Learning with Labeling Induced Abstentions

Kareem Amin
Google Research
New York, NY
kamin@google.com

Giulia DeSalvo
Google Research
New York, NY
giuliad@google.com

Afshin Rostamizadeh
Google Research
New York, NY
rostami@google.com

Abstract

Consider a setting where we wish to automate an expensive task with a machine learning algorithm using a limited labeling resource. In such settings, examples routed for labeling are often out of scope for the machine learning algorithm. For example, in a spam detection setting, human reviewers not only provide labeled data but are such high-quality detectors of spam that examples routed to them no longer require machine evaluation. As a consequence, the distribution of examples routed to the machine is intimately tied to the process generating labels. We introduce a formalization of this setting, and give an algorithm that simultaneously learns a model and decides when to request a label by leveraging ideas from both the abstention and active learning literatures. We prove an upper bound on the algorithm's label complexity and a matching lower bound for any algorithm in this setting. We conduct a thorough set of experiments including an ablation study to test different components of our algorithm. We demonstrate the effectiveness of an efficient version of our algorithm over margin sampling on a variety of datasets.

1 Introduction

In this paper, we consider a system that relies on automated predictions made by a machine learning model. We assume this system has a limited budget for requesting ground-truth labels (e.g. from a domain expert). In practice, such request can be used, among other purposes, to gather additional training data for the machine learning model. If the system asks for a label for an example, then it no longer needs the model's prediction for that particular example. Thus, the pattern of label queries effectively defines the distribution that the model will learn with and predict on.

Take for example a large-scale video-hosting website. The website wants to automatically detect videos that violate its community guidelines. In order to acquire labels for this task, some of these videos are evaluated by a finite pool of human reviewers. There are two consequences of this evaluation. Firstly, the reviewer provides a training label for the model. Secondly, the human intervention makes it so that the machine learning algorithm is no longer tasked with making predictions on these examples. As a result, the goal is then optimize the the model's performance in the domain where it will be executed.

We take the perspective of a system designer who wants to understand (and optimize for) the performance of the automated system on the examples that it will be asked to evaluate. Let $r : \mathcal{X} \to \{0, 1\}$ be a rule governing whether the system requests a label for $x \in \mathcal{X}$. Let $h : \mathcal{X} \to \mathbb{R}$ be a hypothesis describing the automated system. We seek to minimize $\mathbb{E}[L(h(x), y) \mid r(x) = 0]$, where L is a loss function. We can think of this setting as combining the objective studied in *abstention learning* (minimizing $E[L(h(x), y) \mid r(x) = 0]$) with the feedback model studied in *active learning* (labels are only available when $r(x) = 1$). We introduce a new framework, which we call *dual purpose learning* framework, that combines these elements.

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

We first analyze the *proper* dual purpose labeling framework in an online setting, which proceeds as follows. At the start of each round t, the learner selects both a requester function r_t from some class $\mathcal{R}_\rho$, where ρ is an upper bound on the request-rate of functions in this class, and a hypothesis h_t from some class $\mathcal{H}$ using all the feedback available from the past. The learner's expected loss on this round is given by $E[L(h_t(x), y) \mid r_t(x) = 0]$. The function r_t also determines the feedback available to the algorithm. If $r_t(x) = 1$, the learner observes (x, y), which was drawn i.i.d. from an unknown distribution, and otherwise, only x is revealed and y is censored. The goal of the learner is to compete with the optimal choice of hypothesis and requester by minimizing the excess loss $\mathcal{L}_{\mathcal{H},\mathcal{R}_\rho}(h_t, r_t) = E[L(h_t(x), y) \mid r_t(x) = 0] - \inf_{(h^*,r^*)\in\mathcal{H}\times\mathcal{R}_\rho} \mathbb{E}[L(h^*(x), y) \mid r^*(x) = 0]$.

Our first main result is the surprising fact that, under mild assumptions, bounds on the excess loss that match the $O(1/\sqrt{t})$ generalization rates of full-feedback passive learning are not possible in the proper dual purpose labeling framework. This lower bound also suggests a relaxation of the proper dual purpose labeling framework, which we call *improper dual purpose labeling*. In the improper setting, the learner is still interested in learning h_t, r_t that minimize $\mathcal{L}_{\mathcal{H},\mathcal{R}_\rho}$. However, the learner is allowed to request labels using a more powerful requester class than $\mathcal{R}_\rho$ during training. As in classical PAC-learning results, improper learning allows the circumvention of the impossibility result.

As a practical matter, the improper setting is useful when the system designer is willing to spend more resources during training. In our motivating example, the designer of the abuse-detection system might be willing to implement a more complicated system during training, which might include a larger budget (in dollars or man-power) for human intervention. However, after a time horizon T, training stops, and the designer commits to some h_T, r_T for $r_T \in \mathcal{R}_\rho$. From then on, examples satisfyng $r_T(x) = 0$ are routed to h_T, and thus, we wish to characterize $\mathcal{L}_{\mathcal{H},\mathcal{R}_\rho}(h_T, r_T)$.

In the improper setting, we demonstrate that IWAL, an algorithm from the active learning literature [Beygelzimer et al., 2009], can be adapted to our setting into an algorithm which we call DPL-IWAL. Since our objective is no longer an expectation of some loss function, but rather, the conditional expectation evaluated on the event that $r(x) = 0$, IWAL's standard analysis does not apply. A key technical hurdle is proving that estimates of this conditional loss concentrate at the right rate in order to attain generalization guarantees for the pair (h, r) returned by DPL-IWAL. We show that over a time horizon T, DPL-IWAL algorithm requests $O((\rho + \eta)T)$ examples where $\eta = \min_{(h,r)\in\mathcal{H}\times\mathcal{R}_\rho} \mathbb{E}[L(h(x), y) \mid r(x) = 0]$ is the optimum value of our objective. At the same time our main lower bound demonstrates that $\Omega((\rho + \eta)T)$ requests are in fact necessary to compete with the best policy in $\mathcal{H} \times \mathcal{R}_\rho$.

Finally, we conduct a thorough exploration of these techniques on a number of datasets. We first undertake an ablation study, using a finite hypothesis class, showing that DPL-IWAL outperforms baselines that either ignore the active learning or abstention learning aspects of the problem. DPL-IWAL is not computationally efficient to implement since just like IWAL, it maintains a version space, that is a set of candidate hypotheses, which is non-trivial to optimize over in general. We therefore also conduct a number of experiments with an efficient heuristic inspired by our results using continuous hypothesis classes, outperforming natural baselines including margin sampling [Lewis and Gale, 1994, Balcan et al., 2007], which admits state-of-the-art performance for active learning problems in practice [Yang and Loog, 2016, Mussmann and Liang, 2018, Chuang et al., 2019].

1.1 Related Work

Our setting encompasses an objective considered in abstention learning (sometimes called selective classification), where the learner controls its evaluation region, that is the region of the domain where the learner is evaluated, by abstaining on the complement. In our setting, the evaluation region is where the learner does not request a label and the complementing abstention region is where the learner makes a request. At the same time, our setting considers the feedback mechanism from active learning. These are tied in a specific way in our framework: feedback is only available on the abstention (i.e. requesting) region and no information is revealed about the evaluation region. Neither an abstention algorithm nor a standard active learning algorithm would work in our setting and the algorithms in these settings can lead to incorrect solutions (see Appendix C). Nevertheless, we survey some of the relevant literature.

Learning with abstention was first studied by Chow [1957, 1970] for specific practical applications. Subsequently, several authors analyzed algorithms [Bartlett and Wegkamp, 2008, Grandvalet et al., 2008, Yuan and Wegkamp, 2010, Yuang and Wegkamp, 2011] for this setting with an emphasis on developing margin-based rules for abstention. Along a different line of work, El-Yaniv and Wiener [2010, 2011] analyze the theoretical trade-off between the coverage of an abstention function and a classifier's performance when not abstaining. All these works share in common the assumption that the learner has offline access to fully labeled samples, unlike the setting considered in this work.

A more recent line of work Cortes et al. [2016a,b, 2018] considers a setting where the algorithms learn over two classes of functions, a hypothesis class and an abstention class. These work either assume full feedback is always available (e.g. Cortes et al. [2016a,b]), or like in Cortes et al. [2018], the feedback is only available in the evaluation region, which is the exact opposite feedback mechanism than that of our setting since we receive feedback in the abstention region. Similarly, Shekhar et al. [2021] consider a setting where feedback is always available.

In the active learning literature, several authors focus on analyzing margin-based active learning which requests the labels for points close to a learned model classification surface [Dasgupta et al., 2005, Balcan et al., 2007, Balcan and Long, 2013, Awasthi et al., 2014, 2015, Zhang, 2018, Huang et al., 2019, Zhang et al., 2020]. Other algorithms admit generalization guarantees on the same order as passive learning while proving that their algorithm's label complexity, i.e. the number of points requested during learning, is bounded by a favorable rate. Beygelzimer et al. [2009] derived an algorithm, called IWAL, for general loss functions with strong theoretical guarantees. We wish to use this algorithm for our setting with the abstention learning objective. However, the loss function we consider is in fact a conditional expectation evaluated on an event, which is not amenable to standard IWAL. Our analysis and corresponding algorithm, DPL-IWAL, describes how to apply IWAL to such a setting.

Finally, ideas from the abstention framework have been applied to the active learning previously. Zhang and Chaudhuri [2014] used confidence-based predictors as a subroutine of an active learning algorithm. El-Yaniv and Wiener [2012] applied an abstention strategy from El-Yaniv and Wiener [2010] to the CAL algorithm of Cohn et al. [1994] proving theoretical guarantees, but only under specific model and distributional assumptions, which we do not make in our setting.

2 Setting and Preliminaries

In the dual purpose labeling framework, a learner is given a hypothesis class $\mathcal{H}$ with finite VC-dimension d, and a class of deterministic requester functions $\mathcal{R} \subset \{X \to \{0,1\}\}$. Nature fixes a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, unknown to the learner. Given the marginal over $\mathcal{X}$, and $\rho > 0$, we denote $\mathcal{R}_\rho \subset \mathcal{R}$ as the subset of $\mathcal{R}$ with bounded request-rate $\mathbb{E}_x[r(x) = 1] \leq \rho$ for all $r \in \mathcal{R}_\rho$. $\mathbb{E}_x[r(x) = 1]$ can be well estimated using unlabeled data, which is generally readily available, allowing a bound on request-rate to be enforced in practice.

The interaction between the learner and nature proceeds through a sequence of rounds t. The learner first selects $(h_t, r_t) \in \mathcal{H} \times \mathcal{R}_\rho$ as a function of the past. Nature then draws an independent sample (x_t, y_t) from $\mathcal{D}$, where x_t is revealed to the learner. If $r_t(x_t) = 1$, the learner additionally observes y_t, but the performance of h_t is not evaluated. If $r_t(x_t) = 0$, the learner does not observe y_t, but the performance of h_t is evaluated. Thus, the choice of r_t serves dual purposes: it determines whether the algorithm receives feedback, and whether its performance will be evaluated.

Formalizing this further, we suppose that the learner is given a loss function $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$. The learner seeks to output h_t that generalizes well on the region where r_t dictates that h_t should be evaluated. We therefore seek to bound the (conditional) excess loss $\mathcal{L}_{\mathcal{H},\mathcal{R}_\rho}(h_t, r_t) = E[L(h_t(x), y) \mid r_t(x) = 0] - \inf_{(h^*,r^*)\in\mathcal{H}\times\mathcal{R}_\rho} \mathbb{E}[L(h^*(x), y) \mid r^*(x) = 0]$. In particular we seek $O(1/\sqrt{t})$ bounds on $\mathcal{L}_{\mathcal{H},\mathcal{R}_\rho}(h_t, r_t)$, matching the generalization rate of full-feedback passive learning.

The conditional loss of the best pair in $\mathcal{H} \times \mathcal{R}_\rho$ plays an important role in our lower bounds, and so it is useful to define $\eta = \inf_{(h^*,r^*)\in\mathcal{H}\times\mathcal{R}_\rho} E_{x,y}[L(h^*(x), y) \mid r^*(x) = 0]$.

Finally, we call this the *proper dual purpose framework* since the algorithm is attempting to generalize well with respect to the class $\mathcal{H} \times \mathcal{R}_\rho$ and labels are generated according to functions in $\mathcal{R}_\rho$. In the

subsequent section, we will see that $O(1/\sqrt{t})$ bounds on $\mathcal{L}_{\mathcal{H},\mathcal{R}_\rho}(h_t, r_t)$ are impossible in general. This motivates the *improper dual purpose framework* as introduced in the following section.

3 Lower Bound

In this section, we present a lower bound stating that it is impossible for any algorithm to achieve an $O(1/\sqrt{t})$ generalization rate in the proper dual purpose setting. Thus, we here introduce the notion of *improper algorithms*. On each round t, an improper algorithm selects $h_t \in \mathcal{H}$ and an $R_t : X \to \{0, 1\}$ that is unconstrained for the purposes of showing a lower bound (e.g. R_t is not necessarily in the set $\mathcal{R}_\rho$). As in the proper setting, label requests are tied to the evaluation region. That is, the algorithm sees y_t i.f.f. $R_t(x_t) = 1$, and wishes to minimize $\mathcal{L}_{\mathcal{H},\mathcal{R}_\rho}(h_t, R_t)$. Notice that $\mathcal{L}$ is still defined with respect to the reference class $\mathcal{R}_\rho$, and that the proper setting is a special case of the improper setting where R_t is equal to an $r_t \in \mathcal{R}_\rho$. The lower bound below shows that any algorithm with the desired generalization rate must satisfy $\frac{1}{T}\sum_{t=1}^{T} E[R_t(x_t)] > \rho$. Thus, since $E[r(x)] \leq \rho$ for every $r \in \mathcal{R}_\rho$, no proper algorithm can attain the desired rate.

We prove a bound that holds for almost any possible classes of functions $\mathcal{H}, \mathcal{R}$, with some restrictions on $\mathcal{R}$. We say that $\mathcal{R}$ separates $\mathcal{X}$ if given any finite set of examples in $x_0, \ldots, x_n \in \mathcal{X}$, there exists a point $\hat{x}$ outside this finite set of points and a requester in $\mathcal{R}$ such $r(\hat{x}) = 1$ while $r(x_0) = \cdots = r(x_n) = 0$. This condition is much stronger than is necessary for the lower bound, but is already satisfied by simple classes such as when $\mathcal{R}$ contains linear separators and $\mathcal{X}$ is a ball in any dimension ≥ 2, and becomes easier to satisfy as $\mathcal{R}$ becomes more complex. We need a weaker condition that requires only the separation property hold for *some* set shattered by $\mathcal{H}$.

Definition 1. $\mathcal{R}$ separates $\mathcal{X}$ *with respect to* $\mathcal{H}$ *if* $d = \mathrm{VCD}(\mathcal{H})$*, and* there exists *a set of* d *examples* $x_0, \ldots, x_{d-1}$ *shattered by* $\mathcal{H}$*,* $\hat{x} \in \mathcal{X}$*, and* $r \in \mathcal{R}$ *such that* $r(\hat{x}) = 1$ *and* $r(x_0) = \ldots r(x_{d-1}) = 0$*.*

We first describe a distribution that follows the basic construction used to demonstrate lower bounds for pure active learning [Beygelzimer et al., 2009]. We then augment this distribution to include a region of mass ρ that contains random noise. Intuitively, one can think of an algorithm as requesting a label either to solve the active learning problem or to avoid loss on examples that it is uncertain on in the region of mass ρ. We argue that the optimal algorithm, when not solving the active learning problem, spends ρT labels requesting on the random noise.

While this is the basic idea, the proof needs to preclude the possibility that an algorithm with a suboptimal prediction h benefits from requesting labels outside the region of mass ρ purely for the purpose of avoiding loss (and not to solve the active learning problem). An algorithm can also request labels at a rate greater than ρ on any given round with the hope of decreasing its overall label complexity over T rounds. The proof shows that neither of these strategies benefits an algorithm enough to deviate from the optimal strategy outlined in the previous paragraph.

Definition 2. *Fix any* $\rho \geq 0$*. We say that a round* t *is a* failed round *if the algorithm selects a requesting strategy* $R_t : \mathcal{X} \to \{0, 1\}$*, and hypothesis* $h_t \in \mathcal{H}$ *satisfying* $\mathbb{E}[L(h_t(x), y) \mid R_t(x) = 0] \geq \min_{(h,r)\in\mathcal{H}\times\mathcal{R}_\rho} \mathbb{E}[L(h(x), y) \mid r(x) = 0] + \sqrt{\frac{d\eta}{t}}$*.*

As a practical matter, we will also require that an improper algorithm outputs a model $(h_T, r_T) \in \mathcal{H} \times \mathcal{R}_\rho$ at the end of its time horizon, where r_T comes from the reference class. This paves the way for algorithms discussed in the subsequent sections which are applicable in systems that are able to tolerate a higher labeling overhead during a finite training horizon, but eventually need to converge on a model with request rate ρ. Crucially, the lower bound establishes the minimum additional overhead during training as η (defined in Section 2), which is matched by our upper bounds.

Theorem 1. *Let* $L(h(x), y) = 1[yh(x) \leq 0]$ *be the misclassification loss. Given* $\mathcal{R}$ *that* separates $\mathcal{X}$ *with respect to* $\mathcal{H}$ *with* $d = \mathrm{VCD}(\mathcal{H})$*, let* $\mathcal{R}_\rho \subset \mathcal{R}$ *consist of requesters with bounded request-rate* ρ*. For any* $\eta \leq 1/4$*,* $\rho \leq 1/2$*, there exists a distribution on* $\mathcal{X} \times \{-1, 1\}$ *such that* $\eta = \min_{(h,r)\in\mathcal{H}\times\mathcal{R}_\rho} \mathbb{E}[L(h(x), y) \mid r(x) = 0]$ *.*

Furthermore, there exists a sufficiently large $T \geq 0$ *such that with probability at least* $1/2$ *any algorithm that, (A) outputs* $(h_T, r_T) \in \mathcal{H}\times\mathcal{R}_\rho$*, such that* $\mathbb{E}[L(h_T(x), y) \mid r_T(x) = 0] \leq \eta + \sqrt{d\eta/T}$*, and (B) suffers no more than* $T/2$ *failed rounds, requires that:* $\mathbb{E}[\sum_t R_t] \geq \Omega((\eta + \rho)T)$*.*

Algorithm 1 DPL-IWAL Algorithm

Require: Max iteration $T > 0$, $V_1 = \mathcal{H} \times \mathcal{R}_\rho$, t_0 be the first time t such that $t \geq 16\log(t/\delta)$
  for $t \in [1, t_0]$ **do**
    Observe x_t in order to construct estimates $\frac{1}{t-1}\sum_{s=1}^{t-1} \mathbf{1}[r(x_s) = 0]$
  for $t \in [t_0 + 1, T]$ **do**
    $(h_t, r_t) \leftarrow \operatorname{argmin}_{(h,r)\in V_t} \widehat{L}_{t-1}(h, r)$
    Receive x_t
    if $r_t(x_t) = 1$ **then** Request label y_t
    $p_t(x_t) \leftarrow \min\left(1, \max_{(h,r),(h',r')\in V_t} \max_{y\in Y} \left| \frac{L(h(x_t),y)\mathbf{1}[r(x_t)=0]}{\frac{1}{t-1}\sum_{s=1}^{t-1}\mathbf{1}[r(x_s)=0]} - \frac{L(h'(x_t),y)\mathbf{1}[r'(x_t)=0]}{\frac{1}{t-1}\sum_{s=1}^{t-1}\mathbf{1}[r'(x_s)=0]} \right| \right)$
    $q_t \sim$ Bernoulli(p_t)
    if $q_t = 1$ **then** Request or re-use label y_t
    $V_{t+1} \leftarrow \{(h, r) \in V_t : \widehat{L}_t(h, r) \leq \min_{(h,r)\in V_t} \widehat{L}_t(h, r) + \tilde{\Delta}_t\}$
    if $r_t(x_t) = 0 \wedge q_t = 0$ **then** Predict label using $\operatorname{sgn}(h_t(x_t))$
  Return: (h_T, r_T)

The above theorem states that an algorithm must request the labels of at least $\Omega((\eta + \rho)T)$ examples (in expectation) if we require that the pair returned by the algorithm generalizes at a rate approximating that of standard supervised learning. This then directly implies that R_t must be selected outside of the class $\mathcal{R}_\rho$ since this class only contains functions with a requesting rate of at most ρ, resulting in label complexity at most $O(\rho T)$ in expectation. All proofs can be found in Appendix A.

Although we state the lower bound for classification loss, it can be extended to any loss function where mispredicting the sign of an example, $yh(x) \leq 0$, implies $L(h(x), y) \geq C$ for some constant C. This is true for the logistic, hinge, squared, and absolute losses.

4 Dual Purpose Labeling Algorithm

In this section, we present our algorithm, DPL-IWAL (see Algorithm 1 for the pseudo-code), in the improper dual purpose framework. At a high level, DPL-IWAL finds the pair $(h, r) \in \mathcal{H} \times \mathcal{R}_\rho$ that minimizes the conditional loss, $\mathbb{E}[L(h(x), y)|r(x) = 0]$, i.e. the expected loss of h conditioned on the event that the label is not requested by r. Intuitively, the best pair (h, r) requests the label of the point whenever the prediction of $\operatorname{sgn}(h(x))$ is likely to be incorrect.

To find such a pair, at each round t, the algorithm first constructs an importance weighted estimate, $\widehat{L}_t(h, r)$, of $\mathbb{E}[L(h(x), y)|r(x) = 0]$ by using the fewest number of labeled points as possible and then chooses the pair (h_t, r_t) that minimizes $\widehat{L}_t(h, r)$. Ideally, the importance weighted estimates $\widehat{L}_t(h, r)$ could be constructed from the set of points whose labels have been requested by $r_1, \ldots, r_t$, but these sets of points are a non-trivially biased sample of the underlying distribution. Moreover, these points could reside in regions of the space that are not useful for calculating $\mathbb{E}[L(h(x), y)|r(x) = 0]$. To see this more clearly, consider a simple case when $\mathcal{R}_\rho$ is contains just one function and the algorithm is then simply finding the $\operatorname{argmin}_{h\in\mathcal{H}} \mathbb{E}[L(h(x), y)|r(x) = 0]$. The points the r functions request the label for, meaning points where $r(x) = 1$, do not reveal any information necessary to estimate $\mathbb{E}[L(h(x), y)|r(x) = 0]$ for $h \in \mathcal{H}$. Thus, the algorithm must label other regions in the space. Below, we describe how the algorithm uses a subset of the points requested by the $r_t \in \mathcal{R}_\rho$ in conjunction with some additional carefully chosen points, via a function q_t outside $\mathcal{R}_\rho$, to construct unbiased estimators, $\widehat{L}_t(h, r)$. This fact thus makes DPL-IWAL an improper dual purpose algorithm.

Similarly to IWAL [Beygelzimer et al., 2009], the DPL-IWAL algorithm constructs an importance weighted estimate, but instead of estimating the expected loss as is done in IWAL, we craft an estimate of the conditional losses for $t > t_0$,

$$\widehat{L}_t(h, r) = \frac{1}{t - t_0 - 1} \sum_{s=t_0+1}^{t} \frac{q_s}{p_s(x_s)} \frac{L(h(x_s), y_s)\mathbf{1}[r(x_s) = 0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'}) = 0]},$$

where a coin $q_s \in \{0, 1\}$ is flipped with a bias probability $p_s(x_s)$ and where t_0 is the first time t such that $t \geq 16\log(t/\delta)$. Note that for the first $s \in [1, t_0]$, we simply observe the features

x_s in order to construct $\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]$ that are non-zero with high probability since these are needed in definition of the denominator of $\widehat{L}_t(h,r)$. Ignoring the q_s and $p_s(x_s)$ for now, the numerator $\frac{1}{t-t_0-1}\sum_{s=t_0+1}^{t}L(h(x_s),y_s)\mathbf{1}[r(x_s)=0]$ is a measure of the joint expectation $\mathbb{E}[L(h(x),y)\mathbf{1}[r(x)=0]]$ while the denominator contains running averages of the $\mathbb{E}[r(x)=0]$. Roughly speaking, by considering the ratio of these two terms, we estimate the conditional expected loss, $\mathbb{E}[L(h(x),y)|r(x)=0]=\frac{\mathbb{E}[L(h(x),y)\mathbf{1}[r(x)=0]]}{\mathbb{E}[r(x)=0]}$.

The algorithm maintains a version space, V_t, as defined in Algorithm 1, which it reduces at each round. We prove that it suffices to use a slack term $\tilde{\Delta}_t=\tilde{O}\big(\sqrt{(1/t)\log(1/\delta)}\big)$, in order to ensure with high probability that (h^*,r^*) remain within the version space as it shrinks. The $\tilde{O}(\cdot)$ hides constants and $\log(t|\mathcal{H}\times\mathcal{R}_\rho|)$ factors; see the appendix for exact constants. In order to reduce the number of labeled points used to construct the importance-weighted estimates, the probability of requesting a point p_t is defined by the (estimated) conditional loss difference between pairs of functions in this shrinking set V_t. Given the above, the algorithm's overall requesting rule, R_t, is thus defined by the following condition: $R_t(x_t)=1$ if and only if $r_t(x_t)=1\vee q_t=1$ where $r_t\in\mathcal{R}_\rho$ and $q_t\notin\mathcal{R}_\rho$.

5 Generalization and Label Complexity Guarantees

In this section, we present a series of theoretical guarantees that analyze the performance of our approach as compared to different baselines as well as prove an upper bound on the expected number of label requests by the DPL-IWAL algorithm.

In our framework, we seek to select a hypothesis h_t which incurs minimal loss whenever a label request is not made. Below, we prove an upper bound on this type of loss that is in terms of the best pair of functions, $(h^*,r^*)=\operatorname{argmin}_{(h,r)\in\mathcal{H}\times\mathcal{R}_\rho}\mathbb{E}[L(h(x),y)|r(x)=0]$.

Theorem 2. *Given any $\rho<\frac{1}{2}$, for any $\delta>0$, with probability at least $1-\delta$, for all $t\geq 16\log(3t/\delta)$,* $\mathbb{E}[L(h_t(x),y)|r_t(x)=0]\leq\mathbb{E}[L(h^*(x),y)|r^*(x)=0]+\tilde{O}\big(\sqrt{(1/t)\log(1/\delta)}\big)$.

This guarantee states that the pair (h_t,r_t) chosen by the algorithm is converging to the best pair (h^*,r^*) as a function of the time t with respect to the conditional loss. The assumption $t\geq 16\log(3t/\delta)$ is mild condition, for example, if $\delta=0.0001$ we then require $t>104$. Also, in most standard applications, only a small fraction of examples can be labeled and it is natural to assume that $\rho<\frac{1}{2}$. Nevertheless, this assumption can be reduced by increasing the constraint on t in the bound.

Overall, the theoretical analysis, which is given in Appendix B, departs from standard derivations since we need to carefully deal with conditional losses and the constructed estimates, $\widehat{L}_t(h,r)$. More concretely, we first must ensure that the denominator of the estimate $\widehat{L}_t(h,r)$ is non-zero with high probability as otherwise the estimate would not be well defined. To do so, we start the labeling process of our algorithm only after t_0 examples have been observed. After t_0 examples have been observed and using the fact that $\mathbb{E}[r(x)=0]>1-\rho$, we can then prove that the condition $\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]>0$ holds with high probability (Lemma 1). Then, in order to apply Azuma's inequality on conditional losses, we need to prove that $\frac{L(h(x_s),y_s)\mathbf{1}[r(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]}$ is bounded by a favorable constant despite the variable denominator term (Lemma 2). Azuma's inequality implies that the estimates are converging to $\mathbb{E}[\widehat{L}_t(h,r)]$, but this is not enough since we want to prove guarantees in terms of $\mathbb{E}[L(h(x),y)|r(x)=0]$. Thus, using a series of concentration inequalities, we prove that expected value of the estimate, $\mathbb{E}[\widehat{L}_t(h,r)]$, converges to the expected conditional loss, $\mathbb{E}[L(h(x),y)|r(x)=0]$ at the desired rate (see proof of Theorem 2).

Next, we compare the quality of the predictions of our approach to that of simply predicting according to the best-in-class as measured by the non-conditional loss, $h_b=\operatorname{argmin}_{h\in\mathcal{H}}\mathbb{E}[L(h(x),y)]$. This comparison quantifies the potential benefits of our framework as compared to that of supervised learning since h_b is the hypothesis that an algorithm in the supervised setting is attempting to learn.

In the next corollary, we consider the never-requester function, $r_\diamond$, that is $\mathbf{1}[r_\diamond(x)=0]=1$ for all $x\in\mathcal{X}$. The never-requester is in $\mathcal{R}_\rho$ for any value of $\rho>0$. The never-requester trivially does not increase the label complexity and since it's a single function, it also does not discernibly augment the complexity of class, $\mathcal{R}_\rho$, and so it can be included in $\mathcal{R}_\rho$ at effectively no cost.

Corollary 1. *Given any $\rho < \frac{1}{2}$, for any $\delta > 0$, with probability at least $1-\delta$, for all $t \geq 16\log(3t/\delta)$, $\mathbb{E}[L(h_t(x),y)|r_t(x)=0] \leq \mathbb{E}[L(h_b(x),y)] + \gamma + \tilde{O}\big(\sqrt{(1/t)\log(1/\delta)}\big)$, where $\gamma = \mathbb{E}[L(h^*(x),y)|r^*(x)=0] - \mathbb{E}[L(h_b(x),y)]$. Furthermore, if $r_\diamond \in \mathcal{R}_\rho$, then $\gamma \leq 0$.*

The above corollary states the predictions of the chosen function h_t when not requesting admit strictly fewer mistakes as compared to the predictions of h_b, the best-in-class in $\mathcal{H}$, whenever $\gamma < -\tilde{O}\big(\sqrt{(1/t)\log(1/\delta)}\big)$. The value γ characterizes the difference between the best-in-class in our setting versus the best-in-class in the supervised learning. The more negative this term is, the fewer number of mistakes are made. In an empirical study in Appendix D, we show that typically γ is significantly smaller than 0.

To derive label complexity guarantees, we define a disagreement coefficient for conditional losses, directly derived from the coefficient definitions in Henneke [2007], Beygelzimer et al. [2009]. Let $\rho((h,r),(h',r')) = \mathbb{E}[|\frac{L(h(x),y)\mathbf{1}[r(x)=0]}{\mathbb{E}[r(x)=0]} - \frac{L(h'(x),y)\mathbf{1}[r'(x)=0]}{\mathbb{E}[r'(x)=0]}|]$ be a measure of the distance between two pairs (h,r) and (h',r'). Based on this metric, we define the ball around the best pair (h^*,r^*) as follows: $B(h^*,r^*,\Lambda) = \{(h,r) \in \mathcal{H}\times\mathcal{R}_\rho : \rho((h,r),(h^*,r^*)) \leq \Lambda\}$. The disagreement coefficient is the infimum value of $\theta > 0$ such that for all $\Lambda \geq 0$: $\mathbb{E}\left[\max_{(h,r)\in B(h^*,r^*,\Lambda)} \max_y \left|\frac{L(h(x),y)\mathbf{1}[r(x)=0]}{\mathbb{E}[r(x)=0]} - \frac{L(h^*(x),y)\mathbf{1}[r^*(x)=0]}{\mathbb{E}[r^*(x)=0]}\right|\right] \leq \theta\Lambda$. The next theorem bounds the expected number of points requested needed to construct the estimates, $\hat{L}_t(h,r)$ in terms of the coefficient θ.

Theorem 3. *Given any $\rho < \frac{1}{2}$, for all $\delta > 0$, with probability at least $1-\delta$, $\sum_{s=1}^T \mathbb{E}[p_s(x_s)] = \tilde{O}(\theta\eta T + \theta\sqrt{T})$, where θ is the disagreement coefficient.*

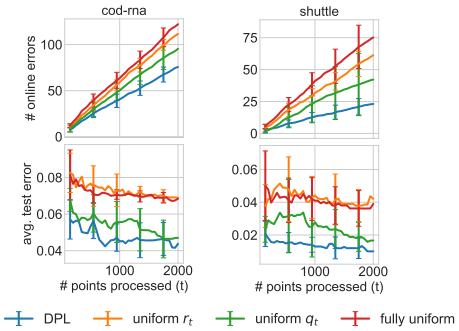
Since the labeling rate of r_t is at most ρ, the label complexity of the DPL-IWAL algorithm is then given by $\sum_{s=1}^T \mathbb{E}[r_s(x_s)=1] + \mathbb{E}[p_s(x_s)] = \tilde{O}((\theta\eta+\rho)T + \theta\sqrt{T})$. Assume that $\mathbb{E}[L(h_t(x),y) \mid R_t(x)=0] \leq \mathbb{E}[L(h_t(x),y)|r_t(x)=0]$; intuitively, this implies using q_t in addition to r_t to make requests helps reduce our conditional loss and it holds in practice as shown by our experiments in Appendix D. Then it follows, by Theorem 2, that $\mathbb{E}[L(h_t(x),y) \mid R_t(x)=0] \leq \eta + \tilde{O}(\sqrt{(1/t)\log(1/\delta)})$, i.e. with high probability failed rounds do not occur. Thus, in this case, DPL-IWAL exhibits an upper bound on the label complexity that matches the lower bound stated in the previous section, apart for $o(T)$ terms, namely $\sqrt{T}$. An even sharper bound for the $o(T)$ term is possible, using analysis similar to that of the EIWAL algorithm in Cortes et al. [2019], resulting in rate of $\sqrt{\eta T}$.

The analysis that proves the label complexity bound carefully deals with the denominator term of the estimates of the conditional loss as well as the fact that we are working with two function classes. Similarly to the generalization bound analysis, we first prove that the denominator term is well behaved and is not too far form its mean. Then, we leverage the disagreement coefficient θ and shrinking version space over the two classes.

In this paper, we analyzed the scenario where the requesting functions $r_t \in \mathcal{R}_\rho$ are constrained to request at most ρ times and minimize the conditional loss over this set of requesters. One could instead consider a Lagrangian relaxation of this constraint and pay a fixed cost c for each request. That is, consider the cost-based loss defined by $1_{r(x)=0}L(h(x),y) + c1_{r(x)=1}$. Both of these loss views are important and in fact have been analyzed in the abstention setting (e.g., see El-Yaniv and Wiener [2010] for conditional loss and see Cortes et al. [2016a] for cost-based loss). Despite focusing on conditional loss over constrained requester functions, the theory and algorithms in this paper can be extended to the cost-based loss. In particular, for the algorithm not to incur too many mistakes during training and to return a pair of function that generalized well, the algorithm must select R_t outside of the class of requesters. The version of DPL-IWAL for cost-based loss will thus require requesting according to both q_t and r_t.

6 Empirical Investigation

We start our empirical investigation by corroborating the theoretical insights made in the previous sections with an ablation study of DPL-IWAL. Since IWAL needs to solve a computationally intractable constrained optimization problem over its version space, which is only made more challenging in our setting by the joint optimization over $\mathcal{H}\times\mathcal{R}_\rho$, we use finite classes for these initial



DPL-Simplified Algorithm

Require: Max iteration $T > 0$, classes $\mathcal{H} \times \mathcal{R}_\rho$
for $t \in [1, T]$ **do**
  $h_t \leftarrow \text{argmin}_{h \in \mathcal{H}} \sum_{s=1}^{t-1} q_s L(h(x_s), y_s)$
  $r_t \leftarrow \text{argmin}_{r \in \mathcal{R}_\rho} \sum_{s=1}^{t-1} q_s \frac{L(h_t(x_s), y_s)\mathbf{1}[r(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]}$
  Receive x_t
  if $r_t(x_t) = 1$ **then** $q_t \leftarrow 1$, request label y_t
  else $q_t \leftarrow 0$, predict label using $\text{sgn}(h_t(x_t))$
Return: (h_T, r_T)

Figure 1: On the left, the number of online mistakes made while processing a stream of data and the held-out conditional loss on non-requested points made by DPL-IWAL and baselines comparators. The plots show the mean and standard deviation over 10 trials. On the right, the pseudo-code of DPL-Simplified Algorithm.

experiments. Then, we turn to a more practical algorithm inspired by the DPL-IWAL algorithm, but that leverages readily available optimization routines and empirically effective heuristics.

Ablation Study: We compare the performance of DPL-IWAL against baselines which ignore either the active learning and/or the abstention aspects of the dual purpose labeling, in order to demonstrate that indeed both aspects are necessary. To do so, we compare our algorithm to several variants, defined as follows. The *fully uniform* baseline simply decides to request a label using a random biased coin-flip independent of the example, thereby mimicking standard passive supervised learning. The *uniform* r_t baseline uses the DPL-IWAL algorithm in conjunction with trivial requester class that fixes its output uniformly at random, independent of the input x_t. The *uniform* q_t baseline is similar to DPL-IWAL, with outcomes q_t determined by coin-flips with a fixed bias, independent of x_t.

For this experiment, we define a set of requester regions near the margin of the classification surface, each covering a mass ρ of the overall distribution. Specifically, for any set of real-valued hypotheses set $\mathcal{H}$, where the classification of point x made by $h \in \mathcal{H}$ is defined as $\text{sgn}(h(x))$, we define a margin-based requester function class as: $\mathcal{R}_{\rho,\mathcal{H}} = \left\{r_h(x) \mapsto \mathbf{1}[|h(x)| \leq \tau_{h,\rho}(\mathcal{D}_\mathcal{X})] \colon h \in \mathcal{H}\right\}$, where $\mathcal{D}_\mathcal{X}$ is marginal distribution on $\mathcal{X}$ and $\tau_{h,\rho}(\mathcal{D}_\mathcal{X})$ is the largest threshold value that satisfies $\mathbb{E}_{\mathcal{D}_\mathcal{X}}[r_h(x) = 1] \leq \rho$. Note, the $\tau_{h,\rho}$ threshold can be estimated using unlabeled data.

We test six publicly available datasets [Chang and Lin] and for each, we use linear logistic regression models trained using the Python `scikit-learn` library. For all datasets, we construct a finite hypothesis class $\mathcal{H}$ and a matching finite margin-based requester set $\mathcal{R}_{\rho,\mathcal{H}}$. We then stream unlabeled examples to each algorithm by sampling without replacement from a training split. The process is repeated 10 times, using a different random train/test split for each trial. For details, see Appendix D.

In Figure 1 (left), we compare each method using two metrics. First, we consider the number of incorrect predictions (i.e. "online mistakes") that the model makes while processing the stream of unlabeled training data, where label-requests are spared mistakes. Second, we consider the conditional loss of the currently selected pair (h_t, r_t) by measuring the average misclassification loss on non-requested points using a held-out test split. See Appendix D for our results on all datasets, which all show a similar pattern.

Overall these results show that our DPL-IWAL algorithm, with both a non-trivial sampling function q_t and requesting function r_t, outperforms all baseline methods. This indicates that both active learning and abstention aspects of DPL-IWAL are necessary since naively applying an active learning algorithm without abstention (e.g. uniform r_t) or naively applying an abstention algorithm without active learning (e.g. uniform q_t) admit suboptimal results.

DPL-Simplified Algorithm: Here, we consider the DPL-Simplified Algorithm (Figure 1 right) where, the joint optimization problem over $\mathcal{H} \times R_\rho$ is split into two separate optimizations. This piece-wise optimization may not arrive at the same solution as the joint optimization, but we find it to be an empirically effective proxy. The first optimization over $\mathcal{H}$ is a standard learning problem over the currently labeled examples and any off-the-shelf hypothesis class and training algorithm can be used. The second optimization over $\mathcal{R}_\rho$ is still non-trivial to solve. However, the objective

	# Online Mistakes			Conditional Test Error (%)		
dataset	KNN	MAR	MIX	KNN	MAR	MIX
a9a	2017 ± 35	$\mathbf{1531 \pm 44}$	1713 ± 51	12.1 ± 0.39	$\mathbf{8.9 \pm 0.44}$	10.3 ± 0.30
acoustic	1654 ± 58	$\mathbf{1458 \pm 52}$	$\mathbf{1418 \pm 51}$	9.4 ± 0.35	$\mathbf{8.1 \pm 0.13}$	$\mathbf{7.8 \pm 0.28}$
cifar	904 ± 30	849 ± 38	$\mathbf{701 \pm 26}$	5.0 ± 0.23	5.1 ± 0.14	$\mathbf{3.8 \pm 0.09}$
cod-rna	644 ± 27	360 ± 13	$\mathbf{329 \pm 12}$	3.1 ± 0.11	2.2 ± 0.06	$\mathbf{1.5 \pm 0.07}$
covtype	3471 ± 89	$\mathbf{3365 \pm 89}$	$\mathbf{3318 \pm 78}$	18.6 ± 0.30	19.3 ± 0.18	$\mathbf{17.9 \pm 0.28}$
HIGGS	5862 ± 79	$\mathbf{5500 \pm 90}$	5689 ± 69	35.2 ± 0.31	$\mathbf{32.7 \pm 0.25}$	34.0 ± 0.27
ijcnn	654 ± 52	420 ± 20	$\mathbf{310 \pm 30}$	2.2 ± 0.28	2.4 ± 0.16	$\mathbf{0.7 \pm 0.14}$
mnist	1489 ± 69	$\mathbf{1032 \pm 26}$	$\mathbf{958 \pm 50}$	6.7 ± 0.31	5.1 ± 0.09	$\mathbf{4.1 \pm 0.12}$
shuttle	$\mathbf{29 \pm 16}$	122 ± 27	$\mathbf{21 \pm 8}$	$\mathbf{0.1 \pm 0.05}$	0.8 ± 0.09	$\mathbf{0.1 \pm 0.05}$
skin	$\mathbf{61 \pm 64}$	662 ± 110	$\mathbf{36 \pm 31}$	$\mathbf{0.1 \pm 0.02}$	4.0 ± 0.89	$\mathbf{0.02 \pm 0.01}$

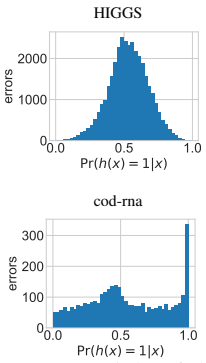


Figure 2: The left side of the table displays the mean and standard deviation over 10 trials of the number of total online mistakes after processing 20,000 examples for each of the requestor strategies (all limited to requesting labels for 20% of examples). The right side of the table shows the mean and standard deviation of the conditional misclassification loss $\Pr(h(x) \neq y | r(x) = 0)$, measured on the test set, for (h_T, r_T). On the right, histograms show the number of errors made by a partially-trained linear hypothesis h as a function of the model confidence for two datasets.

does provide the intuition that an optimal requester seeks to cover all of classifier h_t's mistakes. This suggests an approximate solution, where we train a requester r that seeks to classify the incorrect predictions of a fixed hypothesis h_t over the set of labeled examples thus far. At the same time, notice that the simplified algorithm fixes $q_t = r_t(x_t)$ (and no longer needs to solve IWAL's constrained optimization problem). This makes the requester function responsible for not just sampling regions of the space that $\mathcal{H}$ cannot correctly capture, but also sampling examples that are effective for training.

To cover the regions where the classifier is incorrect, we leverage what we call a **KNN-Requester** function. In particular, we use scikit-learn's `KNeighborsClassifier` to train a non-parametric model to predict hypothesis h_t's training mistakes. The resulting requester is $r_{\text{KNN},h,\rho}(x) = \mathbf{1}[|1 - \Pr_\theta(h(x) \neq y|x)| < \tau_\theta]$, where $\Pr_\theta(h(x) \neq y|x)$ denotes the probability that h makes a mistake according to the KNN model and ρ indicates the classifier's threshold τ_θ has been tuned so that the requester labels approximately a ρ fraction examples. To select points that are effective for training the classifier, we borrow intuition from the simple yet empirically very effective *margin* (or *uncertainty*) active learning algorithm, which samples examples that the current model is least confident on (i.e., example closest to the decision surface).

This leads us to the **Mixture-Requester** (**MIX**) function which merges the margin and KNN strategies. Specifically, we uniformly combine the probability score produced by the KNN model underlying $r_{\text{KNN},h,\rho}$ and the margin score derived from $h_t(x)$ as follows: $r_{\text{MIX},h,\rho}(x) = \mathbf{1}[|1 - \Pr_\theta(h(x) \neq y|x)| + |\sigma(h(x)) - 0.5| < \tau_{h,\theta}]$, where σ is a normalizing function that maps the input into $[0, 1]$, which in our case is the output of scikit-learn's `predict_proba` method. Thus, this requester seeks to sample points that are both covering mistakes of the classifier as well as sampling points that are effective for training.

In addition to the above, we evaluate the simpler **Margin-Requester** to serve as a natural, yet effective baseline: $r_{\text{MAR},h,\rho}(x) = \mathbf{1}[|\sigma(h(x)) - 0.5| < \tau_{h,\theta}]$. This requester is essentially mimicking the behavior of using uncertainty-based active learning without any regard to the DPL setting.

In the following experiments, the hypothesis class $\mathcal{H}$ is the set of linear models with bounded L_2-norm, trained using scikit-learn's `LogisticRegression` implementation, and we use 10 publicly available datasets, all which we cast as binary classification problems (see Appendix D for details). We execute a batch variant of DPL-Simplified, where at each iteration we process a batch of 5,000 examples, querying 20% of the examples for their labels and making prediction for the rest. Upon the completion of the iteration, we receive the requested labels and update the choice of (h, r). This larger batch size, more closely reflects practical learning settings, where it is impractical to re-train the model after every single label query [Amin et al., 2020]. All methods are seeded with 500 randomly sampled initial examples and each experiment is run for 10 trials.

In Figure 2, we show both the number of online errors incurred during the iterative labeling/training procedure as well as the average conditional misclassification loss of the final (h, r) pair on a test set.

In Appendix D, we plot the full learning curves in Figure 8 associated with Figure 2. Overall, MIX is very effective, outperforming the baseline methods in 8 out of 10 datasets. To better understand when MIX outperforms the margin-requester, we present a histogram of errors as a function the confidence of a model trained with a set of 3,500 examples sampled uniformly at random (Figure 2 right). The histogram for the HIGGS dataset, where the margin baseline performs well, shows that most of the errors are highly concentrated around the minimum model certainty region, i.e. where the model prediction score is close to 0.5. In contrast, the histogram for cod-rna, where the DPL-inspired mixture-requester excels, shows that while there are some errors concentrated around the 0.5 threshold, there are also a number of errors far away from the model decision boundary.

7 Conclusion

We introduced a new setting which models the relationship between labeling and learning in real systems. We derived a lower bound in terms of the abstention-rate of a reference class and the optimum value of our objective. We presented an algorithm DPL-IWAL that admits strong generalization and label complexity guarantees and a more efficient variant, DPL-Simplified. Finally, we reported experiments which corroborate our theoretical findings and demonstrate that our algorithm outperforms natural baselines, including the ubiquitous margin sampling algorithm.

References

Kareem Amin, Corinna Cortes, Giulia DeSalvo, and Afshin Rostamizadeh. Understanding the effects of batching in online active learning. In *The 23nd International Conference on Artificial Intelligence and Statistics*, 2020.

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.

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, Andrei Broder, and Tong Zhang. Margin based active learning. In *Proceeedings of COLT*, pages 35–50. Springer, 2007.

Peter Bartlett and Marten Wegkamp. Classification with a reject option using a hinge loss. *JMLR*, pages 291–307, 2008.

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.

Chih-Chung Chang and Chih-Jen Lin. Libsvm. `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`. Accessed: 2021-05-28.

C.K. Chow. An optimum character recognition system using decision function. *IEEE T. C.*, 1957.

C.K. Chow. On optimum recognition error and reject trade-off. *IEEE T. C.*, 1970.

Galen Chuang, Giulia DeSalvo, Lazarus Karydas, Jean-francois Kagy, Afshin Rostamizadeh, and A Theeraphol. Active learning empirical study. In *NeurIPS2019 LIRE Workshop*, 2019.

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

Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Learning with rejection. In *ALT*, pages 67–82. Springer, Heidelberg, Germany, 2016a.

Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Boosting with abstention. In *NIPS*. MIT Press, 2016b.

Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Online learning with abstention. In *ICML*, 2018.

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

Ido Dagan and Sean P Engelson. Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings 1995*, pages 150–157. Elsevier, 1995.

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.

Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *JMLR*, 2010.

Ran El-Yaniv and Yair Wiener. Agnostic selective classification. In *NIPS*, 2011.

Ran El-Yaniv and Yair Wiener. Active learning via perfect selective classification. In *JMLR*, 2012.

Yves Grandvalet, Joseph Keshet, Alain Rakotomamonjy, and Stephane Canu. Suppport vector machines with a reject option. In *NIPS*, 2008.

Stephen Henneke. A bound on the label complexity of agnostic active learning. In *ICML*, 2007.

Boshuang Huang, Sudeep Salgia, and Qing Zhao. Disagreement-based active learning in online settings. *arXiv preprint arXiv:1904.09056*, 2019.

David Lewis and William Gale. A sequential algorithm for training text classifiers. In *Proceedings of SIGIR*, 1994.

Stephen Mussmann and Percy Liang. Uncertainty sampling is preconditioned stochastic gradient descent on zero-one loss. In *Proceedings of NeurIPS*, 2018.

Shubhanshu Shekhar, Mohammad Ghavamzadeh, and Tara Javidi. Active learning for classification with abstention. *IEEE Journal on Selected Areas in Information Theory*, 2021.

Yazhou Yang and Marco Loog. A benchmark and comparison of active learning for logistic regression. In *arXivpreprint arXiv:1611.08618*, 2016.

Ming Yuan and Marten Wegkamp. Classification methods with reject option based on convex risk minimizations. In *Journal of Machine Learning Research*, 2010.

Ming Yuang and Marten Wegkamp. Support vector machines with a reject option. In *Bernoulli*, 2011.

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

Chicheng Zhang and Kamalika Chaudhuri. Beyond disagreement-based agnostic active learning. In *Advances in Neural Information Processing Systems*, pages 442–450, 2014.

Chicheng Zhang, Jie Shen, and Pranjal Awasthi. Efficient active learning of sparse halfspaces with arbitrary bounded noise. *Advances in Neural Information Processing Systems*, 33, 2020.

A Lower Bound Proof

In this appendix, we present the proof of the lower bound for the dual propose labeling problem analyzed in this paper.

Theorem 1. *Let $L(h(x), y) = 1[yh(x) \leq 0]$ be the misclassification loss. Given $\mathcal{R}$ that* separates *$\mathcal{X}$ with respect to $\mathcal{H}$ with $d = \mathrm{VCD}(\mathcal{H})$, let $\mathcal{R}_\rho \subset \mathcal{R}$ consist of requesters with bounded request-rate ρ. For any $\eta \leq 1/4$, $\rho \leq 1/2$, there exists a distribution on $\mathcal{X} \times \{-1, 1\}$ such that $\eta = \min_{(h,r)\in\mathcal{H}\times\mathcal{R}_\rho} \mathbb{E}[L(h(x), y) \mid r(x) = 0]$.*

Furthermore, there exists a sufficiently large $T \geq 0$ such that with probability at least $1/2$ any algorithm that, (A) outputs $(h_T, r_T) \in \mathcal{H}\times\mathcal{R}_\rho$, such that $\mathbb{E}[L(h_T(x), y) \mid r_T(x) = 0] \leq \eta + \sqrt{d\eta/T}$, and (B) suffers no more than $T/2$ failed rounds, requires that: $\mathbb{E}[\sum_t R_t] \geq \Omega((\eta + \rho)T)$.

Proof. Given an algorithm, define $Q = \mathbb{E}\left[\sum_t R_t\right]$, the expected number of labels requested by the algorithm. We begin by showing that under the conditions of the theorem, $Q = \Omega(\eta T)$. We use a similar construction as used for the lower bounds in standard active learning. However, we must be careful to ensure that the ability the learner has to abstain using $\mathcal{R}$ does not affect the bounds by too much. We then show that when $\rho > 2c\eta$, $Q = \Omega(\rho T)$, where c is the constant in the definition of a failed round. Together, these two facts imply the theorem.

Let $x_0, x_1, \ldots, x_{d-1}$ and $\hat{x}$ be a set of examples satisfying Definition 1. Let $r_0 \in \mathcal{R}$ satisfy $r_0(\hat{x}) = 1$ and $r_0(x_i) = 0$, $i \geq 0$.

Fix an $\epsilon > 0$, and set $\beta = 2(\eta + 2\epsilon)$. Our marginal distribution on $\mathcal{X}$ will be supported on $\hat{x}, x_0, x_1, \ldots x_{d-1}$. The instance $\hat{x}$ will have probability mass ρ. x_0 will have probability mass $(1-\rho)(1-\beta)$. The remaining x_i each have mass $(1-\rho)\beta/(d-1)$. Since $\mathbb{E}[r_0(x) = 1] = \mathbb{P}[x = \hat{x}] = \rho$, $r_0 \in \mathcal{R}_\rho$.

Let $\mathrm{Rad}(p)$ denote a Rademacher random variable with p the probability of $+1$. Let $b_i \sim \mathrm{Rad}(1/2)$ for $i \in \{1, \ldots, d-1\}$. These are determined once at the beginning of time.

For each round t if x_i, $i \geq 1$, is selected by the marginal then y is distributed as $\mathrm{Rad}(1/2 + \frac{\epsilon b_i}{\eta+2\epsilon})$. If x_0 is selected by the marginal then $y = 1$ with probability 1. If $\hat{x}$ is selected by the marginal then y is distributed as $\mathrm{Rad}(1/2)$.

For this distribution, the optimal hypothesis h^* (regardless of which requester is used) labels x_0 as 1, labels each x_i, $i \geq 1$, as b_i and labels $\hat{x}$ arbitrarily (say $h^*(\hat{x}) = 1$ wlog).

Intuitively, r_0 is the optimal requester in $\mathcal{R}_\rho$ for h^* since, conditioned on x, h^* suffers a loss of $1/2$ when $x = \hat{x}$, a loss of $(1/2 - \lambda)$ when $x = x_i, i \geq 1$ and a loss of 0 when $x = x_0$, where we define $\lambda = \frac{\epsilon}{\eta+2\epsilon}$. (We make this fact precise below). However, for a *suboptimal* h, h could in principle reduce its loss by carefully abstaining on examples it is uncertain on, avoiding examples where its conditional loss is $(1/2 + \lambda)$. In the following we bound the effectiveness of such a strategy.

Define $\mathcal{R}' = \{r : \mathcal{X} \to \{0, 1\} \mid r(\hat{x}) = 0\}$ as the set of all requesting strategies, not necessarily satisfying $\mathbb{P}[R'(x) = 0] \leq \rho$, and not necessarily belonging to $\mathcal{R}$, but that do not request a label for $\hat{x}$. $\mathcal{R}'_\rho = \{r \in \mathcal{R}' \mid \mathbb{P}[r(x) = 1] \leq \rho\}$.

Fix an arbitrary $h \in \mathcal{H}$, and $r \in \mathcal{R}'_\rho$. Let $F_i = \mathbf{1}[h(x_i) \neq b_i]$, and $G_i = 1 - F_i$. Define $Z = \mathbb{P}[r(x) = 1]$. Let $N = \frac{(1-\rho)\beta}{d-1}$ be the probability mass placed on each x_i, $i \geq 1$.

$$
\begin{aligned}
&\mathbb{E}[L(h(x), y) \wedge r(x) = 0] \\
&\qquad \geq \frac{\rho}{2} + N\sum_{i=1}^{d-1} \mathbf{1}[r(x_i) = 0]G_i(1/2 - \lambda) + N\sum_{i=1}^{d-1} \mathbf{1}[r(x_i) = 0]F_i(1/2 + \lambda) \\
&\qquad = \frac{\rho}{2} + N\sum_{i=1}^{d-1} \mathbf{1}[r(x_i) = 0](1/2 - \lambda) + 2\lambda N\sum_{i=1}^{d-1} \mathbf{1}[r(x_i) = 0]F_i
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{\rho}{2} + N\sum_{i=1}^{d-1}(1/2-\lambda) + 2\lambda N\sum_{i=1}^{d-1}F_i \\
&- N\sum_{i=1}^{d-1}\mathbf{1}[r(x_i)=1](1/2-\lambda) - 2\lambda N\sum_{i=1}^{d-1}\mathbf{1}[r(x_i)=1]F_i \\
&\geq \frac{\rho}{2} + N\sum_{i=1}^{d-1}(1/2-\lambda) + 2\lambda N\sum_{i=1}^{d-1}F_i \\
&- \frac{Z}{2} + \lambda N\sum_{i=1}^{d-1}\mathbf{1}[r(x_i)=1] - 2\lambda N\sum_{i=1}^{d-1}\mathbf{1}[r(x_i)=1]F_i \\
&\geq \frac{\rho}{2} - \frac{Z}{2} + N\sum_{i=1}^{d-1}(1/2-\lambda) + \lambda N\sum_{i=1}^{d-1}F_i \\
&= \frac{\rho}{2} - \frac{Z}{2} + (1-\rho)\beta(1/2-\lambda) + (1-\rho)\lambda\beta\frac{1}{d-1}\sum_{i=1}^{d-1}F_i
\end{aligned}
$$

The first inequality follows since h may also mislabel x_0. The second inquality follows because $N\sum_{i=1}^{d-1}[r(x_i)=1] \leq Z$. The third inequality follows because $\sum_{i=1}^{d-1}\mathbf{1}[r(x_i)=1]F_i$ is less than both $\sum_{i=1}^{d-1}F_i$ and $\sum_{i=1}^{d-1}\mathbf{1}[r(x_i)=1]$.

By definition $\beta(1/2-\lambda)=\eta$ and $\lambda\beta = 2\epsilon$. Therefore:

$$
\begin{aligned}
\mathbb{E}[L(h(x),y) \mid r(x)=0] &\geq \frac{1}{(1-Z)}\left[\frac{\rho}{2} - \frac{Z}{2} + (1-\rho)\eta + (1-\rho)\frac{2\epsilon}{d-1}\sum_{i=1}^{d-1}F_i\right] \\
&\geq \eta + \frac{1-\rho}{1-Z}\frac{2\epsilon}{d-1}\sum_{i=1}^{d-1}F_i \geq \eta + \frac{\epsilon}{d-1}\sum_{i=1}^{d-1}F_i \qquad (1)
\end{aligned}
$$

The second inequality above can be verified by some algebra when $\eta \leq 1/2$. In particular, $\frac{\frac{\rho-Z}{2}+\eta-\eta\rho}{1-Z} \geq \eta \Leftrightarrow \frac{\rho-Z}{2}+\eta \geq (\rho-Z)\eta+\eta \Leftrightarrow \eta \leq 1/2$ when $Z \leq \rho$. The third inequality follows since $\rho \leq 1/2$ and so $\frac{1-\rho}{1-Z} \geq (1-\rho) \geq 1/2$.

Since our distribution places ρ mass on $\hat{x}$, it's clear that r_0 is the only function in $\mathcal{R}_\rho$ that is not in $\mathcal{R}'_\rho$.

Under the requester r_0, which avoids loss on $\hat{x}$, a hypothesis that correctly labels x_0 incurs a loss of:

$$
\begin{aligned}
\mathbb{E}[L(h(x),y \mid r_0(x)=0] &= \frac{1}{1-\rho}\left[N\sum_{i=1}^{d-1}G_i(1/2-\lambda) + N\sum_{i=1}^{d-1}F_i(1/2+\lambda)\right] \\
&= \frac{1}{1-\rho}\left[(d-1)N(\frac{1}{2}-\lambda) + N\sum_{i=1}^{d-1}2\lambda F_i\right] \\
&= \beta(1/2-\lambda) + \frac{2\lambda\beta}{d-1}\sum_{i=1}^{d-1}F_i = \eta + \frac{4\epsilon}{d-1}\sum_{i=1}^{d-1}F_i. \qquad (2)
\end{aligned}
$$

Any hypothesis that incorrectly labels x_0 is strictly worse, and so equations (1) and (2), confirm that the optimal hypothesis requester pair is indeed (h^*, r_0), with a loss of η (since equation (2 holds with equality). Moreover, any suboptimal hypothesis requester pair has a loss of more than $\eta + \frac{\epsilon}{d-1}\sum_{i=1}^{d-1}F_i$, the minimum of (1) and (2).

We can now leverage Theorem 12 of Beygelzimer et al. [2009] . The theorem states that any algorithm that queries $x_i, i \geq 1$ fewer than $c'd\eta^2/\epsilon^2$ times , for some constant c', will incorrectly predict more than $1/4$ of the bits b_i with probability at least $1/2$, and thus will output $(\hat{h}, \hat{r})$ with error at least

$\eta + \epsilon/4$ with probability at least $1/2$. Setting $\epsilon = 4\sqrt{\frac{d\eta}{T}}$ tells us that any algorithm satisfying condition (A) in the statement of the theorem, must query at at least $Q \geq \frac{c'}{16}\eta T = \Omega(\eta T)$ examples before time T.

We next prove that if $\rho \geq 2c\eta$, then any algorithm satisfying condition (B) in the statement of the theorem must query $\Omega(\rho T)$ examples.

Recall that all $r \in \mathcal{R}'$, satisfy $r(\hat{x}) = 0$. Let h be an arbitrary hypothesis satisfying $h(x_0) = 1$. The optimal $r' \in \mathcal{R}'$ for h, satisfies $r'(\hat{x}) = 0$, $r'(x_0) = 0$, $r'(x_i) = 1$ for all $i \geq 1$. In other words, r' is forced to suffer loss on $\hat{x}$ because $r' \in \mathcal{R}'$, but avoids all other error otherwise.

$$\begin{aligned}
\mathbb{E}[L(h(x), y) \mid r'(x) = 0] &= \mathbb{E}[L(h(x), y) \mid r'(x) = 0, x = \hat{x}]\mathbb{P}[x = \hat{x} \mid r'(x) = 0] \\
&\quad + \mathbb{E}[L(h(x), y) \mid r'(x) = 0, x = x_0]\mathbb{P}[x = x_0 \mid r'(x) = 0] \\
&= \mathbb{E}[L(h(x), y) \mid r'(x) = 0, x = \hat{x}]\mathbb{P}[x = \hat{x} \mid r'(x) = 0] \\
&= \frac{1}{2}\frac{\mathbb{P}(x = \hat{x}, r'(x) = 0)}{\mathbb{P}(r'(x) = 0)} = \frac{1}{2}\frac{\rho}{\rho + (1-\rho)(1-\beta)} \\
&\geq \frac{c\eta}{1 - \beta + \rho\beta} > c\eta
\end{aligned}$$

Since $\mathbb{E}[L(h(x), y) \mid r'(x) = 0] > c\eta$, there is a sufficiently large T such that $c\eta + \sqrt{\frac{2d\eta}{T}} < \mathbb{E}[L(h(x), y) \mid r'(x) = 0]$. Thus any $t > T/2$ is a failed round if the algorithm plays a requesting strategy in $\mathcal{R}'$. Any algorithm satisfying (B) in the statement of the theorem must therefore with probability at least $1/2$ select strategies $R_t \notin \mathcal{R}'$, satisfying $R_t(\hat{x}) = 1$ for all rounds $t > T/2$. For each such R_t, $\mathbb{E}[R_t(x) = 1] \geq \rho$, and thus $Q = \Omega(\rho T)$ when $\rho \geq 2c\eta$, completing the proof. □

B Proofs of Generalization and Label Complexity Guarantees

In this appendix, we first prove generalization guarantees and then label complexity guarantees. For simplicity below, let t_0 be the first time t such that $t \geq 16\log(t/\delta)$, let $\Delta_t = \sqrt{\frac{\log(3t/\delta)}{t}}$, $\Delta'_t = (\frac{4}{1-\rho} + 2)\sqrt{\frac{2\log(6(t-t_0)(t-t_0+1)|\mathcal{H}\times\mathcal{R}|^2/\delta)}{t-t_0}}$ for any $t > t_0$, and $\tilde{\Delta}_t = 2\Delta'_t + \frac{8}{1-\rho}\Delta_{t-1}$.

B.1 Generalization Guarantees

Lemma 1. *Assume $\rho < \frac{1}{2}$. For any $\delta > 0$, with probability at least $1 - \delta$, for all $s \geq 16\log(3s/\delta)$, and any $r \in \mathcal{R}$,*

$$\frac{1}{\mathbb{E}[r(x) = 0] + \Delta_{s-1}} \leq \frac{1}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'}) = 0]} \leq \frac{1}{\mathbb{E}[r(x) = 0] - \Delta_{s-1}}.$$

Proof. By Hoeffding's inequality, $\mathbb{E}[r'(x) = 0] \leq \hat{\mathbb{E}}_{s-1}[\mathbf{1}[r'(x) = 0]] + \Delta_{s-1}$ and $\mathbb{E}[r'(x) = 0] \geq \hat{\mathbb{E}}_{s-1}[\mathbf{1}[r'(x) = 0]] - \Delta_{s-1}$ hold concurrently with probability at least $1 - \delta$.

The inequality $\mathbb{E}[r'(x) = 1] \leq \rho$ directly implies that $\mathbb{E}[r'(x) = 0] > 1 - \rho \geq \frac{1}{2}$. It also holds by assumption that $s - 1 \geq 16\log(3(s-1)/\delta)$ which can be rewritten as $\frac{1}{2} - \Delta_{s-1} > \frac{1}{4}$. Hence, $\mathbb{E}[r'(x) = 0] - \Delta_{s-1} > \frac{1}{2} - \Delta_{s-1} > \frac{1}{4} > 0$. The statement of the lemma follows by inverting the concentration inequalities and then dividing by $\mathbb{E}[r'(x) = 0] - \Delta_{s-1}$ and by $\mathbb{E}[r'(x) = 0] + \Delta_{s-1}$, separately. □

Lemma 2. *Assume $\rho < \frac{1}{2}$. For any $\delta > 0$, with probability at least $1 - \delta$, for all $s \geq 16\log(3s/\delta)$, and any $(h, r), (h', r') \in \mathcal{H} \times \mathcal{R}$,*

$$\left|\frac{L(h(x_s), y_s)\mathbf{1}[r(x_s) = 0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'}) = 0]} - \frac{L(h'(x_s), y_s)\mathbf{1}[r'(x_s) = 0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r'(x_{s'}) = 0]}\right| \leq \frac{4}{1-\rho} + 2.$$

Proof. By an application of Lemma 1, with probability at least $1-\delta$,

$$\begin{aligned}
&\frac{L(h(x_s),y_s)\mathbf{1}[r(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]} - \frac{L(h'(x_s),y_s)\mathbf{1}[r'(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r'(x_{s'})=0]} \\
&\qquad \le \frac{L(h(x_s),y_s)\mathbf{1}[r(x_s)=0]}{\mathbb{E}[r(x)=0]-\Delta_{s-1}} - \frac{L(h'(x_s),y_s)\mathbf{1}[r'(x_s)=0]}{\mathbb{E}[r'(x)=0]+\Delta_{s-1}} \\
&\qquad \le \frac{\mathbb{E}[r(x)=0]-\mathbb{E}[r'(x)=0]+2\Delta_{s-1}}{(\mathbb{E}[r'(x)=0]+\Delta_{s-1})(\mathbb{E}[r(x)=0]-\Delta_{s-1})} \\
&\qquad = \frac{\mathbb{E}[r(x)=0]-\mathbb{E}[r'(x)=0]}{(\mathbb{E}[r'(x)=0]+\Delta_{s-1})(\mathbb{E}[r(x)=0]-\Delta_{s-1})} \qquad (3) \\
&\qquad\quad + \frac{2\Delta_{s-1}}{(\mathbb{E}[r'(x)=0]+\Delta_{s-1})(\mathbb{E}[r(x)=0]-\Delta_{s-1})} \qquad (4)
\end{aligned}$$

where we used the fact that $L(h'(x_s),y_s)\mathbf{1}[r'(x_s)=0]\le 1$. First, we analyze the term (3). By the assumptions, it holds that $\mathbb{E}[r'(x)=1]\le\rho$ which implies $\mathbb{E}[r'(x)=0]>1-\rho\ge\frac{1}{2}$ and that $s-1\ge 16\log(3(s-1)/\delta)$ which can be rewritten as $\frac{1}{2}-\Delta_{s-1}>\frac{1}{4}$. Hence, $\mathbb{E}[r'(x)=0]-\Delta_{s-1}>\frac{1}{2}-\Delta_{s-1}>\frac{1}{4}>0$. Thus,

$$(3)\le\frac{4}{1-\rho},$$

where we also used the fact that $\mathbb{E}[r'(x)=0]+\Delta_{s-1}\ge 1-\rho$.

Next, we turn to term (4). For simplicity, let $a=\mathbb{E}[r(x)=0]$ and $b=\mathbb{E}[r'(x)=0]$ and consider the following

$$\begin{aligned}
(a+\Delta_{s-1})(b-\Delta_{s-1})/\Delta_{s-1} &= ab/\Delta_{s-1}-a+b-\Delta_{s-1} \\
&> \frac{1}{4\Delta_{s-1}}-1+\frac{1}{4}=\frac{1}{4\Delta_{s-1}}-\frac{3}{4}>1
\end{aligned}$$

where we used the fact that $b-\Delta_{s-1}>1/4$, $ab>\frac{1}{4}$, and $a<1$ by the same reasoning as term (3) and that $1/\Delta_{s-1}-3>1$ since by assumption $\frac{1}{4}>\Delta_{s-1}$. Hence,

$$(4)=\frac{2}{ab/\Delta_{s-1}-a+b-\Delta_{s-1}}\le 2.$$

Putting the above together, (3) + (4) $\le\frac{4}{1-\rho}+2$ and taking absolute values concludes the proof. □

Theorem 2. *Given any $\rho<\frac{1}{2}$, for any $\delta>0$, with probability at least $1-\delta$, for all $t\ge 16\log(3t/\delta)$,* $\mathbb{E}[L(h_t(x),y)|r_t(x)=0]\le\mathbb{E}[L(h^*(x),y)|r^*(x)=0]+\tilde{O}\big(\sqrt{(1/t)\log(1/\delta)}\big)$.

Proof. Recall that t_0 is the first time t such that $t\ge 16\log(t/\delta)$ and consider only $t>t_0$. For any pair $(h,r)\in V_t$ and $(h',r')\in V_t$, let

$$\begin{aligned}
Z_s = &\mathbb{E}\left[\frac{q_s}{p_s(x_s)}\frac{L(h(x_s),y_s)\mathbf{1}[r(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]}\right]-\mathbb{E}\left[\frac{q_s}{p_s(x_s)}\frac{L(h'(x_s),y_s)\mathbf{1}[r'(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r'(x_{s'})=0]}\right] \\
&-\frac{q_s}{p_s(x_s)}\left(\frac{L(h(x_s),y_s)\mathbf{1}[r(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]}-\frac{L(h'(x_s),y_s)\mathbf{1}[r'(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r'(x_{s'})=0]}\right),
\end{aligned}$$

for $s\in[t]$. If $p_s<1$, then $Z_s\le 2$ by definition. If $p_s=1$, then $|Z_s|\le\frac{8}{1-\rho}+4$ follows by Lemma 2 with probability at least $1-\delta$. Under this high probability event, we apply Azuma's inequality to Z_s to attain:

$$\begin{aligned}
&\left|\frac{1}{t-t_0}\sum_{s=t_0}^{t}\mathbb{E}\left[\frac{q_s}{p_s(x_s)}\frac{L(h(x_s),y_s)\mathbf{1}[r(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]}\right]-\frac{1}{t-t_0}\sum_{s=t_0}^{t}\mathbb{E}\left[\frac{q_s}{p_s(x_s)}\frac{L(h'(x_s),y_s)\mathbf{1}[r'(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r'(x_{s'})=0]}\right]\right. \\
&\left.-\frac{1}{t-t_0}\sum_{s=t_0}^{t}\frac{q_s}{p_s(x_s)}\left(\frac{L(h(x_s),y_s)\mathbf{1}[r(x)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]}-\frac{L(h'(x_s),y_s)\mathbf{1}[r'(x)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r'(x_{s'})=0]}\right)\right|\le 2\Delta'_t,
\end{aligned}$$

where we take a union bound over all $(h, r) \in V_t$. Equivalently, this can be written as:

$$\left| \mathbb{E}[\widehat{L}_t(h,r)] - \mathbb{E}[\widehat{L}_t(h',r')] - \widehat{L}_t(h,r) - \widehat{L}_t(h',r') \right| \leq 2\Delta'_t, \tag{5}$$

To attain our desired bound, we then relate $\mathbb{E}[\widehat{L}_t(h,r)]$ to $\mathbb{E}[L(h(x),y)|r(x)=0]$. First, we rewrite the former expectation:

$$\begin{aligned}
\mathbb{E}[\widehat{L}_t(h,r)] &= \frac{1}{t-t_0}\sum_{s=t_0}^{t} \mathbb{E}\left[\frac{q_s}{p_s(x_s)} \frac{L(h(x_s),y_s)\mathbf{1}[r(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]}\right] \\
&= \frac{1}{t-t_0}\sum_{s=t_0}^{t} \mathbb{E}\left[\frac{L(h(x_s),y_s)\mathbf{1}[r(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]}\right] \\
&= \mathbb{E}[L(h(x),y)\mathbf{1}[r(x)=0]]\frac{1}{t-t_0}\sum_{s=t_0}^{t} \mathbb{E}\left[\frac{1}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]}\right] \qquad (6)
\end{aligned}$$

where the first inequality follows since $\mathbb{E}[q_s|p_s(x_s)] = p_s(x_s)$ and where the second inequality follows by the fact that the data is i.i.d. Combining this with Lemma 1 multiplied by $\mathbb{E}[L(h(x),y)\mathbf{1}[r(x)=0]]$, it follows that

$$\begin{aligned}
&\mathbb{E}[L(h(x),y)\mathbf{1}[r(x)=0]]\frac{1}{t-t_0}\sum_{s=t_0}^{t} \mathbb{E}\left[\frac{1}{\mathbb{E}[r(x)=0]+\Delta_{s-1}}\right] \qquad (7) \\
&\leq \mathbb{E}[\widehat{L}_t(h,r)] \\
&\leq \mathbb{E}[L(h(x),y)\mathbf{1}[r(x)=0]]\frac{1}{t-t_0}\sum_{s=t_0}^{t} \mathbb{E}\left[\frac{1}{\mathbb{E}[r(x)=0]-\Delta_{s-1}}\right]. \qquad (8)
\end{aligned}$$

Hence, in order to relate $\mathbb{E}[\widehat{L}_t(h,r)]$ to $\mathbb{E}[L(h(x),y)|r(x)=0]$, we also need that $\frac{1}{\mathbb{E}[r(x)=0]\pm\Delta_{s-1}}$ is close to $\frac{1}{\mathbb{E}[r(x)=0]}$ and so consider the following:

$$\begin{aligned}
\left|\frac{1}{t-t_0}\sum_{s=t_0}^{t}\left(\frac{1}{\mathbb{E}[r(x)=0]-\Delta_{s-1}} - \frac{1}{\mathbb{E}[r(x)=0]}\right)\right| &= \frac{1}{t-t_0}\sum_{s=t_0}^{t}\frac{\Delta_{s-1}}{(\mathbb{E}[r(x)=0]-\Delta_{s-1})\,\mathbb{E}[r(x)=0]} \\
&= \frac{1}{\mathbb{E}[r(x)=0]}\frac{1}{t-t_0}\sum_{s=t_0}^{t}\frac{\Delta_{s-1}}{(\mathbb{E}[r(x)=0]-\Delta_{s-1})]} \\
&\leq \frac{1}{\mathbb{E}[r(x)=0]}\left(4\frac{\Delta_{t-1}}{\mathbb{E}[r(x)=0]}\right) \qquad (9)
\end{aligned}$$

where in the last inequality we used the fact that $\mathbb{E}[r'(x)=0]-\Delta_{s-1} > \frac{1}{2}-\Delta_{s-1} > \frac{1}{4}$ via the same reasoning as in Lemma 1. Similarly,

$$\begin{aligned}
\left|\frac{1}{t-t_0}\sum_{s=t_0}^{t}\left(\frac{1}{\mathbb{E}[r(x)=0]+\Delta_{s-1}} - \frac{1}{\mathbb{E}[r(x)=0]}\right)\right| &= \frac{1}{t-t_0}\sum_{s=t_0}^{t}\frac{\Delta_{s-1}}{(\mathbb{E}[r(x)=0]+\Delta_{s-1})\,\mathbb{E}[r(x)=0]} \\
&= \frac{1}{\mathbb{E}[r(x)=0]}\frac{1}{t-t_0}\sum_{s=t_0}^{t}\frac{\Delta_{s-1}}{(\mathbb{E}[r(x)=0]+\Delta_{s-1})} \\
&\leq \frac{1}{\mathbb{E}[r(x)=0]}\left(4\frac{\Delta_{t-1}}{\mathbb{E}[r(x)=0]}\right). \qquad (10)
\end{aligned}$$

Multiplying (9) and (10) by $\mathbb{E}[L(h(x),y)\mathbf{1}[r(x)=0]]$ and using the fact that $\mathbb{E}[r(x)=0] \geq 1-\rho$ and $\mathbb{E}[L(h(x),y)|r(x)=0] \leq 1$, it follows that

$$\begin{aligned}
&\left|\mathbb{E}[L(h(x),y)\mathbf{1}[r(x)=0]]\frac{1}{t-t_0}\sum_{s=t_0}^{t}\mathbb{E}\left[\frac{1}{\mathbb{E}[r(x)=0]+\Delta_{s-1}}\right] - \mathbb{E}[L(h(x),y)|r(x)=0]\right| \\
&\leq \mathbb{E}[L(h(x),y)|r(x)=0]\left(4\frac{\Delta_{t-1}}{\mathbb{E}[r(x)=0]}\right) \leq \frac{4}{1-\rho}\Delta_{t-1} \qquad (11)
\end{aligned}$$

and

$$\left| \mathbb{E}[L(h(x), y)\mathbf{1}[r(x) = 0]] \frac{1}{t - t_0} \sum_{s=t_0}^{t} \mathbb{E}\left[\frac{1}{\mathbb{E}[r(x) = 0] - \Delta_{s-1}} \right] - \mathbb{E}[L(h(x), y)|r(x) = 0] \right|$$
$$\leq \frac{4}{1 - \rho} \Delta_{t-1}. \tag{12}$$

Using the series of inequalities (8), (11), and (12) into the result of the concentration inequality of (5) along with some algebra,

$$|\,\mathbb{E}[L(h(x), y)|r(x) = 0] - \mathbb{E}[L(h'(x), y)|r'(x) = 0] - \widehat{L}_t(h, r) + \widehat{L}_t(h', r')|$$
$$\leq 2\Delta'_t + \frac{8}{1 - \rho} \Delta_{t-1}. \tag{13}$$

Recalling that $\tilde{\Delta}_t = 2\Delta'_t + \frac{8}{1-\rho}\Delta_{t-1}$. Since $V_t \subseteq V_{t-1}$, it holds that for any pair $(h, r) \in V_t$ and $(h', r') \in V_t$ via inequality (13):

$$\begin{aligned}
&\mathbb{E}[L(h(x), y)|r(x) = 0] - \mathbb{E}[L(h'(x), y)|r'(x) = 0] \\
&\leq \widehat{L}_{t-1}(h, r) - \widehat{L}_{t-1}(h', r') + \tilde{\Delta}_{t-1} \\
&\leq \min_{(h,r) \in V_{t-1}} \widehat{L}_{t-1}(h, r) + \tilde{\Delta}_{t-1} - \min_{(h,r) \in V_{t-1}} \widehat{L}_{t-1}(h, r) + \tilde{\Delta}_{t-1} \\
&= 2\tilde{\Delta}_{t-1}.
\end{aligned}$$

We then prove that $(h^*, r^*) \in V_t$. By induction, suppose that $(h^*, r^*) \in V_{t-1}$. Let $(h'_{t-1}, r'_{t-1}) = \min_{(h,r) \in V_{t-1}} \widehat{L}_{t-1}(h, r)$. Then, by inequality (13):

$$\begin{aligned}
&\widehat{L}_{t-1}(h^*, r^*) \\
&\leq \widehat{L}_{t-1}(h'_{t-1}, r'_{t-1}) - \mathbb{E}[L(h'_{t-1}(x), y)|r'_{t-1}(x) = 0] + \mathbb{E}[L(h^*(x), y)|r^*(x) = 0] + \tilde{\Delta}_{t-1} \\
&\leq \widehat{L}_{t-1}(h'_{t-1}, r'_{t-1}) + \tilde{\Delta}_{t-1}
\end{aligned}$$

which by definition means that $(h^*, r^*) \in V_t$. □

Corollary 1. *Given any $\rho < \frac{1}{2}$, for any $\delta > 0$, with probability at least $1 - \delta$, for all $t \geq 16 \log(3t/\delta)$, $\mathbb{E}[L(h_t(x), y)|r_t(x) = 0] \leq \mathbb{E}[L(h_b(x), y)] + \gamma + \tilde{O}\big(\sqrt{(1/t)\log(1/\delta)}\big)$, where $\gamma = \mathbb{E}[L(h^*(x), y)|r^*(x) = 0] - \mathbb{E}[L(h_b(x), y)]$. Furthermore, if $r_\diamond \in \mathcal{R}_\rho$, then $\gamma \leq 0$.*

Proof. Since $h_b \in \mathcal{H}$ and since $r_\diamond \in \mathcal{R}$, it holds that $\mathbb{E}[L(h^*(x), y)|r^*(x) = 0] \leq \mathbb{E}[L(h_b(x), y)]$. □

B.2 Label Complexity

Proposition 1. *For all $\delta > 0$, with probability at least $1 - \delta$, for all $t \geq 16 \log(3t/\delta)$,*

$$\mathbb{E}[p_s(x_s)] \leq 4\theta\, \mathbb{E}[L(h^*(x), y)|r^*(x) = 0] + \tilde{O}\left(\theta \sqrt{\frac{\log(1/\delta)}{t}}\right),$$

where θ is the disagreement coefficient.

Proof. By Lemma (1) in conjunction with Inequalities (9) and (10), it holds that

$$\begin{aligned}
\mathbb{E}[p_s(x_s)] &\leq \mathbb{E}\left[\max_{(h,r),(h',r')\in V_s} \max_y \frac{L(h(x_s),y)\mathbf{1}[r(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r(x_{s'})=0]} - \frac{L(h'(x_s),y)\mathbf{1}[r'(x_s)=0]}{\frac{1}{s-1}\sum_{s'=1}^{s-1}\mathbf{1}[r'(x_{s'})=0]}\right] \\
&\leq 2\,\mathbb{E}\left[\max_{(h,r)\in V_s} \max_y \frac{L(h(x_s),y)\mathbf{1}[r(x_s)=0]}{\mathbb{E}[r(x)=0]-\Delta_{s-1}} - \frac{L(h'(x_s),y)\mathbf{1}[r'(x_s)=0]}{\mathbb{E}[r'(x)=0]+\Delta_{s-1}}\right] \\
&= 2\,\mathbb{E}\left[\max_{(h,r)\in V_s} \max_y \frac{L(h(x_s),y)\mathbf{1}[r(x_s)=0]}{\mathbb{E}[r(x)=0]-\Delta_{s-1}} - \frac{L(h(x_s),y)\mathbf{1}[r(x_s)=0]}{\mathbb{E}[r(x)=0]}\right. \\
&\quad + \frac{L(h(x_s),y)\mathbf{1}[r(x_s)=0]}{\mathbb{E}[r(x)=0]} - \frac{L(h'(x_s),y)\mathbf{1}[r'(x_s)=0]}{\mathbb{E}[r'(x)=0]+\Delta_{s-1}} - \frac{L(h'(x_s),y)\mathbf{1}[r'(x_s)=0]}{\mathbb{E}[r'(x)=0]} \\
&\quad \left. + \frac{L(h'(x_s),y)\mathbf{1}[r'(x_s)=0]}{\mathbb{E}[r'(x)=0]}\right] \\
&\leq \mathbb{E}\left[\max_{(h,r),(h',r')\in V_s} \max_y \left|\frac{L(h(x_s),y)\mathbf{1}[r(x_s)=0]}{\mathbb{E}[r(x)=0]} + \frac{L(h'(x_s),y)\mathbf{1}[r'(x_s)=0]}{\mathbb{E}[r'(x)=0]}\right|\right] \\
&\quad + \frac{8}{(1-\rho)^2}\Delta_{s-1}.
\end{aligned}$$

We then focusing on bounding the first term above. By Theorem 2, $V_s \subseteq \{(h,r) \in \mathcal{H}\times\mathcal{R} : \mathbb{E}[L(h(x),y)|r(x)=0] \leq \mathbb{E}[L(h^*(x),y)|r^*(x)=0] + 2\tilde{\Delta}_{s-1}\}$. Using this fact in conjunction with $\rho((h,r),(h^*,r^*)) \leq \mathbb{E}[L(h(x),y)|r(x)=0] + \mathbb{E}[L(h^*(x),y)|r^*(x)=0]$ implies that $V_s \subseteq B(h^*,r^*,\Lambda)$ where $\Lambda = 2\,\mathbb{E}[L(h^*(x),y)|r^*(x)=0] + 2\tilde{\Delta}_{s-1}$. where we used the definition of disagreement coefficient in the last inequality.

Using the above it holds that:

$$\begin{aligned}
&\mathbb{E}\left[\max_{(h,r),(h',r')\in V_s} \max_y \left|\frac{L(h(x_s),y)\mathbf{1}[r(x_s)=0]}{\mathbb{E}[r(x)=0]} - \frac{L(h'(x_s),y)\mathbf{1}[r'(x_s)=0]}{\mathbb{E}[r'(x)=0]}\right|\right] \\
&\leq 2\,\mathbb{E}\left[\max_{(h,r)\in V_s} \max_y \left|\frac{L(h(x_s),y)\mathbf{1}[r(x_s)=0]}{\mathbb{E}[r(x)=0]} - \frac{L(h^*(x_s),y)\mathbf{1}[r^*(x_s)=0]}{\mathbb{E}[r^*(x)=0]}\right|\right] \\
&\leq 2\,\mathbb{E}\left[\max_{(h,r)\in B(h^*,r^*,\Lambda)} \max_y \left|\frac{L(h(x_s),y)\mathbf{1}[r(x_s)=0]}{\mathbb{E}[r(x)=0]} - \frac{L(h^*(x_s),y)\mathbf{1}[r^*(x_s)=0]}{\mathbb{E}[r^*(x)=0]}\right|\right] \\
&\leq 2\theta(2\,\mathbb{E}[L(h^*(x),y)|r^*(x)=0] + 2\tilde{\Delta}_{s-1})
\end{aligned}$$

where we used the definition of disagreement coefficient in the last inequality. □

Theorem 3. *Given any $\rho < \frac{1}{2}$, for all $\delta > 0$, with probability at least $1-\delta$, $\sum_{s=1}^{T}\mathbb{E}[p_s(x_s)] = \tilde{O}(\theta\eta T + \theta\sqrt{T})$, where θ is the disagreement coefficient.*

Proof. The theorem follows directly by summing Proposition 1 over the rounds. □

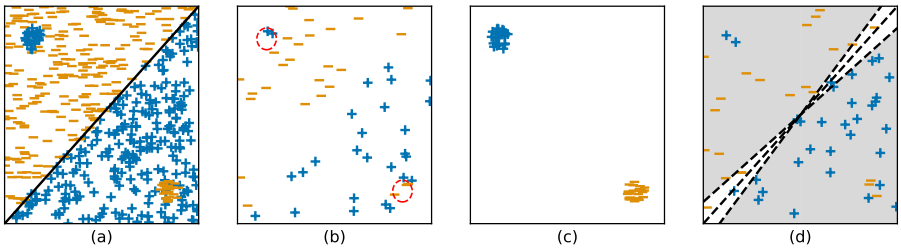


Figure 3: Example distribution: (a) A distribution in $\mathbb{R}^2$. (b) Region selected by an optimal abstention algorithm is depicted by the red circles. (c) Dataset induced by the abstention region in (b). (d) Disagreement region for active learning is shown in white.

C Comparisons to Active and Abstention Learning:

We now provide a simple example to help distinguish our setting from both active and abstention learning. Consider the distribution depicted in Figure 3 (a), and suppose the learner is given a hypothesis class consisting of halfspaces.

Figure 3 (b) depicts the optimal abstention region as dotted red circles. Suppose the learner were handed this region by an oracle (ignoring any computation and sample complexity concerns in actually finding this region). Then recall that in our setting the mechanism for avoiding loss (i.e., "abstaining") is also the same mechanism for generating labeled examples. Thus, if the learner were to solely request labels from the optimal abstention region, it would generate the dataset depicted in Figure 3 (c) and this dataset would induce precisely the wrong hypothesis.

Figure 3 (d) depicts a finite dataset drawn according to the distribution and the dotted lines correspond to a set of "good" hypotheses. Active learning algorithms generally label examples as long as there is disagreement between good hypotheses on the label of the example. While the exact details vary by algorithm, an optimal active learner gains no information by labeling an example that all good hypotheses agree on. Thus, the grayed region in Figure 3 (d) will never be labeled. Moreover, as the active learner hones in on the optimal hypothesis, and the set of candidate good hypotheses shrinks, it will eventually stop requesting labels entirely, even though there is value in requesting labels purely for evading the loss of an incorrect prediction (e.g. requesting labels inside the two dotted red circles).

dataset	# features	# train	# test	$\lvert H \rvert$	$1/C$
a9a	123	20,000	12,561	64	2^{-11}
cod-rna	8	25,000	34,535	64	2^{-13}
mnist	780	30,000	30,000	32	2^{-10}
phishing	68	9,000	2,055	46	2^{-11}
shuttle	9	20,000	23,499	64	2^{-13}
skin	3	150,000	95,065	46	2^{-11}

Table 1: Dataset characteristics.

D Empirical Investigation Appendix

In this section, we first give additional details on the DPL-IWAL ablation study, then present a study on estimating γ for margin-based requesters, and finally present additional details for the evaluation of DPL-Simplified.

D.1 DPL-IWAL Ablation Study

In this study, we test the following publicly available datasets: `mnist`, `shuttle`, `cod-rna`, `phishing`, `skin` and `a9a`. Details of dataset training/test split sizes, number features, and the size of finite hypothesis sets used in the ablation study is shown in Table 1. All datasets can be found at the LIBSVM dataset repository: https://www.csie.ntu.edu.tw/∼cjlin/libsvmtools/datasets/, released under the 3-clause BSD license. For the `mnist` dataset we learn the binary classification of odd vs. even digits and for the `shuttle` dataset we classify the majority class vs. the rest. For all datasets we normalize features by first centering each feature (subtracting the mean value of each feature column) and then scaling to unit variance (dividing by the standard deviation of each feature column). After this, the entire data matrix is scaled uniformly so that the maximum instance feature vector has unit norm, i.e. $\max_i \|x_i\| = 1$.

For the $\mathcal{H}$ and $\mathcal{R}_\rho$ set construction, we use logistic regression models trained using the `scikit-learn` library 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. Then to create the margin-based requesters $\mathcal{R}_\rho$, for each hypothesis $h \in \mathcal{H}$, we use the *unlabeled* training fold examples to estimate a threshold τ that captures ρ fraction of the distribution.

For each of the baseline methods, we adjust the uniform sampling rate so that the overall number of requested labels matches that of our proposed algorithm. See Figure 5 for the labeling budget as well as a breakdown of whether the request came from q_t and/or r_t. Similar to IWAL Beygelzimer et al. [2009], the sampling probability p_t of DPL-IWAL can be adjusted by rescaling the loss function by a constant. In these experiments, we found it effective to downweight the loss difference of Algorithm 1 by a factor 0.1 when the signs of $h(x)$ and $h'(x)$ agree. In order to increase efficiency, at some cost in adaptivity, we update the version space and best model/requester pair after every 25 examples, which is a standard technique when applying active learning in practice Amin et al. [2020].

Figure 6 shows the results of our ablation study on all datasets. We observe that the uniform random r_t attains similar test error to the fully uniform baseline. This is because if r is a uniform random requestor, then for all $h \in \mathcal{H}$, it holds that $\mathbb{E}[L(h(x), y)|r(x) = 0] = \mathbb{E}[L(h(x), y)]$. Thus, uniform r_t behaves like the standard active learner, IWAL, which will generalize as well as a passive learner, meaning that with respect to the number of points processed, it will attain a test error close to that of a passive learner. At the same time, online errors for uniform r_t are smaller than that of the fully uniform baseline. This suggests that the samples that are selected by an active learning strategy are also somewhat correlated with mistakes. Selecting a non-trivial requestor, but constraining q_t to sample uniformly demonstrates that there is indeed significant value to selecting a good request region, as demonstrated by lower online mistakes and smaller conditional test errors.

Our experiments with the uniform r_t baseline demonstrate that naively deploying IWAL in a dual purpose setting will yield suboptimal results. This shows the considerable benefits of DPL-IWAL, which uses an IWAL inspired sampling scheme carefully adapted for the dual purpose setting. A natural question is whether DPL-IWAL can use different active learning algorithms for its sampling

dataset	# features	# examples	$1/C$	notes
a9a	123	32,561	10^{-3}	
acoustic	50	78,823	10^{-4}	class 1 vs. rest
cifar	10	60,000	10^{-3}	project to 10-dim w/ PCA; class 1 vs. rest
cod-rna	8	59,535	10^{-3}	
covtype	54	581,012	10^{-5}	called "covtype.binary" on LIBSVM site
HIGGS	28	100,000	10^{-4}	randomly subsampled from full 11M dataset
ijcnn	22	49,990	10^{-4}	
mnist	780	60,000	10^{-2}	odd vs. even
shuttle	9	43,499	10^{-5}	class 1 vs. rest
skin	3	245,065	10^{-2}	

Table 2: Dataset characteristics.

procedure (such as uncertaity/margin sampling, DHM, or query-by-committee Balcan et al. [2007], Dasgupta et al. [2008], Dagan and Engelson [1995]), and whether these result in even more effective algorithms for the dual purpose setting. These directions pose new technical challenges, for example, if using a margin-based sampler q_t in addition to a margin-based requestor r_t, the two of which may be highly correlated, it becomes non-trivial to estimate the loss conditioned on $r_t = 0$ using samples selected by q_t. Nevertheless, we look forward to investigating these potentially fruitful directions in future work.

Additionally, in Figure 7, we measure the validity of the assumption $E[L(h_t(x), y)|R_t(x) = 0] \leq E[L(h_t(x), y)|r_t(x) = 0]$, which is used in the discussion of the label complexity upper bound. This figures shows that cross these benchmark datasets, the assumption clearly holds.

D.2 Estimating γ for Margin-based Requestors

Recall that in Corollary 1 the bound on the mistakes of our algorithm's chosen hypothesis as compared to the best-in-class is more favorable the more negative γ is. In this study, we measure an upper bound on the value of γ for a margin-based requester function r used in the ablation study and use a continuous hypothesis space to estimate $h_b = \operatorname{argmin}_h \mathbb{E}[L(h(x), y)]$.

We can measure a (potentially pessimistic) upper bound on γ for the class $\mathcal{R}_{\rho,\mathcal{H}}$ by using: $\gamma = \mathbb{E}[L(h^*(x), y) \mid r^*(x) = 0] - \mathbb{E}[L(h_b(x), y)] \leq \mathbb{E}[L(h_b(x), y) \mid r_{h_b}(x) = 0] - \mathbb{E}[L(h_b(x), y)] = \mathbb{E}[L(h_b(x), y) \mid |h_b(x)| > \tau_{h,\rho}(\mathcal{D}_\mathcal{X})] - \mathbb{E}[L(h_b(x), y)]$. For each dataset, we first estimate h_b minimizing the log-loss over the full dataset, then we measure the empirical estimate of $\mathbb{E}[L(h_b(x), y) \mid |h_b(x)| > \tau]$ for various choices of τ. In Figure 4, we show measurements of this upper bound on γ for varying values of the C regularization parameter. The left-most data-point of the plot is the expected loss of h_b, that is $\mathbb{E}[L(h_b(x), y) \mid |h_b(x)| > 0] = \mathbb{E}[L(h_b(x), y)]$. As the graph progresses to the right, for each non-zero threshold value, the conditional loss outside of the threshold is always smaller than the value at $\tau = 0$. This implies a strictly negative upper bound on the value of γ. We also observe that γ tends to grow as τ, or equivalently ρ, increases. This empirical finding of a strictly negative γ verifies the hypothesis returned by our algorithm admits a favorable bound in Corollary 1 as compared to the best-in-class.

D.3 DPL-Simplified Study

In Section 6 we evaluate the DPL-Simplified algorithm and baselines using the 10 datasets described in Table 2, all of which can be found on the LIBSVM dataset website: https://www.csie.ntu.edu.tw/∼cjlin/libsvmtools/datasets/. Some of the datasets which are multi-class case into binary classification problems, which is indicated in the notes column of Table 2 (along with any other pre-processing).

For all datasets we normalize features by first centering each feature (subtracting the mean value of each feature column) and then scaling to unit variance (dividing by the standard deviation of each feature column). After this, the entire data matrix is scaled uniformly so that the maximum instance feature vector has unit norm, i.e. $\max_i \|x_i\| = 1$.

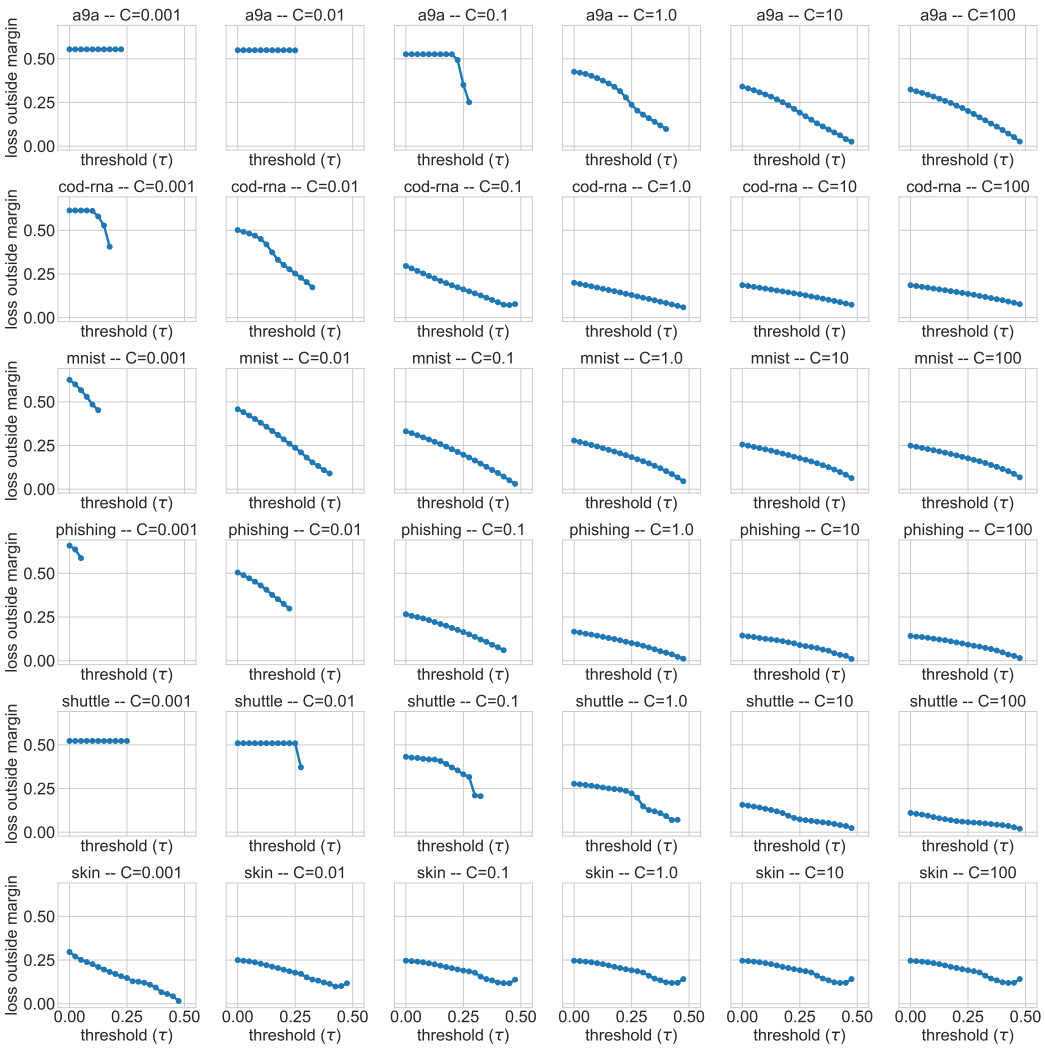


Figure 4: Empirical estimate of $\mathbb{E}[L(h_b(x), y) \mid |h_b(x)| > \tau]$ as a function of τ, which can be used to lower bound γ. As explained in the text, the fact that the value at $\tau = 0$ is strictly larger than the values at any $\tau > 0$, implies a strictly negative upper bound on γ for the linear model family, $\mathcal{H}$, and margin-based requester class $\mathcal{R}_{\rho,\mathcal{H}}$, across these distributions.

The scikit-learn library is used to train the logistic regression model, h, using `solver='saga'`, `max_iter=10000`, and setting the L_2 regularization parameter indicated in Table 2 for each dataset. In order to train the KNN model needed for the KNN- and Mixture-Requesters, we use scikit-learn's `KNeighborsClassifier`, with `n_neighbors=10`, `weights='distance'`, and `algorithm='brute'`.

Finally, Figure 8 shows the full learning curves associated with Table 2 in the main paper.

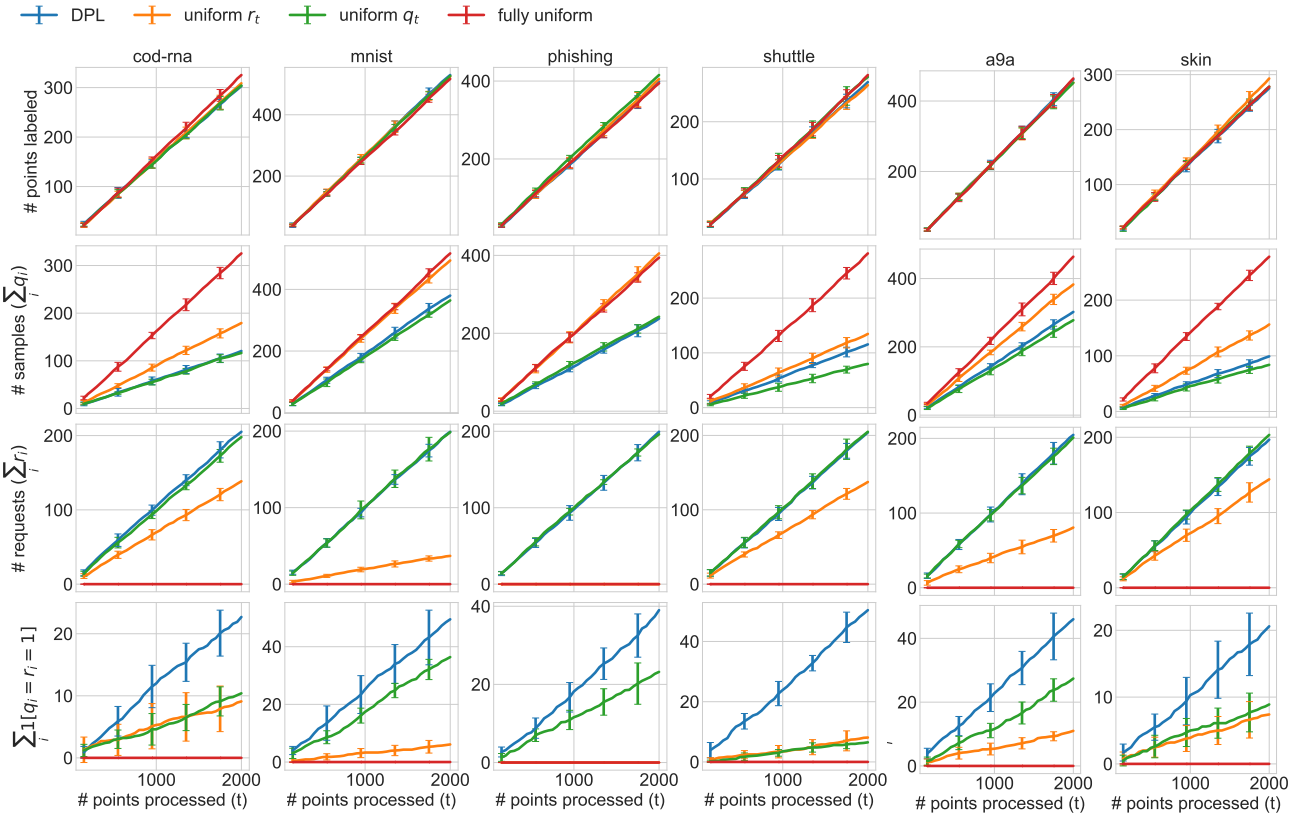


Figure 5: This figure shows for each dataset (each column) the overall number of labels requested throughout the training process (first row), the number examples that where $q_t = 1$ (second row), the number of examples where $r_t(x_t) = 1$ (third row), an finally, the number of cases where are label was requested due to both q_t and $r_t(x_t)$ making a request.

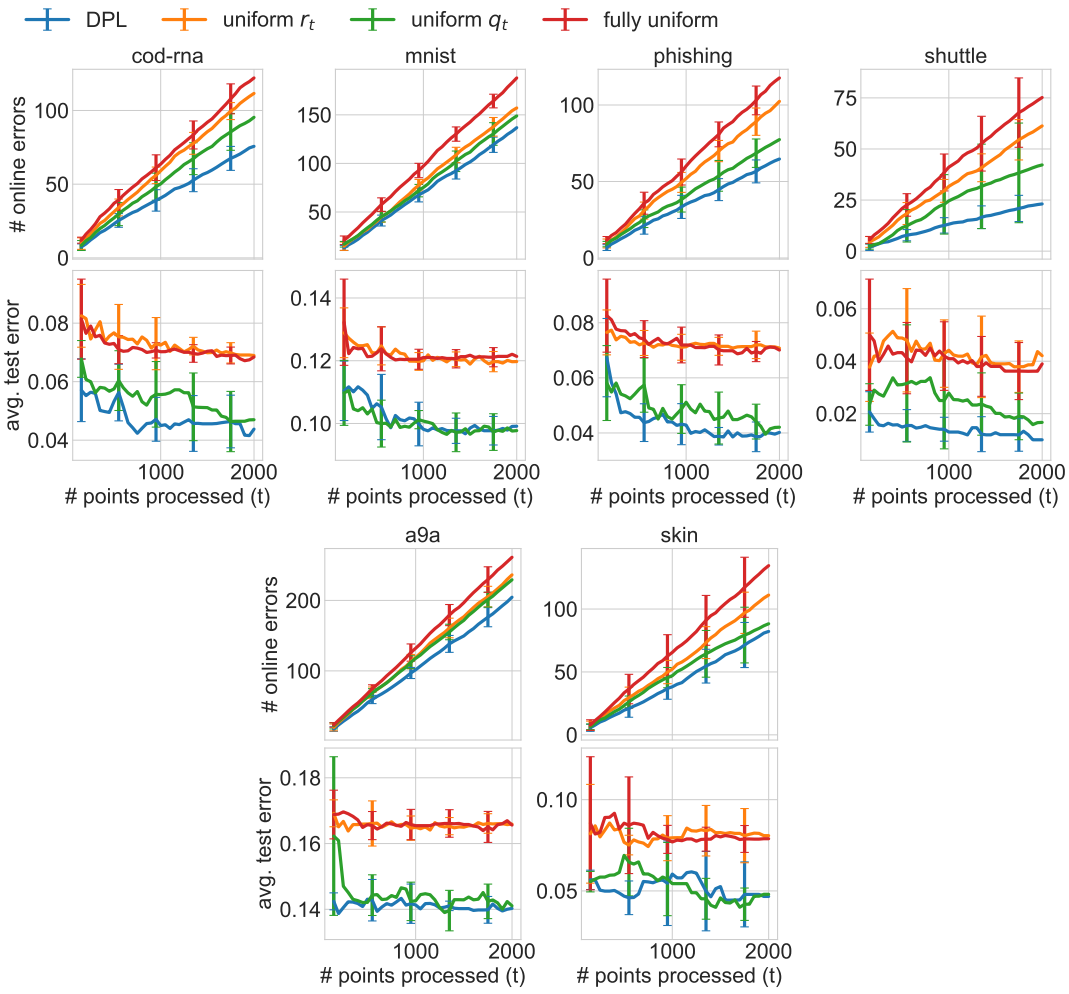


Figure 6: The number of online mistakes made while processing a stream of data as well as the held-out conditional loss on non-requested points made by DPL-IWAL and baselines comparators. The plots show the mean and standard deviation over 10 trials.

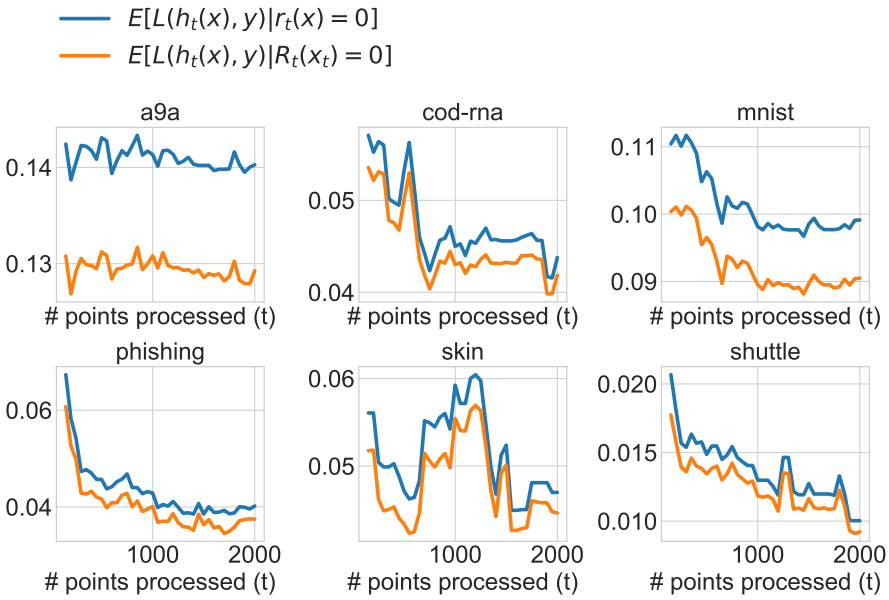


Figure 7: The blue curve is equivalent to the conditional test error of the DPL-IWAL algorithm, while the orange curve display the condition loss on the set of points where $R_t(x) = 0$, i.e. $q_t = 0 \wedge r_t(x) = 0$.

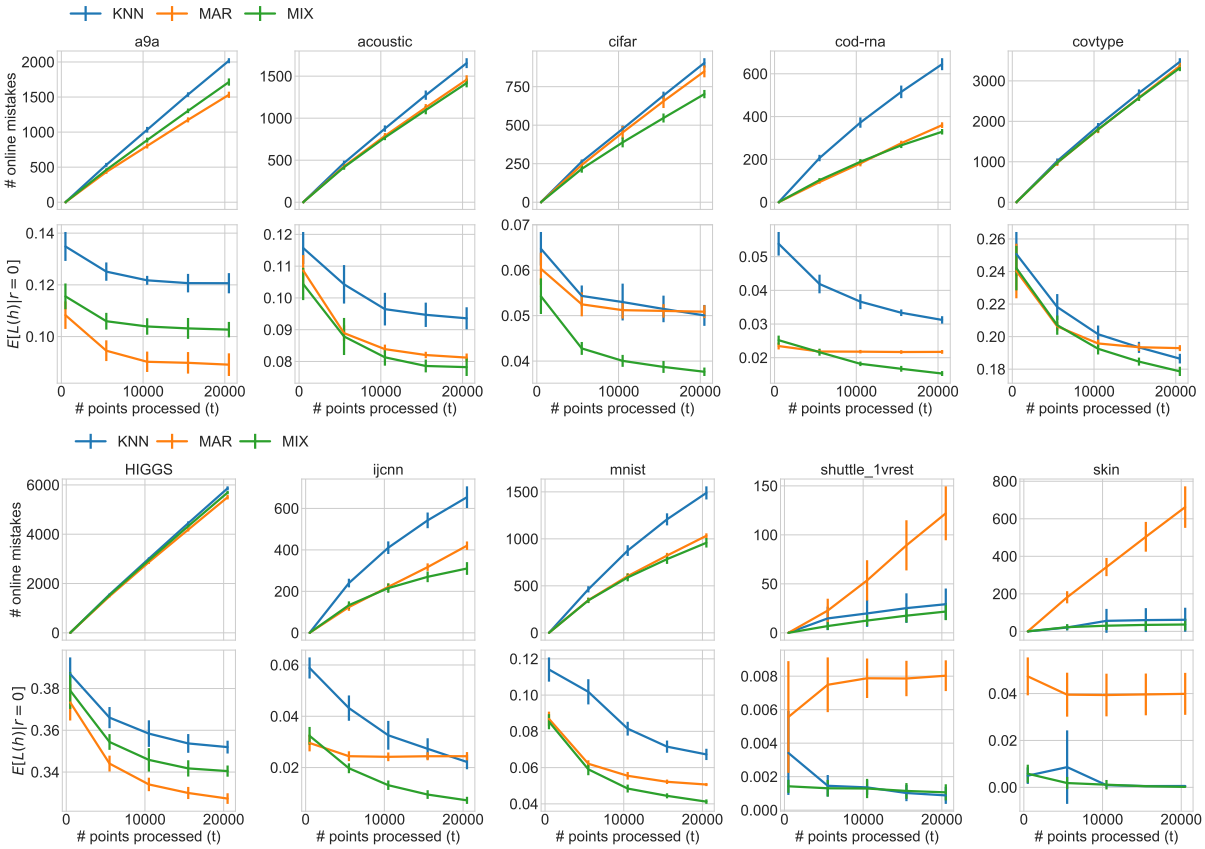


Figure 8: The full learning curves associated to the results presented in Table 2.