Activized Learning with Uniform Classification Noise

Liu Yang LIUY@CS.CMU.EDU
Machine Learning Department, Carnegie Mellon University

Steve Hanneke STEVE.HANNEKE@GMAIL.COM

Abstract

We prove that for any VC class, it is possible to transform any passive learning algorithm into an active learning algorithm with strong asymptotic improvements in label complexity for every nontrivial distribution satisfying a uniform classification noise condition. This generalizes a similar result proven by (Hanneke, 2009; 2012) for the realizable case, and is the first result establishing that such general improvement guarantees are possible in the presence of restricted types of classification noise.

1. Introduction

In many machine learning applications, there is an abundance of cheap unlabeled data, while obtaining enough labels for supervised learning requires significantly more time, effort, or other costs. It is therefore important to try to reduce the total number of labels needed for supervised learning. One of the most appealing approaches to this problem is *active learning*, a protocol in which the learning algorithm itself selects which of the unlabeled data points should be labeled, in an interactive (sequential) fashion. There is now a well-established literature full of compelling theoretical and empirical evidence indicating that active learning can significantly reduce the number of labels required for learning, compared to learning from randomly selected points (passive learning). However, there remain a number of fundamental open questions regarding how strong the theoretical advantages of active learning over passive learning truly are, particularly when faced with the challenge of noisy labels.

At present, there is already a vast literature on the design and analysis of passive learning algorithms, built up over several decades by a substantial number of researchers. In approaching the problem of designing effective active learning algorithms, we might hope to circumvent the need for a commensurate amount of effort, by directly building upon the existing tried-and-true passive learning methods. By leveraging the increased power afforded by the active learning protocol, we may hope to further reduce the number of labels required to learn with these same methods.

Toward this end, (Hanneke, 2009; 2012) recently proposed a framework called *activized learning*, in which a passive learning algorithm is provided as a subroutine to an active meta-algorithm, which constructs data sets to feed into the passive subroutine, and uses the returned classifiers to inform the active learning process. The objective is to design this meta-algorithm in such a way as to guarantee that the number of label requests required to learn to a desired accuracy will always be significantly reduced compared to the number of random labeled examples the given passive learning algorithm would require to obtain a similar accuracy; in this case, we say the active meta-algorithm *activizes* the given passive algorithm. This reduction-based framework captures the typical approach to the design of active learning algorithms in practice (see e.g., (Tong and Koller, 2001; Baldridge and Palmer, 2009; Settles, 2010)), and is appealing because it may inherit the tried-and-true properties (e.g., learning bias) of the given passive learning algorithm, while further reducing the number of labels required for learning.

If an active meta-algorithm activizes *every* passive learning algorithm, under some stated conditions, then it is called a *universal activizer* under those conditions. In the original analysis, (Hanneke, 2009) proved that such universal activizers do exist under the condition that the target concept resides in a known space of finite VC dimension and that there is no label noise (the so-called *realizable case*). (Hanneke, 2012) also proved that there exist classification noise models under which there typically do *not* exist universal activizers, even with the Bayes optimal classifier in a known space of finite VC dimension. Thus, there is a question of what types of noise admit the existence of universal activizers for a given type of concept space.

In this work, we study the classic *uniform classification noise* model of (Angluin and Laird, 1988). In this model,

Proceedings of the 30^{th} International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28. Copyright 2013 by the author(s).

there is a target concept residing in a known concept space of finite VC dimension, and the labels in the training data are corrupted by independent and identically distributed noise variables. The probability that a given label in the training set differs from that of the target concept is referred to as the *noise rate*, and is always strictly less than $1/2$ so that the target concept is also the unique Bayes optimal classifier. Below, we find that there *do* exist universal activizers for any VC classes under the uniform classification noise model. This represents the first general result establishing the existence of universal activizers for VC classes in the presence of classification noise. Our proof of this result builds upon the established methods of (Hanneke, 2012), but requires several novel technical contributions in addition, including a rather interesting technique for handling the problem of adapting to the value of the noise rate.

The paper is structured as follows. In Section 2, we formalize the setting and objective. This is followed in Section 3 with a description of a helpful method and result of (Hanneke, 2012). We then proceed to construct two useful subroutines in Section 4, proving a relevant guarantee for each. Finally, in Section 5, we present our meta-algorithm and prove the main result: that the proposed meta-algorithm is indeed a universal activizer for VC classes under the uniform classification noise model.

2. Notation and Definition

We are interested in a statistical learning setting for binary classification, in which there is some joint distribution $\mathcal{D}_{XY}$ on $\mathcal{X} \times \{-1, +1\}$, and we denote by $\mathcal{D}$ the marginal distribution of $\mathcal{D}_{XY}$ on $\mathcal{X}$. For any classifier $h : \mathcal{X} \to \{-1, +1\}$, denote by $\mathrm{er}(h) = \mathcal{D}_{XY}(\{(x, y) : h(x) \neq y\})$ the *error rate* of h. There is additionally a set $\mathbb{C}$ of classifiers, called the *concept space*, and we denote by d the *VC dimension* of $\mathbb{C}$ (Vapnik and Chervonenkis, 1971; Vapnik, 1982); throughout this work, we will suppose $d < \infty$, in which case $\mathbb{C}$ is called a *VC class*. We also denote by $\nu(\mathbb{C}; \mathcal{D}_{XY}) = \inf_{h \in \mathbb{C}} \mathrm{er}(h)$ the *noise rate* of $\mathcal{D}_{XY}$ with respect to $\mathbb{C}$. We will be interested in the set of distributions satisfying the uniform classification noise assumption of (Angluin and Laird, 1988), which supposes there is an element $h^*_{\mathcal{D}_{XY}} \in \mathbb{C}$ for which the Y values are simply the $h^*_{\mathcal{D}_{XY}}(X)$ values, except corrupted by independently flipping each Y to equal $-h^*_{\mathcal{D}_{XY}}(X)$ with some constant probability (less than $1/2$).

Definition 1. *For a given concept space* $\mathbb{C}$*, define the set of* uniform classification noise *distributions* $\mathrm{UniformNoise}(\mathbb{C}) = \{\mathcal{D}_{XY} : \exists h^*_{\mathcal{D}_{XY}} \in \mathbb{C}, \eta(\mathcal{D}_{XY}) \in [0, 1/2)$ *such that for* $(X, Y) \sim \mathcal{D}_{XY}$*,* $\mathbb{P}(Y \neq h^*_{\mathcal{D}_{XY}}(X) | X) = \eta(\mathcal{D}_{XY})\}$.

For $\mathcal{D}_{XY} \in \mathrm{UniformNoise}(\mathbb{C})$, the classifier $h^*_{\mathcal{D}_{XY}}$ is referred to as the *target function*; note that we clearly have $\nu(\mathbb{C}; \mathcal{D}_{XY}) = \eta(\mathcal{D}_{XY})$, and $h^*_{\mathcal{D}_{XY}} = \mathrm{argmin}_h \mathrm{er}(h)$.

In the learning problem, there is a sequence $\mathcal{Z} = \{(X_i, Y_i)\}_{i=1}^{\infty}$ where the (X_i, Y_i) are independent and $\mathcal{D}_{XY}$-distributed; we denote by $\mathcal{Z}_m = \{(X_i, Y_i)\}_{i=1}^{m}$. The $\{X_i\}_{i=1}^{\infty}$ sequence is referred to as the *unlabeled* data sequence, while the Y_i values are referred to as the *labels*. An active learning algorithm has direct access to the X_i values, but must request to observe the labels Y_i one at a time. In the specific active learning protocol we study here, the active learning algorithm is given as input a *budget* $n \in \mathbb{N}$, and is allowed to request the values of at most n labels; based on the X_i values, the algorithm selects an index $i_1 \in \mathbb{N}$, receives the value Y_{i_1}, then selects another index i_2, receives the value Y_{i_2}, etc. This continues for up to n rounds, after which the algorithm returns a classifier.

Definition 2. *An active learning algorithm* $\mathcal{A}$ *achieves label complexity* $\Lambda_a(\cdot, \cdot)$ *if, for any joint distribution* $\mathcal{D}_{XY}$*,* $\forall \varepsilon > 0$*,* $\forall n \geq \Lambda_a(\varepsilon, \mathcal{D}_{XY})$*,* $\mathbb{E}\left[\mathrm{er}(\mathcal{A}(n))\right] \leq \varepsilon$*.*

Since some $\mathcal{D}_{XY}$ have *no* classifiers h with $\mathrm{er}(h) \leq \varepsilon$ for small $\varepsilon > 0$, we will be interested in analyzing the quantity $\Lambda_a(\nu(\mathbb{C}; \mathcal{D}_{XY}) + \varepsilon, \mathcal{D}_{XY})$, the number of labels sufficient to achieve expected error rate within ε of the best error rate achievable by classifiers in $\mathbb{C}$.

In the present context, we define a *passive learning algorithm* as any function $\mathcal{A}_p(\cdot)$ mapping any finite sequence of labeled examples to a classifier.

Definition 3. *A passive learning algorithm* $\mathcal{A}_p$ *achieves label complexity* $\Lambda_p(\cdot, \cdot)$ *if, for any joint distribution* $\mathcal{D}_{XY}$*,* $\forall \varepsilon > 0$*,* $\forall n \geq \Lambda_p(\varepsilon, \mathcal{D}_{XY})$*,* $\mathbb{E}[\mathrm{er}(\mathcal{A}_p(\mathcal{Z}_n))] \leq \varepsilon$*.*

For any $m \in \mathbb{N}$ and sequence $\mathcal{L} \in (\mathcal{X} \times \{-1, +1\})^m$, we additionally define the *empirical error rate* of a classifier h as $\mathrm{er}_{\mathcal{L}}(h) = m^{-1} \sum_{(x,y) \in \mathcal{L}} \mathbb{1}[h(x) \neq y]$; when $\mathcal{L} = \mathcal{Z}_m$, we abbreviate $\mathrm{er}_m(h) = \mathrm{er}_{\mathcal{Z}_m}(h)$. We also use the notation $V[(x, y)] = \{h \in V : h(x) = y\}$ for any $V \subseteq \mathbb{C}$.

Following (Hanneke, 2009; 2012), we now formally define what it means to *activize* a passive algorithm. An *active meta-algorithm* is a procedure $\mathcal{A}_a$ taking as input two arguments, namely a passive learning algorithm $\mathcal{A}_p$ and a label budget $n \in \mathbb{N}$, and returning a classifier $\hat{h} = \mathcal{A}_a(\mathcal{A}_p, n)$, such that $\mathcal{A}_a(\mathcal{A}_p, \cdot)$ is an active learning algorithm. Define the class of functions $\mathrm{Polylog}(1/\varepsilon)$ as those g s.t. $\exists k \in [0, \infty)$, $g(\varepsilon) = O\left(\log^k(1/\varepsilon)\right)$. Here, and in all contexts below, the asymptotics are always considered as $\varepsilon \to 0$ (from above) when considering a function of ε, and as $n \to \infty$ when considering a function of n; all other quantities are considered constants in these asymptotics. In particular, we write $g_1(\varepsilon) = o(g_2(\varepsilon))$ (or equivalently $g_2(\varepsilon) = \omega(g_1(\varepsilon))$) to mean $\lim_{\varepsilon \to 0} g_1(\varepsilon)/g_2(\varepsilon) = 0$.

For a label complexity Λ_p, we will consider any $\mathcal{D}_{XY}$ for which $\Lambda_p(\cdot, \mathcal{D}_{XY})$ is relatively small as being *triv-*

ial, indicating that we need not concern ourselves with improving the label complexity for that $\mathcal{D}_{XY}$ since it is already very small; for our purposes, "relatively small" means polylog. Furthermore, keeping with the reduction style of the framework, we will only require our active learning methods to be effective when the given passive algorithm has "reasonable" behavior. Formally, define the set $\mathrm{Nontrivial}(\Lambda_p; \mathbb{C})$ as those $\mathcal{D}_{XY}$ for which, $\forall \varepsilon > 0$, $\Lambda_p(\nu(\mathbb{C}; \mathcal{D}_{XY}) + \varepsilon, \mathcal{D}_{XY}) < \infty$, and $\forall g \in \mathrm{Polylog}(1/\varepsilon)$, $\Lambda_p(\nu(\mathbb{C}; \mathcal{D}_{XY}) + \varepsilon, \mathcal{D}_{XY}) = \omega(g(\varepsilon))$. Finally, define an *activizer* under uniform classification noise as follows.

Definition 4. *(Hanneke, 2009; 2012) We say an active meta-algorithm $\mathcal{A}_a$* activizes *a passive algorithm $\mathcal{A}_p$ for $\mathbb{C}$ under* $\mathrm{UniformNoise}(\mathbb{C})$ *if the following condition holds. For any label complexity Λ_p achieved by $\mathcal{A}_p$, the active learning algorithm $\mathcal{A}_a(\mathcal{A}_p, \cdot)$ achieves a label complexity Λ_a such that $\forall \mathcal{D}_{XY} \in \mathrm{UniformNoise}(\mathbb{C}) \cap \mathrm{Nontrivial}(\Lambda_p; \mathbb{C})$, $\exists c \in [1, \infty)$ s.t. (letting $\nu = \nu(\mathbb{C}; \mathcal{D}_{XY})$)*

$$\Lambda_a(\nu + c\varepsilon, \mathcal{D}_{XY}) = o\left(\Lambda_p(\nu + \varepsilon, \mathcal{D}_{XY})\right).$$

In this case, $\mathcal{A}_a$ is called an activizer *for $\mathcal{A}_p$ with respect to $\mathbb{C}$ under* $\mathrm{UniformNoise}(\mathbb{C})$*, and the active learning algorithm $\mathcal{A}_a(\mathcal{A}_p, \cdot)$ is called the $\mathcal{A}_a$-activized $\mathcal{A}_p$. If $\mathcal{A}_a$ activizes* every *passive algorithm for $\mathbb{C}$ under* $\mathrm{UniformNoise}(\mathbb{C})$*, we say $\mathcal{A}_a$ is a* universal activizer *for $\mathbb{C}$ under* $\mathrm{UniformNoise}(\mathbb{C})$.

This definition says that, for all nontrivial distributions satisfying the uniform classification noise model, the activized $\mathcal{A}_p$ algorithm has a label complexity with a strictly slower rate of growth compared to that of the original $\mathcal{A}_p$ algorithm. For instance, if the original label complexity of $\mathcal{A}_p$ was $\Theta(1/\varepsilon)$, then a label complexity of $O(\log(1/\varepsilon))$ for the activized $\mathcal{A}_p$ algorithm would suffice to satisfy this condition (as would, for instance, $O(1/\varepsilon^{1/2})$). The two slight twists on this interpretation are the restriction to nontrivial distributions and the factor of c loss in the ε argument. As noted by (Hanneke, 2012), the restriction to some notion of "nontrivial" $\mathcal{D}_{XY}$ is necessary, since we clearly cannot hope to improve over passive in certain trivial scenarios, such as when $\mathcal{D}$ has support on a single point; passive learning can have $O(\log(1/\varepsilon))$ label complexity in this case. The implication of this definition is that the activized algorithm's label complexity is superior to any nontrivial upper bound on the passive method's label complexity. It is not known whether the loss in the ε argument, by a constant c, is really necessary in general (even for the realizable case). However, this only really makes a difference for rather strange passive learning methods; in most cases, $\Lambda_p(\nu + \varepsilon; \mathcal{D}_{XY}) = \mathrm{poly}(1/\varepsilon)$, in which case we can set $c = 1$ by increasing the leading constant on Λ_a. Our analysis below reveals we can set this c arbitrarily close to 1, or even to a certain $(1 + o(1))$ function of ε.

2.1. Summary of Results

In this work, we construct an active meta-algorithm, referred to as Meta-Algorithm 1 below, and prove that it is a universal activizer for $\mathbb{C}$ under $\mathrm{UniformNoise}(\mathbb{C})$. This applies to *any* VC class $\mathbb{C}$. The significance of this result is primarily a deeper understanding of the advantages of active learning over passive learning. This first step beyond the realizable case in activized learning is particularly interesting in light of established negative results indicating that there exist noise models under which there do not exist universal activizers for certain VC classes (Hanneke, 2012).

The proof is structured as follows. We first review a technique of (Hanneke, 2012) for active learning based on shatterable sets, represented by Subroutine 1 below. For our purposes, the important property of this technique is that it produces a set of labeled examples, where each example has either its true (noisy) label, or else has the label of the target function itself (i.e., the de-noised label). It also has the desirable property that the number of examples in this set is significantly larger than the number of label requests used to produce it. These properties, originally proved by (Hanneke, 2012), are summarized in Lemma 1.

We may then hope that if we feed this labeled sample into the given passive learning algorithm, then as long as this sample is larger than the label complexity of that algorithm, it will produce a good classifier; since we used a much smaller number of label requests compared to the size of this sample, we would therefore have the desired improvements in label complexity. Unfortunately, it is not always so simple. The fact that some of the examples are de-noised turns out to be a problem, as there are passive algorithms whose performance may be highly dependent on the uniformity of the noise, and their performance can actually degrade from denoising select data points. For instance, there are several algorithms in the literature on efficiently learning linear separators under uniform classification noise, which may produce worse classifiers if given a partially-denoised set of examples instead of the original noisy examples. Even common methods such as logistic regression can be made to perform worse by denoising select instances. So our next step is to alter this sample to appear more like a typical sample from $\mathcal{D}_{XY}$; that is, oddly enough, we need to re-noise the de-noised examples.

The difficulty in re-noising the sample is that we do not know the value of the noise rate $\eta(\mathcal{D}_{XY})$. Furthermore, estimating $\eta(\mathcal{D}_{XY})$ to the required precision would require an excessive number of labeled examples, too large to obtain the desired performance gains. So we devise a means of getting what we need, without actually estimating $\eta(\mathcal{D}_{XY})$, by a combination of coarse estimation and a kind of brute-force search. With this approximate noise rate in hand, we simply flip each de-noised label with probability

$\approx \eta(\mathcal{D}_{XY})$, so that the sample now appears to be a typical sample for $\mathcal{D}_{XY}$. This method for re-noising the sample is referred to as Subroutine 2 below, and its effectiveness is described in Lemma 2. Feeding this sample to the passive algorithm then achieves the desired result.

However, before we can conclude, there is some clean-up to be done, as the above techniques generate a variety of undesirable by-products that we must sort through to find the aforementioned re-noised de-noised labeled sample. Specifically, in addition to the large partially de-noised labeled sample, Subroutine 1 also generates several spurious sets of labeled examples, with no detectable way to determine which one is the sample we are interested in. Furthermore, in addition to the re-noised data set resulting from adding noise at rate $\approx \eta(\mathcal{D}_{XY})$, Subroutine 2 also generates several samples re-noised with noise rates differing significantly from $\eta(\mathcal{D}_{XY})$. As such, our approach is to take all of the sets produced by Subroutine 1, run them all through Subroutine 2, and then run the passive algorithm with each of the resulting samples. This results in a large collection of classifiers. We know that at least one of them has the required error rate, and our task is simply to find one of comparable quality. So we perform a kind of tournament, comparing pairs of classifiers by querying for the labels of points where they disagree, and taking as the winner the one that makes fewer mistakes, reasoning that the one with better error rate will likely make fewer mistakes in such a comparison. After several rounds of this tournament, we emerge with an overall winner, which we then return. This tournament technique is referred to as Subroutine 3 below, and the guarantee on the quality of the classifier it selects is given in Lemma 3.

The sections below include the details of these methods, with rigorous analyses of their respective behaviors.

3. Active Learning Based on Shatterable Sets

This section describes an approach to active learning investigated by (Hanneke, 2009; 2012), along with a particular result of (Hanneke, 2012) useful for our purposes. Recall that we say a set of classifiers V *shatters* $\{x_1, \ldots, x_m\} \in \mathcal{X}^m$ if, $\forall y_1, \ldots, y_m \in \{-1, +1\}$, $\exists h \in V$ s.t. $\forall i \leq m, h(x_i) = y_i$. To simplify notation, define $\mathcal{X}^0 = \{\varnothing\}$, and say V shatters $\varnothing$ iff $V \neq \{\}$; also suppose $\mathbb{P}(\mathcal{X}^0) = 1$.

Now consider the definition of Subroutine 1 given in Figure 1, based on a similar method of (Hanneke, 2012). For our purposes, for $m \in \mathbb{N}$, the quantity $\hat{U}_m(\delta)$ is defined as follows, based on an analysis by (Vapnik and Chervonenkis, 1971) of excess risk bounds.

$$\hat{U}_m(\delta) = \frac{2}{m} + 2\sqrt{\frac{\ln(12/\delta) + d\ln(2em/d)}{m}}.$$

The results below would also hold for certain other choices of $\hat{U}_m(\delta)$, which may sometimes yield smaller label complexity guarantees; see (Hanneke, 2012) for one such alternative. The quantities $\hat{\mathbb{P}}(\cdots)$ in Subroutine 1 are estimators for their respective analogous quantities $\mathbb{P}(\cdots)$, based only on *unlabeled* examples. Their specific definitions are not particularly relevant to the present discussion, but for completeness are included in an appendix available online.

Subroutine 1 operates as follows. We first request a number of labels for random points. We then use these labeled examples to prune away any classifiers that make significantly more mistakes than some other classifier in $\mathbb{C}$, leaving a subset V of classifiers from $\mathbb{C}$ whose empirical error rates are relatively small. We then proceed to construct $d + 1$ different pairs $(\mathcal{L}_k, Q_k)$ of labeled data sets. For each k, each data point X_m in the sequence will be inserted into either $\mathcal{L}_k$ or Q_k, along with a corresponding label. If it is determined (in Step 6) that, for most sequences $S \in \mathcal{X}^{k-1}$ that V shatters, V also shatters $S \cup \{X_m\}$, then we request the label Y_m and add the pair (X_m, Y_m) to Q_k. For each $S \in \mathcal{X}^{k-1}$ shattered by V for which V does not shatter $S \cup \{X_m\}$, there is some $y \in \{-1, +1\}$ and some classification of S such that every $h \in V$ that classifies S in that way has $h(X_m) = y$; we let $\hat{y}$ denote the value of y for which this happens on a larger fraction of sequences S of this type; if X_m was not already inserted into Q_k, then we insert the pair $(X_m, \hat{y})$ into $\mathcal{L}_k$. Thus, the set Q_k represents the set of labeled examples for which we explicitly requested the labels, while $\mathcal{L}_k$ represents the set of examples we did not request the labels of, along with a kind of *inferred* label. The motivation for this technique arises from the work of (Hanneke, 2012), where it is shown that, for appropriate values of k, with high probability this $\hat{y}$ will agree with $h^*_{\mathcal{D}_{XY}}(X_m)$. The number of data points processed in this way (specified in Step 5) is chosen to be small enough so that, with high probability, the "$t < n$" condition in Step 6 is redundant, and so that the sets $\mathcal{L}_k$ do not grow too large (for technical reasons arising below).

The following result was essentially proven by (Hanneke, 2012) (more precisely, it can easily be established by a combination of various lemmas of (Hanneke, 2012), along similar lines as the proofs of Lemmas 59 and 60 of (Hanneke, 2012)); for completeness, we include a full proof in an appendix available on the web.

Lemma 1. *(Hanneke, 2012) For any VC class* $\mathbb{C}$ *and* $\mathcal{D}_{XY} \in \text{UniformNoise}(\mathbb{C})$*, there exist constants* $k^* \in \{1, \ldots, d+1\}$*,* $c, c' \in (1, \infty)$*, and a monotone sequence* $\phi_1(n) = \omega(n)$ *such that,* $\forall n \in \mathbb{N}$*, with probability at least* $1 - c \cdot \exp\{-c' n^{1/3}\}$*, running Subroutine 1 with label budget* $\lfloor n/2 \rfloor$ *and confidence parameter* $\delta_n = \exp\{-\sqrt{n}\}$ *results in* $|\mathcal{L}_{k^*} \cup Q_{k^*}| \geq \phi_1(n)$ *and* $\text{er}_{\mathcal{L}_{k^*}}(h^*_{\mathcal{D}_{XY}}) = 0$.

In other words, the set $\mathcal{L}_{k^*} \cup Q_{k^*}$ has size $\gg n$, and every $(x, y) \in \mathcal{L}_{k^*}$ has $y = h^*_{\mathcal{D}_{XY}}(x)$.

```
Subroutine 1:
Input : label budget $n$, confidence parameter $\delta$
Output: pairs of labeled data sets $(\mathcal{L}_1, Q_1), (\mathcal{L}_2, Q_2), \ldots, (\mathcal{L}_{d+1}, Q_{d+1})$
Request the first $m_n = \lfloor n/2 \rfloor$ labels, $\{Y_1, \ldots, Y_{m_n}\}$, and let $t \leftarrow m_n$
Let $V = \left\{ h \in \mathbb{C} : \mathrm{er}_{m_n}(h) - \min_{h' \in \mathbb{C}} \mathrm{er}_{m_n}(h') \leq \hat{U}_{m_n}(\delta) \right\}$
For $k = 1, 2, \ldots, d+1$
    $\hat{\Delta}^{(k)} \leftarrow \hat{\mathbb{P}}\left( x : \hat{\mathbb{P}}\left( S \in \mathcal{X}^{k-1} : V \text{ shatters } S \cup \{x\} \middle| V \text{ shatters } S \right) \geq 1/2 \right)$
    $Q_k \leftarrow \{\}, \mathcal{L}_k \leftarrow \{\}$
    For $m = m_n + 1, \ldots, m_n + \min\left\{ \left\lfloor n / \left( 6 \cdot 2^k \hat{\Delta}^{(k)} \right) \right\rfloor, n^{33/32} \right\}$
        If $\hat{\mathbb{P}}\left( S \in \mathcal{X}^{k-1} : V \text{ shatters } S \cup \{X_m\} \middle| V \text{ shatters } S \right) \geq 1/2$ and $t < n$
            Request the label $Y_m$ of $X_m$, and let $Q_k \leftarrow Q_k \cup \{(X_m, Y_m)\}$ and $t \leftarrow t + 1$
        Else, let $\hat{y} \leftarrow \underset{y \in \{-1,+1\}}{\operatorname{argmax}} \hat{\mathbb{P}}\left( S \in \mathcal{X}^{k-1} : V[(X_m, -y)] \text{ does not shatter } S \middle| V \text{ shatters } S \right)$
            $\mathcal{L}_k \leftarrow \mathcal{L}_k \cup \{(X_m, \hat{y})\}$
Return $(\mathcal{L}_1, Q_1), \ldots, (\mathcal{L}_{d+1}, Q_{d+1})$
```

Figure 1. Subroutine 1 produces pairs of labeled samples by a combination of queries and inferences.

4. An Active Meta-algorithm for Uniform Classification Noise

Re-noising the Sample At first glance, Lemma 1 might seem to have almost solved the problem already, aside from the fact that we need to specify a method for selecting the appropriate value of k. For $k = k^*$, the sample $\mathcal{L}_k \cup Q_k$ represents a partially *de-noised* collection of labeled examples, which might intuitively seem even *better* to feed into the passive algorithm than a sample with the true (noisy) labels. However, this reasoning is naïve, since we are seeking a *universal* activizer for $\mathbb{C}$, applicable to *any* passive learning algorithm. In particular, there are many passive learning algorithms that actually use the properties of the noise to their *advantage* in the learning process, so that altering the noise distribution of the sample may alter the behavior of the passive algorithm to ill effects. Indeed, there exist passive learning algorithms whose performance can be made *worse* by de-noising select examples from a given sample. For instance, this is the case for several algorithms in the computational learning theory literature on efficiently learning linear separators under uniform classification noise. It is also true for many methods based on statistical models, such as logistic regression. Another idea might be to simply feed one of the Q_k sets to the passive learning algorithm. However, this suffers from an essentially similar problem, as there are many passive learning algorithms designed for specific distributions over $\mathcal{X}$, which simply do not work when the data has a different distribution. For instance, this is the case for many algorithms in the computational learning theory literature, which are often designed specifically for certain highly-symmetric distributions (e.g., so that one has concentration guarantees on the coefficients in a high-dimensional representation of the target, such as in Fourier analysis). Therefore, if we are interested in designing an active learning algorithm capable of achieving strict improvements over the label complexities of passive learning methods such as these, then we cannot simply use the de-noised labels, nor can we use only the subset of labels actually requested, as input to the passive algorithm.

Thus, in light of the above considerations, we are tasked with the somewhat unusual problem of *re-noising* the de-noised labels, so that the labeled sample appears to be a typical sample one might expect for the supervised learning problem: that is, essentially similar to an iid sample with distribution roughly $\mathcal{D}_{XY}$. Of course, if we had knowledge of the noise rate $\eta(\mathcal{D}_{XY})$, it would be a simple matter to re-noise the sample by independently corrupting each of the de-noised labels with probability $\eta(\mathcal{D}_{XY})$. In the absence of such direct information, it is natural to try to estimate $\eta(\mathcal{D}_{XY})$. However, it turns out it would require too many labeled examples to estimate the noise rate to the precision necessary for re-noising the sample in a way that appears close enough to the $\mathcal{D}_{XY}$ distribution to work well when we feed it into the passive algorithm. So instead, we employ a combination of estimation and search, which turns out to be sufficient for our purposes. Specifically, consider Subroutine 2 in Figure 2. The algorithm first produces a confidence interval for $\eta(\mathcal{D}_{XY})$ of width $2n^{-5/16}$, and then picks a sequence of evenly-spaced values η_j in this range, at increments of $2n^{-17/16}$; for each of these η_j values, it flips the label of each example in $\mathcal{L}$ independently with probability η_j, and produces a labeled data set R_j by merg-

```
Subroutine 2:
Input : label budget $n$, pair of labeled data sets $(\mathcal{L}, Q)$
Output : sequence of $1 + n^{3/4}$ labeled data sets $R_0, R_1, \ldots, R_{n^{3/4}}$
----------------------------------------------------------------
0. Let $\{(X_{\ell_1}, \hat{y}_{\ell_1}), \ldots, (X_{\ell_s}, \hat{y}_{\ell_s})\}$ denote the first $s = \min\{|\mathcal{L}|, n\}$ elements of $\mathcal{L}$
1. Request the labels $Y_{\ell_1}, Y_{\ell_2}, \ldots, Y_{\ell_s}$
2. Calculate $\hat{\eta} = s^{-1} \sum_{i=1}^{s} \mathbb{1}[Y_{\ell_i} \neq \hat{y}_{\ell_i}]$
3. For each $j \in \{1, 2, \ldots, n^{3/4}\}$
4.     Let $\eta_j = \hat{\eta} - n^{-5/16} + 2jn^{-17/16}$, $\mathcal{L}^{(j)} = \{\}$
5.     Let $\{\chi_{ij}\}_{i \in \mathbb{N}}$ be a collection of iid $\{-1, +1\}$ random variables with $\mathbb{P}(\chi_{ij} = -1) = \eta_j$,
       (independent from $\{\chi_{ij'}\}_{i \in \mathbb{N}, j' \neq j}$ and $\mathcal{Z}$)
6.     Let $\mathcal{L}^{(j)} = \{(X_{\ell_i}, \chi_{ij} \cdot \hat{y}_{\ell_i}) : (X_{\ell_i}, \hat{y}_{\ell_i}) \in \mathcal{L}\}$
7. Let $R_0 = \mathcal{L} \cup Q$, and for each $j \in \{1, 2, \ldots, n^{3/4}\}$, let $R_j = \mathcal{L}^{(j)} \cup Q$
8. Return the sequence $R_0, R_1, \ldots, R_{n^{3/4}}$
```

Figure 2. Subroutine 2 returns several data sets, each re-noised according to a guess η_j of $\eta(\mathcal{D}_{XY})$.

ing this corrupted set $\mathcal{L}$ with the set Q. When constructing the samples R_j, we assume the union operator merges the two data sets in a way that preserves their original order in the unlabeled sequence (supposing each X_i also implicitly records its index i). We have the following lemma.

Lemma 2. *Suppose $\mathcal{D}_{XY} \in \mathrm{UniformNoise}(\mathbb{C})$, $\phi_2(n) = \omega(n)$, $Q_X \subset \{X_1, \ldots, X_m\}$, and $\mathcal{L}_X = \{X_1, \ldots, X_m\} \setminus Q_X$. Further suppose that $Q = \{(X_i, Y_i) : X_i \in Q_X\}$, $\mathcal{L} = \{(X_i, h^*_{\mathcal{D}_{XY}}(X_i)) : X_i \in \mathcal{L}_X\}$, $n^{33/32} \geq |\mathcal{L}| \geq \phi_2(n)$, and that $\mathcal{A}_p$ is a passive learning algorithm. Then there is a function $q_1(n) = o(1)$ s.t., if $\{R_i\}_{i=1}^{n^{3/4}}$ is the sequence of data sets returned by Subroutine 2 when provided n and $(\mathcal{L}, Q)$ as inputs, then (letting $\nu = \nu(\mathbb{C}; \mathcal{D}_{XY})$)*

$$\begin{aligned}\mathbb{E}&\left[\min_j \mathrm{er}\left(\mathcal{A}_p(R_j)\right) - \nu\right] \\ &\leq (1 + q_1(n))\mathbb{E}\left[\mathrm{er}\left(\mathcal{A}_p(\mathcal{Z}_m)\right) - \nu\right] \\ &\qquad + (1 + q_1(n)) \cdot \exp\left\{-n^{1/4}\right\}.\end{aligned}$$

Proof. We consider two cases. First, if $\eta(\mathcal{D}_{XY}) = 0$, then $R_0 = \mathcal{Z}_m$, so that the result clearly holds with $q_1(n) = 0$. For the remainder of the proof, suppose $\eta(\mathcal{D}_{XY}) > 0$.

Let $N_1 = \min\{n' \in \mathbb{N} : \min_{m > n'} \phi_2(m) \geq n\}$; this exists because $\phi_2(n) = \omega(n)$. Since $|\mathcal{L}| \geq \phi_2(n)$, if $n > N_1$ we must have $s = n$. If this is the case, then by Hoeffding's inequality, with probability $1 - \exp\left\{-n^{1/4}\right\}$, $|\hat{\eta} - \eta(\mathcal{D}_{XY})| \leq c_1 \cdot n^{-3/8}$ for some (universal) constant $c_1 \in (0, \infty)$. Thus, on this event, letting $N_2 = \max\{N_1, c_1^{16}\}$, if $n > N_2$, we have $\eta(\mathcal{D}_{XY}) \in \left[\hat{\eta} - n^{-5/16}, \hat{\eta} + n^{-5/16}\right]$. In particular, this means $j^* = \mathrm{argmin}_j\, |\eta_j - \eta(\mathcal{D}_{XY})|$ has $|\eta_{j^*} - \eta(\mathcal{D}_{XY})| \leq 2n^{-17/16}$.

Now for any sequence of labels $y_1, \ldots, y_m \in \{-1, +1\}$ s.t. $X_i \in Q_X \implies y_i = Y_i$, we have

$$\begin{aligned}&\frac{\mathbb{P}\left(R_{j^*} = \{(X_i, y_i)\}_{i=1}^m \middle| \{X_i\}_{i=1}^m, Q, j^*\right)}{\mathbb{P}\left(\mathcal{Z}_m = \{(X_i, y_i)\}_{i=1}^m \middle| \{X_i\}_{i=1}^m, Q\right)} \\ &\leq \max_{0 \leq r \leq |\mathcal{L}|} \frac{\eta_{j^*}^r (1 - \eta_{j^*})^{|\mathcal{L}| - r}}{\eta(\mathcal{D}_{XY})^r (1 - \eta(\mathcal{D}_{XY}))^{|\mathcal{L}| - r}} \\ &\leq \max_{0 \leq r \leq |\mathcal{L}|} \left(1 + \tfrac{2n^{-17/16}}{\eta(\mathcal{D}_{XY})}\right)^r \cdot \left(1 + \tfrac{2n^{-17/16}}{1 - \eta(\mathcal{D}_{XY})}\right)^{|\mathcal{L}| - r} \\ &= \left(1 + \tfrac{2n^{-17/16}}{\eta(\mathcal{D}_{XY})}\right)^{|\mathcal{L}|} \\ &\leq \left(1 + \tfrac{2n^{-17/16}}{\eta(\mathcal{D}_{XY})}\right)^{n^{33/32}} \leq \exp\left\{2n^{-1/32}/\eta(\mathcal{D}_{XY})\right\}.\end{aligned}$$

This final quantity approaches 1 as $n \to \infty$, and we therefore define, for any $n > N_2$,

$$q_1(n) = \exp\left\{2n^{-1/32}/\eta(\mathcal{D}_{XY})\right\} - 1 = o(1).$$

In particular, when the above inequalities hold, we have

$$\begin{aligned}&\mathbb{E}\left[\mathrm{er}\left(\mathcal{A}_p(R_{j^*})\right) - \nu \middle| \{X_i\}_{i=1}^m, Q, j^*\right] \\ &\quad \leq (1 + q_1(n))\mathbb{E}\left[\mathrm{er}\left(\mathcal{A}_p(\mathcal{Z}_m)\right) - \nu \middle| \{X_i\}_{i=1}^m, Q\right].\end{aligned}$$

Since we have established that this holds with probability at least $1 - \exp\left\{-n^{1/4}\right\}$ when $n > N_2$, we have, by the law of total expectation, and since we always have $\mathrm{er}(\mathcal{A}_p(R_{j^*})) - \nu \in [0, 1]$, if $n > N_2$,

$$\begin{aligned}&\mathbb{E}\left[\mathrm{er}\left(\mathcal{A}_p(R_{j^*})\right) - \nu\right] \\ &\leq (1 + q_1(n))\mathbb{E}\left[\mathrm{er}\left(\mathcal{A}_p(\mathcal{Z}_m)\right) - \nu\right] + \exp\left\{-n^{1/4}\right\}.\end{aligned}$$

For completeness, we may define $q_1(n) = \exp\left\{N_2^{1/4}\right\}$ for any $n \leq N_2$, and the inequality in the lemma statement trivially holds in this case since then $(1 + q_1(n)) \exp\left\{-n^{1/4}\right\} \geq 1$ for all $n \leq N_2$. □

```
Subroutine 3:
Input : label budget $n$, sequence of classifiers $h_1, h_2, \ldots, h_N$
Output : classifier $h_{\hat{j}}$
0. If $N = 1$, return the single classifier $h_1$
1. For each $i \in \{1, 2, \ldots \lfloor N/2 \rfloor\}$
2.     Let $T_{iN}$ be the next $\lfloor n/N \rfloor$ (previously untouched) unlabeled examples $X_i$
        in the unlabeled sequence for which $h_{2i-1}(X_i) \neq h_{2i}(X_i)$ (if they exist)
3.     Request the label $Y_t$ for each $X_t \in T_{iN}$ and let $S_{iN} = \{(X_t, Y_t) : X_t \in T_{iN}\}$
4.     Let $h'_i = \mathrm{argmin}_{h \in \{h_{2i-1}, h_{2i}\}} \mathrm{er}_{S_{iN}}(h)$
5. If $N$ is odd, let $h'_{\lceil N/2 \rceil} = h_N$
6. Recursively call Subroutine 3 with label budget $n/2$ and classifiers $h'_1, h'_2, \ldots h'_{\lceil N/2 \rceil}$
    and return the classifier it returns
```

Figure 3. Subroutine 3 selects among a set of classifiers by a tournament of pairwise comparisons.

A Tournament for Classifier Selection Lemmas 1 and 2 together indicate that, if we feed each $(\mathcal{L}_k, Q_k)$ pair into Subroutine 2 (using an appropriate fraction of the overall label budget for each call), then we need only find a way to select from among the returned labeled samples R_j, or equivalently from among the set of classifiers $h_j = \mathcal{A}_p(R_j)$, so that the selected $\hat{j}$ has $\mathrm{er}(h_{\hat{j}}) - \nu(\mathbb{C}; \mathcal{D}_{XY})$ not too much larger than $\mathrm{er}(h_{j^*}) - \nu(\mathbb{C}; \mathcal{D}_{XY})$, for j^* as above. Here we develop and analyze such a procedure, based on running a tournament among these classifiers by pairwise comparisons. Specifically, consider Subroutine 3 specified in Figure 3. This algorithm groups the classifiers into pairs, and for each pair it requests a number of labels for points on which the two classifiers disagree; it then discards whichever of these classifiers makes more mistakes, and makes a recursive call on the set of surviving classifiers (the number of which is smaller than the original set by a factor of 2). This procedure admits the following lemma.

Lemma 3. *Suppose $\mathcal{D}_{XY} \in \mathrm{UniformNoise}(\mathbb{C})$. Then there exists a constant $c \in (0, \infty)$ and a function $q_2(n) = o(1)$ such that, for any $n \in \mathbb{N}$ and any sequence of classifiers $h_1, h_2, \ldots, h_N$ with $1 \leq N \leq (d+1)(1 + (4n)^{3/4})$, with probability at least $1 - \exp\left\{-cn^{1/12}\right\}$, the classifier $h_{\hat{j}}$ returned from calling Subroutine 3 with label budget n and classifiers $h_1, h_2, \ldots, h_N$ satisfies $\mathrm{er}(h_{\hat{j}}) - \nu(\mathbb{C}; \mathcal{D}_{XY}) \leq (1 + q_2(n)) \min_j \left(\mathrm{er}(h_j) - \nu(\mathbb{C}; \mathcal{D}_{XY})\right)$.*

Proof. We proceed inductively (it is clear for $N = 1$). Suppose some $i \in \{1, \ldots, \lfloor N/2 \rfloor\}$ has $\mathbb{P}(h_j(X) \neq h^*_{\mathcal{D}_{XY}}(X)) > (1 + n^{-1/12})\mathbb{P}(h_k(X) \neq h^*_{\mathcal{D}_{XY}}(X))$, where $j, k \in \{2i-1, 2i\}$. Then

$$
\begin{aligned}
&\mathbb{E}[\mathrm{er}_{S_{iN}}(h_k)] \\
&\leq (1 - \eta(\mathcal{D}_{XY}))(2 + n^{-1/12})^{-1} + \eta(\mathcal{D}_{XY}) \frac{1 + n^{-1/12}}{2 + n^{-1/12}} \\
&= \frac{1 + \eta(\mathcal{D}_{XY}) n^{-1/12}}{2 + n^{-1/12}}.
\end{aligned}
$$

Denoting by p_1 this latter quantity, and letting $\varepsilon_1 = \frac{(1/2 - \eta(\mathcal{D}_{XY}))n^{-1/12}}{1 + \eta(\mathcal{D}_{XY})n^{-1/12}}$, a Chernoff bound implies

$$
\begin{aligned}
\mathbb{P}(\mathrm{er}_{S_{iN}}(h_k) > 1/2) &= \mathbb{P}\left(\mathrm{er}_{S_{iN}}(h_k) > (1 + \varepsilon_1)p_1\right) \\
&\leq \exp\left\{-c_1 n^{1/4} p_1 \varepsilon_1^2\right\},
\end{aligned}
$$

for an appropriate choice of constant $c_1 \in (0, \infty)$. Simplifying this last expression, we find that it is at most $\exp\left\{-c_2 n^{1/12}\right\}$ for an appropriate constant $c_2 \in (0, \infty)$. A union bound then implies that with probability at least $1 - (N/2)\exp\left\{-c_1 n^{1/12}\right\}$, for each $i \in \{1, \ldots, \lfloor N/2 \rfloor\}$, we have $\mathbb{P}(h'_i(X) \neq h^*_{\mathcal{D}_{XY}}(X)) \leq (1 + n^{-1/12}) \min_{j \in \{2i-1, 2i\}} \mathbb{P}(h_j(X) \neq h^*_{\mathcal{D}_{XY}}(X))$.

Note that, although both n and N are reduced in the recursive calls, they will still satisfy the constraint on the size of N (i.e., $1 \leq N \leq (d+1)(1 + (4n)^{3/4})$), and the sample sizes $|S_\cdot| = \lfloor n/N \rfloor$ remain essentially constant over recursive calls, so that this result can be applied to the recursive calls as well (tweaking the constant in the exponent can compensate for the variability due to the floor function). Thus, applying this argument inductively, combined with a union bound over the $O(\log N)$ recursive calls, we have that there exists a constant $c_2 \in (0, \infty)$ s.t. with probability $\geq 1 - (N \log_2 N)\exp\left\{-c_2 n^{1/12}\right\} \geq 1 - \exp\left\{-cn^{1/12}\right\}$ (for an appropriate $c > 0$), the returned classifier $h_{\hat{j}}$ satisfies (for an appropriate constant $c_3 \in (0, \infty)$)

$$
\begin{aligned}
&\mathbb{P}\left(h_{\hat{j}}(X) \neq h^*_{\mathcal{D}_{XY}}(X)\right) \\
&\leq \left(1 + n^{-1/12}\right)^{c_3 \log n} \min_{1 \leq j \leq N} \mathbb{P}\left(h_j(X) \neq h^*_{\mathcal{D}_{XY}}(X)\right).
\end{aligned}
$$

Note $\forall h$, $\mathrm{er}(h) - \nu(\mathbb{C}; \mathcal{D}_{XY}) = (1 - 2\eta(\mathcal{D}_{XY}))\mathbb{P}(h(X) \neq h^*_{\mathcal{D}_{XY}}(X))$. Also, as $(1 + n^{-1/12})^{c_3 \log n} \leq \exp\left\{c_3 n^{-1/12} \log n\right\}$ approaches 1 as $n \to \infty$, we can define $q_2(n) = \exp\left\{c_3 n^{-1/12} \log n\right\} - 1 = o(1)$ to get

```
Meta-Algorithm 1:
Input : passive learning algorithm $\mathcal{A}_p$, label budget $n$
Output : classifier $\hat{h}$
0. Execute Subroutine 1 with label budget $\lfloor n/2 \rfloor$ and confidence parameter $\delta = \exp\{-\sqrt{n}\}$
1. Let $(\mathcal{L}_1, Q_1), \ldots, (\mathcal{L}_{d+1}, Q_{d+1})$ be the returned pairs of labeled data sets
2. For each $k \in \{1, \ldots, d+1\}$, execute Subroutine 2 with budget $\lfloor n/(4(d+1)) \rfloor$ and $(\mathcal{L}_k, Q_k)$
3. Let $R_{k0}, R_{k1}, \ldots, R_{kM}$ denote the sequence of returned data sets ($M = \lfloor n/(4(d+1)) \rfloor^{3/4}$)
4. Execute Subroutine 3 with label budget $\lfloor n/4 \rfloor$ and classifiers
                $\{\mathcal{A}_p(R_{kj}) : k \in \{1, \ldots, d+1\}, j \in \{0, 1, \ldots, M\}\}$
5. Return the classifier $\hat{h}$ selected by this execution of Subroutine 3
```

Figure 4. Meta-Algorithm 1 is a universal activizer for any VC class under uniform noise.

$\mathrm{er}(h_{\hat{j}}) - \nu(\mathbb{C}; \mathcal{D}_{XY}) \leq (1 + q_2(n)) \min_{1 \leq j \leq N}(\mathrm{er}(h_j) - \nu(\mathbb{C}; \mathcal{D}_{XY}))$. □

5. Main Result: A Universal Activizer for $\mathbb{C}$ under Uniform Classification Noise

We are now finally ready for our main result, establishing the existence of universal activizers for VC classes under uniform classification noise. Specifically, consider the active meta-algorithm described in Figure 4, which combines the above arguments (setting the label budgets for each subroutine appropriately). We have the following result.

Theorem 1. *For any VC class $\mathbb{C}$, Meta-Algorithm 1 is a universal activizer for $\mathbb{C}$ under* $\mathrm{UniformNoise}(\mathbb{C})$.

Proof. Lemma 1 implies that with probability $1 - \exp\{-\Omega(n^{1/3})\}$, some pair $(\mathcal{L}_k, Q_k)$ will satisfy the conditions of Lemma 2; more precisely, on this event, and conditioned on $|\mathcal{L}_k \cup Q_k|$, the pair $(\mathcal{L}_k, Q_k)$ will be distributionally equivalent to a pair $(\mathcal{L}, Q)$ that satisfies these conditions. Thus, combining Lemmas 1 and 2, and the law of total expectation, combined with the fact that $\mathrm{er}(h) - \nu(\mathbb{C}; \mathcal{D}_{XY}) \in [0, 1]$ for $\mathcal{D}_{XY} \in \mathrm{UniformNoise}(\mathbb{C})$, we have that (letting $\nu = \nu(\mathbb{C}; \mathcal{D}_{XY})$)

$$\begin{aligned}\mathbb{E}\left[\min_{k,j} \mathrm{er}(\mathcal{A}_p(R_{kj})) - \nu\right] \\ \leq (1 + o(1)) \sup_{m \geq \phi_1(n)} \mathbb{E}\left[\mathrm{er}(\mathcal{A}_p(\mathcal{Z}_m)) - \nu\right] \\ + \exp\left\{-\Omega(n^{1/4})\right\}.\end{aligned}$$

Finally, combining this with Lemma 3 implies that

$$\begin{aligned}\mathbb{E}\left[\mathrm{er}(\hat{h}) - \nu\right] \leq (1 + o(1)) \sup_{m \geq \phi_1(n)} \mathbb{E}\left[\mathrm{er}(\mathcal{A}_p(\mathcal{Z}_m)) - \nu\right] \\ + \exp\left\{-\Omega(n^{1/12})\right\}. \qquad (1)\end{aligned}$$

For an $n = \Omega(\log^{12}(1/\varepsilon))$, the second term on the right hand side of (1) is $< \varepsilon$. For the first term, note that if $\mathcal{A}_p$ achieves label complexity Λ_p, then in order to make $\sup_{m \geq \phi_1(n)} \mathbb{E}\left[\mathrm{er}(\mathcal{A}_p(\mathcal{Z}_m)) - \nu\right] \leq \varepsilon$, it suffices to take n large enough so that $\phi_1(n) \geq \Lambda_p(\varepsilon + \nu; \mathcal{D}_{XY})$. Thus, since $\phi_1(n) = \omega(n)$, for $\mathcal{D}_{XY} \in \mathrm{Nontrivial}(\Lambda_p; \mathbb{C})$, the smallest $N_{2\varepsilon} \in \mathbb{N}$ such that every $n \geq N_{2\varepsilon}$ has

$$(1 + o(1)) \sup_{m \geq \phi_1(n)} \mathbb{E}\left[\mathrm{er}(\mathcal{A}_p(\mathcal{Z}_m)) - \nu\right] \leq 2\varepsilon$$

satisfies $N_{2\varepsilon} = o(\Lambda_p(\varepsilon + \nu, \mathcal{D}_{XY}))$. Therefore, since any $O(\log^{12}(1/\varepsilon))$ function is also $o(\Lambda_p(\varepsilon + \nu, \mathcal{D}_{XY}))$ for $\mathcal{D}_{XY} \in \mathrm{Nontrivial}(\Lambda_p; \mathbb{C})$, we see that applying Meta-Algorithm 1 to $\mathcal{A}_p$ results in an active learning algorithm that achieves a label compleity Λ_a with the property that, for $\mathcal{D}_{XY} \in \mathrm{UniformNoise}(\mathbb{C}) \cap \mathrm{Nontrivial}(\Lambda_p; \mathbb{C})$, $\Lambda_a(3\varepsilon + \nu, \mathcal{D}_{XY}) \leq \max\left\{N_{2\varepsilon}, O(\log^{12}(1/\varepsilon))\right\} = o\left(\Lambda_p(\varepsilon + \nu, \mathcal{D}_{XY})\right)$, as required.

Thus, Meta-Algorithm 1 activizes $\mathcal{A}_p$ for $\mathbb{C}$ under $\mathrm{UniformNoise}(\mathbb{C})$. Since $\mathcal{A}_p$ was arbitrary, Meta-Algorithm 1 is a universal activizer for $\mathbb{C}$ under $\mathrm{UniformNoise}(\mathbb{C})$, as claimed. □

6. Conclusions

We established the existence of universal activizers for arbitrary VC classes in the presence of uniform classification noise. This is the first result of this generality regarding the advantages active learning over passive learning in the presence of noise.

Previously, (Hanneke, 2009; 2012) has argued that even seemingly benign noise models typically do not permit the existence of universal activizers for arbitrary VC classes. Thus, in an investigation of the existence of universal activizers for VC classes, the key question going forward is whether there are more general noise models, nontrivially subsuming uniform classification noise, under which universal activizers for VC classes still exist.

References

Angluin, D. and Laird, P. Learning from noisy examples. *Machine Learning*, 2:343–370, 1988.

Baldridge, J. and Palmer, A. How well does active learning *actually* work? Time-based evaluation of cost-reduction strategies for language documentation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2009.

Hanneke, S. *Theoretical Foundations of Active Learning*. PhD thesis, Machine Learning Department, School of Computer Science, Carnegie Mellon University, 2009.

Hanneke, S. Activized learning: Transforming passive to active with improved label complexity. *Journal of Machine Learning Research*, 13(5):1469–1587, 2012.

Settles, B. Active learning literature survey. *http://active-learning.net*, 2010.

Tong, S. and Koller, D. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2, 2001.

Vapnik, V. *Estimation of Dependencies Based on Empirical Data*. Springer-Verlag, New York, 1982.

Vapnik, V. and Chervonenkis, A. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16: 264–280, 1971.