sqrt{1/e}$, constant $c > 0$, time horizon $T \geq 1$, and batch size $B > \max\{16b, 16, 2\log(T)\}$. Then with probability at least $1-2\delta$, the total labels requested by* B-DHM *is bounded by:* $O\big(c\theta' L(h^*)T + B\log_2(T) + B\log(1/\delta) + B\big)$.

Since the label requesting probability decreases at a faster rate than that of IWAL, the resulting label complexity bound is much more favorable in that it admits a logarithmic dependency on time horizon $T$ while IWAL admitted a $\sqrt{T}$ dependency.

## C Further Empirical Results

In this section we present the full empirical results for the remainder of the datasets mentioned in Section 5. For completeness, we also again present the datasets found in the body of the paper.

The details for each dataset are presented below.

| dataset | # features | # “unlabeled” | # test | $\lvert H\rvert$ | $1/C$ |
|---|---|---|---|---|---|
| a9a | 123 | 20000 | 12,561 | 2048 | $2^{-11}$ |
| cod-rna | 8 | 25000 | 34535 | 4096 | $2^{-13}$ |
| covtype | 54 | 400000 | 181012 | 4096 | $2^{-13}$ |
| HIGGS | 28 | 90000 | 10000 | 2048 | $2^{-13}$ |
| mnist | 780 | 30000 | 30000 | 1024 | $2^{-10}$ |
| phishing | 68 | 9000 | 2055 | 2048 | $2^{-11}$ |

**Additional Dataset Preprocessing:** The HIGGS data was uniformly at random subsampled to 100,000 points as a preprocessing step. The `mnist` dataset is converted to a binary classification problem by classifying odd vs. even digits.

**Finite Hypothesis Set Construction:** In all cases, the finite hypothesis set consists of logistic regression models trained using the scikit-learn library [Pedregosa et al., 2011] with `solver=‘‘liblinear’’` and with L2 regularization parameter $C$ set as indicated in the table. The hypotheses are each trained using a small random sample of data, with a size uniformly selected between 30 and 500 data points.

**Details on Plotting Methodology** All plots are shown as a function of the number of observed datapoints, $T$. Since the number of observations made before a batch label request is made is a random variable, the number of observations needed before a first batch request is made is also random. Thus, the first datapoint for each method displayed in the plots occurs at the smallest value $T$ at which *all* 10 trials have already made a batch label request.

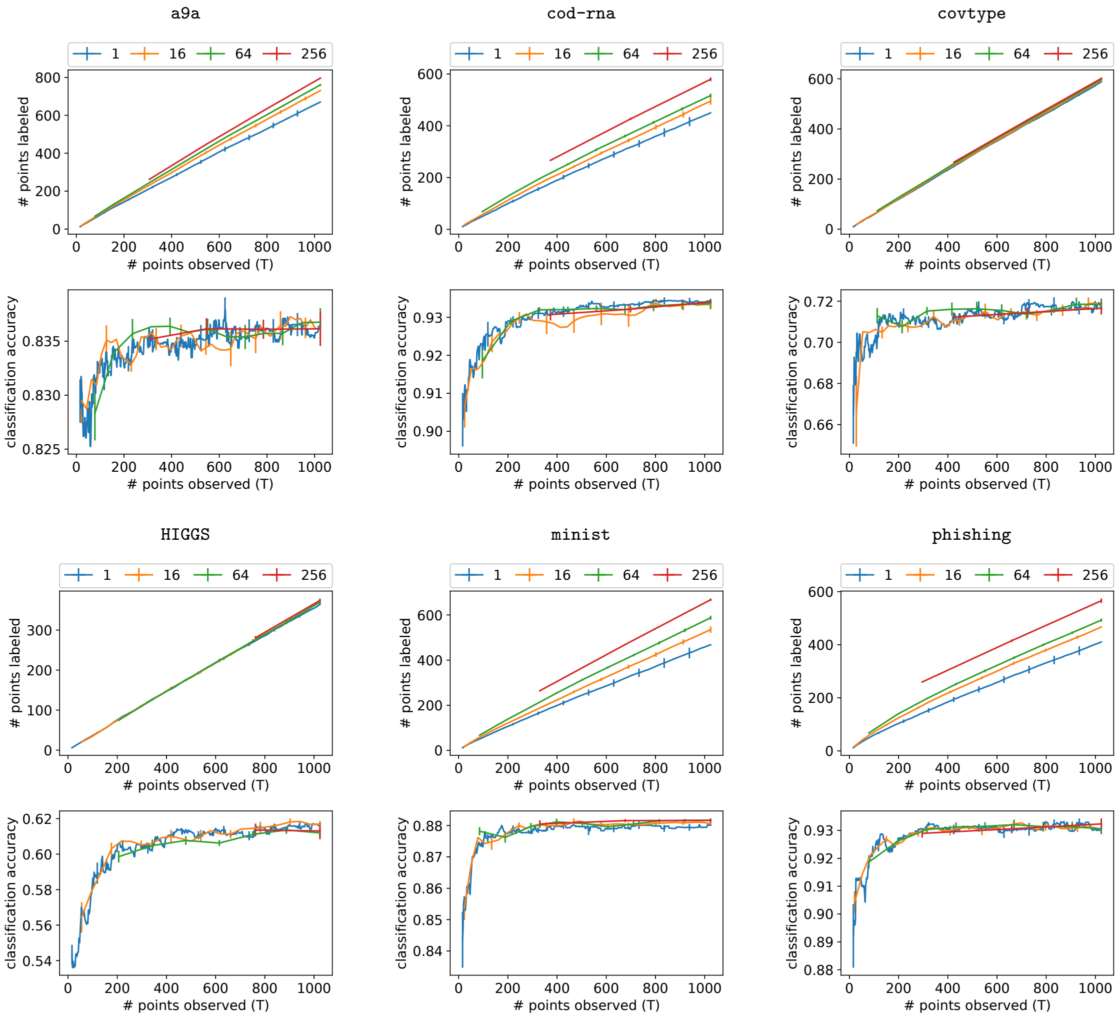


Figure 2: Here we plot the number of points labeled as well as the accuracy of the B-IWAL algorithm for different label query batch sizes $B \in \{1, 16, 64, 256\}$. Note in the case $B = 1$ the learner receives the label of an example as soon as a request is made, i.e. no batching occurs.

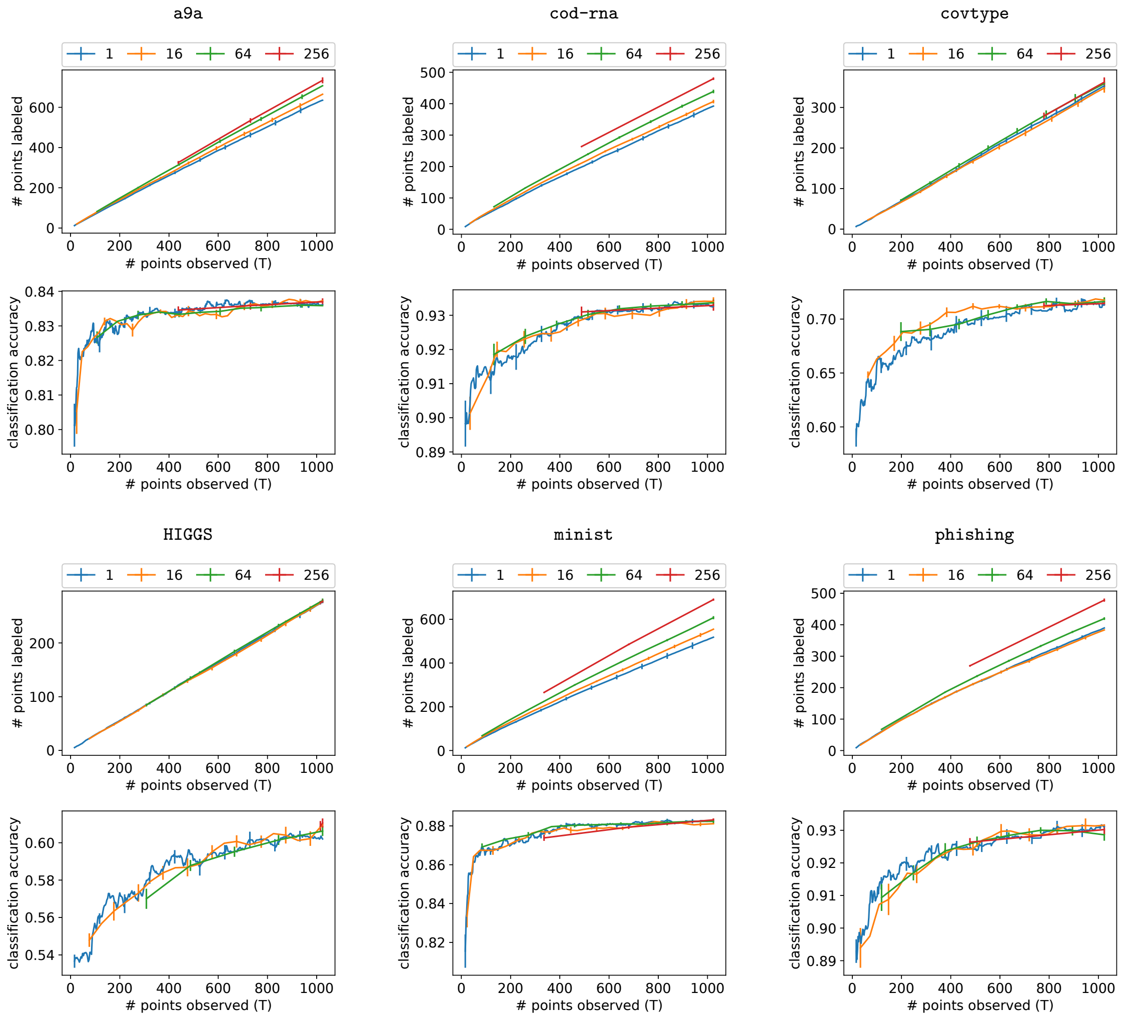


Figure 3: Here we plot the number of points labeled as well as the accuracy of the B-ORIWAL algorithm for different label query batch sizes $B \in \{1, 16, 64, 256\}$. Note in the case $B = 1$ the learner receives the label of an example as soon as a request is made, i.e. no batching occurs.

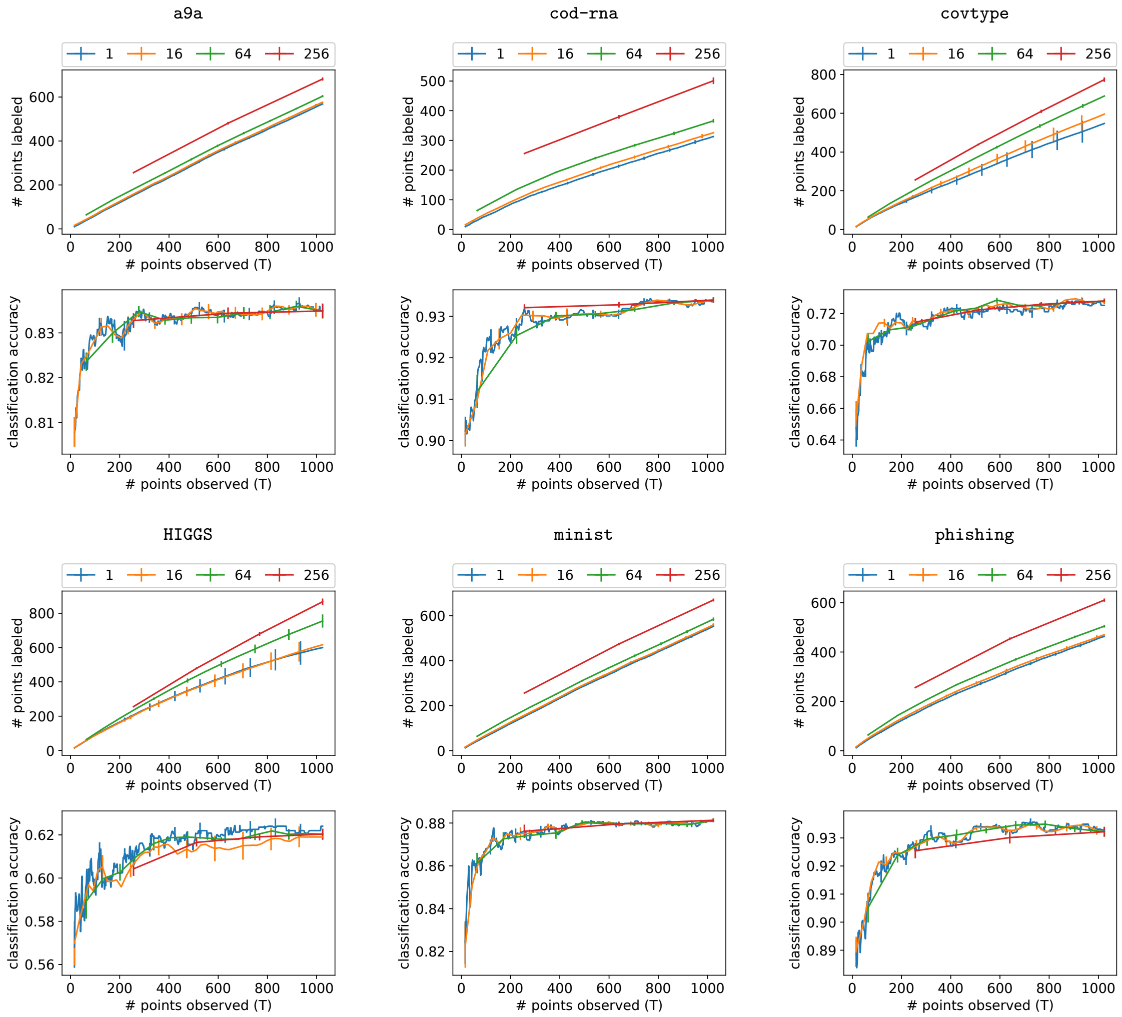


Figure 4: Here we plot the number of points labeled as well as the accuracy of the B-DHM algorithm for different label query batch sizes $B \in \{1, 16, 64, 256\}$. Note in the case $B = 1$ the learner receives the label of an example as soon as a request is made, i.e. no batching occurs.