A PAC-style Model for Learning from Labeled and Unlabeled Data

Maria-Florina Balcan and Avrim Blum

Computer Science Department, Carnegie Mellon University
{ninamf,avrim}@cs.cmu.edu

Abstract. There has been growing interest in practice in using unlabeled data together with labeled data in machine learning, and a number of different approaches have been developed. However, the assumptions these methods are based on are often quite distinct and not captured by standard theoretical models. In this paper we describe a PAC-style framework that can be used to model many of these assumptions, and analyze sample-complexity issues in this setting: that is, how much of each type of data one should expect to need in order to learn well, and what are the basic quantities that these numbers depend on. Our model can be viewed as an extension of the standard PAC model, where in addition to a concept class C, one also proposes a type of compatibility that one believes the target concept should have with the underlying distribution. In this view, unlabeled data can be helpful because it allows one to estimate compatibility over the space of hypotheses, and reduce the size of the search space to those that, according to one's assumptions, are a-priori reasonable with respect to the distribution. We discuss a number of technical issues that arise in this context, and provide sample-complexity bounds both for uniform convergence and ϵ-cover based algorithms. We also consider algorithmic issues, and give an efficient algorithm for a special case of co-training.

1 Introduction

There has recently been substantial interest in using unlabeled data together with labeled data for machine learning. The motivation is that unlabeled data can often be much cheaper and more plentiful than labeled data, and so if useful information can be extracted from it that reduces the need for labeled examples, this can be a significant benefit. A number of techniques have been developed for doing this, along with experimental results on a variety of different learning problems. These include label propagation for word-sense disambiguation [23]; co-training for classifying web pages [5], parsing [15], improving visual detectors [17], and document classification [19]; transductive SVM [16] and EM [18] for text classification; graph-based methods [3, 24] and others.

The difficulty from a theoretical point of view, however, is that standard discriminative learning models do not really capture how and why unlabeled data can be of help. In particular, in the PAC model there is a complete disconnect between the data distribution D and the target function f being learned [6,

21]. The only prior belief is that f belongs to some class $\mathcal{C}$: even if D is known fully, any function $f \in \mathcal{C}$ is still possible. For instance, it is perfectly natural (and common) to talk about the problem of learning a concept class over the uniform distribution; but clearly in this case unlabeled data is useless — you can just generate it yourself. For learning over an unknown distribution (the standard PAC setting), unlabeled data can help somewhat, by allowing one to use distribution-specific sample-complexity bounds, but this does not seem to fully capture the power of unlabeled data in practice.

In *generative*-model settings, one *can* easily talk theoretically about the use of unlabeled data, e.g., [9, 10]. However, these results typically make strong assumptions that essentially imply that there is only one natural distinction to be made for a given (unlabeled) data distribution. For instance, a typical generative-model setting would be that we assume positive examples are generated by one Gaussian, and negative examples are generated by another Gaussian. In this case, given enough unlabeled data, we could recover the Gaussians and would need labeled data only to tell us which Gaussian is the positive one and which is the negative one.[1] This is too strong an assumption for most real-world settings. Instead, we would like our model to allow for a distribution over data (e.g., documents we want to classify) where there are a number of plausible distinctions we might want to make. In addition, we would like a general framework that can be used to model many different uses of unlabeled data.

The goal of this paper is to provide a PAC-style framework that bridges between these positions and captures many of the ways unlabeled data is typically used. We extend the PAC model in a way that allows one to express relationships that one hopes the target function and underlying distribution will possess, but without going so far as is done in generative models. We then analyze sample-complexity issues in this setting: that is, how much of each type of data one should expect to need in order to learn well, and also give a few algorithmic results.

The idea of the proposed model is to augment the notion of a *concept class* with a notion of *compatibility* between a target function and the data distribution. That is, rather than talking of "learning a concept class $\mathcal{C}$," we will talk of "learning a concept class $\mathcal{C}$ under compatibility notion χ." Furthermore, we require that the degree of compatibility be something that can be estimated from a finite sample. More specifically, we will require that χ is actually a function from $\mathcal{C} \times X$ to $[0, 1]$, where the compatibility of h with D is $\mathbf{E}_{x \in D}[\chi(h, x)]$. The degree of *in*compatibility is then something we can think of as a kind of "unlabeled error rate" that measures how a-priori unreasonable we believe some proposed hypothesis to be. For example,

Example 1 (margins): Suppose examples are points in R^n and $\mathcal{C}$ is the class of linear separators. A natural belief in this setting is that data should be "well-separated": not only should the target function separate the positive and negative examples, but it should do so by some reasonable *margin* γ

[1] Castelli and Cover [9, 10] do not assume Gaussians in particular, but they do assume the distributions are distinguishable, which from our perspective has the same issue.

[16]. In this case, we could define $\chi(h, x) = 1$ if x is farther than distance γ from the hyperplane defined by h, and $\chi(h, x) = 0$ otherwise. So, the incompatibility of h with D is probability mass within distance γ of $h \cdot x = 0$. Or we could define $\chi(h, x)$ to be a smooth function of the distance of x to the separator, if we do not want to commit to a specific γ in advance. (In contrast, defining compatibility of a hypothesis based on the largest γ such that D has probability mass *exactly zero* within distance γ of the separator would *not* fit our model: it cannot be written as an expectation over individual examples and indeed one cannot distinguish "zero" from "exponentially close to zero" with a small sample.)

Example 2 (Co-training): In co-training [5], we assume examples come as pairs $\langle x_1, x_2 \rangle$, and our goal is to learn a pair of functions $\langle h_1, h_2 \rangle$. For instance, if our goal is to classify web pages, x_1 might represent the words on the page itself and x_2 the words attached to links pointing *to* this page from other pages. The hope that underlies co-training is that the two parts of the example are consistent, which then allows the co-training algorithm to bootstrap from unlabeled data.[2] In this case, we might naturally define the incompatibility of some hypothesis $\langle h_1, h_2 \rangle$ as $\mathbf{Pr}_{\langle x_1, x_2 \rangle \in D}[h_1(x_1) \neq h_2(x_2)]$.

Example 3 (Linear separator graph cuts): As a special case of Example 2 above, suppose examples are *pairs* of points in R^n, $\mathcal{C}$ is the class of linear separators, and we believe the two points in each pair should both be on the *same* side of the target function (i.e., like co-training but we are requiring $h_1 = h_2$).[3] Again we can define the incompatibility of some h to be the probability mass on examples $\langle x_1, x_2 \rangle$ such that $h(x_1) \neq h(x_2)$. One thing that makes this problem interesting is that we can view examples as edges, view the data as a graph embedded in R^n, and given a set of labeled and unlabeled data, view our objective as finding a linear separator minimum s-t cut.

This setup allows us to analyze the ability of a finite unlabeled sample to reduce our need for labeled data, as a function of the compatibility of the target function and various measures of the "helpfulness" of the distribution. In particular, in our model we find that unlabeled data can help in several distinct ways.

- If the target function is highly compatible with D, then if we have enough unlabeled data to estimate compatibility over all $h \in \mathcal{C}$, we can in principle

[2] For example, *iterative co-training* uses a small amount of labeled data to get some initial information (e.g., if a link with the words "my advisor" points to a page then that page is probably a faculty member's home page) and then when it finds an unlabeled example where one half is confident (e.g., the link says "my advisor"), it uses that to label the example for training its hypothesis over the other half.

[3] As a motivating example, consider the problem of *word-sense disambiguation*: given the text surrounding some target word (like "plant") we want to determine which dictionary definition is intended (tree or factory?). Yarowsky [23] uses the fact that if a word appears twice in the same document, it is probably being used in the *same* sense both times.

reduce the size of the search space from $\mathcal{C}$ down to just those $h \in \mathcal{C}$ whose estimated compatibility is high.
- By providing an estimate of D, unlabeled data can allow us to use a more refined distribution-specific notion of "hypothesis space size" such as Annealed VC-entropy [11] or the size of the smallest ϵ-cover [2], rather than VC-dimension. In fact, for natural cases (such as those above) we find that the sense in which unlabeled data reduces the "size" of the search space is best described in these distribution-specific measures.
- Finally, if the distribution is especially nice, we may find that not only does the set of compatible $h \in C$ have a small ϵ-cover, but also the elements of the cover are far apart. In that case, if we assume the target function is fully compatible, we may be able to learn from even fewer labeled examples than the $1/\epsilon$ needed just to *verify* a good hypothesis!

Our framework also allows us to address the issue of how much *unlabeled* data we should expect to need. Roughly, the "VCdim/ϵ^2" form of standard PAC sample complexity bounds now becomes a bound on the number of *unlabeled* examples we need. However, technically, the set whose VC-dimension we now care about is not C but rather a set defined by both C and χ: that is, the overall complexity depends both on the complexity of C and the complexity of the notion of compatibility (see Section 4).

Relationship to the luckiness framework. There is a strong connection between our approach and the luckiness framework [20]. In both cases, the idea is to define an ordering of hypotheses that depends on the data, in the hope that we will be "lucky" and find that not too many other functions are as compatible as the target. There are two main differences, however. The first is that the luckiness framework uses labeled data both for estimating compatibility and for learning: this is a more difficult task, and as a result our bounds on labeled data can be significantly better. For instance, in Example 3 above, for any non-degenerate distribution, a dataset of $n/2$ pairs can with probability 1 be completely shattered by fully-compatible hypotheses, so the luckiness framework does not help. In contrast, with a larger (unlabeled) sample, one can potentially reduce the space of compatible functions quite significantly depending on the distribution – see Section 5 and 6. Secondly, the luckiness framework talks about compatibility between a hypothesis and a *sample*, whereas we define compatibility with respect to a distribution. This allows us to talk about the amount of unlabeled data needed to estimate true compatibility. There are also a number of differences at the technical level of the definitions.

Outline of results. We begin by describing our formal framework, and then in Section 3 we give the simplest version of our sample-complexity bounds, for the case of finite hypothesis spaces. In Section 4 we give uniform-convergence bounds for infinite hypothesis spaces. To achieve tighter bounds, in Section 5 we consider ϵ-cover size, and give bounds that hold for algorithms that first use the unlabeled data to choose a small set of "representative" hypotheses (every compatible $h \in \mathcal{C}$

is close to at least one of them), and then choose among the representatives based on the labeled data. In Section 6, we give our algorithmic results. We begin with a particularly simple $\mathcal{C}$ and χ for illustration, and then give our main algorithmic result: an efficient algorithm for learning linear separators in the Co-training model using just a *single* labeled example, under the assumption that the distribution satisfies independence given the label. In the process, we simplify the noisy halfspace learning algorithm of [4] somewhat.

2 A Formal Framework

We assume that examples (both labeled and unlabeled) come according to a fixed unknown distribution D over an instance space X, and they are labeled by some unknown target function c^*. As in the standard PAC model, a *concept class* or *hypothesis space* is a set of functions over the instance space X, and we will often make the assumption (the "realizable case") that the target function belongs to a given class $\mathcal{C}$. For a given hypothesis h, the (true) error rate of h is defined as $err(h) = err_D(h) = \mathbf{Pr}_{x \in D}[h(x) \neq c^*(x)]$. For any two hypotheses $h_1, h_2 \in C$, the distance with respect to D between h_1 and h_2 is defined as $d(h_1, h_2) = d_D(h_1, h_2) = \mathbf{Pr}_{x \in D}[h_1(x) \neq h_2(x)]$. We will use $\widehat{err}(h)$ to denote the empirical error rate of h on a given labeled sample and $\hat{d}(h_1, h_2)$ to denote the empirical distance between h_1 and h_2 on a given unlabeled sample.

We define a *notion of compatibility* to be a mapping from a hypothesis h and a distribution D to $[0, 1]$ indicating how "compatible" h is with D. In order for this to be estimable from a finite sample, we require that compatibility be an expectation over individual examples.[4] Specifically, we define:

Definition 1. *A* legal notion of compatibility *is a function $\chi : \mathcal{C} \times X \to [0, 1]$ where we (overloading notation) define $\chi(h, D) = \mathbf{E}_{x \in D}[\chi(h, x)]$. Given a sample S, we define $\chi(h, S)$ to be the empirical average over the sample.*

Definition 2. *Given compatibility notion χ, the* incompatibility *of h with D is $1 - \chi(h, D)$. We will also call this its* unlabeled error rate*, $err_{unl}(h)$, when χ and D are clear from context. For a given sample S, we use $\widehat{err}_{unl}(h)$ to denote the empirical average over S.*

Finally, we need a notation for the set of functions whose incompatibility is at most some given value τ.

Definition 3. *Given threshold τ, we define $\mathcal{C}_{D,\chi}(\tau) = \{h \in \mathcal{C} : err_{unl}(h) \leq \tau\}$. So, e.g., $\mathcal{C}_{D,\chi}(1) = \mathcal{C}$. Similarly, for a sample S, we define $\mathcal{C}_{S,\chi}(\tau) = \{h \in \mathcal{C} : \widehat{err}_{unl}(h) \leq \tau\}$*

3 Finite hypothesis spaces

We now illustrate how unlabeled data, together with a suitable compatibility notion, can reduce the need for labeled examples. We begin with the case of

[4] Though one could imagine more general notions with this property as well.

finite hypothesis spaces where we measure the "size" of a set of functions by just the number of functions in it. In the standard PAC model, one typically talks of either the realizable case, where we assume that $c^* \in \mathcal{C}$, or the agnostic case where we do not. In our setting, we have the additional issue of *unlabeled* error rate, and can either make an a-priori assumption that the target function's unlabeled error is low, or else aim for a more "Occam-style" bound in which we have a stream of labeled examples and halt once they are sufficient to justify the hypothesis produced. We first give a bound for the "doubly realizable" case.

Theorem 1. *If we see m_u unlabeled examples and m_l labeled examples, where*

$$m_u \geq \frac{1}{\epsilon}\left[\ln|\mathcal{C}| + \ln\frac{2}{\delta}\right] \quad \textit{and} \quad m_l \geq \frac{1}{\epsilon}\left[\ln|\mathcal{C}_{D,\chi}(\epsilon)| + \ln\frac{2}{\delta}\right],$$

then with probability $1-\delta$, all $h \in \mathcal{C}$ with $\widehat{err}(h) = 0$ and $\widehat{err}_{unl}(h) = 0$ have $err(h) \leq \epsilon$.

Proof. Notice that the probability that a given hypothesis h with $err_{unl}(h) > \epsilon$ has $\widehat{err}_{unl}(h) = 0$ is at most $(1-\epsilon)^{m_u} < \delta/(2|\mathcal{C}|)$ for the given value of m_u. Therefore, by the union bound, the number of unlabeled examples is sufficient to ensure that with probability $1-\delta/2$, only hypotheses in $\mathcal{C}_{D,\chi}(\epsilon)$ have $\widehat{err}_{unl}(h) = 0$. The number of labeled examples then similarly ensures that with probability $1-\delta/2$, none of those whose true error is at least ϵ have an empirical error of 0, yielding the theorem. □

So, if the target function indeed is perfectly correct and compatible, Theorem 1 gives sufficient conditions on the number of examples needed to ensure that an algorithm that optimizes both quantities over the observed data will, in fact, achieve a PAC guarantee. To emphasize this, we will say that an algorithm efficiently PAC$_{unl}$-learns the pair $(\mathcal{C}, \chi)$ if it is able to achieve a PAC guarantee using time and sample sizes polynomial in the bounds of Theorem 1.

We can think of Theorem 1 as bounding the number of labeled examples we need as a function of the "helpfulness" of the distribution D with respect to our notion of compatibility. That is, in our context, a helpful distribution is one in which $\mathcal{C}_{D,\chi}(\epsilon)$ is small, and so we do not need much labeled data to identify a good function among them. We can get a similar bound in the situation when the target function is not fully compatible:

Theorem 2. *Given $t \in [0,1]$, if we see m_u unlabeled examples and m_l labeled examples, where*

$$m_u \geq \frac{2}{\epsilon^2}\left[\ln|\mathcal{C}| + \ln\frac{4}{\delta}\right] \quad \textit{and} \quad m_l \geq \frac{1}{\epsilon}\left[\ln|\mathcal{C}_{D,\chi}(t+2\epsilon)| + \ln\frac{2}{\delta}\right],$$

then with probability $1-\delta$, all $h \in \mathcal{C}$ with $\widehat{err}(h) = 0$ and $\widehat{err}_{unl}(h) \leq t+\epsilon$ have $err(h) \leq \epsilon$, and furthermore all $h \in \mathcal{C}$ with $err_{unl}(h) \leq t$ have $\widehat{err}_{unl}(h) \leq t+\epsilon$.

In particular, this implies that if $err_{unl}(c^*) \leq t$ and $err(c^*) = 0$ then with high probability the $h \in \mathcal{C}$ that optimizes $\widehat{err}(h)$ and $\widehat{err}_{unl}(h)$ has $err(h) \leq \epsilon$.

Proof. Same as Theorem 1 except apply Hoeffding bounds to the unlabeled error rates. □

Finally, we give a simple Occam/luckiness type of bound for this setting. Given a sample S, let us define $\mathsf{desc}_S(h) = \ln |\mathcal{C}_{S,\chi}(\widehat{err}_{unl}(h))|$. That is, $\mathsf{desc}_S(h)$ is the description length of h (in "nats") if we sort hypotheses by their empirical compatibility and output the index of h in this ordering. Similarly, define $\epsilon\text{-}\mathsf{desc}_D(h) = \ln |\mathcal{C}_{D,\chi}(err_{unl}(h) + \epsilon)|$. This is an upper-bound on the description length of h if we sort hypotheses by an ϵ-approximation to the their true compatibility.

Theorem 3. *For any set S of unlabeled data, given m_l labeled examples, with probability $1-\delta$,* all $h \in \mathcal{C}$ *satisfying $\widehat{err}(h) = 0$ and $\mathsf{desc}_S(h) \leq \epsilon m_l - \ln(1/\delta)$ have $err(h) \leq \epsilon$. Furthermore, if $|S| \geq \frac{2}{\epsilon^2}[\ln |\mathcal{C}| + \ln \frac{2}{\delta}]$, then with probability $1-\delta$, all $h \in \mathcal{C}$ satisfy $\mathsf{desc}_S(h) \leq \epsilon\text{-}\mathsf{desc}_D(h)$.*

The point of this theorem is that an algorithm can use observable quantities to determine if it can be confident, and furthermore if we have enough unlabeled data, the observable quantities will be no worse than if we were learning a slightly less compatible function using an infinite-size unlabeled sample.

4 Infinite hypothesis spaces: uniform convergence bounds

To reduce notation, we will assume in the rest of this paper that $\chi(h,x) \in \{0,1\}$ so that $\chi(h,D) = \mathbf{Pr}_{x\in D}[\chi(h,x) = 1]$. However, all our sample complexity results can be easily extended to the case that $\chi(h,x) \in [0,1]$.

For infinite hypothesis spaces, the first issue that arises is that in order to achieve uniform convergence of *unlabeled* error rates, the set whose complexity we care about is not $\mathcal{C}$ but rather $\chi(\mathcal{C}) = \{\chi_h : h \in \mathcal{C}\}$ where we define $\chi_h(x) = \chi(h,x)$. For instance, suppose examples are just points on the line, and $\mathcal{C} = \{h_a(x) : h_a(x) = 1 \text{ iff } x \leq a\}$. In this case, $\mathrm{VCdim}(\mathcal{C}) = 1$. However, we could imagine a compatibility function such that $\chi(h_a, x)$ depends on some complicated relationship between the real numbers a and x. In this case, $\mathrm{VCdim}(\chi(\mathcal{C}))$ is much larger, and indeed we would need many more unlabeled examples to estimate compatibility over all of $\mathcal{C}$.

A second issue is that we need an appropriate measure for the "size" of the set of surviving functions. VC-dimension tends not to be a good choice: for instance, if we consider the case of Example 1 (margins), then even if data is concentrated in two well-separated "blobs", the set of compatible separators still has as large a VC-dimension as the entire class even though they are all very similar with respect to D. Instead, we consider the *expected* number of splits of a sample of size m drawn from D (its logarithm is *annealed VC-entropy*) which exhibits better behavior. Specifically, for any $\mathcal{C}$, we denote by $\mathcal{C}[m,D]$ the expected number of splits of m points (drawn i.i.d.) from D with concepts in $\mathcal{C}$. Also, for a given (fixed) $S \subseteq X$, we will denote by $\overline{S}$ the uniform distribution over S, and by $\mathcal{C}[m,\overline{S}]$ the expected number of splits of m points (drawn i.i.d.) from $\overline{S}$ with concepts in $\mathcal{C}$. We can now get a bound as follows:

Theorem 4. *An unlabeled sample of size*

$$m_u = \mathcal{O}\left(\frac{VCdim\left(\chi(\mathcal{C})\right)}{\epsilon^2}\log\frac{1}{\epsilon} + \frac{1}{\epsilon^2}\log\frac{2}{\delta}\right)$$

and a labeled sample of size

$$m_l > \frac{2}{\epsilon}\left[\log(2s) + \log\frac{2}{\delta}\right], \text{ where } s = \mathcal{C}_{D,\chi}(t+2\epsilon)[2m_l, D]$$

is sufficient so that with probability $1-\delta$*, all* $h \in \mathcal{C}$ *with* $\widehat{err}(h) = 0$ *and* $\widehat{err}_{unl}(h) \leq t+\epsilon$ *have* $err(h) \leq \epsilon$*, and furthermore all* $h \in \mathcal{C}$ *have* $|err_{unl}(h) - \widehat{err}_{unl}(h)| \leq \epsilon$*.*

This is the analog of Theorem 2 for the infinite case. In particular, this implies that if $err(c^*) = 0$ and $err_{unl}(c^*) \leq t$, then with high probability the $h \in \mathcal{C}$ that optimizes $\widehat{err}(h)$ and $\widehat{err}_{unl}(h)$ has $err(h) \leq \epsilon$.

Proof Sketch: By standard VC-bounds [11, 22], the number of unlabeled examples is sufficient to ensure that with probability $1-\delta/2$ we can estimate, within ϵ, $\mathbf{Pr}_{x\in D}[\chi_h(x) = 1]$ for all $\chi_h \in \chi(\mathcal{C})$. Since $\chi_h(x) = \chi(h, x)$, this implies we have can estimate, within ϵ, the unlabeled error rate $err_{unl}(h)$ for all $h \in \mathcal{C}$, and so the set of hypotheses with $\widehat{err}_{unl}(h) \leq t+\epsilon$ is contained in $\mathcal{C}_{D,\chi}(t+2\epsilon)$.

The bound on the number of labeled examples follows from [11] (where it is shown that the expected number of partitions can be used instead of the maximum in the standard VC proof). This bound ensures that with probability $1-\delta/2$, none of the functions in $\mathcal{C}_{D,\chi}(t+2\epsilon)$ whose whose true (labeled) error is at least ϵ have an empirical (labeled) error of 0. □

We can also give a bound where we specify the number of labeled examples as a function of the *unlabeled sample*; this is useful because we can imagine our learning algorithm performing some calculations over the unlabeled data and then deciding how many labeled examples to purchase.

Theorem 5. *Given* $t \geq 0$*, an unlabeled sample* S *of size*

$$\mathcal{O}\left(\frac{\max[VCdim(\mathcal{C}), VCdim(\chi(\mathcal{C}))]}{\epsilon^2}\log\frac{1}{\epsilon} + \frac{1}{\epsilon^2}\log\frac{2}{\delta}\right)$$

is sufficient so that if we label m_l *examples drawn uniformly at random from* S*, where*

$$m_l > \frac{4}{\epsilon}\left[\log(2s) + \log\frac{2}{\delta}\right] \quad \text{and} \quad s = \mathcal{C}_{S,\chi}(t+\epsilon)\left[2m_l, \overline{S}\right]$$

then with probability $\geq 1-\delta$*, all* $h \in \mathcal{C}$ *with* $\widehat{err}(h) = 0$ *and* $\widehat{err}_{unl}(h) \leq t+\epsilon$ *have* $err(h) \leq \epsilon$*. Furthermore all* $h \in \mathcal{C}$ *have* $|err_{unl}(h) - \widehat{err}_{unl}(h)| \leq \epsilon$*.*

Proof. Standard VC-bounds (in the same form as for Theorem 4) imply that the number of *labeled* examples m_l is sufficient to guarantee the conclusion of the theorem with "$err(h)$" replaced by "$err_{\overline{S}}(h)$" (the error with respect to $\overline{S}$) and "ϵ" replaced with "$\epsilon/2$". The number of *unlabeled* examples is enough to ensure that, with probability $\geq 1-\delta/2$, for all $h \in \mathcal{C}$, $|err(h) - err_{\overline{S}}(h)| \leq \epsilon/2$. Combining these two statements yields the theorem. □

So, if $err(c^*) = 0$ and $err_{unl}(c^*) \leq t$, then with high probability the $h \in \mathcal{C}$ that optimizes $\widehat{err}(h)$ and $\widehat{err}_{unl}(h)$ has $err(h) \leq \epsilon$. If we assume $err_{unl}(c^*) = 0$ then we can use $\mathcal{C}_{S,\chi}(0)$ instead of $\mathcal{C}_{S,\chi}(t+\epsilon)$.

Notice that for the case of Example 1, in the worst case (over distributions D) this will essentially recover the standard margin sample-complexity bounds. In particular, $\mathcal{C}_{S,\chi}(0)$ contains only those separators that split S with margin $\geq \gamma$, and therefore, s is no greater than the maximum number of ways of splitting $2m_l$ points with margin γ. However, if the distribution is nice, then the bounds can be much better because there may be many fewer ways of splitting S with margin γ. For instance, in the case of two well-separated "blobs" discussed above, if S is large enough, we would have just $s = 4$. We also mention that using [7, 8] we can give versions of these bounds using other complexity measures such as Rademacher averages.

5 ϵ-Cover-based Bounds

The bounds in the previous section are for uniform convergence: they provide guarantees for *any* algorithm that optimizes well on the observed data. In this section, we consider stronger bounds based on ϵ-covers that can be obtained for algorithms that behave in a specific way: they first use the unlabeled examples to choose a "representative" set of compatible hypotheses, and then use the labeled sample to choose among these. Bounds based on ϵ-covers exist in the classical PAC setting, but in our framework these bounds and algorithms of this type are especially natural and convenient.

Recall that a set $C_\epsilon \subseteq 2^X$ is an ϵ-cover for $\mathcal{C}$ with respect to D if for every $c \in \mathcal{C}$ there is a $c' \in C_\epsilon$ which is ϵ-close to c. That is, $\mathbf{Pr}_{x \in D}(c(x) \neq c'(x)) \leq \epsilon$.

To illustrate how this can produce stronger bounds, imagine examples are *pairs* of points in $\{0,1\}^n$, $\mathcal{C}$ is the class of linear separators, and compatibility is determined by whether both points are on the same side of the separator (i.e., the case of Example 3). Now suppose for simplicity that the target function just splits the hypercube on the first coordinate, and the distribution is uniform over pairs having the same first coordinate (so the target is fully compatible). It is not hard to show that given polynomially many unlabeled examples U and $\frac{1}{4}\log n$ labeled examples L, with high probability there will exist high-error functions consistent with L and compatible with U.[5] So, we do not yet have uniform convergence. In contrast, the cover-size of the set of functions compatible with U is constant, so ϵ-cover based bounds allow learning from just a constant number of labeled examples.

[5] Proof: Let V be the set of all variables that (a) appear in *every* positive example of L and (b) appear in *no* negative example of L. Over the draw of L, each variable has a $(1/2)^{2|L|} = 1/\sqrt{n}$ chance of belonging to V, so with high probability V has size at least $\frac{1}{2}\sqrt{n}$. Now, consider the hypothesis corresponding to the conjunction of all variables in V. This correctly classifies the examples in L, and whp it classifies *every* other example in U negative because each example in U has only a $1/2^{|V|}$ chance of satisfying every variable in V, and the size of U is much less than $2^{|V|}$. So, this means it is compatible with U and consistent with L, even though its true error is high.

Theorem 6. *If t is an upper bound for $err_{unl}(c^*)$ and p is the size of a minimum $\epsilon-$* cover *for $\mathcal{C}_{D,\chi}(t+4\epsilon)$, then using m_u unlabeled examples and m_l labeled examples for*

$$m_u = \mathcal{O}\left(\frac{VCdim\left(\chi(\mathcal{C})\right)}{\epsilon^2}\log\frac{1}{\epsilon}+\frac{1}{\epsilon^2}\log\frac{2}{\delta}\right) \quad \textit{and} \quad m_l = \mathcal{O}\left(\frac{1}{\epsilon}\ln\frac{p}{\delta}\right),$$

we can with probability $1-\delta$ identify a hypothesis which is 10ϵ close to c^*.

Proof Sketch: First, given the unlabeled sample U, define $H_\epsilon \subseteq \mathcal{C}$ as follows: for every labeling of U that is consistent with some h in $\mathcal{C}$, choose a hypothesis in $\mathcal{C}$ for which $\widehat{err}_{unl}(h)$ is smallest among all the hypotheses corresponding to that labeling. Next, we obtain C_ϵ by eliminating from H_ϵ those hypotheses f with the property that $\widehat{err}_{unl}(f) > t+3\epsilon$. We then apply a greedy procedure on C_ϵ, and we obtain $G_\epsilon = \{g_1, \cdots, g_s\}$, as follows:

Initialize $H_\epsilon^1 = C_\epsilon$ and $i = 1$.

1. Let $g_i = \operatorname{argmin}_{f\in H_\epsilon^i} \widehat{err}_{unl}(f)$.
2. Using unlabeled data, determine H_ϵ^{i+1} by crossing out from H_ϵ^i those hypotheses f with the property that $\hat{d}(g_i, f) < 3\epsilon$.
3. If $H_\epsilon^{i+1} = \emptyset$ then set $s = i$ and stop; else, increase i by 1 and goto 1.

Our bound on m_u is sufficient to ensure that, with probability $\geq 1-\delta/2$, H_ϵ is an ϵ-cover of $\mathcal{C}$, which implies that, with probability $\geq 1-\delta/2$, C_ϵ is an ϵ-cover for $\mathcal{C}_{D,\chi}(t)$. It is then possible to show G_ϵ is, with probability $\geq 1-\delta/2$, a 5ϵ-cover for $\mathcal{C}_{D,\chi}(t)$ of size at most p. The idea here is that by greedily creating a 3ϵ-cover of C_ϵ with respect to distribution $\overline{U}$, we are creating a 4ϵ-cover of C_ϵ with respect to D, which is a 5ϵ-cover of $\mathcal{C}_{D,\chi}(t)$ with respect to D. Furthermore, we are doing this using no more functions than would a greedy 2ϵ-cover procedure for $\mathcal{C}_{D,\chi}(t+4\epsilon)$ with respect to D, which is no more than the optimal ϵ-cover of $\mathcal{C}_{D,\chi}(t+4\epsilon)$.

Now to learn c^* we use labeled data and we do empirical risk minimization on G_ϵ. By standard bounds [2], the number of labeled examples is enough to ensure that with probability $\geq 1-\delta/2$ the empirical optimum hypothesis in G_ϵ has true error at most 10ϵ. This implies that overall, with probability $\geq 1-\delta$, we find a hypothesis of error at most 10ϵ. □

As an interesting case where unlabeled data helps substantially, consider a co-training setting where the target c^* is fully compatible *and D* satisfies the independence given the label property. As shown by [5], one can boost any weak hypothesis from unlabeled data in this setting (assuming one has enough labeled data to produce a weak hypothesis). We show here that given enough unlabeled data, in fact we can learn from just a single labeled example. Specifically it is possible to show that, for any concept classes $\mathcal{C}_1$ and $\mathcal{C}_2$, we have:

Theorem 7. *Assume that $err(c^*) = err_{unl}(c^*) = 0$ and D satisfies independence given the label. Then using m_u unlabeled examples and m_l labeled examples we can find a hypothesis that with probability $1-\delta$ has error at most ϵ,*

provided that $m_u = \mathcal{O}\left(\frac{1}{\epsilon} \cdot \left[(VCdim(\mathcal{C}_1) + VCdim(\mathcal{C}_2)) \cdot \ln\left(\frac{1}{\epsilon}\right) + \ln\left(\frac{1}{\delta}\right)\right]\right)$ *and* $m_l = \mathcal{O}(\log_{\frac{1}{\epsilon}} \frac{1}{\delta})$.

In particular, by reducing ϵ to poly(δ), we can reduce the number of labeled examples needed m_l to 1. In fact, our argument can be extended to the case considered in [1] that D^+ and D^- merely satisfy constant expansion. In section 6.2, we give an *efficient* algorithm for the case that $\mathcal{C}_1$ and $\mathcal{C}_2$ are the class of linear separators (though that requires true independence given the label).

6 Algorithmic results

6.1 A simple computational example

We give here a simple example to illustrate the bounds in Section 3, and for which we can give a polynomial-time algorithm that takes advantage of them. Let the instance space $X = \{0,1\}^n$, and for $x \in X$, let $\mathsf{vars}(x)$ be the set of variables set to 1 by x. Let $\mathcal{C}$ be the class of monotone disjunctions (e.g., $x_1 \vee x_3 \vee x_6$), and for $h \in \mathcal{C}$, let $\mathsf{vars}(h)$ be the set of variables disjoined by h. Now, suppose we say an example x is compatible with function h if either $\mathsf{vars}(x) \subseteq \mathsf{vars}(h)$ or $\mathsf{vars}(x) \cap \mathsf{vars}(h) = \phi$. This is a very strong notion of "margin": it says, in essence, that every variable is either a positive indicator or a negative indicator, and no example should contain both positive and negative indicators.

Given this setup, we can give a simple efficient PAC$_{unl}$-learning algorithm for this pair $(\mathcal{C}, \chi)$. We begin by using our unlabeled data to construct a graph on n vertices (one per variable), putting an edge between two vertices i and j if there is any example x in our unlabeled sample with $i, j \in \mathsf{vars}(x)$. We now use our labeled data to label the components. If the target function is fully compatible, then no component will get multiple labels (if some compnent does get multiple labels, we halt with failure). Finally, we produce the hypothesis h such that $\mathsf{vars}(h)$ is the union of the positively-labeled components. This is fully compatible with the unlabeled data and has zero error on the labeled data, so by Theorem 1, if the sizes of the datasets are as given in the bounds, with high probability the hypothesis produced will have error $\leq \epsilon$.

Notice that if we want to view the algorithm as "purchasing" labeled data, then we can simply examine the graph, count the number of connected components k, and then request $\frac{1}{\epsilon}[k \ln 2 + \ln \frac{2}{\delta}]$ labeled examples. (Here, $2^k = |\mathcal{C}_{S,\chi}(0)|$.) By the proof of 1, with high probability $2^k \leq |\mathcal{C}_{D,\chi}(\epsilon)|$, so we are purchasing no more than the number of labeled examples in the theorem statement.

Also, it is interesting to see the difference between a "helpful" and "non-helpful" distribution for this problem. An especially non-helpful distribution would be the uniform distribution over all examples x with $|\mathsf{vars}(x)| = 1$, in which there are n components. In this case, unlabeled data does not help at all, and one still needs $\Omega(n)$ labeled examples (or, even $\Omega(n/\epsilon)$ if the distribution is a non-uniform as in VC-dimension lower bounds [13]). On the other hand, a helpful distribution is one such that with high probability the number of components is small, such as the case of features appearing independently given the label.

6.2 Co-training with linear separators

We now consider the case of co-training where the hypothesis class is the class of linear separators. For simplicity we focus first on the case of Example 3: the target function is a linear separator in R^n and each example is a *pair* of points both of which are assumed to be on the same side of the separator (i.e., an example is a line-segment that does not cross the target plane).

As in the previous example, a natural approach is to try to solve the "consistency" problem: given a set of labeled and unlabeled data, our goal is to find a separator that is consistent with the labeled examples and compatible with the unlabeled ones. Unfortunately, this consistency problem is NP-hard: given a graph G embedded in R^n with two distinguished points s and t, it is NP-hard to find the linear separator that cuts the minimum number of edges, even if the minimum is 0 [14]. For this reason, we will make an additional assumption, that the two points in an example are each drawn *independently given the label*. That is, there is a single distribution D over R^n, and with some probability p_+, two points are drawn iid from D_+ (D restricted to the positive side of the target function) and with probability $1 - p_+$, the two are drawn iid from D_- (D restricted to the negative side of the target function). Blum and Mitchell [5] have also given positive algorithmic results for co-training when (a) the two halves of an example are drawn independently given the label (which we are assuming now), (b) the underlying function is learnable via Statistical Query algorithms (which is true for linear separators by [4]), and (c) we have enough labeled data to produce a weakly-useful hypothesis on one of the halves to begin with.[6] Thus, our key contribution here is to show how we can run that algorithm with only *a single labeled example*. In the process, we also simplify the results of [4] somewhat.

Theorem 8. *There is a polynomial-time algorithm (in n and b, where b is the number of bits per example) to learn a linear separator under the above assumptions, using polynomially many unlabeled examples and a single labeled example.*

Proof Sketch: Assume for convenience that the target separator passes through the origin, and let us denote the separator by $c^* \cdot x = 0$. We will also assume for convenience that $p_+ \in [\epsilon/2, 1 - \epsilon/2]$; that is, the target function is not overwhelmingly positive or overwhelmingly negative (if it is, this is actually an easy case, but it makes the arguments more complicated). Define the *margin* of some point x as the distance of $x/|x|$ to the separating plane, or equivalently, the cosine of the angle between c^* and x.

We begin by drawing a large unlabeled sample $S = \left\{\langle x_1^i, x_2^i\rangle\right\}$; denote by S_j the set $\left\{x_j^i\right\}$, for $j = 1, 2$. (We describe our algorithm as working with the fixed unlabeled sample S, since we just need to apply standard VC-dimension arguments to get the desired result.) The first step is to perform a transformation

[6] A weakly-useful predictor is a hypothesis h such that $\mathbf{Pr}[h(x) = 1|c^*(x) = 1] > \mathbf{Pr}[h(x) = 1|c^*(x) = 0] + \epsilon$; it is equivalent to the usual notion of a "weak hypothesis" when the target function is balanced, but requires the hypothesis give more information when the target function is unbalanced.

T on S_1 to ensure that some reasonable $(1/poly)$ fraction of $T(S_1)$ has margin at least $1/poly$, which we can do via the Outlier Removal Lemma of [4, 12].[7] The Outlier Removal Lemma states that one can algorithmically remove an ϵ' fraction of S_1 and ensure that for the remainder, for any vector w, $\max_{x \in S_1}(w \cdot x)^2 \leq poly(n, b, 1/\epsilon')\mathbf{E}_{x \in S_1}[(w \cdot x)^2]$, where b is the number of bits needed to describe the input points. We reduce the dimensionality (if necessary) to get rid of any of the vectors for which the above quantity is zero. We then determine a linear transformation (as described in [4]) so that in that in the transformed space for all unit-length w, $\mathbf{E}_{x \in T(S_1)}[(w \cdot x)^2] = 1$). Since the maximum is bounded, this guarantees that at least a $1/poly$ fraction of the points in $T(S_1)$ have at least a $1/poly$ margin with respect to the separating hyperplane.

To avoid cumbersome notation in the rest of the discussion, we drop our use of "T" and simply use S and c^* to denote the points and separator in the transformed space. (If the distribution originally had a reasonable probability mass at a reasonable margin from c^*, then T could be the identity anyway.)

The second step is we argue that a *random* halfspace has at least a $1/poly$ chance of being a weak predictor on S_1. ([4] uses the perceptron algorithm to get weak learning; here, we need something simpler since we do not yet have any labeled data.) Specifically, consider a point x such that the angle between x and c^* is $\pi/2 - \gamma$, and imagine that we draw h at random subject to $h \cdot c^* \geq 0$ (half of the h's will have this property). Then,

$$\mathbf{Pr}_h(h(x) \neq c^*(x) | h \cdot c^* \geq 0) = (\pi/2 - \gamma)/\pi = 1/2 - \gamma/\pi.$$

Since at least a $1/poly$ fraction of the points in S_1 have at least a $1/poly$ margin this implies that:

$$\mathbf{Pr}_{h,x}[h(x) = 1 | c^*(x) = 1] > \mathbf{Pr}_{h,x}[h(x) = 1 | c^*(x) = 0] + 1/poly.$$

This means that a $1/poly$ probability mass of functions h must in fact be weakly-useful predictors.

The final step of the algorithm is as follows. Using the above observation, we pick a random h, and plug it into the bootstrapping theorem of [5] (which, given unlabeled pairs $\langle x_1^i, x_2^i \rangle \in S$, will use $h(x_1^i)$ as a noisy label of x_2^i, feeding the result into an SQ algorithm), repeating this process $poly(n)$ times. With high probability, our random h was a weakly-useful predictor on at least one of these steps, and we end up with a low-error hypothesis. For the rest of the runs of the algorithm, we have no guarantees. We now observe the following. First of all, any function h with small $err(h)$ must have small $err_{unl}(h)$. Secondly, because of the assumption of independence given the label, as shown in theorem 7, the *only* functions with low unlabeled error rate are functions close to c^*, close to $\neg c^*$, close to the "all positive" function, or close to the "all negative" function.[8] So, if we simply examine all the hypotheses produced by this procedure, and

[7] If the reader is willing to allow running time polynomial in the margin of the data set, then this part of the argument is not needed.

[8] I.e., exactly the case of the generative models we maligned at the start of this paper.

pick some h with a low unlabeled error rate that is at least $\epsilon/2$-far from the "all-positive" or "all-negative" functions, then either h or $\neg h$ is close to c^*. We can now just draw a single labeled example to determine which case is which. □

We can easily extend our algorithm to the standard co-training (where c_1^* can be different from c_2^*) as follows: we repeat the procedure in a symmetric way, and then, in order to find a good pair of functions, just try all combinations of pairs of compatible functions to find one of small unlabeled error rate, not close to "all positive", or "all negative" functions; finally use a constant number of labeled examples to produce a low error hypothesis (and here we use only one part of the example and only one of the functions in the pair).

7 Conclusions

We have provided a PAC-style model that incorporates both labeled and unlabeled data, and have given a number of sample-complexity bounds. The intent of this model is to capture many of the ways unlabeled data is typically used, and to provide a framework for thinking about when and why unlabeled data can help. The main implication of our analysis is that unlabeled data is useful if (a) we have a good notion of compatibility so that the target function indeed has a low unlabeled error rate, (b) the distribution D is *helpful* in the sense that not too many other hypotheses also have a low unlabeled error rate, and (c) we have enough *unlabeled* data to estimate unlabeled error rates well.

Our best (ϵ-cover based) bounds apply to strategies that use the unlabeled data first to select a small set of "reasonable" rules and then use labeled data to select among them, as do our algorithms of Section 6.2. It is interesting to consider how this relates to algorithms (like the original co-training algorithm) that use labeled data first, and then use unlabeled data to bootstrap from them.

Another open problem generally would be to better understand the space of efficient algorithms in this context. In particular, even though we present two positive algorithmic results, even for fairly simple pairs $(\mathcal{C}, \chi)$, it seems difficult to efficiently make full use of unlabeled data without additional assumptions on the distribution. A specific open problem is whether there exist efficient algorithms for the simple problem in Section 6.1 if we allow irrelevant variables. That is, we assume the set of variables is partitioned into 3 groups A, B, and C, each positive example has $|\mathsf{vars}(x) \cap A| \geq 1$ and $|\mathsf{vars}(x) \cap B| = 0$, and each negative example has $|\mathsf{vars}(x) \cap B| \geq 1$ and $|\mathsf{vars}(x) \cap A| = 0$, but we allow $|C| > 0$.

Acknowledgements. We thank Santosh Vempala for a number of useful discussions. This work was supported in part by NSF grants IIS-0312814 and CCR-0105488.

References

1. M. F. Balcan, A. Blum, and K. Yang. Co-training and expansion: Towards bridging theory and practice. In *NIPS*, 2004.
2. G.M. Benedek and A. Itai. Learnability with respect to a fixed distribution. *Theoretical Computer Science*, 86:377–389, 1991.

3. A. Blum and S. Chawla. Learning from labeled and unlabeled data using graph mincuts. In *Proc. ICML*, pages 19–26, 2001.
4. A. Blum, A. Frieze, R. Kannan, and S. Vempala. A polynomial-time algorithm for learning noisy linear threshold functions. *Algorithmica*, 22:35–52, 1998.
5. A. Blum and T. M. Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. 11th Annual Conf. Computational Learning Theory*, pages 92–100, 1998.
6. A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth. Learnability and the Vapnik Chervonenkis dimension. *Journal of the ACM*, 36(4):929–965, 1989.
7. S. Boucheron, O. Bousquet, and G. Lugosi. Theory of classification: a survey of recent advances. Manuscript, 2004.
8. S. Boucheron, G. Lugosi, and P. Massart. A sharp concentration inequality with applications. *Random Structures and Algorithms*, 16:277–292, 2000.
9. V. Castelli and T.M. Cover. On the exponential value of labeled samples. *Pattern Recognition Letters*, 16:105–111, 1995.
10. V. Castelli and T.M. Cover. The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter. *IEEE Transactions on Information Theory*, 42(6):2102–2117, 1996.
11. L. Devroye, L. Gyorfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer-Verlag, 1996.
12. J. Dunagan and S. Vempala. Optimal outlier removal in high-dimensional spaces. In *Proceedings of the 33rd ACM Symposium on Theory of Computing*, 2001.
13. A. Ehrenfeucht, D. Haussler, M. Kearns, and L. Valiant. A general lower bound on the number of examples needed for learning. *Inf. and Comput*, 82:246–261, 1989.
14. A. Flaxman. Personal communication, 2003.
15. R. Hwa, M. Osborne, A. Sarkar, and M. Steedman. Corrected co-training for statistical parsers. In *ICML-03 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*, Washington D.C., 2003.
16. T. Joachims. Transductive inference for text classification using support vector machines. In *Proc. ICML*, pages 200–209, 1999.
17. A. Levin, P. Viola, and Y. Freund. Unsupervised improvement of visual detectors using co-training. In *Proc. 9th Int. Conf. Computer Vision*, pages 626–633, 2003.
18. K. Nigam, A. McCallum, S. Thrun, and T.M. Mitchell. Text classification from labeled and unlabeled documents using EM. *Mach. Learning*, 39(2/3):103–134, 2000.
19. S.-B. Park and B.-T. Zhang. Co-trained support vector machines for large scale unstructured document classification using unlabeled data and syntactic information. *Information Processing and Management*, 40(3):421 – 439, 2004.
20. J. Shawe-Taylor, P. L. Bartlett, R. C. Williamson, and M. Anthony. Structural risk minimization over data-dependent hierarchies. *IEEE Transactions on Information Theory*, 44(5):1926–1940, 1998.
21. L.G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, 1984.
22. V. N. Vapnik. *Statistical Learning Theory*. John Wiley and Sons Inc., 1998.
23. D. Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Meeting of the Association for Computational Linguistics*, pages 189–196, 1995.
24. X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proc. ICML*, pages 912–912, 2003.