Precision-Recall versus Accuracy and the Role of Large Data Sets

Brendan Juba and **Hai S. Le**
Washington University in St. Louis
1 Brookings Dr., St. Louis, MO 63130
bjuba@wustl.edu, hsle@wustl.edu

Abstract

Practitioners of data mining and machine learning have long observed that the imbalance of classes in a data set negatively impacts the quality of classifiers trained on that data. Numerous techniques for coping with such imbalances have been proposed, but nearly all lack any theoretical grounding. By contrast, the standard theoretical analysis of machine learning admits no dependence on the imbalance of classes at all. The basic theorems of statistical learning establish the number of examples needed to estimate the accuracy of a classifier as a function of its complexity (VC-dimension) and the confidence desired; the class imbalance does not enter these formulas anywhere. In this work, we consider the measures of classifier performance in terms of precision and recall, a measure that is widely suggested as more appropriate to the classification of imbalanced data. We observe that whenever the precision is moderately large, the worse of the precision and recall is within a small constant factor of the accuracy weighted by the class imbalance. A corollary of this observation is that a larger number of examples is necessary and sufficient to address class imbalance, a finding we also illustrate empirically.

1 Introduction

One of the primary concerns of statistical learning theory is to quantify the amount of data needed to reliably and accurately learn an unknown function that belongs to a given family, such as halfspaces, decision trees, neural networks, and so on. This quantity is often referred to as the *sample complexity* of learning the family of functions. The foundational results of statistical learning theory, first established by Vapnik and Chervonenkis (1971), assert that accurate learning of functions is possible using an amount of training data that depends only on a parameter of the family of functions known as the *VC-dimension*. In particular, for discrete functions, this VC-dimension parameter is no larger than the number of bits needed to specify a member of the family in a given domain; often, it is proportional to this "description size" or the "number of parameters." Moreover, it is known that (up to the leading constant) the same number of examples is also *necessary* to reliably identify a function that fits the data accurately (Blumer et al. 1989; Ehrenfeucht et al. 1989; Anthony and Bartlett 1999; Hanneke 2016). Thus, the central message of statistical learning theory is that this notion of model size controls the amount of data inherently needed for learning. In particular, the same quantity suffices regardless of the distribution of the actual data, and thus regardless of the degree of class imbalance ("rarity" of positive examples) and so on.

Copyright © 2019, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

In spite of the seemingly complete theoretical picture that these theorems paint, practitioners in data mining in particular have been long concerned with the effects of class imbalance or rarity of classes in training data (Weiss 2004; He and Garcia 2009). These concerns led to much work on attempts to reduce the class imbalance by either dropping examples or synthesizing new ones. A crucial clue to the nature of the problem was provided by a substantial empirical study by Weiss and Provost (2003): In a study of a fixed decision tree learning algorithm on twenty-six data sets, they determined that indeed, the use of the natural class distribution provided close to the optimal accuracy (error rate) across almost all of the data sets, as standard learning theory suggests. But, on the other hand, they also found that for *another metric* – the area under the ROC curve (AUC) – the quality of the learned decision tree was significantly affected by the class imbalance of the data. Thus, their findings suggested that the sample complexity of some metrics of classifier quality, such as AUC, might depend on class imbalance, while others such as accuracy do not. And indeed, Agarwal and Roth (2005) established that the sample complexity of learning under the AUC metric does depend on the class imbalance. More recently, Raeder et al. (2012) evaluated a variety of metrics and found empirically that every metric they considered *except* for accuracy had a dependence on the class imbalance. While the empirical studies leave open the question of whether these observations are artifacts of the particular algorithms used, we establish here that such a dependence is also inherent for the *precision* and *recall* of the learned classifier.

Our main technical contribution is actually a simple observation: We note that a simultaneous bound on the precision and recall when precision is greater than 50% is *equivalent* to a bound on the accuracy that is scaled by the base positive rate. As a corollary, we obtain tight bounds on the sample complexity of precision-recall bounds. In particular, we find that the base positive rate (i.e., the class imbalance)

enters these expressions as an additional penalty over the usual expressions for accuracy.

A precise relationship between precision, recall, and accuracy (that holds with equality) was discovered previously by Alvarez (2002), and our bounds could also be derived easily from Alvarez's equation. Interestingly, however, Alvarez did not interpret the consequences of his relation for the amount of data needed, which is our main contribution. Davis and Goadrich (2006) similarly showed that classifiers that dominate in the usual ROC space also dominate in precision-recall and vice-versa. Actually, results by Cortes and Mohri (2004) imply that precision-recall bounds are fundamentally different from AUC: for a fixed class imbalance and accuracy, various ranking functions may feature substantially different AUC scores, in contrast to precision-recall bounds. Hence, Davis and Goadrich's result is incomparable to ours.

Theoretical upper bounds on the sample complexity that is sufficient to achieve a given bound on the precision and recall were established previously by Liu et al. (2002) as part of a more general bound for partially supervised classification and by Valiant (2006); similar upper bounds were established for the related "Neyman-Pearson classification task," which is essentially an agnostic variant of learning with precision-recall bounds, by Cannon et al. (2002). Our contribution is that we show that the first set of bounds (for the realizable setting) cannot be substantially asymptotically improved and such a large amount of data is inherently necessary.[1] To our knowledge, the only lower bound for any such task was obtained by Scott and Nowak (2005), who note that the false-negative and false-positive rates converge no faster than the overall error rate. Our quantitative conclusions are also similar to those obtained under the heursitic calculation used by Raeder et al. (2012) which simply calculated the number of examples necessary to get, e.g., a training set containing 100 positive examples. This turns out to be a reasonable rule of thumb for, e.g., simple classifiers on data with 10–100 attributes, but we stress that it is not simply the number of positive examples that are relevant, as the vast number of negative examples is necessary to rule out the various classifiers with an unacceptably high false alarm rate. In any case, we provide formal justification, showing that this number of examples is *inherently necessary* for success at the task in general, whereas the "100 positive examples" heuristic was largely grounded in intuition until now.

At a high level, our main observation is that achieving high precision and recall demands that our training data scales with the class imbalance; in particular, the methods for attempting to "correct" the imbalance by sub-sampling, or over-sampling, or generating new synthetic data discussed above (all of which are topics of intense interest in certain areas) fundamentally should not provide any improvement in precision-recall in general. By contrast, if we have sufficient training data (to achieve high accuracy), then any standard learning method will do, without need for these special "imbalance-correcting" methods. We illustrate this finding empirically: We observe that for several standard off-the-shelf methods and several data sets featuring significant class imbalance, these imbalance-correcting methods do not provide reliable improvements in precision and recall, but adding more data always does significantly improve precision and recall. We found that the only classifiers for which the imbalance correction methods frequently improve precision and recall are decision tree methods (here, Random Forest), where the techniques were originally developed.

But, we argue that in many applications, high precision is of the utmost importance, and in particular, in applications that suffer a severe class imbalance. We discuss two example domains here, one of hospital early-warning systems, and another in machine translation. The upshot is, unfortunately, these applications simply inherently require enormous data sets. This also partially explains a finding from the work on machine translation (Brants et al. 2007) that, from the usual standpoint of accuracy, is a bit puzzling: namely, as the size of the training data increased into the hundreds of billions of examples, Brants et al. found continued improvement in the quality of translations produced by a relatively simple method. The usual accuracy bounds suggest that the amount of training data necessary should only scale proportionally to the size of the model trained, and it is not clear why the rule for translation should need to be so large. But, we observe that to even solve the problem of identifying which words to use in machine translation, we require high precision (since most words are infrequently used), and hence our bounds establish that such an enormous number of examples is precisely what we should expect to require.

The rest of our paper is structured as follows: in Section 2, we discuss the importance of high-precision classifiers; in Section 3, we quantify the relationship between precision-recall and accuracy, derive sample complexity bounds for precision-recall, and discuss the need to use a large training data set; in Section 4, we present our experiments illustrating how using large training data improves the precision-recall on an imbalanced data set, while methods designed to simply correct the class imbalance (e.g., from different data sets) generally do not achieve satisfactory precision and/or recall; in Section 5, we give our conclusions plus some advice for practitioners and finally, in Section 6, we sketch some possible directions for future work.

2 The Importance of Precision

Accuracy is the simplest and most widely used metric to measure the performance of a classifier. However, accuracy is not always a good metric, especially when the data is imbalanced. The basic problem is that when the negative class is dominant, we can achieve high accuracy merely so long as we predict negative most of the time. As a consequence, even when a highly accurate classifier produces a positive prediction, it may still be that the negative class is (much) more likely. Such situations are characterized by precision of 50% or less. In many applications this is highly undesirable.

[1] We observe that a small improvement over the bound obtainable from Liu et al. follows immediately from Hanneke's (2016) improved bound for accuracy, and we likewise strengthen Valiant's upper bound from representation length to VC-dimension.

A first example concerns learning for alarm systems. Following a review by Amaral (2014), we find that early-warning systems for use in hospitals feature significant class imbalance and crucially rely on nontrivial precision, or else the alarms are ignored. We expect that this is broadly true of alarm systems in general.

A second example where the data is highly imbalanced (in a sense) and high precision is important can be found in machine translation. Here, we suppose that our training data consists of parallel texts in two languages, and we would like to train a system that, when given text in a source language, produces an equivalent text in the target language. Brants et al. (2007) designed a fairly successful system that, at its heart, simply counts the co-occurrence of n-grams between the two texts and then performs some smoothing. The striking finding of Brants et al. was that a relatively simplistic scheme could achieve state-of-the-art performance merely so long as the amount of training data was enormous; in particular, they found constant improvement when training their model on up to hundreds of billions of examples. It would not be so clear why so much data would be required if accuracy were the only issue, as there is no clear reason why many gigabytes of data should be necessary to specify the rules for translation.

The answer, of course, is that accuracy is not the key measure in this case. Let us simply focus on the question of whether or not it is appropriate to use a given word in a sentence of the translated document, perhaps given a sentence in the source language together with its context. In this case we have a simple binary classification task per word; but, we observe that *most sentences contain infrequently used words.* Intuitively, words that are informative and specific should be used in specific circumstances. We confirmed this intuition in practice by investigating the New York Times Annotated Corpus (Sandhaus 2008), a standard corpus containing more than 1.8 million articles. We first counted the frequency (with respect to sentences) of words in the corpus, that is, what fraction of sentences used that word. We take this fraction as an estimate of the imbalance of correct classifications for our function that predicts whether or not words should be used in the translation. Then, in order to translate a sentence, we must be able to correctly predict the words that should be used in the translated sentence. In particular, it will follow from our bounds in Section 3.2 that it is the frequency of the *rarest* word in the target sentence that determines the amount of data we require to reliably translate the sentence. As anticipated, for most sentences, this is quite high.

Concretely, we examine the fraction of sentences in which the rarest word appears in at least a μ-fraction of the sentences in the corpus; equivalently, as we will see in Section 3.2, this is the fraction of sentences for which an overhead of $1/\mu$ in the sample complexity (over what would suffice for high-accuracy learning) suffices to achieve high precision and recall. Since we were interested in the implications for translation, we used the Natural Language Toolkit (Bird, Klein, and Loper 2009) to filter out proper nouns for which (1) usage is likely abnormally infrequent and (2) various methods of directly rendering the word in a target language by sound are frequently used for translation.[2] The results appear in Figure 1: as a simple point of reference, observe that fewer than five in ten sentences have an *additional overhead* of less than 1.2 million. This still gives no clear indication of how much data would be necessary to reach 95% of sentences, for example, except that it is surely well beyond 1.2 million examples, and likely beyond 12 billion (if we suppose $\approx 10^4$ examples is necessary for learning such a model to high accuracy).

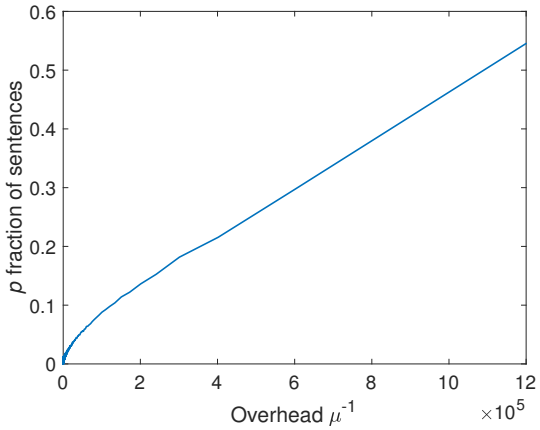

Figure 1: Proportion of sentences using only words occurring in a μ-fraction of sentences (and thus, learnable with overhead $1/\mu$) in the New York Times corpus (Sandhaus 2008). Fewer than 55% of sentences use only words that are sufficiently common to be learned with an overhead of any less than a factor of 1.2 million (see Section 3.2).

Now, we observe that unfortunately, at least moderately high precision and recall is important for this task. Indeed, given the high degree of imbalance, it is possible to achieve very high accuracy by simply predicting that these rare words are never used, or perhaps only used once in 50 million sentences—in other words, by sacrificing recall. This is of course undesirable, since then our machine translation system may not learn to use such informative words at all. It is also possible for the predictor to achieve high accuracy but low precision if it produces even a small number of false positives, that is nevertheless of the same order as the frequency with which the word should actually be used. In this case, the use of the word in the translation is uninformative. That is, it may not provide any indication that the word should actually have appeared in the target text. This is also clearly undesirable, since it means that the use of a word is not connected to its context, i.e., it is "meaningless" (from an information-theoretic standpoint). As we will see next, as a consequence of our sample complexity lower bounds, if the training set does not contain billions of examples, then it is likely that either the recall is very low and hence the word is almost never used, or else the prediction that a word should

[2] Although this may not have filtered out 100% of proper nouns, we surely cannot fall back on a simpler method of translation if we cannot detect that such a proper noun is being used.

be used in the translated document may have low precision, and hence our word choice may be meaningless.

3 Precision-Recall Is Equivalent to Imbalance-Weighted Accuracy

We now derive our main techincal bounds relating precision and recall to accuracy, and relate these to the sample complexity of classification tasks. We stress that while these are technically very simple, they have serious consequences for settings such as those discussed in Section 2.

3.1 Relating Accuracy, Imbalance, Precision, and Recall

Suppose that D is a distribution over examples with Boolean labels $(x, b) \in X \times \{0, 1\}$ with a *base positive rate* of $\mu = \Pr_D[b = 1]$. For a hypothesis $h : X \to \{0, 1\}$, the *accuracy* of h is $\Pr[h(x) = b] \stackrel{\text{def}}{=} 1 - \epsilon_{\text{acc}}$; the *precision* of h is $\Pr_D[b = 1|h(x) = 1] \stackrel{\text{def}}{=} 1 - \epsilon_{\text{prec}}$, and the *recall* of h is $\Pr_D[h(x) = 1|b = 1] \stackrel{\text{def}}{=} 1 - \epsilon_{\text{rec}}$. One often uses the expressions

$$tpr \stackrel{\text{def}}{=} \Pr_D[b = 1 \wedge h(x) = 1] \quad fpr \stackrel{\text{def}}{=} \Pr_D[b = 0 \wedge h(x) = 1]$$

$$tnr \stackrel{\text{def}}{=} \Pr_D[b = 0 \wedge h(x) = 0] \quad fnr \stackrel{\text{def}}{=} \Pr_D[b = 1 \wedge h(x) = 0]$$

and it may be verified that our definitions above are equivalent to the perhaps more common expressions

$$1 - \epsilon_{\text{prec}} = \frac{tpr}{tpr + fpr} \qquad 1 - \epsilon_{\text{rec}} = \frac{tpr}{tpr + fnr}$$

$$1 - \epsilon_{\text{acc}} = tpr + tnr \qquad \mu = tpr + fnr$$

Motivated by settings such as those discussed in Section 2, we will focus on the problem of learning classifiers with precision at least 50%. It turns out that in this case, precision and recall are closely related to the (accuracy) error, scaled by the class imbalance:

Theorem 1 *Suppose $h : X \to \{0, 1\}$ has precision greater than $1/2$ in D. Then $\epsilon_{\max} \stackrel{\text{def}}{=} \max\{\epsilon_{\text{prec}}, \epsilon_{\text{rec}}\}$ satisfies $\mu\epsilon_{\max} \leq \epsilon_{\text{acc}} \leq \mu(\epsilon_{\text{rec}} + \frac{1}{1-\epsilon_{\text{prec}}}\epsilon_{\text{prec}})$. In particular, $\epsilon_{\text{acc}} \leq 3\mu\epsilon_{\max}$.*

Proof: Note that $\epsilon_{\text{rec}} = \frac{fnr}{tpr+fnr} = \frac{fnr}{\mu}$, and since $\epsilon_{\text{prec}} = \frac{fpr}{tpr+fpr}$, $(1 - \epsilon_{\text{prec}})fpr = \epsilon_{\text{prec}}tpr$ and thus $fpr \leq \mu\frac{1}{1-\epsilon_{\text{prec}}}\epsilon_{\text{prec}}$ since $tpr \leq \mu$. So, together,

$$\epsilon_{\text{acc}} = fpr + fnr \leq \mu(\epsilon_{\text{rec}} + \frac{1}{1 - \epsilon_{\text{prec}}}\epsilon_{\text{prec}}).$$

For the lower inequality, we already immediately have $\epsilon_{\text{rec}} = \frac{fnr}{\mu} \leq \frac{\epsilon_{\text{acc}}}{\mu}$. Furthermore, since $1 - \epsilon_{\text{prec}} > 1/2$, $tpr + fpr < 2tpr$, and hence $tpr > fpr$. We also find, since $\frac{\epsilon_{\text{acc}}}{\mu} = \frac{fpr+fnr}{tpr+fnr}$ and for $c \geq 0$ and $tpr > fpr \geq 0$ generally $\frac{fpr+c}{tpr+c} \geq \frac{fpr}{tpr}$, $\frac{\epsilon_{\text{acc}}}{\mu} \geq \frac{fpr}{tpr} \geq \frac{fpr}{tpr+fpr} = \epsilon_{\text{prec}}$. And so therefore, together, we obtain $\epsilon_{\max} \leq \frac{\epsilon_{\text{acc}}}{\mu}$. ■

3.2 Sample Complexity Bounds For Precision-Recall

We now obtain the asymptotic sample complexity of simultaneously achieving high precision and high recall as a corollary of Theorem 1 and the known bounds for accuracy. Again, the *VC-dimension*, introduced by Vapnik and Chervonenkis (1971), is the parameter controlling these bounds:

Definition 2 (VC-dimension) *We say that a set of points in a domain X is* shattered *by a set of Boolean classifiers $\mathcal{C}$ on the domain X if every possible labeling of the set is obtained by some classifier in $\mathcal{C}$. We then say that $\mathcal{C}$ has* VC-dimension *$d \in \mathbb{N}$ if some set of d points in X is shattered by $\mathcal{C}$, and no set of $d + 1$ points is shattered. If for every $d \in \mathbb{N}$ every set of d points is shattered by $\mathcal{C}$, then we say that $\mathcal{C}$ has infinite VC-dimension ("$d = \infty$").*

The asymptotic sample complexity of obtaining a given accuracy is now known exactly:

Theorem 3 (Hanneke 2016) *For a family of classifiers of VC-dimension d, with probability $1 - \delta$, every classifier that is consistent with $O\left(\frac{1}{\epsilon}(d + \log\frac{1}{\delta})\right)$ examples drawn from a distribution D on labeled examples, has error rate at most ϵ (accuracy $1 - \epsilon$) on D.*

A matching lower bound was proved in two parts; first, a $\Omega(\frac{1}{\epsilon}\log\frac{1}{\delta})$ lower bound was obtained by Blumer et al. (1989), and subsequently an $\Omega(d/\epsilon)$ bound was proved by Ehrenfeucht et al. (1989). Together, these yield a matching lower bound:

Theorem 4 (Blumer et al. and Ehrenfeucht et al.) *For a family of classifiers of VC-dimension d, there is a constant δ_0 such that for $\delta \leq \delta_0$, unless $\Omega\left(\frac{1}{\epsilon}(d + \log\frac{1}{\delta})\right)$ examples are drawn from a distribution D on labeled examples, with probability at least δ some classifier in the family with accuracy less than $1 - \epsilon$ on D is consistent with all of the examples.*

Since any classifier with precision greater than $1/2$ and accuracy $1 - \epsilon_{\text{acc}}$ must have precision and recall at least $1 - \frac{\epsilon_{\text{acc}}}{\mu}$, we find that to obtain precision and recall $1 - \epsilon_{\max}$, it suffices to simply learn to accuracy $1 - \mu\epsilon_{\text{acc}}$, i.e., $O\left(\frac{1}{\mu\epsilon_{\max}}(d + \log\frac{1}{\delta})\right)$ examples suffice.[3]

Corollary 5 (Upper bound for precision-recall) *For any family of Boolean classifiers $\mathcal{C}$ of VC-dimension d on a domain X and any distribution D on X, suppose we are given examples drawn from D labeled by $c \in \mathcal{C}$ such that $\Pr[c(x) = 1] = \mu$. Then with probability $1 - \delta$ any classifier $c' \in \mathcal{C}$ that is consistent with $O\left(\frac{1}{\mu\epsilon_{\max}}(d + \log\frac{1}{\delta})\right)$ examples achieves precision and recall $1 - \epsilon_{\max}$.*

And conversely, since (unless the precision is less than $1/2$) either the precision or recall is at most $1 - \frac{\epsilon_{\text{acc}}}{3\mu}$, we find that unless $\Omega\left(\frac{1}{\mu\epsilon_{\max}}(d + \log\frac{1}{\delta})\right)$ examples are drawn from D, if $\delta < \delta_0$, then with probability at least δ, some

[3] Note that although Theorem 1 would not apply to $\epsilon_{\max} > 1/2$, the big-O can hide a factor of 2, so that we still get precision $> 1/2$.

hypothesis that is consistent with all of the examples has accuracy less than $1 - 3\mu\epsilon_{\max}$; hence some such hypothesis has either precision or recall that is less than $1 - \epsilon_{\max}$:

Corollary 6 (Lower bound for precision-recall) *For a family of classifiers of VC-dimension d, there is a constant δ_0 such that for $\delta \leq \delta_0$ and $\epsilon_{\max} < 1/2$, unless $\Omega\left(\frac{1}{\mu\epsilon_{\max}}(d + \log\frac{1}{\delta})\right)$ examples are drawn from a distribution D on labeled examples, with probability at least δ some classifier in the family with precision or recall less than $1 - \epsilon_{\max}$ on D is consistent with all of the examples.*

Recap As we noted in the introduction, similar upper bounds have been proved for a variety of cases by many authors, and here we can exploit the recent work by Hanneke to unify and/or improve all such bounds for the "realizable" case. A perhaps more important novelty here is the (matching) lower bound, which had not been studied previously. As we will discuss shortly, the presence of the $1/\mu$ penalty in these bounds is quite significant: it means that

1. in contrast to learning under the accuracy objective, when we seek precision greater than 50%, class imbalance does impose a cost on learning, and
2. the problem of coping with imbalanced data is entirely one of learning to unusually high accuracy, and hence one of collecting a large enough data set since training set size is what controls generalization error.[4]

We note that these bounds also imply that the algorithms for learning to abduce k-DNFs of Juba (2016) have optimal sample complexity, when we consider the maximum "plausibility" (equal to the true positive rate) for distributions actually labeled by a k-DNF. The same sample complexity lower bound also applies to the related problem of conditional linear regression under the sup norm (Juba 2017).

We can similarly obtain bounds for the sample complexity of achieving a given bound $\epsilon_{\max}$ on the precision and recall in *agnostic* (a.k.a. *non-realizable*) learning, using the known bounds in that setting. Specifically, we obtain the following bounds for approximate agnostic learning

Corollary 7 (c.f. Anthony and Bartlett 1999) *There is a constant $\delta_0 > 0$ such that the following holds. For any family of Boolean classifiers $\mathcal{C}$ of VC-dimension d on a domain X and any distribution D over $X \times \{0,1\}$, suppose we are given examples drawn from D such that $\Pr[b = 1] = \mu$ and $\min_{c^* \in \mathcal{C}} \Pr[c^*(x) \neq b] = \epsilon^*_{\text{acc}}$. Put $\epsilon^*_{\text{prec}} = \Pr[b \neq 1 | c^*(x) = 1]$, $\epsilon^*_{\text{rec}} = \Pr[c^*(x) \neq 1 | b = 1]$, and $\epsilon^*_{\max} = \max\{\epsilon^*_{\text{prec}}, \epsilon^*_{\text{rec}}\}$. Then for $\delta < \delta_0$, $\alpha > 1$, and $3\alpha\epsilon^*_{\max} < 1/2$, $\Theta\left(\frac{1}{\mu\epsilon^*_{\max}(\alpha-1)^2}(d + \log\frac{1}{\delta})\right)$ examples are necessary to return a classifier that with probability $1 - \delta$ satisfies $\epsilon_{\max} \leq (\alpha/3)\epsilon^*_{\max}$ and sufficient to return a classifier that with probability $1 - \delta$ satisfies $\epsilon_{\max} \leq 3\alpha\epsilon^*_{\max}$.*

Proof: We first observe that indeed, $\epsilon^*_{\text{acc}} \leq 3\mu\epsilon^*_{\max}$ and $\mu\epsilon^*_{\max} \leq \epsilon^*_{\text{acc}}$ since the corresponding $c^* \in \mathcal{C}$ attaining the minimum on the RHS can be used to bound the LHS in each case. Therefore, since $O\left(\frac{1}{\epsilon^*_{\text{acc}}(\alpha-1)^2}(d + \log\frac{1}{\delta})\right)$ examples suffice to learn a classifier with $\epsilon_{\text{acc}} \leq \alpha\epsilon^*_{\text{acc}}$, we find that since $\epsilon^*_{\text{acc}} \leq 3\mu\epsilon^*_{\max}$ and $\mu\epsilon_{\max} \leq \epsilon_{\text{acc}}$, that for such a classifier $\epsilon_{\max} \leq 3\alpha\epsilon^*_{\max}$.

Similarly, if we do not use $\Omega\left(\frac{1}{\epsilon^*_{\text{acc}}(\alpha-1)^2}(d + \log\frac{1}{\delta})\right)$ examples there is a distribution D such that with probability at least δ we return a classifier with $\epsilon_{\text{acc}} > \alpha\epsilon^*_{\text{acc}}$. For this distribution, since $\epsilon^*_{\text{acc}} \geq \mu\epsilon^*_{\max}$, with probability at least δ we are returning a classifier with $\epsilon_{\text{acc}} > \mu\alpha\epsilon^*_{\max}$. Now, for this classifier, $\epsilon_{\max} \geq \frac{\epsilon_{\text{acc}}}{3\mu}$, so it has $\epsilon_{\max} > \alpha\epsilon^*_{\max}/3$. ■

Since the known time-efficient algorithms for agnostic learning (Awasthi, Blum, and Sheffet 2010, e.g.) obtain an approximation factor α that grows polynomially with the dimension (so in particular $\alpha = \omega(1)$), these bounds are adequate to understand how many examples are asymptotically necessary for such guarantees; indeed, Daniely (2016) shows evidence that polynomial-time algorithms for agnostic learning of halfspaces (or any stronger class) must suffer an approximation factor of $\alpha = \omega(1)$, so it seems likely that these bounds address all such algorithms. In particular, we observe that they again pay the same $1/\mu$ penalty as in realizable learning. Nevertheless, it would still be nice to understand the sample complexity of precision and recall for agnostic learning more precisely. In particular, we do not obtain any bounds for the additive-error agnostic learning task. We leave this for future work.

4 Experiment

To verify our theoretical conclusion that using more training data is necessary to cope with class-imbalance, we performed an experiment on a severely imbalanced data set, comparing the performance (i.e. precision and recall) between various techniques that are used for dealing with imbalanced data and training on a larger data set without using these techniques. Such techniques include: Random undersampling, which randomly removes samples from the majority class; Random oversampling, which randomly expands the minority class (He and Garcia 2009); Easy Ensemble (Liu, Wu, and Zhou 2006), which uses random subsets of the majority class to improve balance and form multiple classifiers; NearMiss (Zhang and Mani 2003), which applies K-Nearest Neighbor (K-NN) to choose a number of majority examples around each minority example. In addition, we also applied the Synthetic minority oversampling technique (SMOTE) (Chawla et al. 2002), which makes up data based on the feature space similarities among minority examples, with the integration with condensed nearest neighbor (CNN), edited nearest neighbor (ENN), Tomek links (Batista, Prati, and Monard 2004), and support vector machines (SVM) (Akbani, Kwek, and Japkowicz 2004).

Data Sets In this experiment, we ran all methods on three data sets from the UCI repository, including all of the data sets that were used by Liu et al. (2006). We excluded data sets that have less than 4000 examples or imbalance ratio (the ratio of the size of the majority class to that of the mi-

[4] In particular, this conclusion follows when the generalization error is the primary component of test error. This is true by definition in the "realizable" setting where zero training error is always achievable, and is often true in practice more generally.

Table 1: Data set characteristics

Dataset	Size	Attributes	Target	Maj/Min
abalone	4,177	8	Ring = 7	9.7:1
letter	20,000	16	A	24.3:1
satimage	6,435	36	class 4	9.3:1
drug discovery	1,062,230	152	class 1	16.1:1

nority class) less than 5. In addition, we also added a large data set (about 1 million examples) that was used for a virtual screening task in drug discovery (Garnett et al. 2015). Key characteristics of these data sets are summarized in Table 1. For the scope of this experiment, for all data sets, we merged all majority classes into a single majority class.

Experiment Setup For this experiment, we used the Python scikit-learn library (Pedregosa et al. 2011) with the imbalanced-learn package (Lemaître, Nogueira, and Aridas 2016) that contains a number of re-sampling techniques that are commonly used for classifying imbalanced data. In the first experiment, all data sets were stratified into ten folds. We selected data in one fold as *small* training set, eight folds as *large* training set, and the last fold as *test set*. For the *small* training set, we created different learning models by re-sampling data in that fold using various techniques mentioned earlier (along with an unmodified version) before training them using one of several standard classifiers. Meanwhile, for the *large* training set, we simply trained the model using the same classifier. All models are then tested on the remaining fold. For each model, we recorded the precision and recall values. We used K-Nearest Neighbor, Logistic Regression, SVM, and Random Forest as our choices of standard classifiers. For the preprocessing algorithms, we used the following settings of our hyperparameters: we set the *"sampling_strategy"* to *"float"*, and left other hyperparameters default for the imbalanced-learn package implementation. For K-Nearest Neighbor, we performed validation to determine k. k was set to 5 for most of the data sets, except for *letter* ($k = 3$). For the other classifiers, we set *"class_weight"* to *"balanced"* and left all other hyperparameters as default.

We note that K-Nearest Neighbor is *not* a classifier of bounded VC-dimension, so the analysis of the previous section does not actually apply. Analyses of K-Nearest Neighbor (Gyorfi, Lugosi, and Devroye 1996, e.g.) depend on smoothness properties of the data that would seem to be favorable for the intuitions behind techniques like SMOTE. Nevertheless, we will see that even in cases where K-Nearest Neighbor is rather successful, these techniques for directly correcting class imbalance not only do not strictly help, they may actually harm precision and recall.

In addition, to verify the the relationship between the number of examples and the performance in imbalanced and balanced settings, we ran a fine-grained experiment with increasing sizes of the data fold. We chose two large data sets (*letter* and *drug discovery*); selected the methods that seemed to be working on them from the first experiment; then repeated the first experiment with K-Nearest Neighbor, varying the size of the *small* training set. For both experiments, to get stable results, we repeated the experiment 100 times and recorded the mean and standard deviation values.

Results and Discussion For the first experiment, results are shown in Table 2 (K-NN), Table 3 (Logistic Regression), Table 4 (SVM), and Table 5 (Random Forest), which contains the precision and recall results of training on the results of sampling techniques on the small training set as well as the results of training on two uncorrected data sets, one small and one large. For almost all of the data sets, the results suggest that we can improve the precision and recall simply by increasing the size of the training set, and this effect is not matched by applying the methods for class imbalance. Indeed, the only exception(s) to this were SMOTE+SVM and SMOTE+ENN with Random Forest on the *drug discovery* data set; we will discuss this in more detail later. Meanwhile, compared to simply performing no correction, the methods aimed at correcting the imbalance frequently improve recall at the cost of precision or vice versa. They might even harm either precision, recall, or both. Except when Random Forest was used, the only seven exceptions (out of 96 conditions) where both precision and recall improved were SMOTE+SVM and SMOTE+ENN for *abalone* with K-NN and logistic regression and Oversampling, SMOTE+SVM, and SMOTE+ENN for *letter* with SVM.

In the case of Random Forest, SMOTE+SVM and SMOTE+ENN improved both precision and recall on all data sets except *satimage*. Easy Ensemble improved both precision and recall on *abalone*. Oversampling improved both precision and recall for both *abalone* and *letter*, and SMOTE+Tomek also improved both for *letter*. We note that no method improved both precision and recall for *satimage* using Random Forest. We believe that the reason for these methods' marked success on Random Forest is as suggested by Japkowicz and Stephen (2002): the imbalance correction methods were designed using a decision tree classifier as their benchmark, and hence narrowly address the issues decision tree classifiers face. Nevertheless, as noted by Japkowicz and Stephen in the case of accuracy, these methods do not generalize well across different classifiers (e.g. SVM). What is different for precision-recall is that our theoretical bounds show that class imbalance still poses an inherent problem for all classifiers. In particular, a method like SVM will show the effect of class imbalance under precision-recall but not accuracy. (Indeed, Japkowicz and Stephen observed that the accuracy achieved by SVM was unaffected by the class imbalance.) We stress that the performance of an algorithm on a given data distribution is not simply determined by the class imbalance: it also depends on how well the representations used can fit the data distribution, and how successful the algorithm is at fitting them.

For the second experiment, results are shown in Table 6, containing the precision and recall results of applying different sampling techniques versus no modification on increasing sizes of the *small* training set. For both data sets (*letter* and *drug discovery*), precision and recall improved

as the size of the training set increased, but still imbalance correction methods showed no clear benefit. It suggests that precision and recall indeed scale by the size of the training set, strengthening our observation that the the effect of adding more data dominates the impact of imbalance correction methods in terms of precision and recall performance.

In general, both experiments corroborate our theoretical conclusion that training on a large data set would improve the performance (i.e. precision and recall) of the class imbalance problem, and that the methods aimed at simply "correcting" the class imbalance are not reliable. Thus, our advice in general is not to rely on the imbalance correction methods as a means of achieving high precision and recall whereas acquiring more data, when available, does help.

5 Ramifications and Advice to Practitioners

Our main finding is negative: in the presence of severe class imbalance, it is not possible to reliably achieve high precision and recall unless one possesses a large amount of training data. In some typical applications, such as machine translation, we observed that the amount of training data needed to obtain a meaningful indication of what word to use in a translation by statistical methods alone borders on the implausible. We find that methods that have been proposed to correct the class imbalance directly simply should not work, since there is an inherent statistical barrier. (And conversely, if there were enough data available, then standard machine learning techniques will do and the methods to correct the imbalance are not needed.)

To a practitioner faced with such a task, simply giving up is not an option. Our results suggest inherent limitations of purely data-driven methods for such tasks: one simply should not trust machine learning algorithms to provide rules of high precision and recall on a typical size data set with high imbalance any more than one would trust them to provide high accuracy rules on a data set of a handful of examples (e.g., a training set of size 10-50 in practice). The advice we would give to such a practitioner would be the same in both cases, that is, *to try to exploit some kind of prior knowledge about the domain.*

For example, in the case of natural language tasks, there may not be enough data to learn the appropriate usage of uncommon, informative words. But, *definitions* of these words that approximately indicate their usage are readily available in dictionaries. Indeed, we observe that humans learning a language (whether it is their native language or a foreign language) do not learn many of these words from everyday speech, but rather through definitions that are explicitly provided in language classes or dictionaries. Thus, while statistical methods might be very good at learning the usage of common words or phrases, we believe that the knowledge provided by the rules in a dictionary should be used to complement this understanding of the basic language to achieve adequate precision for tasks such as machine translation.

Similarly, in the case of the hospital early warning systems, much is known about conditions such as cardiac arrest. We hope that this explicit knowledge, properly encoded and integrated into the learning algorithm, could be used to help achieve high precision in the classification task in spite of the relatively small amount of data available to train such systems.

We caution that while more general strategies, such as us-

Table 2: Precision and Recall values for models that apply different imbalance-correction methods on the *small* training set versus K-NN alone using the *large* training set

(a) abalone

Size	Technique	Precision	Recall
417	No correction	0.2500±0.0000	0.1025±0.0000
	Oversampling	0.2439±0.0000	0.5128±0.0000
	Undersampling	0.2000±0.0127	0.8790±0.0419
	Easy Ensemble	0.2470±0.0195	0.5590±0.0671
	NearMiss-3	0.2044±0.0000	0.7179±0.0000
	SMOTE+SVM	0.2640±0.0195	0.6130±0.0546
	SMOTE+ENN	0.2530±0.0183	0.6490±0.0513
	SMOTE+Tomek	0.2290±0.0100	0.6850±0.0402
	SMOTE+CNN	0.2030±0.0000	0.3077±0.0000
16K	No correction	0.5263±0.0000	0.2564±0.0000

(b) letter

Size	Technique	Precision	Recall
2K	No correction	0.9500±0.0000	0.9620±0.0000
	Oversampling	0.9290±0.0000	0.9870±0.0000
	Undersampling	0.2620±0.0305	0.9970±0.0053
	Easy Ensemble	0.7420±0.0400	1.0000±0.0000
	NearMiss-3	0.1850±0.0000	0.9750±0.0000
	SMOTE+SVM	0.8570±0.0097	1.0000±0.0000
	SMOTE+ENN	0.8650±0.0128	1.0000±0.0000
	SMOTE+Tomek	0.8550±0.0170	1.0000±0.0000
	SMOTE+CNN	0.5910±0.0000	0.9873±0.0000
16K	No correction	0.9870±0.0000	0.9870±0.0000

(c) satimage

Size	Technique	Precision	Recall
643	No correction	0.6707±0.0000	0.5308±0.0000
	Oversampling	0.4170±0.0022	0.7770±0.0015
	Undersampling	0.2950±0.0356	0.9100±0.0269
	Easy Ensemble	0.4100±0.0148	0.7770±0.0271
	NearMiss-3	0.3079±0.0000	0.7630±0.0000
	SMOTE+SVM	0.4700±0.0070	0.8100±0.0123
	SMOTE+ENN	0.3850±0.0215	0.8570±0.0135
	SMOTE+Tomek	0.3750±0.0112	0.8630±0.0125
	SMOTE+CNN	0.5000±0.0000	0.6303±0.0000
5149	No correction	0.7513±0.0000	0.6730±0.0000

(d) drug discovery

Size	Technique	Precision	Recall
50K	No correction	0.4825±0.0000	0.4613±0.0000
	Oversampling	0.7222±0.0001	0.3714±0.0001
	Undersampling	0.0017±0.0002	0.7657±0.0911
	Easy Ensemble	0.0042±0.0006	0.5429±0.0313
	NearMiss-3	0.0012±0.0000	0.5429±0.0000
	SMOTE+SVM	0.4074±0.0006	0.3143±0.0002
	SMOTE+ENN	0.1650±0.0001	0.4857±0.0005
	SMOTE+Tomek	0.1652±0.0010	0.4853±0.0005
	SMOTE+CNN	0.6257±0.0000	0.2571±0.0000
950K	No correction	0.9474±0.0000	0.6429±0.0000

Table 3: Precision and Recall values for models that apply different imbalance-correction methods on the *small* training set versus Logistic Regression alone using the *large* training set

(a) abalone

Size	Technique	Precision	Recall
417	No correction	0.2250±0.0000	0.2450±0.0000
	Oversampling	0.2170±0.0000	0.8970±0.0000
	Undersampling	0.2090±0.0060	0.8670±0.0108
	Easy Ensemble	0.2160±0.0033	0.9080±0.0132
	NearMiss-3	0.1940±0.0000	0.7180±0.0000
	SMOTE+SVM	0.2390±0.0350	0.5560±0.0205
	SMOTE+ENN	0.2250±0.0055	0.8380±0.0173
	SMOTE+Tomek	0.2170±0.0024	0.9030±0.0108
	SMOTE+CNN	0.1980±0.0000	0.5070±0.0000
3342	No correction	0.2530±0.0000	0.2650±0.0000

(b) letter

Size	Technique	Precision	Recall
2K	No correction	0.8950±0.0000	0.8610±0.0000
	Oversampling	0.3880±0.0067	0.9750±0.0000
	Undersampling	0.2900±0.0289	0.9810±0.0136
	Easy Ensemble	0.3570±0.0195	0.9750±0.0000
	NearMiss-3	0.2350±0.0000	0.9620±0.0000
	SMOTE+SVM	0.3200±0.0139	0.9920±0.0106
	SMOTE+ENN	0.3960±0.0055	0.9720±0.0080
	SMOTE+Tomek	0.4030±0.0055	0.9750±0.0000
	SMOTE+CNN	0.5340±0.0000	0.8990±0.0000
16K	No correction	0.9210±0.0000	0.8860±0.0000

(c) satimage

Size	Technique	Precision	Recall
643	No correction	0.3090±0.0020	0.1040±0.0000
	Oversampling	0.2000±0.0047	0.5310±0.0132
	Undersampling	0.1750±0.0136	0.6300±0.0460
	Easy Ensemble	0.1970±0.0085	0.5310±0.0280
	NearMiss-3	0.1730±0.0000	0.5170±0.0000
	SMOTE+SVM	0.2130±0.0052	0.5180±0.0186
	SMOTE+ENN	0.2090±0.0080	0.5020±0.0133
	SMOTE+Tomek	0.2070±0.0060	0.4970±0.0145
	SMOTE+CNN	0.1870±0.0000	0.4740±0.0000
5149	No correction	0.4800±0.0000	0.4500±0.0000

(d) drug discovery

Size	Technique	Precision	Recall
50K	No correction	0.8000±0.0000	0.1540±0.0000
	Oversampling	0.0427±0.0002	0.5000±0.0000
	Undersampling	0.0029±0.0002	0.8970±0.0222
	Easy Ensemble	0.0080±0.0010	0.7690±0.0385
	NearMiss-3	0.0010±0.0000	0.8080±0.0000
	SMOTE+SVM	0.0465±0.0020	0.4230±0.0000
	SMOTE+ENN	0.0441±0.0003	0.5000±0.0000
	SMOTE+Tomek	0.0443±0.0000	0.5000±0.0000
	SMOTE+CNN	0.0114±0.0000	0.3850±0.0000
950K	No correction	1.0000±0.0000	0.5000±0.0000

Table 4: Precision and Recall values for models that apply different imbalance-correction methods on the *small* training set versus SVM alone using the *large* training set

(a) abalone

Size	Technique	Precision	Recall
417	No correction	0.1900±0.0000	0.9230±0.0000
	Oversampling	0.1890±0.0003	0.9230±0.0002
	Undersampling	0.1950±0.0060	0.8260±0.0713
	Easy Ensemble	0.1890±0.0031	0.9280±0.0265
	NearMiss-3	0.1890±0.0000	0.6920±0.0000
	SMOTE+SVM	0.1890±0.0042	0.9000±0.0307
	SMOTE+ENN	0.1880±0.0031	0.8900±0.0320
	SMOTE+Tomek	0.1890±0.0045	0.9310±0.0173
	SMOTE+CNN	0.1890±0.0000	0.6920±0.0000
16K	No correction	0.2100±0.0000	0.9280±0.0000

(b) letter

Size	Technique	Precision	Recall
2K	No correction	0.9620±0.0000	0.9490±0.0000
	Oversampling	0.9740±0.0000	0.9490±0.0000
	Undersampling	0.9180±0.0238	0.9700±0.0065
	Easy Ensemble	0.9630±0.0040	0.9510±0.0040
	NearMiss-3	0.9040±0.0000	0.9490±0.0000
	SMOTE+SVM	0.9740±0.0001	0.9660±0.0061
	SMOTE+ENN	0.9640±0.0053	0.9490±0.0000
	SMOTE+Tomek	0.9620±0.0000	0.9490±0.0000
	SMOTE+CNN	0.9730±0.0000	0.9240±0.0000
16K	No correction	1.0000±0.0000	1.0000±0.0000

(c) satimage

Size	Technique	Precision	Recall
643	No correction	0.3490±0.0000	0.3130±0.0000
	Oversampling	0.3490±0.0022	0.3130±0.0000
	Undersampling	0.1740±0.0217	0.6130±0.0763
	Easy Ensemble	0.2710±0.0266	0.3550±0.0385
	NearMiss-3	0.1300±0.0000	0.5120±0.0000
	SMOTE+SVM	0.3270±0.0165	0.3320±0.0211
	SMOTE+ENN	0.3390±0.0113	0.3570±0.0125
	SMOTE+Tomek	0.3400±0.0010	0.3660±0.0120
	SMOTE+CNN	0.1850±0.0000	0.5310±0.0000
5149	No correction	0.5920±0.0000	0.5780±0.0000

(d) drug discovery

Size	Technique	Precision	Recall
50K	No correction	0.1420±0.0000	0.5770±0.0000
	Oversampling	0.2100±0.0000	0.5000±0.0000
	Undersampling	0.0023±0.0005	0.8210±0.0588
	Easy Ensemble	0.0058±0.0005	0.8210±0.0222
	NearMiss-3	0.0007±0.0000	0.5380±0.0000
	SMOTE+SVM	0.1640±0.0025	0.4230±0.0000
	SMOTE+ENN	0.2090±0.0000	0.5380±0.0000
	SMOTE+Tomek	0.2080±0.0018	0.5380±0.0000
	SMOTE+CNN	0.0098±0.0000	0.7310±0.0000
950K	No correction	0.2200±0.0000	0.9230±0.0000

ing simple parametric models, have long been used in data-poor conditions, the catch here is that in order to achieve high precision, we know that we must use a method for classification that is expressive enough to achieve an extremely low error rate in the presence of class imbalance. Thus, we

Table 5: Precision and Recall values for models that apply different imbalance-correction methods on the *small* training set versus Random Forest alone using the *large* training set

(a) abalone

Size	Technique	Precision	Recall
417	No correction	0.3570±0.0000	0.3250±0.0000
	Oversampling	0.3580±0.0340	0.3330±0.0435
	Undersampling	0.2370±0.0169	0.3590±0.0367
	Easy Ensemble	0.3620±0.0254	0.4690±0.0500
	NearMiss-3	0.2580±0.0000	0.7950±0.0000
	SMOTE+SVM	0.3600±0.0146	0.5030±0.0630
	SMOTE+ENN	0.3590±0.0305	0.5410±0.0633
	SMOTE+Tomek	0.3500±0.0220	0.6150±0.0270
	SMOTE+CNN	0.2790±0.0000	0.3080±0.0000
16K	No correction	0.4850±0.0000	0.5130±0.0000

(b) letter

Size	Technique	Precision	Recall
2K	No correction	0.9590±0.0000	0.8860±0.0000
	Oversampling	0.9590±0.0000	0.9040±0.0000
	Undersampling	0.5570±0.0828	0.9890±0.0110
	Easy Ensemble	0.9400±0.0173	0.9590±0.0144
	NearMiss-3	0.3160±0.0000	0.9870±0.0000
	SMOTE+SVM	0.9600±0.0042	0.9410±0.0169
	SMOTE+ENN	0.9600±0.0006	0.9090±0.0144
	SMOTE+Tomek	0.9600±0.0006	0.9110±0.0180
	SMOTE+CNN	0.9480±0.0000	0.9240±0.0000
16K	No correction	0.9740±0.0000	0.9370±0.0000

(c) satimage

Size	Technique	Precision	Recall
643	No correction	0.7480±0.0000	0.4080±0.0000
	Oversampling	0.6580±0.0192	0.5580±0.0000
	Undersampling	0.3840±0.0419	0.8510±0.0404
	Easy Ensemble	0.6110±0.0185	0.6000±0.0111
	NearMiss-3	0.4400±0.0000	0.7300±0.0000
	SMOTE+SVM	0.6250±0.0084	0.5950±0.0182
	SMOTE+ENN	0.6270±0.0104	0.6040±0.0090
	SMOTE+Tomek	0.6260±0.0126	0.6120±0.0130
	SMOTE+CNN	0.6610±0.0000	0.5730±0.0000
5149	No correction	0.8110±0.0000	0.5800±0.0000

(d) drug discovery

Size	Technique	Precision	Recall
50K	No correction	0.8890±0.0000	0.6150±0.0000
	Oversampling	1.0000±0.0001	0.1540±0.0001
	Undersampling	0.0032±0.0005	0.8210±0.0444
	Easy Ensemble	0.0780±0.0136	0.6030±0.0222
	NearMiss-3	0.0008±0.0000	0.8850±0.0000
	SMOTE+SVM	1.0000±0.0000	0.8970±0.0022
	SMOTE+ENN	1.0000±0.0000	0.7690±0.0000
	SMOTE+Tomek	1.0000±0.0000	0.1280±0.0222
	SMOTE+CNN	1.0000±0.0000	0.1540±0.0000
950K	No correction	1.0000±0.0000	0.7690±0.0000

don't generally expect that these "rough approximations" will do for such tasks. Some kind of richer representation

Table 6: Precision and Recall values for models that apply different sampling methods versus K-NN alone on varying sizes of data sets

(a) letter

Size	Technique	Precision	Recall
4K	Oversampling	0.8850±0.0000	0.9750±0.0000
	Easy Ensemble	0.7270±0.0246	0.9890±0.0072
	SMOTE+SVM	0.8490±0.0068	0.9870±0.0000
	SMOTE+Tomek	0.8550±0.0059	0.9870±0.0000
	SMOTE+CNN	0.8520±0.0049	0.9870±0.0000
	No correction	0.9625±0.0000	0.9750±0.0000
8K	Oversampling	0.9260±0.0046	0.9870±0.0000
	Easy Ensemble	0.8120±0.0212	0.9970±0.0053
	SMOTE+SVM	0.9080±0.0000	1.0000±0.0000
	SMOTE+Tomek	0.9060±0.0095	1.0000±0.0000
	SMOTE+CNN	0.9020±0.0053	1.0000±0.0000
	No correction	0.9740±0.0000	0.9620±0.0000
12K	Oversampling	0.9450±0.0000	0.9870±0.0000
	Easy Ensemble	0.9260±0.0278	0.9960±0.0061
	SMOTE+SVM	0.9450±0.0000	0.9870±0.0000
	SMOTE+Tomek	0.9570±0.0055	1.0000±0.0000
	SMOTE+CNN	0.9550±0.0055	1.0000±0.0000
	No correction	0.9870±0.0000	0.9870±0.0000
16K	No correction	0.9870±0.0000	0.9870±0.0000

(b) drug discovery

Size	Technique	Precision	Recall
100K	Oversampling	0.7350±0.0015	0.4084±0.0015
	SMOTE+SVM	0.4791±0.0008	0.3620±0.0015
	SMOTE+Tomek	0.2310±0.0012	0.5230±0.0008
	SMOTE+CNN	0.6733±0.0125	0.3062±0.0020
	No correction	0.6042±0.0015	0.5346±0.0030
250K	Oversampling	0.7413±0.0016	0.4218±0.0012
	SMOTE+SVM	0.5164±0.0005	0.4168±0.0021
	SMOTE+Tomek	0.2857±0.0016	0.5667±0.0010
	SMOTE+CNN	0.7055±0.0105	0.3350±0.0022
	No correction	0.7560±0.0022	0.5873±0.0025
500K	Oversampling	0.7482±0.0013	0.4432±0.0022
	SMOTE+SVM	0.5338±0.0009	0.4542±0.0025
	SMOTE+Tomek	0.3155±0.0012	0.6069±0.0015
	SMOTE+CNN	0.7241±0.0110	0.3782±0.0016
	No correction	0.8437±0.0010	0.6255±0.0017
950K	No correction	0.9474±0.0003	0.6429±0.0001

of knowledge and classification decisions will be necessary to achieve high precision.

6 Future Work

As mentioned earlier, we did not obtain any bounds for the sample complexity of additive-error agnostic learning. This is the one main significant piece missing from the picture for the sample complexity of precision and recall. Indeed, the key question here is, does the overhead scale like $1/\mu$ or $1/\mu^2$ (like the dependence on the additive error parameter)?

There is also a much larger family of questions one might seek to answer along similar lines: in the scope of this paper, we focus only on the precision and recall of the learned classifier. However, in some cases, precision-recall may not be the most desirable metric (e.g. when the number of true negative examples is large). We might also consider examining

the sample complexity of learning under other metrics such as the Brier Score, H-measure, F-measure, etc. The empirical results of Raeder et al. (2012) suggest that these measures likewise will depend on the class imbalance, but the question is (of course) how precisely achieving bounds on these alternative measures of loss depends on these parameters. Relatedly, the top-k metrics are often used in information retrieval to overcome the problem of low precision in such challenging settings. It would be interesting to better understand the sample complexity of these metrics as well.

Acknowledgements

This work was supported by an AFOSR Young Investigator award and NSF award CCF-1718380. We thank R. Garnett, S. Das, and our anonymous reviewers for their suggestions.

References

Agarwal, S., and Roth, D. 2005. Learnability of bipartite ranking functions. In *International Conference on Computational Learning Theory*, 16–31. Springer.

Akbani, R.; Kwek, S.; and Japkowicz, N. 2004. *Applying Support Vector Machines to Imbalanced Datasets*. Berlin, Heidelberg: Springer Berlin Heidelberg. 39–50.

Alvarez, S. A. 2002. An exact analytical relation among recall, precision, and classification accuracy in information retrieval. Technical Report BC-CS-2002-01, Computer Science Department, Boston College.

Amaral, A. C. K. B. 2014. The art of making predictions: Statistics versus bedside evaluation. *American J. Respiratory and Critical Care Medicine* 190(6):598–599.

Anthony, M., and Bartlett, P. 1999. *Neural Network Learning: Theoretical Foundations*. Cambridge: Cambridge University Press.

Awasthi, P.; Blum, A.; and Sheffet, O. 2010. Improved guarantees for agnostic learning of disjunctions. In *Proc. 23rd COLT*, 359–367.

Batista, G. E. A. P. A.; Prati, R. C.; and Monard, M. C. 2004. A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explor. Newsl.* 6(1):20–29.

Bird, S.; Klein, E.; and Loper, E. 2009. *Natural Language Processing with Python*. O'Reilly Media, Inc., 1st edition.

Blumer, A.; Ehrenfeucht, A.; Haussler, D.; and Warmuth, M. K. 1989. Learnability and the Vapnik-Chervonenkis dimension. *J. ACM* 36(4):929–965.

Brants, T.; Popat, A. C.; Xu, P.; Och, F. J.; and Dean, J. 2007. Large language models in machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 858–867. Prague, Czech Republic: Association for Computational Linguistics.

Cannon, A.; Howse, J.; Hush, D.; and Scovel, C. 2002. Learning with the Neyman-Pearson and min-max criteria. Technical Report LA-UR-02-2951, Los Alamos National Laboratory.

Chawla, N. V.; Bowyer, K. W.; Hall, L. O.; and Kegelmeyer, W. P. 2002. SMOTE: synthetic minority over-sampling technique. *JAIR* 16:321–357.

Cortes, C., and Mohri, M. 2004. AUC optimization vs. error rate minimization. In *Advances in neural information processing systems*, 313–320.

Daniely, A., and Shalev-Shwartz, S. 2016. Complexity theoretic limtations on learning DNF's. In *Proc. 29th COLT*, volume 49 of *JMLR Workshops and Conference Proceedings*. 1–16.

Davis, J., and Goadrich, M. 2006. The relationship between precision-recall and ROC curves. In *Proc. 23rd ICML*, 233–240.

Ehrenfeucht, A.; Haussler, D.; Kearns, M.; and Valiant, L. 1989. A general lower bound on the number of examples needed for learning. *Inf. Comp.* 82:247–261.

Garnett, R.; Gärtner, T.; Vogt, M.; and Bajorath, J. 2015. Introducing the 'active search' method for iterative virtual screening. *Journal of Computer-Aided Molecular Design* 29(4):305–314.

Gyorfi, L. D. L.; Lugosi, G.; and Devroye, L. 1996. *A probabilistic theory of pattern recognition*. New York: Springer.

Hanneke, S. 2016. The optimal sample complexity of PAC learning. *JMLR* 17(38):1–15.

He, H., and Garcia, E. A. 2009. Learning from imbalanced data. *IEEE Trans. Knowledge and Data Eng.* 21(9):1263–1284.

Japkowicz, N., and Stephen, S. 2002. The class imbalance problem: A systematic study. *Intelligent data analysis* 6(5):429–449.

Juba, B. 2016. Learning abductive reasoning using random examples. In *Proc. 30th AAAI*, 999–1007.

Juba, B. 2017. Conditional sparse linear regression. In *Proc. 8th ITCS*, volume 67 of *LIPIcs*.

Lemaître, G.; Nogueira, F.; and Aridas, C. K. 2016. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *CoRR* abs/1609.06570.

Liu, B.; Lee, W. S.; Yu, P. S.; and Li, X. 2002. Partially supervised classification of text documents. In *Proc. 19th ICML*, 387–394.

Liu, X.; Wu, J.; and Zhou, Z. 2006. Exploratory under-sampling for class-imbalance learning. In *Sixth International Conference on Data Mining (ICDM'06)*, 965–969.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12:2825–2830.

Raeder, T.; Forman, G.; and Chawla, N. V. 2012. Learning from imbalanced data: Evaluation matters. In *Data Mining: Foundations and Intelligent Paradigms*, volume 23 of *Intelligent Systems Reference Library*. Berlin: Springer. 315–331.

Sandhaus, E. 2008. The New York Times annotated corpus ldc2008t19. DVD.

Scott, C., and Nowak, R. 2005. A Neyman-Pearson approach to statistical learning. *IEEE Trans. Inf. Theory* 51(11):3806–3819.

Valiant, L. G. 2006. Knowledge infusion. In *Proc. AAAI-06*, 1546–1551.

Vapnik, V., and Chervonenkis, A. 1971. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications* 16(2):264–280.

Weiss, G. M., and Provost, F. 2003. Learning when training data are costly: The effect of class distribution on tree induction. *JAIR* 19:315–354.

Weiss, G. M. 2004. Mining with rarity: A unifying framework. *ACM SIGKDD Explorations Newsletter* 6(1):7–19.

Zhang, J., and Mani, I. 2003. KNN Approach to Unbalanced Data Distributions: A Case Study Involving Information Extraction. In *Proceedings of the ICML'2003 Workshop on Learning from Imbalanced Datasets*.