Access to Unlabeled Data can Speed up Prediction Time

Ruth Urner RURNER@CS.UWATERLOO.CA
Shai Ben-David SHAI@CS.UWATERLOO.CA
David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON N2L 3G1 CANADA

Shai Shalev-Shwartz SHAIS@CS.HUJI.AC.IL
School of Computer Science and Engineering, The Hebrew University of Jerusalem, ISRAEL

Abstract

Semi-supervised learning (SSL) addresses the problem of training a classifier using a small number of labeled examples and many unlabeled examples. Most previous work on SSL focused on how availability of unlabeled data can improve the accuracy of the learned classifiers. In this work we study how unlabeled data can be beneficial for constructing *faster* classifiers. We propose an SSL algorithmic framework which can utilize unlabeled examples for learning classifiers from a predefined set of fast classifiers. We formally analyze conditions under which our algorithmic paradigm obtains significant improvements by the use of unlabeled data. As a side benefit of our analysis we propose a novel quantitative measure of the so-called cluster assumption. We demonstrate the potential merits of our approach by conducting experiments on the MNIST data set, showing that, when a sufficiently large unlabeled sample is available, a fast classifier can be learned from much fewer labeled examples than without such a sample.

1. Introduction

In many learning applications unlabeled data is abundantly available while labeled examples are much harder (or more expensive) to obtain. When data is generated by some fixed but unknown underlying distribution, unlabeled samples generated by that distribution reveal information about the distribution and, at least intuitively, such information could be utilized towards the construction of a label predictor. This intuition drives the work on semi-supervised learning (SSL) - the utilization of unlabeled samples in classification prediction tasks.

Semi-supervised learning has become a popular area of research with hundreds of papers proposing a variety of algorithmic methods (see e.g., (Zhu, 2006)). Most of the work in this area is directed towards using the unlabeled data for finding more accurate predictors. In this work we consider a different goal — using unlabeled data for speeding up the runtime of the predictors. In many applications, runtime of the learned classifier is an important factor, and one would prefer a faster classifier even at the expense of slightly poorer predictions.

We formally tackle this problem by studying *proper* SSL, namely, an SSL setting in which we constrain the learner to output a predictor that belongs to some predetermined restricted class of predictors, called the *target class*. If the target class only contains fast computable functions, then the properness of the SSL algorithm guarantees that the algorithm will output a fast predictor. Another scenario to which our formal setup applies arises when a user is interested in the explanatory aspects of the predictor, requiring the output predictor to belong to a family of functions that are readily interpretable. Linear classifiers are an obvious example of such desirable predictors, under both of these scenarios.

Given a target class H of predictors, the goal of an *H-proper learner* is to find some $h \in H$ whose accuracy (with respect to the unknown data-generating distribution) is as close as possible to the most accurate predictor in H. In this paper, we show that unlabeled examples can *provably and empirically* direct the learner towards more accurate classifiers in the class.

Appearing in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

The basic idea of our algorithmic paradigm is a simple two phase procedure. First, we use the labeled sample to learn a classifier that is not necessarily a member of the target class H, but has small prediction error. Note that this classifier might be slow and obscure. In the second stage, we apply the learned classifier to label the unlabeled examples, and feed these now-labeled examples to a fully supervised H-proper learner.

Different algorithms can be derived from our paradigm by specifying how the classifier is being learned in the first phase. One option is to apply the ERM rule using a different class, that we call the *approximation class*. Another option is to apply a non-parametric learning method, such as Nearest Neighbor (NN). In both cases, our algorithm can yield a predictor $h \in H$ with a higher accuracy compared to learning H without unlabeled examples. We formally prove the last claim by deriving upper and lower bounds on the (labeled) sample complexity of the different approaches.

To analyze the NN-based SSL, we propose a novel measure for the relationship between the marginal of a data generating distribution and its conditional label distribution. Our measure, that we name *"Probabilistic Lipschitzness"*, captures the intuition behind the so-called "cluster-assumption" or "smoothness-assumption" that are commonly used to support and motivate SSL paradigms. Our measure allows quantification of the degree to which these assumptions hold for a given learning task.

We demonstrate the potential merits of our approach by applying it to the MNIST digit recognition data set, using linear predictors as the target class. As mentioned before, linear predictors are both fast and interpretable. Our experiments show that an optimal linear predictor can be learned from a modest number of labeled examples, if a large number of unlabeled examples is available.

1.1. Related work

Semi-Supervised learning is a very active research area. Probably the most prolific direction is the introduction of algorithmic approaches and describing their application to real life learning tasks (Chapelle et al., 2006). Most of the work along these lines emphasizes experimental results and is not supported by proven performance guarantees.

On the other end of the spectrum there is purely theoretical research that focuses on abstract models and is not intended to be directly applied to real practical data. One direction of such research are methods that do not impose any prior assumptions about the data-generating distributions, like the work of Kääriäinen (2005), as well as the work of Balcan & Blum (2005). They suggest to use a notion of a compatibility function that assigns a higher score to classifiers which "fit nicely" with respect to the unlabeled distribution. However, the work of Ben-David et al. (2008), provides rigorous results indicating that, under worst-case scenarios, the utility of unlabeled data in that setup is very limited.

We aim to combine rigorous theoretical analysis with practical relevance. Consequently, most related to this paper are previous works that try to bridge the gap between theory and practice. In SSL research, such papers often make strong assumptions about the relationship between the marginal data distribution and the conditional distribution of labels, or about the type of probability distributions that generate the data, and then provide rigorous analysis of the performance of SSL algorithms under these assumptions.

A popular assumption in that context is the so-called cluster assumption, postulating that the data contains clusters that have homogeneous labels. Under such an assumption, SSL works by using the unlabeled observations to identify these clusters, and then considering only classifiers that are constant on each cluster. Closely related to the cluster assumption are the *smoothness assumption* and the related *low density assumption* (Chapelle & Zien, 2005) which suggests that the classification decision boundaries should lie in low density regions. For example, Rigollet (2007) provides a mathematical formalization of (a version of) the cluster assumption. He assumes that the data contains a collection of countably many connected components of high density (w.r.t. some threshold density level) and that for two points in the same cluster, the probability of them having the same label is greater than 0.5. The downside of such paradigms is that, since the set of potential classifiers is trimmed down by the assumptions requirements, if the presumed label-structure relationship fails to hold, the learner may be left with only poorly performing classifiers.

Maybe most relevant to our setup of using unlabeled data to improve the runtime of the learned classifier is work of Liang et al. (2008). They consider learning for NLP tasks, where expressive conditional random field (CRF) predictors have low error but are slow to compute. They propose to use unlabeled data to replace them by fast computable Independent Logistic regression (ILR). Their algorithmic paradigm is similar to ours and, in a sense, our work here can be viewed as generalizing that work. Previous to that, Bucila et al. (2006) showed experimentally how a similar idea can

be applied to replace complex ensembles by compact neural network predictors.

1.2. Outline of the paper

We start, in Section 2, by describing our learning setup, as well as the SSL algorithmic paradigm we propose to employ for it. We then go on to describe two instantiations of that algorithm for two potential scenarios, based on different types of prior knowledge. In Section 3 we provide formal definitions of the learning model, present our formalization of the cluster assumption, and compare the performance of our SSL algorithms with the inherent limitations of fully supervised algorithms. Due to space constraints, most of the proofs are omitted and can be found in a supplementary material (Sup). In Section 4 we describe the experimental results.

2. The algorithmic paradigm

We consider learning algorithms that are confined by a prior requirement concerning the nature of the label predictor they are allowed to come up with. As discussed in the introduction, this could be a requirement for fast computability, or interpretability of that predictor. We model this by specifying a collection of predictors, called the *target class*, consisting of the set of the predictors that meet these prior requirements. Our goal is to find a low error classifier that is a member of that pre-specified class H. In many cases such a restriction renders the learning task harder. We propose to first ignore this restriction and use the labeled sample to solve the unrestricted learning task. Then, use the unlabeled data to transform the resulting learner into a predictor from H. More concretely, our learning algorithms follow a 2-step, rather simple, paradigm:

1. Use the labeled sample to learn a classifier that is not necessarily a member of the target class H, but has small prediction error.
2. Apply that learned classifier to label the points of the unlabeled input sample, and feed that now-labeled sample to a fully supervised H-learner.

Since labeled samples are needed only for the first step of this paradigm, the labeled-sample complexity of our algorithm equals that of the unrestricted learning task. Consequently, whenever the search for an unrestricted, low-error classifier has a relatively low sample complexity, our SSL paradigm allows us to find a low-error predictor in the target class H with less labeled-examples than what is required by any fully supervised algorithm.

We investigate two scenarios where such a saving of labeled examples occurs. First, we analyze our algorithmic paradigm when, on top of the target class H, determined by the task, the leaner is aware of a different (possibly larger) class, the approximation class H', that contains a low error classifier. We then investigate an alternative approach, that does not require the knowledge of such an H' and, instead, uses the nearest neighbor algorithm in the first step of our paradigm.

2.1. SSL of fast classifiers using a large approximation class

One scenario in which we can make use of the unlabeled data in order to learn a classifier from H is when the learner is aware of a richer class of predictors H' that has low approximation error w.r.t the data-generating distribution - much lower than that of the best predictor in the class H.

When H' is a class of finite VC dimension that contains a classifier with close-to-zero error, known learning bounds imply that it can be learned (in the fully supervised setup) with a sample complexity of $O(\frac{\mathrm{VCdim}(H')\ln(1/\epsilon)}{\epsilon})$ (see, e.g., Boucheron et al. (2005)). On the other hand, we prove a lower bound of $\Omega(\frac{1}{\epsilon^2})$ labeled samples, for the task of learning linear separators without unlabeled examples that holds even for distributions that are realizable by some learnable classes H'. It follows that in such scenarios, the labeled-sample complexity of our SSL learning algorithm is strictly lower than that of any fully supervised learner for the target class H of linear classifiers. For details, see Section 3.1.

2.2. SSL using nearest neighbors

We also investigate our paradigm when the first step is carried out using a nearest neighbor algorithm. The obvious advantage of such an algorithm over the SSL algorithm discussed above is that it does not require the prior knowledge of any low-approximation-error class, H'. The nearest neighbor algorithm (NN) takes a labeled sample and, when required to label some domain point, assigns to it the label of its nearest neighbor in the labeled sample (with respect to some underlying metric).

The sample complexity of nearest neighbor prediction algorithms cannot be bounded in a distribution-free manner (independently of the data-generating distribution). To get a handle on the performance of such algorithms, we introduce a new quantitative formulation of the cluster assumption. In contrast with common such assumptions that either hold or fail, our proposed

measure is quantitative. We consider the probability that two points have different labels, as a function of the distance between these points. This function, that we call "probabilistic Lipschitzness", quantifies a relationship between the unlabeled marginal and the conditional label distribution and the metric structure of the underlying data space. It captures the intuition behind assumptions like the "cluster assumption" and the "smoothness assumption" in a formal way that allows quantification of the degree by which these assumptions hold for a given learning task.

We investigate the sample complexity of the NN-based SSL paradigm as a function of the degree by which the learning task conforms with the cluster assumption, as quantified by our probabilistic Lipschitzness measure. In Section 3.2 we show that when the data distribution can be partitioned into homogeneously labeled clusters with some margin separating any pair of differently labeled clusters, then the labeled-sample complexity of our algorithm is $O(1/\epsilon)$ (for any learnable target class H, and, in particular, for the task of prediction with linear half-spaces).

When no such nice partitioning exists, we show that the NN-based SSL algorithm can still be beneficial under milder conditions. In particular, if the underlying data distribution satisfies some probabilistic Lipschitzness condition, then for any learnable target class H (including the task of prediction with linear half-spaces) the labeled-sample complexity of our SSL algorithm is $O\left(\frac{d(\ln(1/\epsilon))^d}{\epsilon}\right)$, where d is the dimension of the Euclidean space in which our data is embedded. For that case as well, there are lower bounds showing that without utilizing unlabeled data, the sample complexity of learning linear separators in such scenarios is $\Omega(\frac{1}{\epsilon^2})$.

3. Formal presentation of our results

In this section we present our results in a more formal setting and provide the main proof ideas. We start by introducing some notation and definitions.

We fix some domain set X and some label set Y, (which, for concreteness, will be the unit cube in $\mathbb{R}^d$ and the binary set $\{0,1\}$, respectively). A *hypothesis* or label predictor, is a function $h : X \to Y$, and a hypothesis class is a set of hypotheses. We assume that the data for a learning problem is generated by some *target distribution* P over $X \times \{0,1\}$. We denote the marginal distribution of P over X by D and let $l : X \to [0,1]$ denote the labeling rule of this distribution, *i.e.* the conditional probability of label 1 at some point: $l(x) = \Pr_{(x',y)\sim P}(y = 1|x' = x)$. For some function $h : X \to \{0,1\}$ we define the *error* of h with respect to P as $\mathrm{Err}_P(h) = \Pr_{(x,y)\sim P}(y \neq h(x))$.

For a class H of hypotheses on X, let the smallest error of a hypothesis $h \in H$ with respect to P be denoted by $\mathrm{opt}_H(P) := \min_{h\in H} \mathrm{Err}_P(h)$. Given a hypothesis class H, an *H-proper SSL-learner* takes a labeled sample, S, and an unlabeled sample, T, and should output a function $h \in H$. We assume that the labeled sample S is sampled *i.i.d.* by P and that the unlabeled sample T is sampled *i.i.d.* by D. For simplicity, we focus on l being a deterministic labeling function, *i.e.* $l : X \to \{0,1\}$, but our analysis can be readily generalized to arbitrary labeling functions.

We use the common definition of agnostic PAC proper learning: An algorithm A is an *agnostic PAC proper learner* for some hypothesis class H over X if for all $\epsilon > 0$ and $\delta > 0$ there exists a sample size $m = m(\epsilon, \delta)$ such that, for all distributions P over $X \times \{0,1\}$, when given an *i.i.d.* sample of size m from P, then with probability at least $1 - \delta$ over the sample, A outputs a classifier $h \in H$ with error at most $\mathrm{opt}_H(P) + \epsilon$. In this case, we also say that the learner (ϵ, δ)-learns H.

Our algorithms make use of the fact that agnostic learners are robust with respect to small changes in the input distribution. We formalize this observation in the following lemma:

Lemma 1. *Let P be a distribution over $X \times \{0,1\}$, let $f : X \to \{0,1\}$ be a function with $\mathrm{Err}_P(f) \leq \epsilon_0$, let A be an agnostic learner for some hypothesis class H over X and let m be the sample size required by A to (ϵ, δ)-learn H. Then, with probability at least $(1-\delta)$ over an i.i.d. sample of size m from P's marginal labeled by f, A outputs a hypothesis h with $\mathrm{Err}_P(h) \leq \mathrm{opt}_H(P) + 2\epsilon_0 + \epsilon$.*

This lemma implies that, in order to prove the success of our algorithms, it suffices to show that the classifier that we learn in the first step of our SSL-algorithm has error smaller than $\epsilon/3$ with confidence at least $1 - \delta/2$. If we then use an agnostic learner for our target-class H in the second step of our paradigm and feed this agnostic learner with a sample of the size it needs to $(\epsilon/3, \delta/2)$-learn H, our paradigm is guaranteed to (ϵ, δ)-learn H.

3.1. SSL using an approximation class

Let H be the target class of our H-proper SSL learner and let H' be an approximation class. We denote by $A_{(H,H')}$ our two stage SSL algorithm, which first learns H' using the labeled examples and then learns H using the predictions of the previous stage on unlabeled examples. In the following we show that if

$\text{opt}_{H'}(P) = 0 < \text{opt}_H(P)$ then our paradigm can provably save labeled samples for properly learning the class H. We first derive an upper bound on the labeled sample complexity of $A_{(H,H')}$.

Theorem 2. *For every pair of hypothesis classes H, H', every $\epsilon, \delta \in (0,1)$ and every target distribution P, if $\text{opt}_{H'}(P) = 0$ then, given access to a labeled sample S of size $\frac{12}{\epsilon}\left(\text{VCdim}(H')\log(36/\epsilon) + \log(4/\delta)\right)$ and an unlabeled sample T of size $\frac{576}{\epsilon^2}\left(2\text{VCdim}(H)\log(36/\epsilon) + \log(4/\delta)\right)$, with probability at least $(1-\delta)$, we have $\text{Err}_P(A_{(H,H')}(S,T)) \leq \text{opt}_H(P) + \epsilon$.*

Proof. To prove the bound, we make use of standard results of VC-theory. Namely, as H' is realizable, Anthony & Bartlett (Theorem 4.8) tells us that a sample size of $\frac{12}{\epsilon}\left(\text{VCdim}(H')\log(36/\epsilon) + \log(4/\delta)\right)$ suffices for $\text{ERM}(H')$ to output a classifier from H' that has error at most $\epsilon/3$ with probability at least $1-\delta/2$. Similarly, for $\text{ERM}(H)$ a labeled sample of size $\frac{576}{\epsilon^2}\left(2\text{VCdim}(H)\log(36/\epsilon) + \log(4/\delta)\right)$ suffices to output a classifier with error at most $\epsilon/3$ with probability at least $1-\delta/2$. Now Lemma 1 implies the claim. □

Note that, even in cases where $\text{VCdim}(H) < \text{VCdim}(H')$, for sufficiently small values of ϵ, this upper bound on the sufficient size of the labeled sample is smaller than the known lower bound of $\Omega\left(\frac{\text{VCdim}(H)+\log(1/\delta)}{\epsilon^2}\right)$ on the required labeled sample size for learning a hypothesis from H in the agnostic setup when no additional unlabeled sample is available. In particular, there are specific distributions in which we can prove a lower bound on the sample complexity of learning from labeled examples only, which is higher than the upper bound on the labeled-sample complexity given in Theorem 2.

Remark 3. *The bound on the size of the labeled sample in Theorem 2 corresponds to the sample complexity of learning the class H'. We apply it to learn a class H that has lower error convergence rate due to having higher approximation error. However, it is easy to see that the same paradigm can be applied when the slow convergence rate of H is due to other reasons, such as having higher VC dimension. In fact, this situation occurs in the scenario discussed by Liang et al. (2008).*

In our lower bound, we let H be the class of half-spaces in $\mathbb{R}^d$ and let H' be the class of all unions of members of H. Further, we let $\mathcal{P}$ be the family of all data probability distributions that are realizable by H'.

Theorem 4. *The sample complexity of (ϵ,δ)-agnostically learning half-spaces in $\mathbb{R}^d$ over the set of distributions $\mathcal{P}$, is bounded below by*

$$\frac{1-(1.5\epsilon)^2}{2(1.5\epsilon)^2}\ln\left(\frac{1}{8\delta(1-2\delta)}\right).$$

Proof sketch. To prove this lower bound, we consider distributions whose support is a discrete set of three labeled points $(x_1,1)$, $(x_2,0)$ and $(x_3,1)$. If these three points are collinear with x_2 being in between x_1 and x_3, then clearly no halfspace can correctly classify these three points, thus it becomes crucial for the learning algorithm to estimate which of x_1 and x_3 has more weight and then to accept misclassification of whichever of the two has less weight. In case the weights of x_1 and x_3 are roughly the same, say the difference is 2ϵ, then we can adapt the lower bound from Lemma 5.1 in (Anthony & Bartlett, 1999) to show that we need at least $\frac{1-(1.5\epsilon)^2}{2(1.5\epsilon)^2}\ln\left(\frac{1}{8\delta(1-2\delta)}\right)$ sample points to decide which point has more weight. □

Note that in the situation we described in the proof, the sample that is needed to estimate the weights can be replaced with an unlabeled sample. The full proof of this results can be found in the supplementary material. Combining this result with Theorem 2, this lower bound implies that for learning halfspaces in $\mathbb{R}^d$ the use of unlabeled data is provably beneficial, as our algorithmic paradigm requires less labeled data as any fully supervised proper learner for that class.

Corollary 5. *Consider the task of proper learning the class H of half-spaces in $\mathbb{R}^d$ w.r.t target distributions that have their labels realizable by some class H' over $\mathbb{R}^d$ that has a finite VC-dimension. For this task, the labeled-sample complexity of our SSL algorithm $A_{(H,H')}$ is strictly below the sample complexity of* any *learning algorithm that outputs linear predictors and does not utilize unlabeled samples.*

3.2. SSL using nearest neighbors

In this section we analyze our paradigm with the usage of a nearest neighbor algorithm in the first stage. In order for the nearest neighbor approach to output a low error-predictor, we need to make assumptions about the relationship between the underlying marginal distribution and the labels. Intuitively, we need to assume, that with high probability close points share the same label. We now present our proposed formalization of this assumption and then proceed with analyzing our paradigm under this assumption.

Our probabilistic Lipschitzness notion: The common λ-Lipschitz condition is stated for real-valued functions, namely, for all x, y, $|f(x)-f(y)| \leq \lambda||x-y||$,

for some constant λ. This condition can be readily applied to probabilistic labeling rules $\ell : X \to [0,1]$. However, if the labeling function is deterministic, this requirement forces any two points with different labels to be at distance of at least $1/\lambda$ from each other. In particular, every binary-valued function satisfying such a condition over a connected domain, must be constant. We propose a relaxation of this strict requirement by the following notion, that we call *probabilistic Lipschitzness*:

Definition 6. For a monotonically increasing $\phi : \mathbb{R}^+ \to [0,1]$, we say that a function $f : X \to \{0,1\}$ is ϕ-Lipschitz w.r.t. a probability distribution P over X, if for all $\lambda > 0$,
$\Pr_{x\sim D}\left[\exists y\middle|\ f(x) \neq f(y) \wedge ||x-y|| \leq \lambda\right] \leq \phi(\lambda)$.

The probabilistic Lipschitzness generalizes the common Lipschitzness definition, since if a function is λ-Lipschitz then it is ϕ-Lipschitz w.r.t. any distribution with the function $\phi(x) = 0$ for $x \leq \lambda$ and 1 for $x > \lambda$. Applied to the labeling rule of a distribution P, our notion of probabilistic Lipschitzness encapsulates the ideas of a "low-density-assumption" or the "cluster assumption" in that it penalizes having high probability density in areas that are label heterogeneous.

Note that if the data lies in separated homogeneous clusters and the distance between clusters of opposing labels is at least $1/\lambda$, then the labeling rule of this distribution is λ-Lipschitz and is therefore also ϕ-Lipschitz for $\phi(x) = \mathbf{1}_{[x\leq\lambda]}$. One can model weaker clusterability requirements by various monotone functions ϕ. In particular, we consider the case that a distribution has a ϕ-Lipschitz labeling rule for $\phi(x) = \exp(-1/x)$.

As an example, consider the following "smoothly clustered" distribution: We let the domain be the unit interval $[0,1]$ and let the labeling rule be 0 for $x \leq 1/2$ and 1 for $x > 1/2$. Now we let the density d of the distribution form clusters by setting $d(x) = c\mathrm{e}^{-1/x}$ for $0 \leq x \leq 1/4$, $d(x) = c\mathrm{e}^{-1/|1/2-x|}$ for $1/4 \leq x \leq 3/4$ and $d(x) = c\mathrm{e}^{-1/|1-x|}$ for $3/4 \leq x \leq 1$ with $c = (2\mathrm{e}^{-1/4})^{-1}$. This distribution is not λ-Lipschitz for any constant λ, since there exist arbitrarily close points with opposing labels. But, we have

$$\Pr_{x\sim D}\left[\exists y\middle|\ f(x) \neq f(y) \wedge ||x-y|| \leq \lambda\right] \leq \int_{1/2-\lambda}^{1/2-\lambda} c\mathrm{e}^{-1/|1/2-x|}dx \leq \mathrm{e}^{-1/\lambda}$$

Our results: Next, we explore our paradigm for proper learning under the probabilistic Lipschitz assumption and show that it is beneficial use a nearest neighbor function w.r.t. the input labeled sample, in the first stage of the SSL algorithm. We denote this SSL-based algorithm by $A_{NN}(S,T)$.

Let us begin by considering the scenario where the data is scattered in small but separated homogeneously-labeled lumps (or clusters). It is easy to see that under this assumption, the nearest neighbor algorithm will have close-to-zero error if the input sample is large enough to hit (almost) each of these clusters. As mentioned above, in such a scenario the labeling function is λ-Lipschitz (where λ is a lower bound on the separation between any two differently labeled clusters).

Theorem 7. *Let X be the unit cube of $\mathbb{R}^d$ and let H be any hypothesis class over X with a sample complexity of $m(\epsilon,\delta)$. Given $\epsilon,\delta \in (0,1)$ and $\lambda > 0$, for every target distribution P over $X \times \{0,1\}$, whose corresponding labeling function is λ-Lipschitz, with probability at least $1-\delta$ over the choice of $4\left(\sqrt{d}/\lambda\right)^d \frac{6}{\epsilon\delta\mathrm{e}}$ i.i.d. labeled samples and $m(\epsilon/3,\delta/2)$ i.i.d. unlabeled samples we have that $\mathrm{Err}_P(A_{NN}(S,T)) \leq \mathrm{opt}_H(P) + \epsilon$.*

Thus, if the data lies in well separated clusters, the required labeled sample size grows linearly with $1/\epsilon$. Next, relaxing this condition, we assume that the probability of two differently labeled points decays smoothly as they get closer.

Theorem 8. *Let X be the unit cube of $\mathbb{R}^d$ and let H be any hypothesis class over X with a sample complexity of $m(\epsilon,\delta)$. Given $\epsilon,\delta \in (0,1)$ and $\lambda > 0$, for every target distribution P over $X \times \{0,1\}$, whose corresponding labeling function is ϕ-Lipschitz, where $\phi(a) = e^{-\frac{1}{a}}$, with probability at least $1-\delta$ over the choice of*

$$\frac{\sqrt{d}^{5d}}{\epsilon\delta}\left(3\ln(3d^{3/2}(\epsilon\delta)^{-1/d}\right)^d$$

i.i.d. labeled samples and $m(\epsilon/3,\delta/2)$ i.i.d. unlabeled samples we have that $\mathrm{Err}_P(A_{NN}(S,T)) \leq \mathrm{opt}_H(P)+\epsilon$.

For proofs we refer the reader to the supplementary material. In contrast to the labeled sample complexity of $A_{NN}(S,T)$ given above, for many concept classes H of finite VC dimension, there exist probability distributions that meet the above probabilistic Lipschitzness requirement and yet, any fully supervised algorithm for properly learning H w.r.t these distributions requires a training data set of size $\Omega(1/\epsilon^2)$.

4. Experiments

As mentioned in the introduction, linear classifiers are a prime example of predictors that are desirable, both

in terms of the speed of evaluating them on an example that needs to be labeled, and due to allowing a clear and intuitive interpretation of the predictor. In this section we demonstrate how unlabeled examples can reduce the label-complexity of learning the class of linear classifiers (when the learned predictors are required to belong to that class). We have conducted experiments with the well-known MNIST digit recognition dataset (Cun et al., 1998), which contains 70,000 images (28×28 pixels each) of the digits $0-9$. While this dataset is designed for multiclass classification (e.g. recognizing the digit in the image), we constructed a binary classification task by assigning the label 0 to digits $0-4$ and the label 1 to digits $5-9$.

4.1. SSL with a complex approximation class

We considered two hypotheses classes over the space of 28×28 images. The first class, denoted H, treats each image as a vector, $x \in \mathbb{R}^{784}$, of gray scale values at each pixel (note that $784 = 28^2$). Then, each $h \in H$ is a linear classifier parameterized by a vector $w \in \mathbb{R}^{784}$ and a scalar $b \in \mathbb{R}$, where the prediction is $h(x) = \text{sign}(\langle w, x \rangle)$. This is our target class of fast linear predictors.

The second hypothesis class, denoted H', is a kernel-based linear predictor with the kernel function being a Chamfer distance between images (Barrow et al., 1977; Felzenszwalb & Huttenlocher, 2004; Gavrila, 2007). In particular, we first constructed a binary image, denoted $B \in \{0,1\}^{28,28}$, from each image (by performing a simple thresholding). We then calculated the distance transform of each image, denoted $D \in \mathbb{R}^{28,28}$, where $D_{i,j}$ contains the Euclidean distance from pixel (i,j) to the closest turned pixel of B. The Chamfer distance between two images represented by B_1, B_1, whose distance transforms are D_1, D_2, is defined to be: $\frac{\langle B_2, D_1 \rangle}{\|B_2\|_1} + \frac{\langle B_1, D_2 \rangle}{\|B_1\|_1}$, where the inner product between matrices A, B is $\sum_{i,j} A_{i,j} B_{i,j}$. Finally, the kernel function, denoted $K(x, x')$, is set to be $e^{-d(x,x')/\sigma}$, where $d(x, x')$ is the Chamfer distance as described previously and σ is a parameter (which we tuned using cross validation). The kernel-based classifier is therefore parameterized by a set of images $x_1, \ldots, x_r$, a vector of coefficients $\alpha_1, \ldots, \alpha_r$, and a scalar $b \in \mathbb{R}$, where $h(x) = \text{sign}\left(\sum_{i=1}^{r} \alpha_i K(x_i, x) + b\right)$.

We compared three algorithms. The first algorithm only relies on a labeled sample, S, and does not use unlabeled examples at all. The algorithm trains a linear classifier from the class H using the regularized least squares method. That is, the algorithm returns a minimizer of $\min_{w,b} \sum_{(x,y) \in S} (\langle w, x \rangle + b - y)^2 + \lambda \|w\|_2^2$, where λ is a regularization parameter we tuned using cross validation. We refer to this algorithm as "Linear". The second algorithm also only relies on a labeled sample, S, without using unlabeled examples. The algorithm learns a kernel-based classifier from H', using the Chamfer kernel described previously, where now we used the kernel-based regularized least squares method. We refer to this algorithm as "Chamfer". Finally, the third method is our semi-supervised learning approach described in Section 3.1 applied on H and H'. That is, we first train a kernel-based classifier from H' using a labeled sample S. Let $h \in H'$ be the output classifier. Next, we predict the labels of an unlabeled sample, T, using h. Last, we train a linear classifier from H using the labeled examples in S plus the examples in T with labels produced by h. We again used regularized least squares for each training task. We refer to this algorithm as "Linear with SSL".

The MNIST dataset is divided into a training set of size 60000 examples and a test set of size 10000. We ran the three algorithms with different sizes of labeled sample, ranging from 500 to 5000. For the "Linear with SSL" method we used the rest of the training set, without the labels, as a set of unlabeled examples, T, of size 55000. The test error of the three algorithms is depicted on the left hand frame of Figure 1. As can be seen from the figure, the class H' performs much better than H. It is also clear that using the unlabeled set greatly improves the performance of the learned linear classifier. To further emphasize this, we ran the "Linear" method with larger sets of labeled examples, ranging from 500 to 60000. It is evident that the performance does not improve significantly when increasing the training size beyond 20000 examples, which may indicate that the algorithm achieves the approximation error of H. Furthermore, running "Linear with SSL" with 3500 labeled examples yields the same performance as running "Linear" with 20000, that is, the unlabeled examples helped us to be very close to the best classifier in H using much less labels.

We mention that in this example, the runtime of evaluating the Chamfer classifier on a new example is unrealistically large, while the runtime of evaluating a linear classifier is negligible. In many applications faster classifier is important even if it leads to poorer predictions, and in that case, our SSL approach leads to the same performance as the vanilla "Linear" approach, while requiring significantly less labels.

4.2. SSL with a Nearest Neighbor algorithm

We have also run the NN-based version of our SSL algorithm on the same MNIST data set (with respect

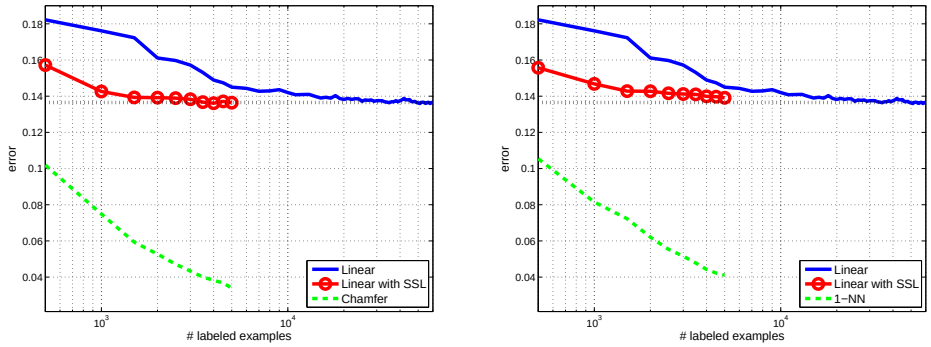

Figure 1. Test error of different algorithms (see description in the text) as a function of the number of labeled examples.

to the basic $L2$ metric over the set of images). The graphs on the right hand frame of Figure 1 show the results of these runs. Once again, they show large label complexity savings of the SSL algorithm compared to fully supervised linear classifiers learning. It should be noticed that in this case, the 1-NN predictor that is used for the first stage of the SSL algorithm has somewhat higher error rate than that of the Chamfer based algorithm that is used at the first stage of the algorithm displayed on the left hand part of the figure. It is therefore not surprising that the resulting "Linear with SSL" algorithm also has a somewhat higher prediction errors in that version of the algorithm.

Acknowledgements We would like to thank Fernando Pereira for the reference to (Liang et al., 2008). Shai Shalev-Shwartz acknowledges the support of the Israeli Science Foundation grant number 598-10.

References

Supplementay material. http://www.cs.uwaterloo.ca/~shai/publications/SupplementICML2011.pdf.

Anthony, Martin and Bartlett, Peter L. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999.

Balcan, Maria-Florina and Blum, Avrim. A pac-style model for learning from labeled and unlabeled data. In *Proceedings of the Conference on Learning Theory (COLT)*, pp. 111–126, 2005.

Barrow, H., Tenenbaum, J., Bolles, R., and Wolf, H. Parametric correspondence and chamfer matching: Two new techniques for image matching. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 659–663, 1977.

Ben-David, Shai, Lu, Tyler, and Pál, Dávid. Does unlabeled data provably help? worst-case analysis of the sample complexity of semi-supervised learning. In *Proceedings of the Conference on Learning Theory (COLT)*, pp. 33–44, 2008.

Boucheron, Stéphane, Bousquet, Olivier, and Lugosi, Gabor. Theory of classification: a survey of recent advances. In *ESAIM: Probability and Statistics*, volume 9, pp. 323–375, 2005.

Bucila, Cristian, Caruana, Rich, and Niculescu-Mizil, Alexandru. Model compression. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 535–541, 2006.

Chapelle, O., Schölkopf, B., and Zien, A. (eds.). *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006.

Chapelle, Olivier and Zien, Alexander. Semi-supervised classification by low density separation. In *Proceedings of the Tenth International Workshop on Articial Intelligence and Statistics*, pp. 57–64, 2005.

Cun, Y. L. Le, Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of IEEE*, 86(11):2278–2324, November 1998.

Felzenszwalb, P.F. and Huttenlocher, D.P. Distance transforms of sampled functions. *Cornell Computing and Information Science Technical Report TR2004-1963*, 2004.

Gavrila, D.M. A bayesian, exemplar-based approach to hierarchical shape matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1408–1421, 2007. ISSN 0162-8828.

Kääriäinen, Matti. Generalization error bounds using unlabeled data. In *Proceedings of the Conference on Learning Theory (COLT)*, pp. 127–142, 2005.

Liang, Percy, III, Hal Daumé, and Klein, Dan. Structure compilation: trading structure for features. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 592–599, 2008.

Rigollet, Philippe. Generalized error bounds in semi-supervised classification under the cluster assumption. *Journal of Machine Learning Research (JMLR)*, 8:1369–1392 (electronic), 2007.

Zhu, Xiaojin. Semi-supervised learning literature survey, 2006.