Unlabeled Data Can Degrade Classification Performance of Generative Classifiers

Fabio G. Cozman, Ira Cohen
Internet Systems and Storage Laboratory
HP Laboratories Palo Alto

HPL-2001-234
September 28th , 2001*

semi-supervised learning, labeled and unlabeled data problem, classification, maximum-likelihood estimation, EM algorithm

This report analyzes the effect of unlabeled training data in generative classifiers. We are interested in classification performance when unlabeled data are added to an existing pool of labeled data. We show that there are situations where unlabeled data can *degrade* the performance of a classifier. We present an analysis of these situations and explain several seemingly disparate results in the literature.

* Internal Accession Date Only
Approved for External Publication

Unlabeled Data Can Degrade Classification Performance of Generative Classifiers

Fabio G. Cozman[*]
Escola Politécnica, Universidade de São Paulo
Av. Prof. Mello Moraes, 2231 - 05508-900
São Paulo, SP - Brazil

Ira Cohen[†]
Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304

September 26, 2001

Abstract

This reports analyzes the effect of unlabeled training data in generative classifiers. We are interested in classification performance when unlabeled data are added to an existing pool of labeled data. We show that there are situations where unlabeled data can *degrade* the performance of a classifier. We present an analysis of these situations and explain several seemingly disparate results in the literature.

1 Introduction

The purpose of this report is to discuss the performance of generative classifiers that are built with labeled and unlabeled records. For the most part we assume that classifiers are obtained using maximum likelihood estimation.

We show that there are cases where unlabeled data can *degrade* the performance of a classifier. Our analysis clarifies several seemingly disparate results that have been reported in the literature, and also explains existing but unpublished experiments in the field.

We review the technical aspects of the *labeled-unlabeled data problem* and present a summary of current results regarding this problem in Sections 2, 3 and 4. Existing empirical results display conflicting evidence on the value of unlabeled data. In Section 5, we discuss extensive tests that we conducted to investigate the behavior of classifiers in the presence of unlabeled data. We then present a mathematical analysis of the labeled-unlabeled data problem, and demonstrate how unlabeled data can sometimes improve and sometimes degrade classification performance (Section 6).

[*] This work was conducted while the first author was with the Internet Systems and Storage Laboratory, Hewlett-Packard Laboratories Palo Alto.

[†] Mailing address: The Beckman Institute, 405 N. Mathews Ave., Urbana, IL 61801.

2 The labeled-unlabeled data problem

Our goal is to label an incoming vector of *features* $\mathbf{X}$. Each instantiation of $\mathbf{X}$ is a *record*, and we assume that we have a database of previously observed records. Some of the records in the database are labeled, and some are unlabeled. We assume that there exists a *class variable* C. The possible values of C are the labels. We focus on a binary situation where we have labels c_0 and c_1; this is done merely to simplify notation but all results carry unchanged to situations with arbitrary number of labels.

We must build a *classifier* that receives a record $\mathbf{x}$ and generates a label $\hat{c}(\mathbf{x})$ for the record. Readers who are familiar with this topic may skip the remainder of this section.

Given a record $\mathbf{x}$, our goal is to label $\mathbf{x}$ so as to minimize the *classification risk* [14]:

$$\begin{cases} r_1 P(C = c_1|\mathbf{X} = \mathbf{x}) & \text{if } \hat{c}(\mathbf{x}) \text{ is } c_0, \\ r_0(1 - P(C = c_1|\mathbf{X} = \mathbf{x})) & \text{if } \hat{c}(\mathbf{x}) \text{ is } c_1, \end{cases}$$

where r_i is the missclassification loss when choosing $\hat{c}(\mathbf{x})$ incorrectly. We assume that r_0 and r_1 are equal; our results do not change substantially if we remove this assumption.

If we knew exactly the joint distribution $p(C, \mathbf{X})$, we could design the optimal *classification rule* to label an incoming record $\mathbf{x}$:

$$\hat{c}(\mathbf{x}) \text{ is } \begin{cases} c_1 & \text{if } P(C = c_1|\mathbf{X} = \mathbf{x}) \geq 1/2, \\ c_0 & \text{otherwise.} \end{cases} \tag{1}$$

Instead of storing the whole joint distribution $p(C, \mathbf{X})$, we could simply store the posterior distribution $p(C|\mathbf{X})$. This strategy is usually termed a *diagnostic* one (for example, diagnostic procedures are often used to "train" neural networks). In a statistical setting, diagnostic procedures may be cumbersome as they require a great number of parameters — essentially the same number of probability values as required to specify the joint distribution $p(C, \mathbf{X})$.

An alternative strategy is to store the class distribution $p(C)$ and the conditional distributions $p(\mathbf{X}|C)$ and then, as we observe $\mathbf{x}$, compute $p(C|\mathbf{X} = \mathbf{x})$ using Bayes rule. This strategy is usually called *generative*. An advantage of generative methods is that unlabeled data do relate to some portions of the model (namely, the *marginal* distribution $p(\mathbf{X})$). If instead we focus solely on $p(C|\mathbf{X})$, there is no obvious and principled way to handle unlabeled data [8, 24, 26]. For this reason, we employ generative schemes in this paper, and leave other approaches for future work.

Normally we will divide our database of previously recorded data in two parts: the *training data* and the *testing data*. First we build a classifier based on the training data. We use the testing data to measure *classification error* (the fraction of incorrect classifications). The best achievable classification error for a problem is called the *Bayes rate*, and it is a property of the problem.

To build a classifier, we normally choose the structure of the classifier and estimate the parameters of the classifier. By *structure* we mean the set of constraints that must be satisfied by the numerical parameters of the classifier. For example, we can assume a fixed number of labels or impose independence relations between features conditional on the class variable. In this paper we focus on parameter estimation under fixed structure. In particular, we assume that all variables (class and features) have a specified and fixed number of values.

Once we fix the structure of a classifier, we must estimate the joint distribution $p(C, \mathbf{X})$. We focus on *maximum-likelihood* estimates, where we choose probability values that maximize the *likelihood* of the training data. If we have training data divided in N_l labeled records and N_u unlabeled records, we have the likelihood:

$$\left(\prod_{i=1}^{N_l} p(\mathbf{x}_i|c_i)\, p(c_i)\right) \left(\prod_{j=1}^{N_u} p(\mathbf{x}_j)\right),$$

which is a function of the probability values themselves, as these values are not fixed in advance. If all training records are labeled, then maximum likelihood estimates can be produced in closed-form for discrete and Gaussian features. There is no general closed-form solution for maximizing likelihood in the presence of unlabeled records. More generally, there is no closed-form solution for maximizing likelihood when we have missing labels or missing features in the training data. Then we must resort to numerical methods for maximizing likelihood. One of the most popular methods is the Expectation-Maximization algorithm (EM) [3, 11]. We have used the EM algorithm in our experiments, as reported in Section 5.

The fact that parameters must be estimated to obtain a classifier leads to two types of error: bias and variance. For a parameter p, the estimation error is usually measured as $E\left[(p-\hat{p})^2\right]$, where $E[\cdot]$ denotes expected value and $\hat{p}$ is the estimator of p. The following decomposition is immediate:

$$E\left[(p-\hat{p})^2\right] = (p - E[\hat{p}])^2 + E\left[(\hat{p} - E[\hat{p}])^2\right].$$

The second term in the right hand side is the variance of $\hat{p}$. It is usually the case that by increasing the number of records used by an estimator, the variance of the estimator decreases. The first term in the right hand side is the square of the *bias*, and it measures the “systematic” error in trying to approximate p with $\hat{p}$. If we add more degrees of freedom to an estimator, we may reduce the bias (more freedom for $\hat{p}$ to approximate p), but the variance of the estimator may then increase for a fixed number of training records. Thus we have a bias-variance trade-off in the design of classifiers.

Classification performance should improve as we have more features — presumably, the more features we have, the more information we can infer about labels. As we add features to our classifier, we may have an increasing number of parameters, an increase on estimator variance, and an eventual degradation in performance (a fact referred to as the *Hughes phenomenon* [25]).

The distinction between classification error and estimation error is important, as a classifier may offer an inaccurate representation for the joint distribution $p(C, \mathbf{X})$, and yet have low classification error. Classification performance is directly affected by the boundary (in feature space) that separates labels. A classification boundary may or may not be close to the optimal boundary defined by (1), regardless of how accurate the probability values are estimated. Friedman uses a Gaussian approximation to show that classification error decreases when the following expression is positive, and increases when the expression is negative [14]:

$$\operatorname{sign}\left(P(C=c_1|\mathbf{x}) - 1/2\right) \frac{E\left[\hat{P}(C=c_1|\mathbf{x})\right] - 1/2}{\sqrt{V\left[\hat{P}(C=c_1|\mathbf{x})\right]}}, \tag{2}$$

where $V[\cdot]$ denotes variance. The variance of the estimator may be small and yet the probability of error may be large if $(P(C=c_1|\mathbf{x}) - 1/2)$ and $\left(E\left[\hat{P}(C=c_1|\mathbf{x})\right] - 1/2\right)$ have different signs.

3 Existing theoretical results for the labeled-unlabeled data problem

Classification problems are usually divided into *supervised* ones (where all training data are labeled) and *unsupervised* ones (where all training data are unlabeled) [13]. The labeled-unlabeled data problem is a combination of both supervised and unsupervised problems. At first we may reason that unlabeled data must always help, as unsupervised problems *can* be solved with unlabeled data alone. We may also reason that more data should normally reduce the variance of estimators and consequently reduce estimation error. Also, it is part of statistical folklore that freely available data always increase expected utility in decision-making [18].

Suppose that we have a classifier with the "correct" structure; that is, the structure of the classifier is identical to the structure that generates training and testing data. Early work has proved that unlabeled data can lead to improved maximum likelihood estimates even in finite sample cases [7]. Also, Shahshahani and Landgrebe emphasize the variance reduction caused by unlabeled data under the assumption that bias is zero; their conclusion is that unlabeled data must help classification [25]. A similar conclusion is reached by Zhang and Oles [26]. In general, unlabeled data can help in providing information for the marginal distribution $p(\mathbf{X})$ (a formal analysis of this argument is given by Cohen et al [8]). Overall, the message of previous work is that unlabeled data must help as long as structure is correct.

Castelli and Cover have investigated the value of unlabeled data in an asymptotic sense, with the assumption that the number of unlabeled records goes to infinity (and do so faster than the number of labeled records) [5, 6, 7]. Under the additional assumption of identifiability, unlabeled data alone are enough to estimate the shape of the marginal distribution for $\mathbf{X}$ [16], and labeled records are trivially necessary to label the decision regions. Castelli and Cover prove that, under various assumptions, classification error decreases exponentially with the number of labeled records, and linearly with the number of unlabeled records. Ratsaby and Venkatesh describe similar results for the particular case of Gaussian features [22]. These results again assume that estimators can replicate the "correct" structure that generated the training data.[1]

4 Existing empirical results for the labeled-unlabeled data problem

In the last few years, several empirical investigations have suggested that unlabeled training data do improve classification performance. Shahshahani and Landgrebe describe classification improvements with spectral data [25]; Mitchell and co-workers report a number of approaches to extract valuable information from unlabeled data, from variations of maximum likelihood estimation [21] to co-training algorithms [20]. Other publications report on EM-like algorithms [1, 4, 19] and co-training approaches [9, 10, 17]. There have been several workshops on the labeled-unlabeled data problem (workshops at NIPS1998, NIPS1999, NIPS2000 and IJCAI2001).

Overall, these publications and meetings advance an optimistic view of the labeled-unlabeled data problem, where unlabeled data can be profitably used whenever available. A more detailed analysis of current results does reveal some puzzling phenomena concerning unlabeled data. In fact, even the

[1] Another aspect of Castelli and Cover's results is that they assume identifiability, a property that fails when features are discrete [13] — note that many classifiers are built just for this situation, and certainly fail identifiability. Lack of identifiability does not seem to be a crucial matter in the labeled-unlabeled problem, as we made extensive tests with discrete models and observed behavior consistent with Gaussian (identifiable) models.

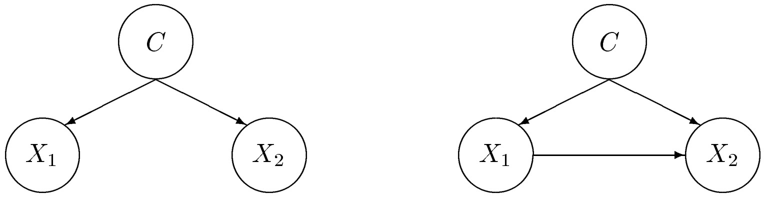

Figure 1: Classifiers with two features: Naive Bayes (left) and TAN (right).

last workshop on the labeled-unlabeled data problem, held during IJCAI2001, witnessed a great deal of discussion on whether unlabeled data are really useful.[2]

We now summarize three results in the literature that should suffice to illustrate the difficulties surrounding unlabeled data. The results we describe use *Naive Bayes* [12, 14] and *TAN* classifiers [15]. The basic assumption of a Naive Bayes classifier is that the joint distribution $p(C, \mathbf{X})$ is

$$p(C, \mathbf{X}) = p(C) \prod_{i=1}^{n} p(X_i | C) \, .$$

We can represent a Naive Bayes classifier by a graph, as depicted in Figure 1. One way to relax the strong independence assumptions in Naive Bayes classifiers is to admit that every feature depends on the class variable and also depend on another feature. The resulting classifier is called a *Tree-Augmented Network (TAN)* classifier [15]. Figure 1 shows a TAN classifier with a class variable and two features.

The following results, presented in chronological order, are of interest.

- Shahshahani and Landgrebe [25] focused on the use of unlabeled data to overcome the Hughes phenomenon (Section 2). They modeled features with Gaussian distributions and did not enforce independence relations among features, and they employed the EM algorithm for estimation. They succeeded in showing that it is possible to add features to a classifier and improve performance when a large number of unlabeled records is used to estimate parameters. It should be noted that, for a small number of features, the performance of their classifier was negatively affected by unlabeled data. They suggest that this apparently strange fact (it contradicts their own theoretic results) was due to deviations from assumed structure; for example, "outliers, . . . , and samples of unknown classes" — they even suggest that unlabeled records should be used with care, and only when the labeled data alone produce a poor classifier.

- Excellent classification results are reported by Baluja [1] using Naive Bayes and TAN classifiers. The classifiers were built from labeled and unlabeled data using EM. The use of unlabeled data generally improved performance, however this was not always true. When a relatively large number of labeled records were present and a Naive Bayes classifier was used, classification performance degraded with the addition of unlabeled records.

- In work aimed at classification documents, Nigam et al [21] used the EM algorithm to estimate parameters of Naive Bayes classifiers with fixed structure and a large number of features. Unlabeled data was treated as missing data in the EM algorithm. The paper describes situations where unlabeled records led to improved performance, but also describes situations where unlabeled records led to degraded performance (in the presence of a large number of labeled records,

[2] This fact was communicated to us by Georges Forman.

> consistently with the results reported by Baluja [1]). In one situation, adding a small number of unlabeled records to a small number of labeled records definitely degraded performance, but adding a larger number of unlabeled records led to substantial improvement. Nigam et al do not attempt to completely explain the reasons for these observations, but suggest that the problem might have been a mismatch between the natural clusters in feature space and the actual labels; they speculate that the fact that they used a large number of features even worsened this mismatch. Overall, their conclusion is that "unlabeled data can significantly increase performance" when properly handled.

This brief summary of previous research raises some questions. Are unlabeled data really useful? Can unlabeled data degrade performance, and if so, how, and why?

5 Experiments with labeled and unlabeled data

Intrigued by the existing results discussed in the previous section, we conducted a series of experiments aimed at understading the value of unlabeled data.

In all experiments, we generated training data from a structure with randomly chosen parameters, and then estimated the parameters of a classifier using the EM algorithm. We used simple structures and simple classifiers, as our goal was to understand the behavior of unlabeled data in controled circumstances. Every classifier was tested with 50000 labeled records drawn from the "correct" model. A complete description of our experiments is available elsewhere [8]; here we just summarize the main points.

We generated two sets of structures, one from structures that follow the Naive Bayes assumptions, another from structures that follow the TAN assumptions. For the latter structures, we added edges from feature X_i to feature X_{i+1}, for $i > 1$ (Figure 1 shows one such structure). We generated structures with 3 to 10 features; for each structure, we observed how a classifier *with the same structure* would recover the model. We considered estimation with 30, 300 and 3000 labeled records, and for each one of these situations, with 0, 30, 300, 3000 and 30000 unlabeled records. Figure 2 shows the result of learning a Naive Bayes classifier when the data was generated by a Naive Bayes structure, and similarly for a TAN classifier. Each point in these graphs is an average of ten trials; each graph in Figure 2 summarizes 150 trials. In this particular problem, estimation was relatively easy so the classification error is only slightly affected by unlabeled data when we already have 300 or more labeled records. We consistently observed that, when classifiers have the correct structure, unlabeled data improve classification on average. We also observed that more labeled data is always better for classification performance.

We then tried to estimate parameters for Naive Bayes classifiers with the data generated from the TAN structures. Here we consistently observed that *more unlabeled data degraded classification performance.* Figure 3 shows a typical graph. Note that performance degrades abysmally when we add 30000 unlabeled records to 30 labeled records. To avoid the possibility that this behavior was an artifact of the EM algorithm, we run a series of Gibbs sampling tests and obtained similar results. In all tests, we always started the EM algorithm with the estimates obtained using labeled data — consequently, the estimates are always better (in terms of likelihood) for the unlabeled data than for the labeled data alone. Despite that, we observe these drops in classification performance.

At this point it is convenient to stop and reflect upon the facts we have presented so far. Firstly, we have theoretical results that guarantee that more labeled data and more unlabeled data help classification when the structure is correct, and we observe this empirically. Secondly, we observe empirically

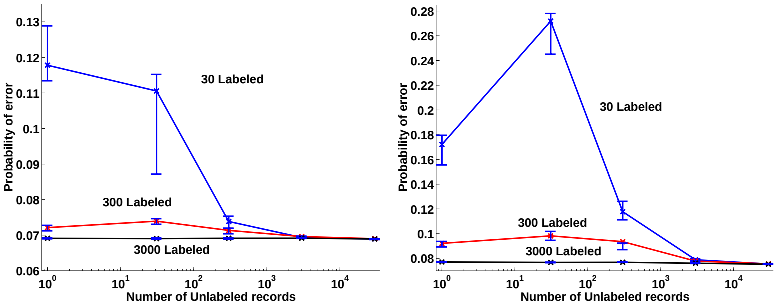

Figure 2: Examples: estimating parameters for a Naive Bayes classifier from data generated from a Naive Bayes structure with 10 features (left), and estimating parameters for a TAN classifier from data generated from a TAN structure with 10 features (right). Bars cover 30 to 70 percentiles.

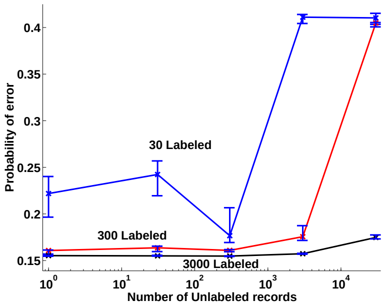

Figure 3: Example: estimating parameters for a Naive Bayes classifier from data generated from a TAN structure with 10 features. Bars cover 30 to 70 percentiles.

that more labeled data help classification when the structure is incorrect. Thirdly, we observe empirically that more unlabeled data may degrade classification when the structure is incorrect. There is no coherent explanation for these observations in current literature. Existing analyses suggest that more training data lead to less variance and less estimation error — and presumably to better classification.

Shahshahani and Landgrebe, and Nigam et al suggest that there might be mismatches between independence assumptions, or presence of outliers, in cases where performance is degraded by unlabeled data. One natural observation is then, if modeling errors degrade classification with unlabeled data, they would seem to degrade classification with labeled data as well — why would these different types of data have different effects? Also, how can we explain that we have cases, as reported by Nigam et al, where adding a few unlabeled records degraded peformance, and adding more unlabeled records led to better performance? The interaction between training data and modeling errors surely require a more detailed analysis.

6 An analysis of classification performance in the labeled-unlabeled data problem

In this section we discuss the effect of unlabeled data to classification error and show how to reconcile empirical results with theoretical analysis. Instead of studying classification error directly, we first show how to explain the performance degradation presented previously, and then why this degradation occurs with unlabeled data.

6.1 How

We propose a new strategy for graphing performance in the labeled-unlabeled data problem. Instead of fixing the number of labeled records and varying the number of unlabeled records, we propose to fix the *percentage* of unlabeled records among all training records. We then plot classification error against the number of training records. Call such a graph a *LU-graph*. It may not be clear at this point why LU-graphs are appropriate visualization tools, so we discuss LU-graphs in an example.

Example 1 Consider a situation where we have a class variable C with labels c_0 and c_1, and probability $p(c_0) = 0.4017$. We have two features X_1 and X_2. The features are real valued with distributions:

$$p(X_1|c_0) = N(2,1),\ p(X_1|c_1) = N(3,1),\ p(X_2|c_0, x_1) = N(2,1),\ p(X_2|c_1, x_1) = N(1 + 2x_1, 1),$$

where $N(\mu, \sigma^2)$ denotes a Gaussian distribution with mean μ and variance σ^2.

Note that there is dependency between X_2 and X_1 (X_2 depends on X_1 when $C = c_1$). Note that this problem is identifiable, and it is the simplest possible departure from the Naive Bayes assumptions. Figure 4 shows a contour plot of the joint density for X_1 and X_2; the figure also shows the optimal classification boundary. The optimal classification rule is to choose c_0 if $\{x_1, x_2\}$ lies below the boundary, and c_1 otherwise.

Suppose we build a Naive Bayes classifier for this problem. Consider now a series of LU-graphs for this problem. Figure 5 shows LU-graphs for 0% unlabeled records, 50% unlabeled records and 99% unlabeled records. For each graph in the figure, we produced points for total numbers of records equal to 50, 100, 500, 1000, 5000, 10000 and 50000. Each point in each graph is the average of 100 trials; classification error was obtained by testing in 10000 labeled records drawn from the "correct" model.

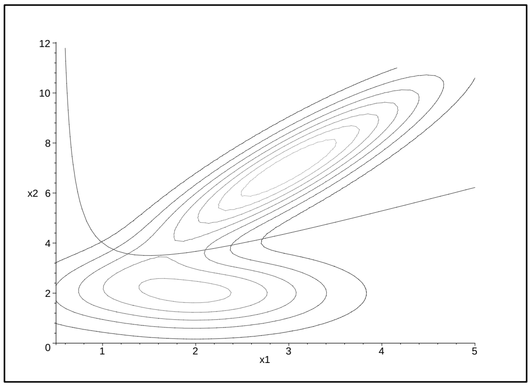

Figure 4: Classification with two Gaussian features.

The figure also shows two additional graphs containing classification performance when we *discard* the unlabeled data and use only the labeled data. We should expect all graphs with just labeled data to converge to the same classification error in the limit. That must happen because the estimates are eventually the same; it just takes longer to reach low classification error when we are discarding 99% of the data.

The LU-graphs for 50% and 99% unlabeled data have an interesting property: their asymptotes do not converge to the same value, and they are both different from the asymptotes for labeled data. Suppose then that we started with 50 labeled records as our training data. Our classification error would be about 7.8%, as we can see in the LU-graph for 0% unlabeled data. Suppose we added 100 labeled records, and we reduced classification error to about 7.2%. Now suppose we added 100 *unlabeled* records. We would move from the 0% LU-graph to the 50% LU-graph. Classification error would increase to 8.2%! And if we then added 9800 unlabeled records, we would move to the 99% LU-graph, with classification error about 16.5% — more than twice the error we had with just 50 labeled records.

The fact that classification error has different asymptotes, for different levels of unlabeled data, leads to possible degradation of classification performance. Note that it is possible to have incorrect structure in the classifier and still for unlabeled data to help — it is enough that we move from one rapidly decreasing LU-graph to another decreasing LU-graph, and the rate of decrease in the graphs is larger than the degradation caused by unlabeled data. These considerations indicate that there are interactions between the Bayes rate of a problem (how hard the problem is), the number of features used in the problem (how many parameters specify the classifier) and the difference between "correct" and "assumed" structure. In a difficult problem with many features, we may need a large amount of data to reach a low Bayes rate; in these cases we can benefit from unlabeled data (to win over the Hughes phenomenon) even if classifier structure is incorrect. Examples discussed by Nigam et al [21] seem to fit this description exactly — while Nigam et al speculate that more features could cause unlabeled data to misbehave, in fact difficult classification problems with more features should profit more consistently from unlabeled data. This observation agrees with the empirical findings of Shahshahani and Landgrebe [25], as they observed that unlabeled data degraded performance in the presence of a small number of features, and unlabeled data improved performance in the presence of a large number of features. The LU-graphs for a particular problem are a useful tool to determine how unlabeled data affects classification performance.

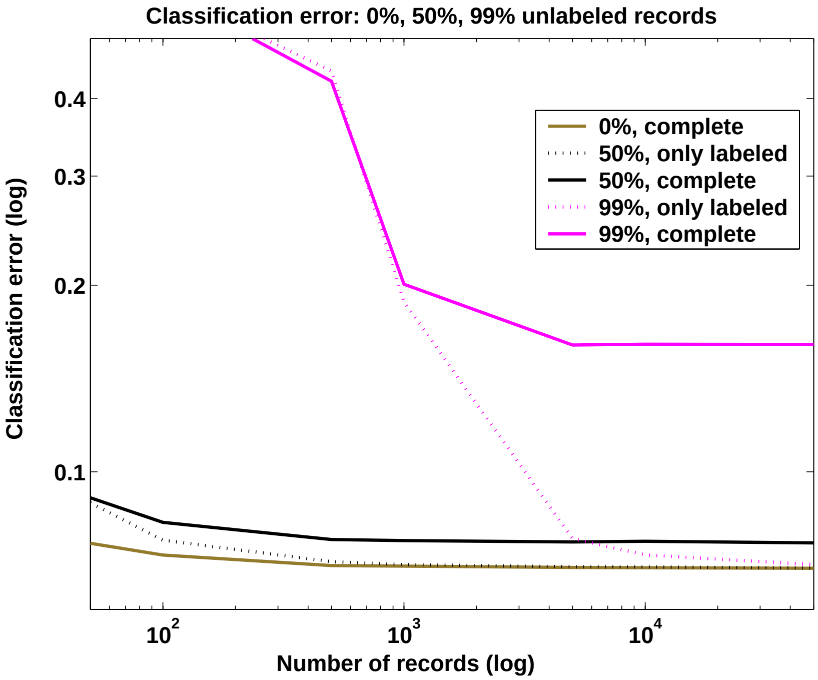

Figure 5: LU-graphs for the example with two Gaussian features.

6.2 Why

With the help of LU-graphs we can visualize the effect of unlabeled data in classification performance. Why are unlabeled data the source of asymptotic differences between LU-graphs?

To proceed with the analysis, assume that we have an infinitely large number of labeled records. Taking the number of unlabeled records to infinity simplifies the problem because we can look at the estimation problem as one of function approximation. In doing this, we are inspired by the strategy in Castelli and Cover's work [5, 6, 7].

Assuming identifiability, we can estimate a complete classifier from an infinite amount of unlabeled data. If we have the correct structure for this classifier, we obtain the exact values of $p(\mathbf{X})$ without bias. If we have incorrect structure for the classifier, we can only estimate a function $g(\mathbf{X})$ that approximates $p(\mathbf{X})$. The fact that $g(\mathbf{X})$ is the "best" possible for estimation does not mean that $g(\mathbf{X})$ leads to the best classification boundary.

Basically, the fact that estimation error is the guiding factor in building a classifier leads us to use estimates that are not optimal with respect to classification error. This seemingly innocuous fact works in subtle ways, as can be seen analyzing LU-graphs. Note that $g(\mathbf{X})$ cannot be equal to $p(\mathbf{X})$ by assumption, so we cannot obtain the optimal classification boundary just with $g(\mathbf{X})$. If we had labeled records, we could alter the classification boundary so as to make it closer to the optimal boundary — we could "damage" the estimate of $p(\mathbf{X})$ so as to obtain a better classification boundary. When we have no labeled record, we cannot affect the boundary, so we obtain a biased boundary with $g(\mathbf{X})$. As we add labeled records to a pool of unlabeled records, we are moving the classification boundary in the direction of the optimal one, even as we move the estimates away from $g(\mathbf{X})$.

These comments are not restricted to maximum likelihood estimation, nor they depend on identi-

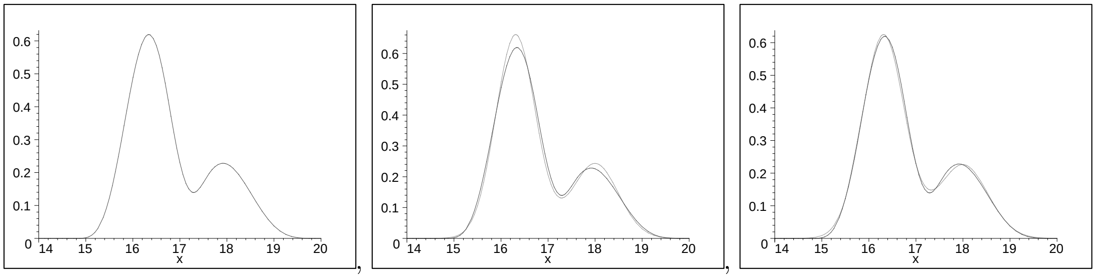

Figure 6: A mixture of Beta distributions $p(X)$ (left); comparison between mixture of Beta distributions and mixture of Gaussian distributions $g(X)$ estimated from labeled data (middle); comparison between mixture of Beta distributions and mixture of Gaussian distribution $h(X)$ estimated from unlabeled data (right).

fiability; the central fact is that we use one criterion to judge estimation, and a different one to judge classification. The following example shows how the effort to reduce estimation error may lead to different estimates when we use different types of training data.

Example 2 Suppose we have a binary class C, a single real-valued feature X, and an infinite amount of training data. We have $p(C = c_0) = 0.3$ and X follows Beta distributions conditional on C:

$$p(X) = 0.3\frac{0.3(0.3X-5)^3(1-(0.3X-5))^5}{\mathrm{Beta}(4,6)} + 0.7\frac{0.34(0.34X-5)^5(1-(0.34X-5))^4}{\mathrm{Beta}(6,5)}.$$

Figure 6 depicts this mixture distribution. For classification, the classification boundary is crucial (the boundary is the value of X for which $p(X, C = c_0) = p(X, C = c_1)$). For $p(X)$, the boundary is defined by $X_o = 17.19053765$. Suppose we are informed about the exact value of $p(C = c_0)$ and also we obtain the exact means for the components of this mixtures (first component has mean 18 and second component has mean 16.31016043), and suppose we take the incorrect assumption that X is Gaussian. Now, if we have completely labeled data, we can estimate the variances of each component with some consistent estimator, and obtain 0.2424242 for the first component and 0.1787296 for the second component. Figure 6 depicts the resulting Gaussian mixture $g(X)$. For $g(X)$, the classification boundary is defined by $X_l = 17.21261916$. Now suppose that training data are unlabeled. We cannot hope to recover the labels (they are not specified), but we can hope to recover the classification boundary — that is, we can distinguish between c_0 and c_1 even if we do not know which features should be labeled with c_0 and otherwise. We can use the fact that the form of the mixture distribution is known exactly for infinitely many training data, and we can approximate $p(X)$ with a mixture of Gaussian distributions using least-squares (unfortunately we cannot obtain closed-form maximum likelihood estimates in this case). We choose the variances so as to minimize the squared error $\int_{-\infty}^{\infty}(p(x) - h(x))^2 dx$, where $h(X)$ is the mixture of Gaussian distributions. By performing this minimization, we obtain 0.2809 for the variance of the first component and 0.200704 for the variance of the second component. Figure 6 depicts the resulting Gaussian mixture $h(X)$; note that $h(X)$ is quite close to $p(X)$ — closer to $p(X)$ than $g(X)$. The classification boundary for $h(X)$ is $X_u = 17.22483179$. Note that $X_o < X_l < X_u$; unlabeled data lead to a boundary that is strictly *worse* than the boundary produced by labeled data.

Classification error depends only on the estimates for $p(C|\mathbf{X})$ (Expression (2)); it is possible to have better overall estimates (with respect to likelihood) but still obtain worse estimates for $p(C|\mathbf{X})$ — some parameters in the classifier may have smaller estimation error while other critical parameters have larger estimation error. Because unlabeled data contains information only on the marginal distribution $p(\mathbf{X})$, unlabeled data may adversely affect estimates of some critical classifier parameters,

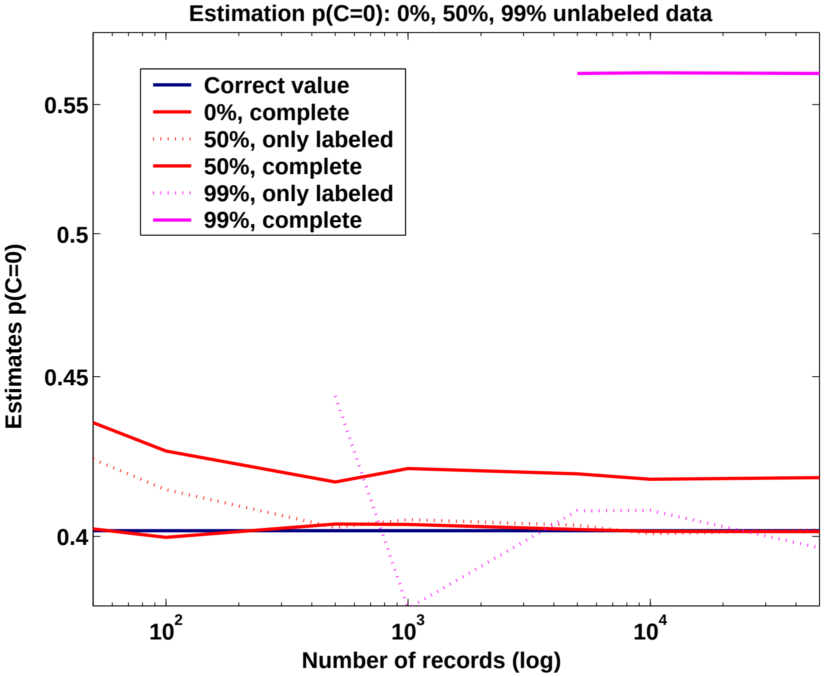

Figure 7: Graphs with estimates for $p(C = c_0)$ in the example with two Gaussian features.

even as unlabeled data reduce estimation error in other parameters. To illustrate these statements, consider the classification problem in Figure 4. The value of the parameter $p(C = c_0)$ can certainly be estimated perfectly with an infinite amount of labeled data — regardless of whether the conditional distributions $p(\mathbf{X}|C)$ have correct functional forms or not. If we have unlabeled training data, then we cannot guarantee that $p(C = c_0)$ has an unbiased estimate; results will depend on structural assumptions. If we have correct structure for $p(\mathbf{X}|C)$, we can still recover $p(C = c_0)$ without bias. Incorrect assumptions about structure can introduce bias into estimates of $p(C = c_0)$. Figure 7 shows estimates for $p(C = c_0)$ for a Naive Bayes classifier when data is generated from the distributions sketched in Figure 4. The graphs in Figure 7 are similar to LU-graphs, but they show estimates as we keep the percentage of unlabeled data constant. Each point in these graphs is the average of 100 trials. *Note that we always obtain unbiased estimates for class probabilities when we only use labeled records. Bias is introduced when we use unlabeled data.* The bias in $p(C)$ can certainly affect $p(C|\mathbf{X})$; an analysis of Expression (2) shows that bias in $p(C|\mathbf{X})$ can degrade classification performance even as variance is essentially zero.[3]

The preceeding discussion also indicates that unlabeled data are fundamentally different from missing feature values. Even though both forms of missing data degrade estimation performance, unlabeled data also affects classification performance directly by introducing bias in critical parameters. This insight clarifies several questions raised by Seeger on the value of unlabeled data [24].

[3] In fact, things are slightly more complicated in the presence of incorrect structure. There may exist a *set* of estimates that maximize likelihood; this set is called the *asymptotic carrier* by Berk [2]. We may experience variation on estimates inside the asymptotic carrier even as the number of training records goes to infinity.

7 Conclusion

The central message of this paper is that unlabeled training data can degrade classification performance if the classifier assumes an incorrect structure. Because in practice we can never be sure about structure, it is necessary to exercise caution when dealing with unlabeled data.

The current literature in the labeled-unlabeled data problem does not seem to be aware of the results reported in this paper. Even though there have been reports of performance degradation with unlabeled data, the explanations that have been offered suggest that degradation occurs in somewhat extreme circunstances. In this paper we show that this is not the case, and in fact problems with less features are more likely to show performance degradation with unlabeled training data. The type of degradation described here is a fundamental property of classification error. Of course, it is possible that additional sources of performance degradation can be found, particularly when there are severe mismatches between real and assumed structure.

Because unlabeled data is affected by classifier structure, we can use unlabeled data to help our search for a "correct" structure. Some of the work in the labeled-unlabeled data problem can be understood from this perspective; for example, Schuurmans and Southey suggest that unlabeled data should help to parameterize classifiers to prevent overfitting [23]. A different proposal is made by Cohen et al [8].

It certainly seems that some creativity must be exercised when dealing with unlabeled data. As discussed in the literature [24], currently there is no coherent strategy for handling unlabeled data with diagnostic classifiers, and generative classifiers are likely to suffer from the effects described in this paper. Future work should investigate whether unlabeled data can degrade performance in different classification approaches, such as decision trees and co-training. Hopefully, the results in this paper will provide a better foundation for algorithms dealing with the labeled-unlabeled data problem.

Acknowledgements

This work was conducted at HP Labs, Palo Alto; we thank Alex Bronstein and Marsha Duro for proposing the research on unlabeled data and for many suggestions and comments during the course of the work. Their help was critical to the results reported here.

We thank Vittorio Castelli for sending us an electronic copy of his PhD dissertation, Charles Elkan for sending us a copy of his BNB software, and George Forman for telling us about the IJCAI workshop on unlabeled data. We generated data for our tests using Matlab and the BNT system coded and distributed by Kevin Murphy; we thank Kevin Murphy for doing it. We coded our own Naive Bayes and TAN classifiers in the Java language, using the libraries of the JavaBayes system (freely available at http://www.cs.cmu.edu/˜javabayes).

References

[1] Shumeet Baluja. Probabilistic modeling for face orientation discrimination: Learning from labeled and unlabeled data. In *Neural and Information Processing Systems (NIPS)*, 1998.

[2] R. H. Berk. Limiting behavior of posterior distributions when the model is incorrect. *Annals of Mathematical Statistics*, pages 51–58, 1966.

[3] Jeff A. Bilmes. A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models. Technical report, U. C. Berkeley, Berkeley, California, United States, April 1998.

[4] Rebecca Bruce. Semi-supervised learning using prior probabilities and EM. In *IJCAI-01 Workshop on Text Learning: Beyond Supervision*, August 2001.

[5] Vittorio Castelli. *The Relative Value of Labeled and Unlabeled Samples in Pattern Recognition.* PhD thesis, Stanford University, December 1994.

[6] Vittorio Castelli and Thomas M. Cover. On the exponential value of labeled samples. *Pattern Recognition Letters*, 16:105–111, 1995.

[7] Vittorio Castelli and Thomas M. Cover. The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter. *IEEE Transactions on Information Theory*, 42(6):2102–2117, November 1996.

[8] Ira Cohen, Fabio Cozman, and Alex Bronstein. On the value of unlabeled data in supervised learning in maximum-likelihood. Technical report, HP labs, 2001.

[9] Michael Collins and Yoram Singer. Unupervised models for named entity classification. In *Proc. 17th International Conf. on Machine Learning*, pages 327–334. Morgan Kaufmann, San Francisco, CA, 2000.

[10] Francesco De Comité, François Denis, Rémi Gilleron, and Fabien Letouzey. Positive and unlabeled examples help learning. In O. Watanabe and T. Yokomori, editors, *Proc. of 10th International Conference on Algorithmic Learning Theory*, pages 219–230. Springer-Verlag, 1999.

[11] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal Royal Statistical Society B*, 44:1–38, 1977.

[12] Pedro Domingos and Michael J. Pazzani. On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning*, 29(2-3):103–130, 1997.

[13] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis.* John Wiley and Sons, New York, 1973.

[14] Jerome H. Friedman. On bias, variance, 0/1-loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, 1(1):55–77, 1997.

[15] Nir Friedman, Dan Geiger, and Moises Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29:131–163, 1997.

[16] S. Ganesalingam and G. J. McLachlan. The efficiency of a linear discriminant function based on unclassified initial samples. *Biometrika*, 65, December 1978.

[17] Sally Goldman and Yan Zhou. Enhancing supervised learning with unlabeled data. In *International Joint Conference on Machine Learning*, 2000.

[18] I. J. Good. *Good Thinking: The Foundations of Probability and its Applications.* University of Minnesota Press, Minneapolis, 1983.

[19] David J. Miller and Hasan S. Uyar. A mixture of experts classifier with learning based on both labelled and unlabelled data. In *Advances in Neural Information Processing Systems*, pages 571–577. 1996.

[20] Tom Mitchell. The role of unlabeled data in supervised learning. In *Proc. of the Sixth International Colloquium on Cognitive Science*, San Sebastian, Spain, 1999.

[21] Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39:103–144, 2000.

[22] Joel Ratsaby and Santosh S. Venkatesh. Learning from a mixture of labeled and unlabeled examples with parametric side information. In *COLT*, pages 412–417, 1995.

[23] Dale Schuurmans and Finnegan Southey. An adaptive regularization criterion for supervised learning. In *Proc. 17th International Conf. on Machine Learning*, pages 847–854. Morgan Kaufmann, San Francisco, CA, 2000.

[24] Matthias Seeger. Learning with labeled and unlabeled data. Technical report, Institute for Adaptive and Neural Computation, University of Edinburgh, Edinburgh, United Kingdom, February 2001.

[25] Behzad M. Shahshahani and David A. Landgrebe. The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon. *IEEE Transactions on Geoscience and Remote Sensing*, 32(5):1087–1095, 1994.

[26] Tong Zhang and Frank Oles. A probability analysis on the value of unlabeled data for classification problems. In *International Joint Conference on Machine Learning*, pages 1191–1198, 2000.