Evaluating Classifiers by Means of Test Data with Noisy Labels

Chuck P. Lam
Department of Electrical Engineering
Stanford University
Stanford, CA 94305
chucklam@stanford.edu

David G. Stork
Ricoh Innovations, Inc.
2882 Sand Hill Road, Suite 115
Menlo Park, CA 94025-7022
stork@rii.ricoh.com

Abstract

Often the most expensive and time-consuming task in building a pattern recognition system is collecting and accurately labeling training and testing data. In this paper, we explore the use of inexpensive noisy *testing* data for evaluating a classifier's performance. We assume 1) the (human) labeler provides category labels with a known mislabeling rate and 2) the trained classifier and the labeler are statistically independent. We then derive the number of "noisy" test samples that are, on average, equivalent to a single perfectly labeled test sample for the task of evaluating the classifier's performance. For practical and realistic error and mislabeling rates, this number of equivalent test patterns can be surprisingly low. We also derive an upper and lower bound for the true error rate when the labeler and the classifier are not independent.

1 Introduction

The overall construction of a modern classification system can be divided into four broad tasks: (1) specifying the classifier type, (2) collecting data, (3) training the classifier (i.e., learning), and (4) evaluating the classifier (i.e., testing) [Duda *et al.*, 2001]. The second stage, data collection, can further be divided into two tasks: gathering samples and labeling them. Recently, the machine learning community has realized that in many practical cases the most expensive part of the whole design process is the labeling of such samples. For example, there is an enormous number of text documents on the internet that can be obtained at very low cost; however, relatively few of these have been labeled — e.g., according to content topic, language, or style — in a consistent way that would facilitate training a classifier. Likewise, there are large databases of recorded speech, handwritten digits, and printed characters but these databases, too, are either not labeled accurately or not labeled at all [Stork, 1999]. To reduce the labeling expense, many researchers have sought ways to modify training algorithms so as to utilize both labeled and unlabeled data [Blum and Mitchell, 1998; Nigam *et al.*, 2000]. This approach has shown surprisingly encouraging results, in some cases reducing the number of labeled samples by a few orders of magnitude [Nigam *et al.*, 2000].

In order to build up and extend this success in reducing the labeling cost, we turn to the problem of reducing the need for accurately labeled data in the classifier *evaluation* stage. In fact, most of the experiments for learning with labeled and unlabeled data use much more labels for testing than training [Nigam *et al.*, 2000]. Thus we now need to address the labeling cost for classifier evaluation.

As with many areas of commerce, the general economics of labeling is such that the higher the quality (accuracy) of labeling, the greater the associated cost. This greater cost may be due to greater expertise of the labeler, or the need for multiple passes of cross-checking, or both. There is thus an additional cost to "clean" or "truth" those data and labels. In some situations, such as marking a text corpus, the labeling task is complicated enough that even experts need several passes to reduce labeling errors [Eskin, 2000]. Furthermore, in some application domains, obtaining accurate labels is simply too cost prohibitive. For example, for some medical diagnostics, the true disease can only be known with expensive or invasive techniques. Similarly, in remote sensing, one must send measuring instruments to the ground location to obtain the "ground truth," and the transportation cost can be astronomical. (It is quite literal for remote sensing of other planets [Smyth, 1997].) For both situations in practice, one must rely on the imperfect judgements of experts [Smyth, 1997].

We propose to lower the labeling cost in classifier evaluation by using cheaper, noisy labels. This paper examines methodologies of estimating the error rate and classifier confusion matrix using test data with noisy labels. We shall see that even a slight labeling inaccuracy (say, 1%) can have a significant effect on the error rate estimate when the classifier performs well. In addition, when data sets used to be small and expensive to collect, it made sense to spend each additional labeling effort to increase label accuracy on that small data set. However, when data sets are large and cheap to collect, it is no longer obvious how one should spend each additional labeling effort. Should one spend it labeling the unlabeled data, or should one spend it increasing the accuracy of already labeled data? We present a preliminary analysis to this question.

2 Preliminaries and notation

Our formulation assumes an object $\mathbf{x}$ possessing a *true* label $y \in \Omega$, where $\Omega = \{\omega_1, \ldots, \omega_c\}$ is the set of possible states of nature (e.g., category membership) for the object. The object is presented to a labeler, who marks it with a label $\tilde{y} \in \Omega$, as his guess of y. The situation that $y \neq \tilde{y}$ is call a *labeling error* (or a mislabeling). The classifier system, on the other hand, is presented with the feature vector $\mathbf{x}$ that represents certain aspects of the object, and the classifier outputs a label $\hat{y}(\mathbf{x}) \in \Omega$, as its guess of y. For notational convenience, we will call the classifier output $\hat{y}$, and its dependence on the feature vector $\mathbf{x}$ is implicit. The situation that $y \neq \hat{y}$ is call a *classification error* (or a misclassification).

The probability of the labeler making mistakes, $\Pr[y \neq \tilde{y}]$, is called the *mislabeling rate*. The probability of the classifier's label being different from the labeler's label, $\Pr[\hat{y} \neq \tilde{y}]$, is called the *apparent error rate*. Our goal is to estimate $\Pr[y \neq \hat{y}]$, which is called the *true error rate*. Note that it is possible to have a high apparent error rate even with a perfect classifier (with a true error rate of zero) simply because of a high mislabeling rate. That is, the classifier can classify all test data perfectly, but will often disagree with the test labels because those labels are incorrect. On the other hand, it is also possible to have a zero apparent error rate even with a high true error rate if the classifier and the labeler make the same kind of mistakes.

The *confusion matrix* for the human labeler is defined as

$$\mathbf{P}_{\tilde{y}|y} \equiv \begin{pmatrix} P(\tilde{y} = \omega_1 | y = \omega_1) & \cdots & P(\tilde{y} = \omega_1 | y = \omega_c) \\ \vdots & \ddots & \vdots \\ P(\tilde{y} = \omega_c | y = \omega_1) & \cdots & P(\tilde{y} = \omega_c | y = \omega_c) \end{pmatrix},$$

which is the identity matrix for a perfect labeler. Similarly the classifier's confusion matrix is defined as

$$\mathbf{P}_{\hat{y}|y} \equiv \begin{pmatrix} P(\hat{y} = \omega_1 | y = \omega_1) & \cdots & P(\hat{y} = \omega_1 | y = \omega_c) \\ \vdots & \ddots & \vdots \\ P(\hat{y} = \omega_c | y = \omega_1) & \cdots & P(\hat{y} = \omega_c | y = \omega_c) \end{pmatrix}.$$

For many two-class cases where one class has a much higher prior probability, the actual error rate is not a good measure of classifer usefulness. For example, in detecting email spams or network intrusions, the undesirable events are so rare that one can easily get an error rate less than 1% by classifying all events as "desirable." In those situations, then, one may want to compute the entire confusion matrix or metrics such as *precision* and *recall* [Frakes and Baeza-Yates, 1992]. We denote ω_2 as the "rare" class (e.g., spams or network intrusions). For the classifier, precision is defined as $P(y = \omega_2 | \hat{y} = \omega_2)$ and recall is defined as $P(\hat{y} = \omega_2 | y = \omega_2)$, and analogously for the labeler. Note that precision and recall can be derived from the confusion matrix and the class prior probabilities.

3 Obtaining the true error rate

In examining the relationship between true and apparent error rates, we make the constraint that we have a two-class problem, that is, $\Omega = \{\omega_1, \omega_2\}$. Thus $\{\hat{y} \neq \tilde{y} \cup y \neq \tilde{y}\} \Rightarrow \{y = \hat{y}\}$. Then we can rewrite the apparent error rate as

$$\begin{aligned} \Pr[\hat{y} \neq \tilde{y}] &= \Pr[\hat{y} \neq \tilde{y}, y \neq \tilde{y}] + \Pr[\hat{y} \neq \tilde{y}, y = \tilde{y}] \\ &= \Pr[y = \hat{y}, y \neq \tilde{y}] + \Pr[y \neq \hat{y}, y = \tilde{y}] \\ &= \Pr[y = \hat{y}] \Pr[y \neq \tilde{y}] \\ &\quad + \Pr[y \neq \hat{y}] \Pr[y = \tilde{y}] \qquad (1) \\ &= (1 - \Pr[y \neq \hat{y}]) \Pr[y \neq \tilde{y}] \\ &\quad + \Pr[y \neq \hat{y}](1 - \Pr[y \neq \tilde{y}]) \\ &= \Pr[y \neq \tilde{y}] + \Pr[y \neq \hat{y}](1 - 2\Pr[y \neq \tilde{y}]) \end{aligned}$$

and thus

$$\Pr[y \neq \hat{y}] = \frac{\Pr[\hat{y} \neq \tilde{y}] - \Pr[y \neq \tilde{y}]}{1 - 2\Pr[y \neq \tilde{y}]} \qquad (2)$$

The above derivation assumed that $\Pr[y \neq \tilde{y}] \neq 1/2$; otherwise the noisy labels are meaningless. In practice, $\Pr[y \neq \tilde{y}]$ is always much less than $1/2$. More important is the independence assumption that the labeler and the classifier make errors independently, or stated succintly, $\Pr[y \neq \hat{y}, y \neq \tilde{y}] = \Pr[y \neq \hat{y}] \Pr[y \neq \tilde{y}]$. That is, knowing that the labeler had made an error on a pattern does not change the probability that the classifier would also make an error, and vice versa. Section 6 will deal with some situations in which the independence assumption does not hold. In the meantime, we argue for this idealization and simplification based on the fact that human and computer generally classify samples using different methodologies, and thus they may not make similar kinds of mistakes.

3.1 Example: Apparent error rate for various true error rates and mislabeling rates

Equation 2 gives us a way to account for noisy labels when calculating the true error rate. A natural question, then, is how important is it to correct for the influence of noisy labels? Let's consider some classification systems with error rates between 2% and 10%[1] and testing data sets with 1% to 5% incorrect labels. Table 1 shows the apparent error rate for classifiers of different accuracy and testing data of different mislabeling rates. The percentage increase over the true error rate is also shown. For example, even when only 1% of the testing labels are wrong, a classifier with true error rate of 6% will have an apparent error rate 15% higher (at 6.88%). The percentage increase is even more dramatic with noisier labels or more accurate classifiers. A quick rule of thumb is that, when the labels have relatively few errors, the denominator in Eq. 2 is approximately 1.0 and can be ignored. The mislabeling rate of the testing labels ($\Pr[y \neq \tilde{y}]$) is then just an additive component to the true error rate. Continuing the previous example, a 1% mislabeling rate for a classifier with true error rate of 6% makes the apparent error rate approximately 7%, when the actual is 6.88%.

4 Noisy labels for estimating true error rate

Above we assumed knowledge of the apparent error rate, $\Pr[\hat{y} \neq \tilde{y}]$, but in practice, we must estimate this rate using test data. In this section, we analyze the effects of

[1] We note that many classifi ers on the UCI datasets have accuracy within this range [Kaynak and Alpaydin, 2000].

True Error Rate, $\Pr[y \neq \hat{y}]$	Mislabeling Rate, $\Pr[y \neq \tilde{y}]$									
	1%		2%		3%		4%		5%	
2%	2.96%	↑48%	3.92%	↑96%	4.88%	↑144%	5.84%	↑192%	6.80%	↑240%
4%	4.92%	↑23%	5.84%	↑46%	6.76%	↑69%	7.68%	↑92%	8.60%	↑115%
6%	6.88%	↑15%	7.76%	↑29%	8.64%	↑44%	9.52%	↑59%	10.40%	↑73%
8%	8.84%	↑11%	9.68%	↑21%	10.52%	↑32%	11.36%	↑42%	12.20%	↑53%
10%	10.80%	↑8%	11.60%	↑16%	12.40%	↑24%	13.20%	↑32%	14.00%	↑40%

Table 1: The left sub-columns of the table show the apparent error rates ($\Pr[\hat{y} \neq \tilde{y}]$) for different true error rates ($\Pr[y \neq \hat{y}]$) and different mislabeling rates ($\Pr[y \neq \tilde{y}]$), based on Eq. 2. It is assumed that the labeler and the classifier make errors independently. The right sub-columns of the table, with up-arrow (↑) signs, show the percentage increase of the apparent error rate over the true error rate (i.e., $(\Pr[\hat{y} \neq \tilde{y}] - \Pr[y \neq \hat{y}])/\Pr[y \neq \hat{y}]$).

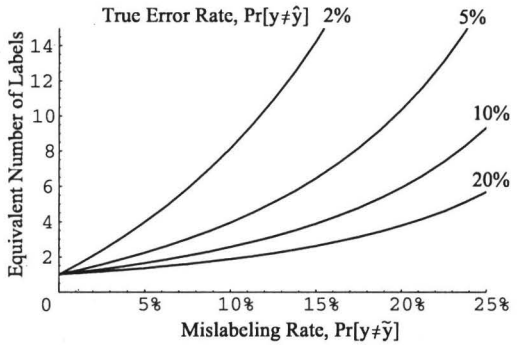

Figure 1: The figure shows the number of noisy labels needed to achieve the same variance in the true error rate estimate as a single perfect label (see Eq. 3). The four plots represent different true error rates. As the mislabeling rate increases, more noisy labels are needed to achieve the same confidence. Note that, in the ranges shown in the figure, when the mislabeling rate is smaller than the true error rate, a single perfect label is equivalent to less than four noisy labels.

such estimates. Assume we have l objects in the test set, each with a feature vector $\mathbf{x}_i$, true but unknown label y_i, noisy label $\tilde{y}_i$, and classification $\hat{y}_i$. Assume further that tuples $\{(\mathbf{x}_1, y_1, \tilde{y}_1, \hat{y}_1), \ldots, (\mathbf{x}_l, y_l, \tilde{y}_l, \hat{y}_l)\}$ are independent and identically distributed as $(\mathbf{x}, y, \tilde{y}, \hat{y})$. The apparent error rate estimate is $\hat{\Pr}[\hat{y} \neq \tilde{y}] = \frac{1}{l}\sum_{i=1}^{l} I[\hat{y}_i \neq \tilde{y}_i]$, in which $I[\cdot]$ is the indicator function (i.e., $I[event] = 1$ if $event$ is true and 0 otherwise). An estimate of the true error rate is

$$\hat{\Pr}[y \neq \hat{y}] \equiv \frac{\hat{\Pr}[\hat{y} \neq \tilde{y}] - \Pr[y \neq \tilde{y}]}{1 - 2\Pr[y \neq \tilde{y}]}.$$

It is straightforward to verify that $\mathcal{E}[\hat{\Pr}[\hat{y} \neq \tilde{y}]] = \Pr[\hat{y} \neq \tilde{y}]$ and $\mathcal{E}[\hat{\Pr}[y \neq \hat{y}]] = \Pr[y \neq \hat{y}]$, thus the estimates are unbiased. Intuitively we know that we have less confidence when the error estimates are based on test data with noisy labels. To formalize this intuition, we examine the variance of the true error rate estimate,

$$\mathrm{Var}[\hat{\Pr}[y \neq \hat{y}]] = \frac{1}{l}\left(\frac{\Pr[y \neq \tilde{y}](1 - \Pr[y \neq \tilde{y}])}{(1 - 2\Pr[y \neq \tilde{y}])^2} + \Pr[y \neq \hat{y}](1 - \Pr[y \neq \hat{y}])\right).$$

The variance of the error estimate given perfectly labeled data is $\Pr[y \neq \hat{y}](1 - \Pr[y \neq \hat{y}])/l$. Thus, to get the same variance, the ratio of noisy labels to perfect labels is

$$\frac{\Pr[y \neq \tilde{y}](1 - \Pr[y \neq \tilde{y}])}{(1 - 2\Pr[y \neq \tilde{y}])^2 \Pr[y \neq \hat{y}](1 - \Pr[y \neq \hat{y}])} + 1. \quad (3)$$

This ratio will help us understand the economic trade-offs between using perfect and noisy labels. Collecting perfect labels (or collecting noisy labels first and cleaning them) is often much more expensive than just collecting noisy labels itself. Therefore it may be economically justified to used noisy labels, as long as one does not need too many more of them.

Unfortunately, applying Eq. 3 requires us to know the true error rate of the classifier, which is exactly what one is trying to estimate. However, we often already have a good idea of a reasonable range for the true error rate. In any case, we examine the ratio for a wide range of true error rate and mislabeling rate, and we found the ratio to fall within a relatively narrow range, as shown in Fig. 1. Even for relatively noisy testing data with 10% incorrect labels, unless the classifier is much more accurate (with true error rate of less than 5%), cleaning the testing data to be perfectly labeled increases its value by less than a factor of four. In other words, one needs much less than four such noisy labels to achieve the same effect as one perfect label. Imagine that perfect labels need to be collected from a domain expert, whereas noisy labels can be collected from a non-expert, the high cost of a domain expert can often justify the use of noisy labels.

4.1 Example: Evaluating with many noisy labels or few reliable labels

In many labeling tasks, experts must make multiple passes through samples to ensure accurate labeling [Eskin, 2000]. We now question the wisdom of that policy when the samples are free but labeling cost is a constraint.

Consider a hypothetical labeling situation with two labelers, each paid to look at l samples, and both labelers have an error rate of ϵ. There are two choices in how to use these two labelers. One is to have them look at completely different samples, thus in the end we have a testing set of size $2l$ and mislabeling rate ϵ. Another choice is to have them look at the exact same samples. Assuming that they make independent

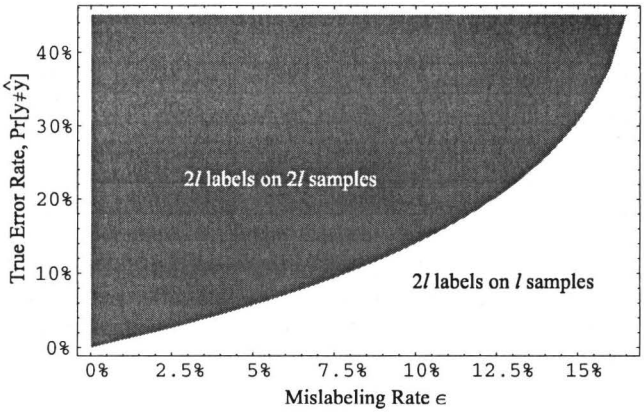

Figure 2: The figure shows which one of the two labeling policies is optimal for a range of mislabeling rate ϵ and true classifier error rate, based on Eq. 4. The problem is posed such that two labelers both have mislabeling rate ϵ and are paid to label l samples. One policy is that they label different samples, creating a test set of "$2l$ labels on $2l$ samples," with ϵ portion mislabeled. The other policy is that they both label the same samples, creating a testing set of "$2l$ labels on l samples," with ϵ^2 portion mislabeled (after various assumptions).

labeling errors, and optimistically assume that a sample has a wrong label only if both labelers err, then we have a testing set of size l and mislabeling rate ϵ^2. Which is the better policy?

Based on the previous discussion, we can have an unbiased estimate of the true error rate from either testing set. We then should prefer one that gives us a lower variance estimate. That is, we go with the "$2l$ labels on $2l$ samples" policy if its variance is lower than the "$2l$ labels on l samples" policy,

$$\begin{aligned} & \tfrac{1}{2l}\left(\tfrac{\epsilon(1-\epsilon)}{(1-2\epsilon)^2} + \Pr[y \neq \hat{y}](1 - \Pr[y \neq \hat{y}])\right) \\ < \; & \tfrac{1}{l}\left(\tfrac{\epsilon^2(1-\epsilon^2)}{(1-2\epsilon^2)^2} + \Pr[y \neq \hat{y}](1 - \Pr[y \neq \hat{y}])\right). \end{aligned}$$

Which, after a little algebra, becomes

$$\left(\tfrac{2\epsilon^2(1-\epsilon^2)}{(1-2\epsilon^2)^2} - \tfrac{\epsilon(1-\epsilon)}{(1-2\epsilon)^2} + \tfrac{1}{4}\right) > \left(\Pr[y \neq \hat{y}] - \tfrac{1}{2}\right)^2. \quad (4)$$

An interesting observation is that in the realistic range of ϵ ($0 \leq \epsilon < 0.5$), the left hand side of Eq. 4 is negative for $\epsilon > 0.166$, which means that one should *always* choose the "$2l$ labels on l samples" policy for such inaccurate labelers, and such high mislabeling rate does occur in practice [Smyth, 1997]. For other cases, we have plotted the policy boundary (i.e., $\frac{1}{2} - \sqrt{\frac{2\epsilon^2(1-\epsilon^2)}{(1-2\epsilon^2)^2} - \frac{\epsilon(1-\epsilon)}{(1-2\epsilon)^2} + \frac{1}{4}} = \Pr[y \neq \hat{y}]$) in Fig. 2.

For fairly accurate labelers (say, $\epsilon < 2.5\%$), Fig. 2 shows that one should prefer the "$2l$ labels on $2l$ samples" policy unless the classifier error rate is very low. The hint for practitioners is that time spent cleaning labels is often not as effective as time spent labeling extra samples.

5 Obtaining True Confusion Matrix

In evaluating classification systems, we often need to know more than just the error rate. When the cost of different mis-

precision/recall (classifier)	precision/recall (labeler)		
	99%	98%	95%
70%	69.3%	68.7%	66.7%
80%	79.2%	78.4%	76.1%
90%	89.1%	88.2%	85.6%
100%	99%	98%	95%

Table 2: Apparent precision recall breakeven points (i.e., $P(\tilde{y} = \omega_2 | \hat{y} = \omega_2)$) for different actual classifier precision/recall and labeler precision/recall. The prior probabilities for ω_1 and ω_2 are 90% and 10% respectively.

classifications (e.g., false positives and false negatives) are not equal, we may want to know the full confusion matrix. In addition, in some domains, such as classifying text or spam, the distribution of classes is highly skewed, and the error rate can be misleadingly low. In those situations, we are more interested in precision and recall statistics, which can be estimated from the confusion matrix.

We define the *joint distribution matrix* between labeler and classifier as

$$\mathbf{P}_{\tilde{y},\hat{y}} \equiv \begin{pmatrix} P(\tilde{y} = \omega_1, \hat{y} = \omega_1) & \cdots & P(\tilde{y} = \omega_1, \hat{y} = \omega_c) \\ \vdots & \ddots & \vdots \\ P(\tilde{y} = \omega_c, \hat{y} = \omega_1) & \cdots & P(\tilde{y} = \omega_c, \hat{y} = \omega_c) \end{pmatrix},$$

which can be estimated from data. Note that unlike our analysis of the error rates, it is not necessary to assume two-class problems.

Our goal is to recover the classifer's confusion matrix given the labeler's confusion matrix and the joint distribution matrix between labeler and classifier. If we make the independence assumption that $P(\tilde{y}, \hat{y}|y) = P(\tilde{y}|y)P(\hat{y}|y)$, then we have the decomposition $P(\tilde{y}, \hat{y}) = \sum_y P(y)P(\tilde{y}|y)P(\hat{y}|y)$. We can rewrite the decomposition in matrix form and solve for the classifier's confusion matrix.

$$\begin{aligned} \mathbf{P}_{\tilde{y},\hat{y}} &= \mathbf{P}_{\tilde{y}|y} diag(\mathbf{p}_y) \mathbf{P}^t_{\hat{y}|y} \\ \mathbf{P}_{\hat{y}|y} &= ((\mathbf{P}_{\tilde{y}|y} diag(\mathbf{p}_y))^{-1} \mathbf{P}_{\tilde{y},\hat{y}})^t \\ &= (diag(\mathbf{p}_y)^{-1} \mathbf{P}^{-1}_{\tilde{y}|y} \mathbf{P}_{\tilde{y},\hat{y}})^t, \end{aligned} \quad (5)$$

in which $\mathbf{p}_y$ is defined to be the column vector of prior probabilities, $(P(y = \omega_1), \ldots, P(y = \omega_c))^t$.

When $\mathbf{p}_y$ is not given, it can be derived. To see this, define the probability vector $\mathbf{p}_{\tilde{y}} \equiv (P(\tilde{y} = \omega_1), \ldots, P(\tilde{y} = \omega_c))^t$. It is the case that $\mathbf{p}_{\tilde{y}} = \mathbf{P}_{\tilde{y},\hat{y}} \mathbf{1}_c$, in which $\mathbf{1}_c$ is a column vector of c 1's. It is also the case that $\mathbf{p}_{\tilde{y}} = \mathbf{P}_{\tilde{y}|y} \mathbf{p}_y$. Combining those two equations we have

$$\mathbf{p}_y = \mathbf{P}^{-1}_{\tilde{y}|y} \mathbf{P}_{\tilde{y},\hat{y}} \mathbf{1}_c \quad (6)$$

5.1 Example: Precision/recall breakeven points

As mentioned earlier, one benefit of being able to recover the confusion matrix is that one can then work with precision and recall measures. We analyze the following system to see some effects of noisy labels on those measures. To reduce the number of variables examined, we only look at precision

recall breakeven points, defined as the points where precision and recall are equal. They will simply be denoted as precision/recall. For a given $\mathbf{p}_y$ and precision/recall, the confusion matrix is uniquely determined. Table 2 shows the apparent precision/recall (i.e., $P(\tilde{y} = \omega_2 | \hat{y} = \omega_2)$ or $P(\hat{y} = \omega_2 | \tilde{y} = \omega_2)$) for different actual classifier precision/recall and labeler precision/recall. Table 2 assumes $\mathbf{p}_y = (.9, .1)^t$, although the values are almost exactly the same for both $\mathbf{p}_y = (.8, .2)^t$ and $\mathbf{p}_y = (.999, .001)^t$.

6 Bounds on true error rate when the classifier and labeler are not independent

In deriving the true error rate (Eq. 2), we have made the assumption that the labeler and the classifier make errors independently. We argue for this assumption because human and computer use different methodologies to classify samples. Even for algorithms inspired by human reasoning (e.g., neural networks), they still do not learn human intuition but they do avoid psychological biases. It is even harder to imagine algorithms based on more abstract models (e.g., support vector machine) to err in similar ways as humans. Furthermore, in many application domains (e.g., speech recognition), humans label the samples based on a full presentation of the object, whereas the feature vector $\mathbf{x}$ used for classification are mathematical notions (e.g., linear vector coefficients) that have little neurological basis. In many other application domains (e.g., statistical text classification), assumptions that blatantly violate how human reasons are often made (e.g., assume words in a text are independently generated, rather than in a grammatical way). Lastly, to make a stronger argument, we can require the training data to be labeled independently from the testing data (or better yet, be perfectly labeled), thus avoiding the possibility that the computer would learn biases and other "bad habits" from the training data that would correlate with labeling errors in the testing data.

However, even with the above reasoning for the independence assumption, it is still conceivable for one to be more conservative and assume some *non-negative* dependency,

$$\Pr[y \neq \hat{y}, y \neq \tilde{y}] \geq \Pr[y \neq \hat{y}] \Pr[y \neq \tilde{y}].$$

That is, the probability of a classifier misclassifying a sample is higher if the labeler has also mislabeled that sample , and vice versa. This can happen, for example, if the training data have been mislabeled in the same way as the testing data, and the classifier has learned to imitate those mislabelings. We have deliberately ignored the case of negative dependency, $\Pr[y \neq \hat{y}, y \neq \tilde{y}] < \Pr[y \neq \hat{y}] \Pr[y \neq \tilde{y}]$, as we are hard-pressed to find a justification for it in practice.

The non-negative dependency assumption is easily incorporated into Eq. 1 by changing the equal sign to a less-than-or-equal-to sign. Propagating that change through the derivation, we have a lower bound on the true error rate,

$$\begin{aligned} \Pr[y \neq \hat{y}] &\geq \frac{\Pr[\hat{y} \neq \tilde{y}] - \Pr[y \neq \tilde{y}]}{1 - 2\Pr[y \neq \tilde{y}]} \\ &\geq \Pr[\hat{y} \neq \tilde{y}] - \Pr[y \neq \tilde{y}] \end{aligned}$$

The second inequality can be tight if the mislabeling rate is small, as the denominator of the first inequality becomes approximately one.

Separately we derive an upper bound for the true error rate.

$$\begin{aligned} \Pr[\hat{y} = \tilde{y}] &= \Pr[\hat{y} = \tilde{y}, y \neq \tilde{y}] + \Pr[\hat{y} = \tilde{y}, y = \tilde{y}] \\ &\leq \Pr[y \neq \tilde{y}] + \Pr[y = \hat{y}] \\ \Pr[y \neq \hat{y}] &\leq \Pr[y \neq \tilde{y}] + \Pr[\hat{y} \neq \tilde{y}]. \end{aligned}$$

Note that no assumption is used in deriving the upper bound (not even limiting to two-class problems). One can easily verify that the bound is exact when the mislabeling rate is zero . The bound is also exact when the apparent error rate is zero, such that the true error rate is equal to the mislabeling rate. Thus with the looser assumption of non-negative dependency, the true error rate is in the range of $\Pr[\hat{y} \neq \tilde{y}] \pm \Pr[y \neq \tilde{y}]$.

6.1 Example: Simulation of non-negative dependency between classifier and labeler

In the above derivation, the lower bound is achieved exactly when the mislabeling rate is small and the independence assumption is true. We examine how tight the upper bound is by simulation. We have taken pairs of classes from UCI's Opt-Digit dataset, which is a handwritten digit recognition dataset, and trained both a naive Bayes classifier and a nearest-neighbor classifier [Duda *et al.*, 2001] on the training set of each pair. The nearest-neighbor classifier is then used to simulate a labeler and labeled the testing set. The naive Bayes classifier is the classifier under evaluation. Since we have the actual labels for the testing set, both the mislabeling rate (of the nearest-neighbor "labeler") and the true error rate (of the naive Bayes classifier) can be determined. The output of the naive Bayes classifier and the nearest-neighbor "labeler"are compared to determine the apparent error rate.

We have chosen the Opt-Digit dataset and the nearest-neighbor algorithm because we know that this combination can give very low error rate [Kaynak and Alpaydin, 2000], thus closely matching the accuracy of many human labelers. In fact, for most pairs of classes, the nearest-neighbor algorithm has zero error. The Opt-Digit dataset is also interesting because the handwritten digit recognition task is a classical example in which much human labeling effort has been applied. The naive Bayes classifier is chosen because it is a popular classifier and is sufficiently different from nearest-neighbor to give interesting results.

Table 3 shows the results for some pairs of classes where the nearest-neighbor "labeler" has non-zero error. Note that an insignificant positive dependence between the naive Bayes classifier and the nearest-neighbor "labeler" should be expected since they both are trained from the same dataset, use the same features, and assume independence of those features (explicitly in naive Bayes and implicitly in nearest-neighbor through its distance metric), even though they are different in other aspects (e.g., naive Bayes is generative while nearest-neighbor classifier is discriminative). The naive Bayes classifier's true error rate is almost exactly the upper bound for the pairs (1,2) and (4,5), but it is much closer to the apparent error rate for the pairs (7,8) and (8,9). The simulation thus shows the upper bound to be tight in some non-trivial situations.

7 Discussion and Future Work

When designing classification systems there are frequently parameters that are not learned automatically from the train-

	classes to discriminate ($\omega_1 - \omega_2$)			
	$1-2$	$4-5$	$7-8$	$8-9$
Upper bound (apparent error rate + mislabeling rate)	9.20%	1.11%	2.27%	6.77%
True error rate (Naive Bayes)	9.19%	1.10%	1.70%	5.65%
Apparent error rate	8.64%	0.83%	1.70%	5.08%
Lower bound (apparent error rate - mislabeling rate)	8.08%	0.55%	1.13%	3.39%
Mislabeling rate (Nearest Neighbor)	0.56%	0.28%	0.57%	1.69%

Table 3: Error rates on pairs of digits from the UCI Opt-Digit dataset. The classifier under evaluation is a Naive Bayes classifier (NB), and a nearest-neighbor classifier (NN) is used to simulate the (human) labeler. The error rate of the NB and NN classifiers are considered to be the true error rate and mislabeling rate, respectively. The fraction of time the two classifiers disagree is the apparent error rate. The upper and lower bounds are derived in Sec. 6, which are simply the apparent error rate plus or minus the mislabeling rate. The true error rate does come very close to the upper bound in some cases ($1-2$ and $4-5$).

ing data. Some examples are the number of hidden units in a feedforward neural network, the number k in a k-nearest-neighbor classifier, and the window width in a Parzen window classifier [Duda *et al.*, 2001]. *Validation* is one technique to estimate those parameters. In validation, one conceptually creates several classifiers with different values of the parameter and train them with the same training set. The trained classifiers are evaluated on the *validation* data set, and the best classifier is chosen. Our results for testing with noisy labels is directly applicable to validation. In fact, validation is not concerned with the actual value of the true error rates, but just their ordering. Therefore the *apparent error rate* $\Pr[\hat{y} \neq \tilde{y}]$ can work just as well, as long as the mislabeling rate $\Pr[y \neq \tilde{y}]$ is less than 0.5.

In a world where (unlabeled) data is cheap, noisy labels are easily obtained (e.g., the Open Mind Initiative [Stork, 1999; Stork and Lam, 2000]), but perfect labels are expensive, the findings in this paper allow one to confidently use noisy labels for testing and validating. An obvious area for future work is to use noisy labels for training as well. Although some works do allow for training with noisy labels, this has not been an active research area [Szummer and Jaakkola, 2000].

So far in our derivations we have assume either knowledge of the mislabeling rate $\Pr[y \neq \tilde{y}]$ or the labeler's confusion matrix $\mathbf{P}_{\tilde{y}|y}$. In practice those information must be estimated and be treated as random. The effects of such estimates and the cost/benefit analysis of obtaining more accurate estimates are unknown. We hope to investigate them in the future.

8 Conclusion

Traditionally test data have been assumed to be perfectly labeled. Increasingly this assumption is becoming a burden. We advocate the use of noisy labels as a cheaper alternative. We have shown that, under the assumption in which the labeler and the classifier make mistakes independently, the true error rate and true confusion matrix can be derived exactly. We have also examined the number of noisy labels to achieve the equivalent estimation confidence as one perfect label, and we found that number to be less than four in many practical situations. Furthermore, if we loosen the independence assumption to the non-negative dependence assumption, the true error rate can be bounded to be between the apparent error rate plus or minus the mislabeling rate.

References

[Blum and Mitchell, 1998] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT '98)*, pages 92–100, 1998.

[Duda *et al.*, 2001] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. John Wiley & Sons, 2nd edition, 2001.

[Eskin, 2000] Eleazar Eskin. Detecting errors within a corpus using anomaly detection. In *Proceedings of the First Conference of the North American Association for Computational Linguistics (NAACL-2000)*, 2000.

[Frakes and Baeza-Yates, 1992] William B. Frakes and Ricardo Baeza-Yates, editors. *Information Retrieval: Data Structures & Algorithms*. Prentice Hall, 1992.

[Kaynak and Alpaydin, 2000] Cenk Kaynak and Ethem Alpaydin. Multistage cascading of multiple classifiers: One man's noise is another man's data. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML)*, pages 455–462, 2000.

[Nigam *et al.*, 2000] Kamal Nigam, Andrew McCallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled examples. *Machine Learning*, 39(2):103–134, 2000.

[Smyth, 1997] Padhraic Smyth. Bounds on the mean classification error rate of multiple experts. *Pattern Recognition Letters*, 1997.

[Stork and Lam, 2000] David G. Stork and Chuck P. Lam. Open Mind *Animals*: Ensuring the quality of data openly contributed over the world wide web. In *AAAI Workshop on Learning with Imbalanced Data Sets*, pages 4–9, 2000.

[Stork, 1999] David G. Stork. The Open Mind Initiative. *IEEE Intelligent Systems & Their Applications*, 14(3):19–20, 1999.

[Szummer and Jaakkola, 2000] Martin Szummer and Tommi Jaakkola. Kernel expansions with unlabeled examples. In *Advances in Neural Information Processing Systems (NIPS)*, 2000.