The effect of unlabeled data on generative classifiers, with application to model selection

Ira Cohen[1], Fabio G. Cozman[2], Alexandre Bronstein
Internet Systems and Storage Laboratory
HP Laboratories Palo Alto
HPL-2002-140
May 14th , 2002*

Email: {icohen, alexbr}@hpl.hp.com, fgcozman@poli.usp.br, iracohen@ifp.uiuc.edu

semi-supervised learning, labeled and unlabeled data problem, classification, machine learning

In this paper we investigate the effect of unlabeled data on generative classifiers in semi-supervised learning. We first characterize situations where unlabeled data cannot change estimates obtained with labeled data, and argue that such situations are unusual in practice. We then report on a large set of experiments involving labeled and unlabeled data, and demonstrate that unlabeled data can degrade classification performance when modeling assumptions are incorrect. To improve classification performance, we propose a method to switch assumed model structure based on the effect of unlabeled data.

* Internal Accession Date Only Approved for External Publication
[1] Beckman Institute, University of Illinois at Urbana Champaign, Urbana, IL
[2] Escola Politecnica, Universidade de Sao Paulo, Brazil

The effect of unlabeled data on generative classifiers, with application to model selection

Ira Cohen[1,3], Fabio G. Cozman[2], Alexandre Bronstein[1]

[1]Hewlett-Packard Laboratories, Palo Alto, CA, {icohen,alexbr}@hpl.hp.com

[2]Escola Politecnica, Universidade de Sao Paulo, Brazil, fgcozman@poli.usp.br

[3]Beckman Institute, University of Illinois at Urbana Champaign, iracohen@ifp.uiuc.edu

Abstract

In this paper we investigate the effect of unlabeled data on generative classifiers in semi-supervised learning. We first characterize situations where unlabeled data cannot change estimates obtained with labeled data, and argue that such situations are unusual in practice. We then report on a large set of experiments involving labeled and unlabeled data, and demonstrate that unlabeled data can degrade classification performance when modeling assumptions are incorrect. To improve classification performance, we propose a method to switch assumed model structure based on the effect of unlabeled data.

1 Introduction

Recently there has been a growing interest in the use of unlabeled data for enhancing classification accuracy in supervised learning settings. Most studies have shown that there is potential in using unlabeled data to enhance the learning process and improve classification [12]. Finding methods for adding unlabeled data to supervised learning is important because in many real world problems it is easy to collect unlabeled data, but expensive to correctly label it.

The task in classification is to predict the value of the class given a set of features (or attributes). We must find a mapping from feature space to labels. Statistical estimation methods are used to learn this mapping when the features and class variables are treated as random variables. A common method in classification is to assume a model believed to represent the joint probability distribution of the variables, estimate the model using a set of training data, and use the estimated model to classify new data. The model is a combination of structure and associated parameters. 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. When the assumed model of the classifier matches the model that generates the data, we say the assumed model is "correct".

For a given fixed structure, estimating the model's parameters can be done using a maximum likelihood (ML) approach — in this paper we focus on the ML approach.

We first discuss the method of maximizing the likelihood of both the labeled and unlabeled data. We first show how it is possible to predict when the addition of an unlabeled data set to a labeled data set does not change the estimates of a model. We argue that usually this would not occur, thus unlabeled data added to a labeled training set will usually change the estimates of a model.

We then look at the expected value of unlabeled data in classification problems. Intuitively, adding more data should improve classification performance because it reduces the variance of the model's estimator. There are proven results showing that unlabeled data improve classification performance when the assumed model is correct [13, 15]. However, we show that when it is not possible to estimate the model that generated the data with the assumed model, unlabeled data can degrade classification performance. We show with experiments on artificially generated data sets and on a real dataset (from the UCI repository) that unlabeled data can both improve and degrade classification performance, depending on different model assumptions and the size of the labeled and unlabeled training sets. We further provide a theoretical explanation for this phenomenon.

That understanding leads to a new use of unlabeled data, namely, in detecting incorrect modeling assumptions. We propose a general method for using unlabeled data to indicate the need to change modeling assumptions. As a first step towards implementation of the method, we look at the case of all discrete variables and changes in the structure of a model. We first present an expectation-maximization (EM) scheme for learning a Tree-Augmented-Naive network (TAN) structure from both labeled and unlabeled data. The EM scheme follows directly from Meila's MixTreeS algorithm [10]and we name the resulting scheme EM-TAN. The algorithm provides an efficient method of learning the best pair-wise dependencies between the features, allowing a richer structure than the simpler Naive-Bayes (NB) models, which are commonly used. We describe a method for switching between a Naive-Bayes model to a TAN model using the EM-TAN algorithm. We show that in cases where unlabeled data degrade the classification performance when learning a NB model, TAN models are more likely to improve classification with unlabeled data.

2 Problem statement

Let C be the class variable with a finite number of labels; the number of labels is denoted by $|C|$. Let X be the feature vector. We are given a dataset, D, of independent records sampled from some unknown joint distribution $P(C, X)$, with N labeled records and M unlabeled records. We assume throughout the paper that there are no missing data other than the labels.

We note the two possible paradigms for modeling the joint distribution when estimating a classifier, namely the generative and diagnostic paradigms. We focus on the generative paradigm for modeling the relationship between the class variable C and the feature vector X. In the generative paradigm, the a-priori probability $P(X|C)$ and prior probability $P(C)$ are estimated directly from the data. Classification of an example $X = d$ involves computing $P(C = c|X = d) = \frac{P(C=c)\cdot P(X=d|C=c)}{P(X=d)}$, using the estimated model. The diagnostic paradigm is not directly useful for dealing with unlabeled data [12, 15], so we adopt the generative paradigm in this paper.

In the next section and in all of the experiments in this report, unless stated otherwise, C and X are discrete, so all distributions are multinomial. The parameters for the classifiers are the probability values, denoted by the vector θ.

Usually θ is selected as the maximum of the likelihood function:

$$L(\theta) = \prod_{i=1}^{N} p(x_i|c_i, \theta)p(c_i|\theta) \prod_{j=N+1}^{N+M} \sum_{k=1}^{|C|} p(x_j|c_k, \theta)p(c_k|\theta), \tag{1}$$

where for notational convenience we index the labeled data as the first N records in D and the unlabeled data as the $N + 1$ to $N + M$ records in D.

We can write this function as the product of two functions:

$$L_l(\theta) = \prod_{i=1}^{N} p(x_i|c_i, \theta)p(c_i|\theta),$$

$$L_u(\theta) = \prod_{j=N+1}^{N+M} \sum_{k=1}^{|C|} p(x_j|c_k,\theta)p(c_k|\theta).$$

We define θ^*, θ_l^* and θ_u^* to be the $\arg\max_\theta$ of $L(\theta)$, $L_l(\theta)$ and $L_u(\theta)$ respectively.

3 Can unlabeled data be discarded? When?

Consider a method that can generate estimates from labeled and unlabeled data. The method can be called to produce estimates from an initial set of labeled data, and then to produce estimates from a mix of labeled and unlabeled data. We should expect the unlabeled data to affect the initial estimates, either positively or negatively (as discussed in the next section) — but can we predict those situations where unlabeled data do not affect the initial estimate? Note that this question focuses on a specific data set, not on expected behavior. The remainder of this section discusses this issue. Our first result proves that under certain conditions, for discrete variables, adding the unlabeled data to the labeled data does not change the estimates of the model. We then argue that in most practical cases these conditions will not be met, and therefore adding the unlabeled data should usually affect the estimates.

To obtain our result, it is instructive to look at $L_u(\theta)$ in a somewhat different form:

$$L_u(\theta) = p(x_{N+1},\ldots,x_{N+M}|\theta) = \prod_{j=N+1}^{N+M} p(x_j|\theta). \tag{2}$$

That is, $L_u(\theta)$ is the marginal of the unlabeled data for a given θ. The crucial observation is that there are no missing data if we want to estimate the probability values $p(X|\theta)$ from unlabeled data only. Hence θ_u^* should be easy to get.

Suppose that all of the features are discrete, the empirical distribution for X in the unlabeled data is:

$$f_u(X=x) = \frac{\#\text{ of times } \{X=x\} \text{ in unlabeled records}}{\#\text{ of unlabeled records}}.$$

Given a particular dataset D and a given joint probability distribution $P(C,X|\theta)$, we prove:

Theorem 1 *Assume that all the features in the vector X are discrete and there exists a θ such that $p(X|\theta) = f_u(X)$. Let $\theta_l^* = \arg\max_\theta L_l(\theta)$ and $\theta^* = \arg\max_\theta L(\theta)$. If for all values of X, $P(X|\theta_l^*) = f_u(X)$, then $\theta^* = \theta_l^*$.*

Proof. We know that if for some θ', $p(X|\theta') = f_u(X)$, then $\theta' = \arg\max_\theta L_u(\theta)$.
So for $\theta_l^* = f_u(x)$ we obtain $\theta_l^* = \arg\max_\theta L_u(\theta)$.
Now we can bound $\max L(\theta)$ from above using:

$$\begin{aligned} \max L(\theta) &= \max L_l(\theta)L_u(\theta) \\ &\leq (\max L_l(\theta))\,(\max L_u(\theta)) \\ &= L_l(\theta_l^*)L_u(\theta_l^*), \end{aligned}$$

and then clearly the way to maximize $L(\theta)$ is to take $\theta = \theta_l^*$. QED

So, if the empirical marginal of the unlabeled data is equal to $p(X|\theta_l^*)$, then the unlabeled data do not change the estimate of the model. Note that it is easy to compute θ_l^* for a given dataset by simple event counting, since the data is fully labeled and we assume no missing data for the features.

When labeled data are available in abundance, then θ_l^* should be enough to provide a good approximation to the empirical marginal, and then the value of unlabeled data is small. It is also important to note that the theorem relates to a specific

dataset. It is expected that for small size datasets, the condition that $p(X|\theta_l^*) = f_u(X)$ is unlikely to be be met, but is certain in the asymptotic case.

If modeling assumptions do not allow estimates of $p(X|\theta)$ to be equal to $f_u(X)$, the theorem above does not hold. The conclusion that we draw from this analysis is that for many practical cases, such as a relatively small labeled training set, the use of unlabeled data does change the distribution that maximizes the likelihood function. We also see that algorithms that maximize likelihood try to establish a balance between forcing θ to comply with the labeled data, while also approximating the empirical marginal provided by the unlabeled data. As the number of unlabeled records increases, the empirical marginal comes closer to the marginal that generated the data, and the variance of the estimator is expected to decrease.

4 Improving classification with unlabeled data

In this section, we are interested in studying when these changes in estimates improve the classification performance. In the next paragraphs, we summarize some of the results in the literature; a more detailed analysis of previous results has been published by the authors [5], but we include a summary for completeness.

We start by discussing some observations from recent studies on the semi-supervised learning problem that have used ML. Some studies reported an occasional increase in the classification error when adding the unlabeled data to the available labeled data [11, 1, 13]. Nigam et al. [11] show that adding unlabeled data sometimes degrades performance in text classification problems when using EM to learn a Naive-Bayes classifier. Baluja [1] reported a similar phenomenon for a face orientation classification problem. Both papers attribute the phenomenon to an unrealistic structure assumption. Baluja [1] then changes the structure to more complex structures, and observes that unlabeled data enhance the classification performance after the change. Shahshahani et al. [13] investigate the mitigation of the Hughes phenomenon by adding unlabeled data. The Hughes phenomenon states that after adding many features to a classifier which is trained using a fixed size training set, the classification performance starts to degrade. They show that adding unlabeled data to the training set can mitigate this phenomenon, allowing the addition of more features. However, in their results there are cases when adding the unlabeled data degraded the classification performance relative to that which used only the labeled data. They attribute the increased error either to structure or to outliers in the unlabeled data.

Seeger [12] also suggests that the use of unlabeled data can degrade the classifier's performance when using EM because of convergence to local maxima. While convergence to local maxima can degrade the classification performance since the estimated model's parameters are not optimal, it does not explain all of the cases reported in the studies. Furthermore, as the experiments in the following sections show, even when EM is initialized with a very good starting point, unlabeled data can still degrade the performance, thus the degraded performance cannot be attributed to convergence to a local maximum. We also would like to stress that the degraded classification performance with unlabeled data is not caused by differences in the underlying distribution of the labeled and unlabeled data. We discuss here a more fundamental case, where the labeled and unlabeled data are sampled from the same distribution.

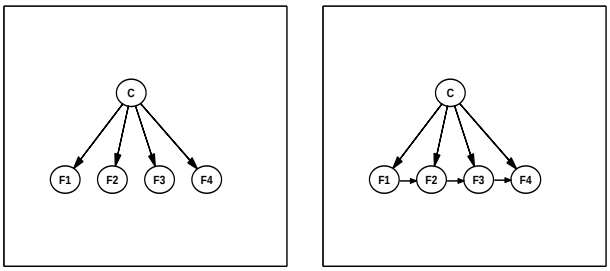
In the following discussion, we restrict ourselves to models that are different only by the structure, and not by other modeling assumptions, such as the type of variables and their distributions(e.g., discrete, Gaussian, etc.). However, the same conclusions carry over to the more general case [4]. For ease of notation, we refer to the structure of the model that generated the data as the "correct structure", and to any other structure as an "incorrect structure".

In the next sections, we describe our extensive experiments that show that an incorrect structure can cause unlabeled data to degrade the classifier's performance. We also show experimentally that unlabeled data do not degrade, and only improve the classification performance when the correct structure is assumed, as existing results suggest. We also show that unlabeled data can improve classification performance even when an incorrect structure is assumed. We link this improvement to models with high complexity with respect to the size of the labeled training set, which is similar to the analysis of Shahshahani et al. [13].

4.1 Experiments with artificial and real data

We generated datasets using two different model structures, Naive-Bayes (NB) and Tree-Augmented-Naive Bayes (TAN) [8], varying the number of features, the number of values per feature (all features are discrete) and the size of the datasets, with different proportions of labeled and unlabeled records in each set. The full description of the experiments, with their results are given in Appendix A.

In these experiments, we focus on two basic structures: Naive-Bayes (NB) and Tree-Augmented-Naive-Bayes (TAN) structures [8]. In the TAN structure, the class node has no parents and each feature has the class node as a parent and at most one other feature, such that the result is a tree structure for the features. examples of Naive-Bayes and TAN structures.



(a) Naive-Bayes structure (b) TAN structure

Figure 1: Example of a Naive-Bayes structure and a TAN structure.

We illustrate the main results of the experiments using three typical examples below.

Figure 2(a-c) shows an example of the probability of error graphs for the three types of tests we performed, where each point in the graph is an average of 10 trials. The graphs correspond to models with 10 features. Figure 2(a) corresponds to learning a NB structure when the correct structure is NB. Figure 2(b) is the result of estimating a TAN structure when the correct structure is TAN and Figure 2(c) is the result of estimating a NB structure when the correct structure is the TAN given in (b).

We see from Figures 2(a) and 2(b) that unlabeled data help significantly in reducing the classification error. We also see that the error is reduced as more unlabeled data is added. When more labeled data are added, the improvement gained by using the unlabeled data is smaller, but that can be explained in two, possibly complementary, ways. First, the classifier learned using only the labeled data is already close to the optimal Bayes error rate. Second, we have seen in the previous section that when there is a large number of labeled data, they provide a good approximation to the marginal of the features. In that case adding the unlabeled data does not change the estimates of the classifier significantly.

The graphs in Figure 2(c) show that unlabeled data degrade the performance when the incorrect structure is assumed. First we see that adding more labeled data improves the classification even with an incorrect structure assumption. Second we see that as we add more unlabeled data, the classification error becomes higher.

We can see that assuming a NB structure to the TAN generated data (Fig 2(c)) results in a higher probability of error over the classifier that uses the correct structure(Fig 2(b)) for both labeled and unlabeled data and comparing with the same size training sets.

These experiments suggest that an incorrect structure assumption can lead to degraded classification performance when using unlabeled data, over using only labeled data. However, will it always degrade the performance? The answer is no. We performed another test, using data generated from a TAN structure with 49 features. The data sets were generated just as in the previous tests. Figure 3(a-b) shows the averaged classification error for both types of experiments, with (a) showing the results assuming the correct structure and (b) the results assuming a NB model.

Again, we see that when we assume the correct structure, adding the unlabeled examples improves the classification result

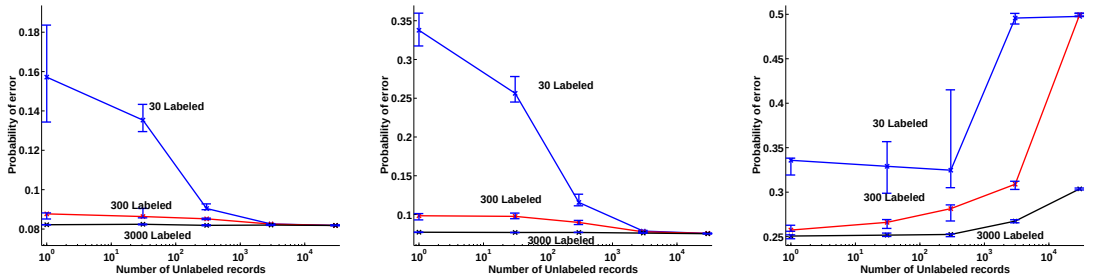


Figure 2: Classification error example for networks with 10 features. (a) Assumed and correct structure is NB, (b) Assumed and correct structure is TAN, (c) Assumed structure is NB, correct structure is TAN. The bars represent 30% and 70% percentiles of the error (statistics computed over 10 trials per point).

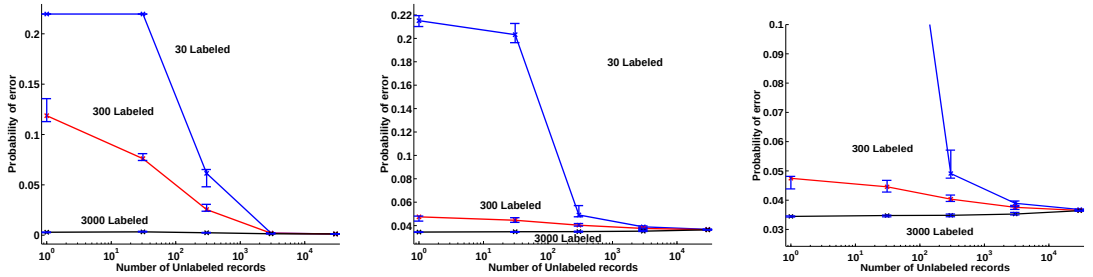


Figure 3: Classification error example for networks with 49 features. (a) Assumed and correct structure are TAN, (b) Assumed structure is NB, correct structure is TAN, (c) Zoom in on the bottom portion of (b).The bars represent 30% and 70% percentiles of the error (statistics computed over 10 trials per point).

at a fast rate, reaching almost the Bayes rate with just 30 labeled records and 30000 unlabeled records. In Figure 3(b) we see that although the structure assumption is incorrect, adding unlabeled data improves the classification results significantly for the cases where 30 and 300 labeled records were used. However, as can be seen in Figure 3(c), with 3000 labeled records, adding unlabeled data degraded the performance. We can conclude that when the estimator using only the labeled data has low variance, adding the unlabeled data can degrade the performance. This means that unlabeled data improve or degrade the classifier's performance depending on both the classifier's complexity and the number of labeled training records.

Further strengthening the experiments above, we performed a similar experiment with the Adult database taken from the UCI repository. The Adult database consists of 30162 labeled records for training and 15060 labeled records for testing. The study by Kohavi [9] using the MLC++ machine learning library showed that the classification error using all of the labeled data set is around 14-17% for the best classifiers. Naive-Bayes was used as one of the classifiers, and it achieved around 16% classification error.

In our experiment, we randomly partition the training data set to create labeled and unlabeled ("LUL") data sets; ranging from 30–3000 for the labeled sets and 0–30000 for the unlabeled sets. When possible, we create 5 sample sets. We use the EM algorithm to learn a NB classifier for each LUL training set.

The classification results are shown in Figure 4. The graphs clearly show that using unlabeled data increases the classification error compared to using only labeled data, except for the cases with only 30 labeled records.

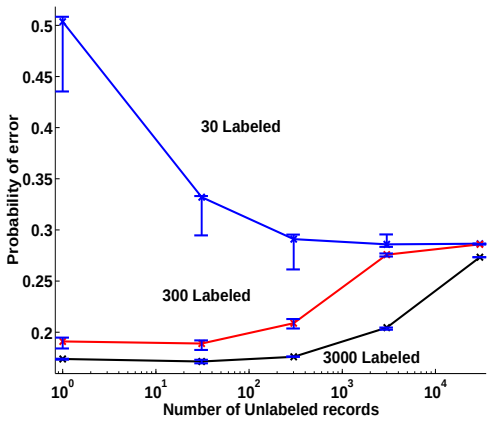


Figure 4: Classification result for the Adult DB experiment. The bars represent 30% and 70% percentiles of the error (statistics computed over the five trials per point).

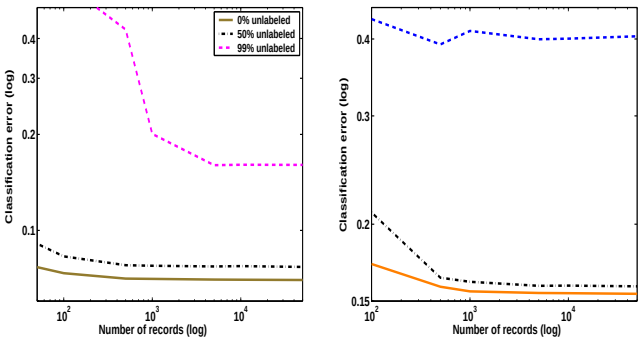
As in the artificially generated data case, when the complexity of the model is high compared to the size of the labeled training set, adding the unlabeled data does improve the classification result, from about 50% error to 30% error. However, as the size of the labeled set increases, adding unlabeled data degrades the classification performance from 19% and 17% to about 30%. This also indicates that the underlying structure is not NB.

4.2 Why does the use of unlabeled data degrade classification performance?

One of the main questions that is probably troubling the reader at this point is why unlabeled data behave differently than labeled data for incorrect structure? We have previously presented a geometric argument for these differences, based on the use of Kullback-Leibler divergence as a distance in the space of probabilities [5]; here we present another argument that can clarify aspects of the problem.

Consider N labeled training records and three estimation procedures. First, take estimates $\hat{p}_1(C, \mathbf{X}) = \arg\max S_1$, where $S_1 = \sum_{i=1}^{N} \log\left(p(C|\mathbf{X})\right)$. Then take $\hat{p}_2(C, \mathbf{X}) = \arg\max\left(S_1 + S_2\right)$, where $S_2 = \sum_{i=1}^{N} \log\left(p(\mathbf{X})\right)$ — we obtain the maximum likelihood estimator for fully labeled data. Finally, ignore the labels in the training data and take $\hat{p}_3(C, \mathbf{X}) = \arg\max S_2$ — we obtain the maximum likelihood estimator for fully unlabeled data. Note that $\hat{p}_1(C, \mathbf{X})$, $\hat{p}_2(C, \mathbf{X})$ and $\hat{p}_3(C, \mathbf{X})$ must always comply with the assumed classifier structure. If the correct structure is assumed, estimates can attain the exact values of $p(C, \mathbf{X})$, and all three estimators produce identical estimates as N grows without bound. If the structure is incorrect, then the set of possible models generated by the estimators may not contain the model that generated the data. We then have two possibilities. The first possibility is that $\hat{p}_1(C|\mathbf{X})\hat{p}_3(\mathbf{X})$ does satisfy the assumed structure; in that case this combination must be a valid estimate $\hat{p}_2(C, \mathbf{X})$ — and the asymptotes for fully labeled and fully unlabeled data are identical. The second possibility is that $\hat{p}_1(C|\mathbf{X})\hat{p}_3(\mathbf{X})$ violates the assumed structure, and so we cannot combine $\hat{p}_1(C, \mathbf{X})$ and $\hat{p}_3(\mathbf{X})$.[1] Now we can have estimates $\hat{p}_2(C, \mathbf{X})$ that are different from $\hat{p}_3(C, \mathbf{X})$, because $\hat{p}_2(C, \mathbf{X})$ must maximize a combination of S_1 and S_2. This is the situation that can produce different classification error asymptotes for labeled and unlabeled data, that is, even with infinite training data, the classification error of the labeled training set will be smaller than that of the unlabeled training set.

[1] We should expect that $\hat{p}_1(C, \mathbf{X})$ yields a better classification error than $\hat{p}_3(C, \mathbf{X})$, because $p(C, \mathbf{X})$ is the relevant quantity for classification and $\hat{p}_1(C, \mathbf{X})$ is obtained by focusing on $p(C, \mathbf{X})$.



(a) (b)

Figure 5: LU-graphs (a) example with two Gaussian features, (b) Discrete features. Each point in each graph is the average of multiple trials.

But what about training sets that have both labeled and unlabeled data? The classification error asymptotes will change as the LUL ratio changes, filling the gap between the asymptote of the fully labeled training set and fully unlabeled training set. To visualize this effect we use a new type of graph. 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*.

Figure 5(a) shows LU-graphs for a classification problem involving two Gaussian features that are not independent given the class. During the learning we assume that they are independent, which is an incorrect structure assumption. Figure 5(b) shows the LU graphs for the discrete feature case, involving 10 features sampled from a TAN model and learned with a NB model. Both figures show the LU-curves for 0% unlabeled records, 50% unlabeled records and 99% unlabeled records. We see that the LU-graphs for 50% and 99% unlabeled data have asymptotes that do not converge to the same value, and they are both different from the asymptote for labeled data.

An additional argument is needed to understand the effect of unlabeled data: there is a difference between classification and estimation. Even though adding more data (labeled and unlabeled) leads to better overall estimation (with respect to various global measures such as likelihood, squared-error, variance, Fisher information), the improvement may be uneven amongst the estimated parameters. Note that for classification, only $p(C|\mathbf{X})$ matters [7]; if the bias in $\hat{p}_u(C|\mathbf{X})$ is larger than the bias in $\hat{p}_l(C|\mathbf{X})$, the asymptotic classification performance for unlabeled data is worse than for labeled data. When this performance gap is present, then unlabeled data can degrade performance and the LU-graphs can be used to capture this phenomenon.

Another important point that follows from the preceding discussion is that missing labels are different from missing feature values, even though algorithms such as EM handle them in the same manner. While both are forms of missing data and degrade estimation performance, unlabeled data also directly affects classification performance by introducing bias in the critical parameters of $p(C|\mathbf{X})$. This insight clarifies questions on missing/unlabeled data raised by Seeger [12].

4.3 How to use unlabeled data

In most real applications, the correct model is not known, so we assume one. As our experiments showed, when using unlabeled data with a small number of labeled records, the classifier's performance could degrade or improve, depending on the complexity of the model. For a particular training set, we can use the fact that unlabeled data degrades the classification performance as an indicator that we have incorrect modeling assumptions.

Thus, we propose a general method for using unlabeled data in searching for modeling assumptions, and subsequently improving the classification performance:

- Take an initial set of modeling assumptions to characterize the joint distribution $P(C, X)$.
- Learn at least two classifiers. One using only the labeled records, denoted as C_l, and the second using both labeled and unlabeled records, denoted as C_{lul}.
- Test both classifiers on a test set.
- If the classification accuracy of C_{lul} is worse than that of C_l, change the modeling assumptions and repeat the previous steps.

The method described above is very general, and requires further details. Each step can have many possible executions. For example, the initial structure can be chosen depending on the training set size and complexity of the model (number of features, cardinality of the parameter space for the features/class). In the second step, we can train several classifiers using all of the unlabeled data and perform an hypothesis test to determine if unlabeled data degraded the performance. Testing the classifiers can be done using cross-validation, bootstrap or other statistical tests, depending on the labeled training set size. In the fourth step, the decision on changing the structure has to be such as to avoid overfitting. In addition, a consistent method for changing the structure has to be applied such that we end up with a classifier that performs better than any of the other ones tested.

As an example of the decisions that must be made when switching models, consider the Adult database. Figure 4 shows the effect of adding unlabeled data. We must verify whether it is statistically meaningful to say that performance is degrading — that is, we must decide whether the performance fluctuations are random or really caused by incorrect modeling assumptions. To do so, we can set up a statistical test, taking as null hypothesis the fact that "the expected classification error decreases after adding unlabeled data". With a few assumptions of normality and equality of variances, we can set up a comparison of means test [6]. To illustrate this type of test with the Adult database, we learned 20 classifiers, using 300 labeled records for each classifier, with and without unlabeled data (24000 records), and ran each one of them in the testing data. The mean of the "labeled classifiers" was 0.1876 (with standard deviation 0.0144), and the mean of the "unlabeled classifiers" was 0.2707 (with standard deviation 0.0225). The statistic used in the appropriate t-test is equal to 13.5691; the test recommends rejection at confidence 99.95% if this statistic is larger than 3.55. So, we can be quite confident that fluctuations are not random in this problem. In practice, we may not have the resources to produce this collection of classifiers; in those cases, we must resort to statistical techniques such as the bootstrap and cross validation methods. We are currently investigating this possibility.

Constructing algorithms that implement the basic method is part of our future research.

4.4 Beyond Naive-Bayes: Using unlabeled data to learn a TAN model via EM-TAN

The results of the previous sections show that it is important to obtain the correct structure when unlabeled data are to be used for learning a classifier. But finding the correct structure is usually a difficult problem. In addition, switching to very complicated structures could result in poor classifiers, depending on the size of the training set. Since the objective is to improve classification result, an alternative to finding the actual correct structure is finding a structure in which adding unlabeled data does not degrade the classification performance. But even for this alternative, searching the whole space of possible structures suffers from the same difficulties.

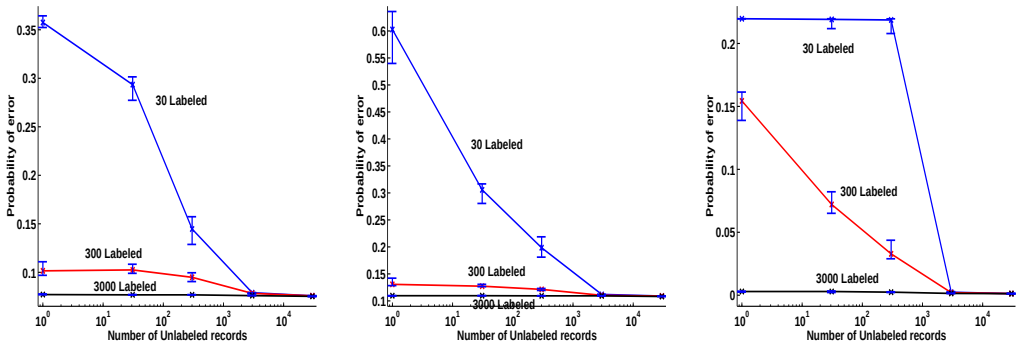
One solution that has been suggested in the supervised learning setting is finding the best TAN model to fit the data [8]. It turns out that searching over the space of TAN structures to find the TAN structure that maximizes the likelihood of the data can be done efficiently using the Chow-Liu algorithm [3] by computing the minimum spanning tree (MST) over the features using the pairwise mutual information of the features. Friedman et al. [8] proposed using the TAN model as a classifier, to enhance the performance over the simple Naive-Bayes classifier. TAN models are more complicated than NB, but are not fully

connected graphs. The existence of an efficient algorithm to compute the best TAN model makes it a good candidate in the search for a better structure. Another motivation for using a slightly more complicated structure instead of the Naive-Bayes model can be drawn from the experiments done by Rebbeca Bruce [2]. In her experiments, the structure of the classifier is varied from the full structure (no independence relations between features) to the simple NB structure. Data sets consisting of a fixed number of unlabeled data and changing number of labeled records are used to train the classifiers. One observation from most of her experiments is that the model slightly more complicated than NB displayed a constant error rate, even as the number of labeled records was reduced. The NB structure performed the worse, and the more complicated structures usually had good results when all the labeled records were used, but the classification performance degraded as the number of labeled records was reduced. This leads to the conclusion that with the available training set size she used, the slightly more complicated structure than NB is the best candidate structure, and would yield good results even for a very small labeled training set.

The algorithm presented by Friedman et al. [8] for computing the best TAN assumes that there are no missing data. To solve the maximization problem when there are missing labels, it is possible to develop an EM algorithm that finds the best TAN given both labeled and unlabeled data. Meila [10] developed the EM equations for the task of building the best set of minimum spanning tree for a classification problem. In her setup, there are no labeled records and the number of classes can vary. The problem she solves is basically a clustering problem and is not directly related to the semi-supervised learning problem, but the EM algorithm she developed applies directly to the semi-supervised learning problem. The EM algorithm for TAN models given labeled and unlabeled data follows directly from Meila's [10] *MixTreeS* algorithm; we call the resulting scheme EM-TAN. The general steps of EM-TAN are as follows:

- Pick an initial TAN structure and parameters. The class variable is always the root node. The initial TAN is either arbitrarily chosen or is found by estimating the TAN model with only the labeled records using Friedman et al.'s *Construct-TAN* procedure [8].
- Iterate until the change in the likelihood between the current iteration and previous falls under a threshold:
  - E-Step: Compute expected value of the missing labels using the current model parameters.
  - M-Step:
    - Compute the class conditional mutual information between all pairs of feature variables using the fractional counts from the E-step.
    - Construct the MST using the Chow-Liu algorithm.
    - Compute all of the parameters of the new model given the MST found in the previous step.

To test the EM-TAN algorithm we use the artificial data generated using a TAN model and learn the best classifier. It is expected that EM-TAN would converge to the correct TAN structure as the number of data records increases. We perform tests using the data from three different TAN structures, two with 10 features and one with 49 features. The results are shown in Figure 6. From the figures we see that unlabeled data helped reduce the classification error to the same results as if the structure was known a-priori. Observing the learned structure we can also report that as the number of records increased, either labeled or unlabeled, the correct TAN structure was learned, thus explaining the good classification results. For the model with 49 features, with 30 and 300 unlabeled records and few unlabeled records, the classification error is higher than the case shown in Figure 3(a), because EM-TAN does not converge to the correct TAN structure with this small training set, however, as the number of unlabeled records increases, the classification error goes down to the same as if the structure was known.



(a) 10 features (b) 10 features (c) 49 features

Figure 6: Probability of error of classifiers learned using EM-TAN for three different network examples.

5 Summary and future work

In this paper we discussed the value of unlabeled data in learning the parameters of a generative model used for classification. We characterized when adding unlabeled data changes estimates.

We also showed that unlabeled data can both degrade and improve performance when we have incorrect modeling assumptions and improve performance otherwise. We presented the results of experiments both on real and artificial data to illustrate this point and provided with a theoretical explanation to this phenomenon.

Because degraded classification performance when adding unlabeled data indicates that we have an incorrect structure, we leverage on that to come up with a structure search method which takes advantage of unlabeled data to improve classification performance.

We further suggested EM-TAN, an algorithm for searching the space of TAN structures using both labeled and unlabeled data, with the purpose of learning a more realistic structure which takes into account dependencies between features. The hope is that because a TAN structure can be closer to the real distribution of variables, unlabeled data will help improve the classification performance.

Future research involves developing algorithms for implementing the general method introduced in this paper. Another direction uses active-learning on a small set of unlabeled records to counteract the possible effect of learning the classifier with a large set of unlabeled data.

It remains to be seen if similar effects occur in different learning settings, such as the modified diagnostic mode settings, co-training, transductive SVM's and others. We can only hypothesize that for any model based approach, when the model that generated the data is not included in the set of models being searched as classifiers, adding unlabeled could degrade the classification performance as we have shown for ML estimation.

Acknowledgements

We thank Marsha Duro for many suggestions and comments during the course of the work, her help and support was critical to the success of this work. We thank Kevin Murphy for the freely available BNT system, which we used to generate examples and data. 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] S. Baluja. Probabilistic modelling for face orientation discrimination: Learning from labeled and unlabeled data. In *NIPS*, 1998.

[2] R. Bruce. Semi-supervised learning using prior probabilities and EM. In *IJCAI-01 Workshop on Text learning: Beyond supervision*, Aug 2001.

[3] C.K. Chow and C.N. Liu. Approximating discrete probability distribution with dependence trees. *IEEE Transactions on Information Theory*, 14:462–467, 1968.

[4] F.G. Cozman and I. Cohen. Unlabeled data can degrade classification performance of generative classifiers. Technical Report HPL-2001-234, Hewlett-Packard Labs, 2001.

[5] F.G. Cozman and I. Cohen. Unlabeled data can degrade classification performance of generative classifiers. In *FLAIRS*, 2002.

[6] M.H. DeGroot. *Probability and statistics*. Addison-Wesley Pub. Co, Reading, Mass., 1986.

[7] J.H. Friedman. On bias, variance, 0/1 - loss, and the curse-of-dimensionality. In *Technical Report*. Stanford University, 1996.

[8] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2):131–163, 1997.

[9] R. Kohavi. Scaling up the accuracy of naive-bayes classifiers: a decision-tree hybrid. In *Proc. Second Int. Conference on Knowledge Discovery and Data Mining*, 1996.

[10] M. Meila. *Learning with mixture of trees*. PhD thesis, Massachusetts Institute of Technology, 1999.

[11] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39:103–134, 2000.

[12] M. Seeger. Learning with labeled and unlabeled data. In *Technical Report*. Edinburgh University, UK, 2001.

[13] B. Shahshahani and D. Landgrebe. 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.

[14] N.L Zhang. Irrelevance and parameter learning in Bayesian networks. *Artificial Intelligence, An International Journal*, 88:359–373, 1996.

[15] T. Zhang and F. Oles. A probability analysis on the value of unlabeled data for classification problems. In *ICML*, 2000.

Appendix A

The purpose of the following experiments is to test the hypothesis that unlabeled data can degrade classification performance when the assumed structure is incorrect, and improve performance when the assumed structure is correct. We can also observe the effect of labeled data on classification in these two cases. The use of artificially generated data allows us to know the ground truth, generate large test samples and arbitrarily large training samples. The data are generated using Bayesian networks (BN's) with known structures and known parameters.

We use the TAN and NB structures to generate the data. For each structure (TAN and NB) we sample different data sets as follows:

- We generate data from TAN and NB structures with increasing number of features; from 3 features to 10 features.
- The number of values for each feature is selected at random from 2-4 values. The class node always has two values[2]. All features are discrete.
- For each structure with n number of features, the CPT's are randomly chosen by uniformly sampling all the entries between 0-1 followed by normalization to get proper probability distributions.
- For each model (structure + CPT assignment),the training data sets are randomly generated with all combinations of labeled and unlabeled defined below. In addition, a test set of 50000 records is generated.
- The training sets consists of all possible combinations of data sets consisting of: Labeled samples= $\{30, 300, 3000\}$, Unlabeled samples= $\{0, 30, 300, 3000, 30000\}$. There are 15 possible combinations. All data sets are generated independently of each other. The only missing data in the unlabeled data sets are the class labels, the features are always observed.
- We generate five such models for each n number of features, each model has different CPT values, and different number of values per feature.
- For each labeled-unlabeled combination and a particular model record, 10 sets are generated.

Overall, for each structure (NB or TAN) there are 40 different models that are generated. From each model there are 150 training sets with different number of labeled-unlabeled combinations, and a test set consisting of 50000 samples.

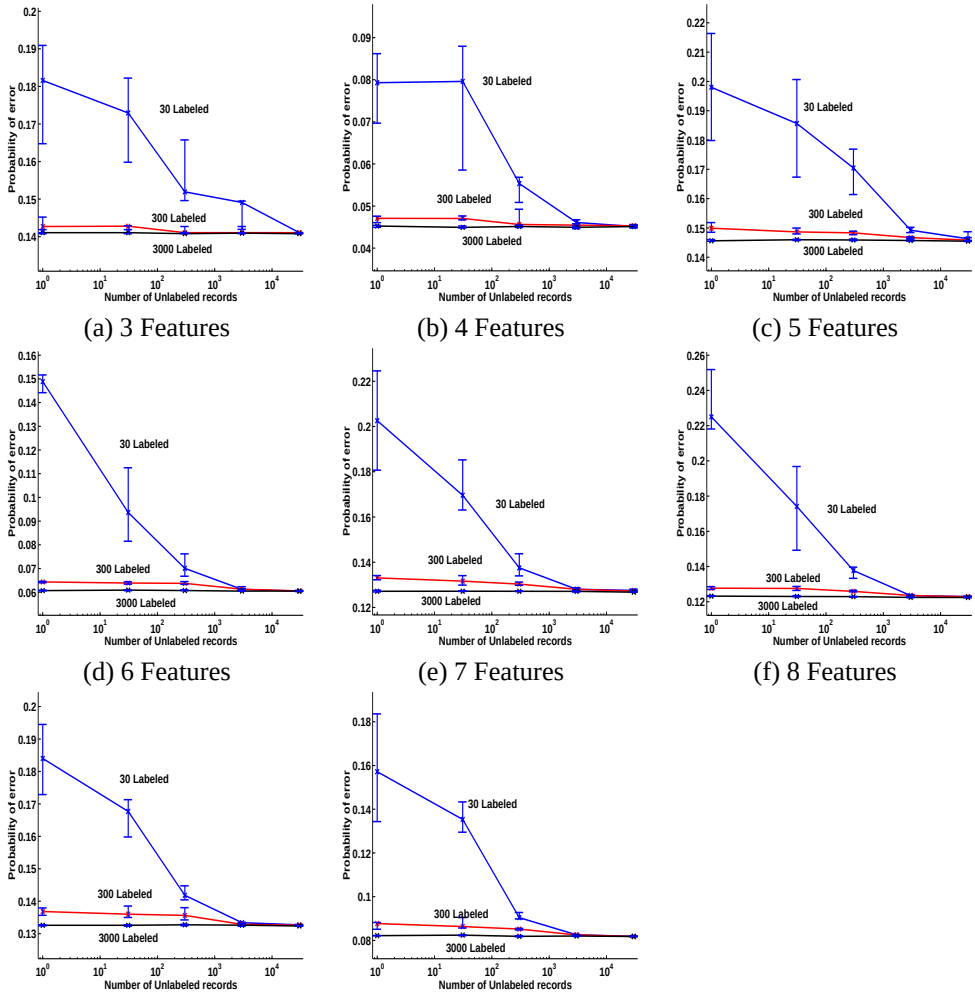
The purpose of using different number of features is to see if there are different effects related to an increase in number of parameters, such as the Hughes phenomenon.

Learning the classifiers (the assumed model's parameters) is done by maximizing the likelihood function over the training data. For data sets with only labeled data this amounts to simple event counting. For data sets with both labeled and unlabeled data, we use the EM algorithm (see [14] for description of EM applied to discrete BN's). The starting point for the EM algorithm is always set with the parameters learned using only the labeled examples of a given dataset. With small number of labeled data, this starting point could lead to a local maximum, however, with enough labeled data, the initial starting point would most likely lead EM to the global maximum. We also performed a few tests using a good starting point (the parameters learned using the 300000 datasets) and observed that the results we discuss below still hold.

We perform two types of tests. In the first type, we use the various training sets to learn classifiers, providing the true structure (TAN or NB). In the second type, we provide an incorrect structure and use the training data to learn the classifier. In our tests we assume a NB structure for the TAN generated data. For the NB generated data, we assume a more complicated TAN structure, and learn both the best TAN structure and the best parameters for that structure given the training data. We name the algorithm for computing the best TAN model with both labeled and unlabeled data, EM-TAN.

We demonstrate the results using just one of the five models in each of the experiments, however we observed similar results for all of the models. Figure 7(a-h) shows the plots of the classification errors for all LUL combinations with the different number of features using the NB data sets and assuming the correct NB structure. Figure 8(a-h) shows the plots of the classification errors using the TAN structure data sets and assuming the correct TAN structure. Figure 9(a-h) shows the classification errors for the tests using the TAN generated data, assuming a NB structure. For all of the figures, the error bars show the 30-70% percentiles of the error computed from the results of the 10 trials per point.

[2]We performed a similar experiment with 4 classes and saw the same results

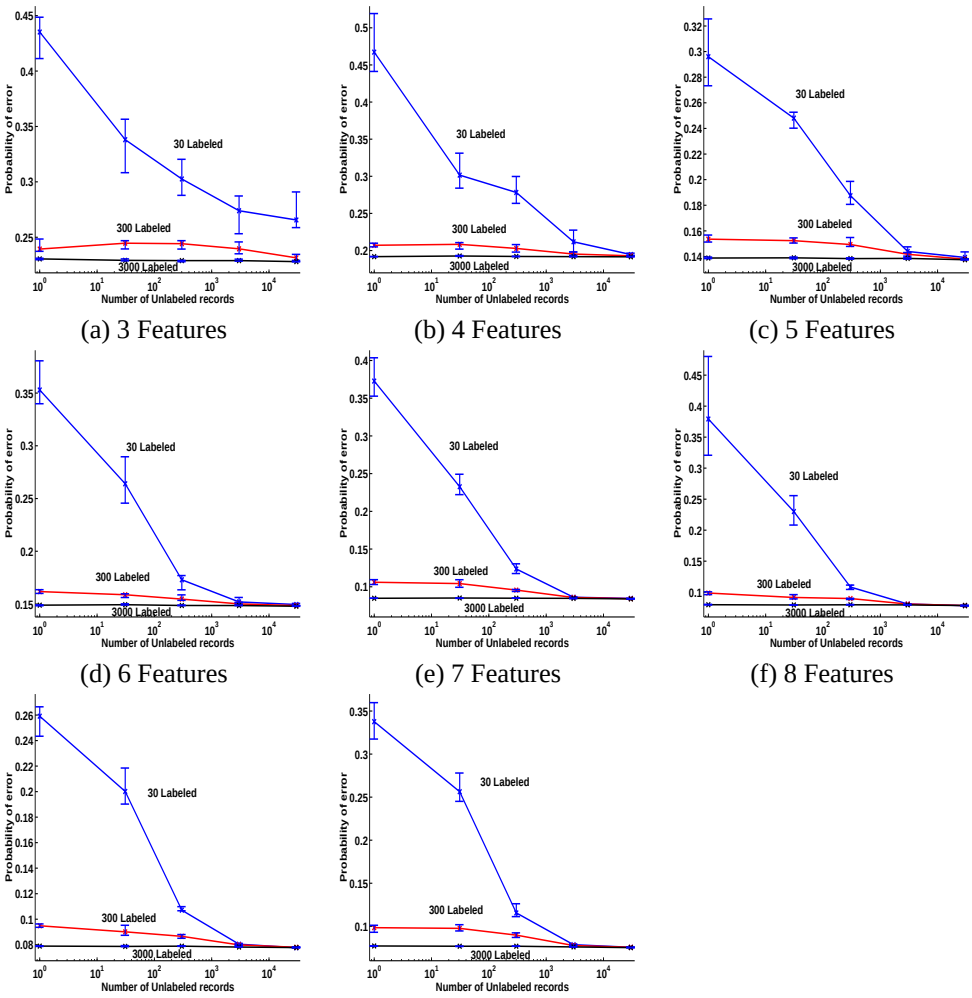


(a) 3 Features (b) 4 Features (c) 5 Features

(d) 6 Features (e) 7 Features (f) 8 Features

(g) 9 Features (h) 10 Features

Figure 7: Classification error in the case of learning with the correct structure. The correct structure is Naive-Bayes.

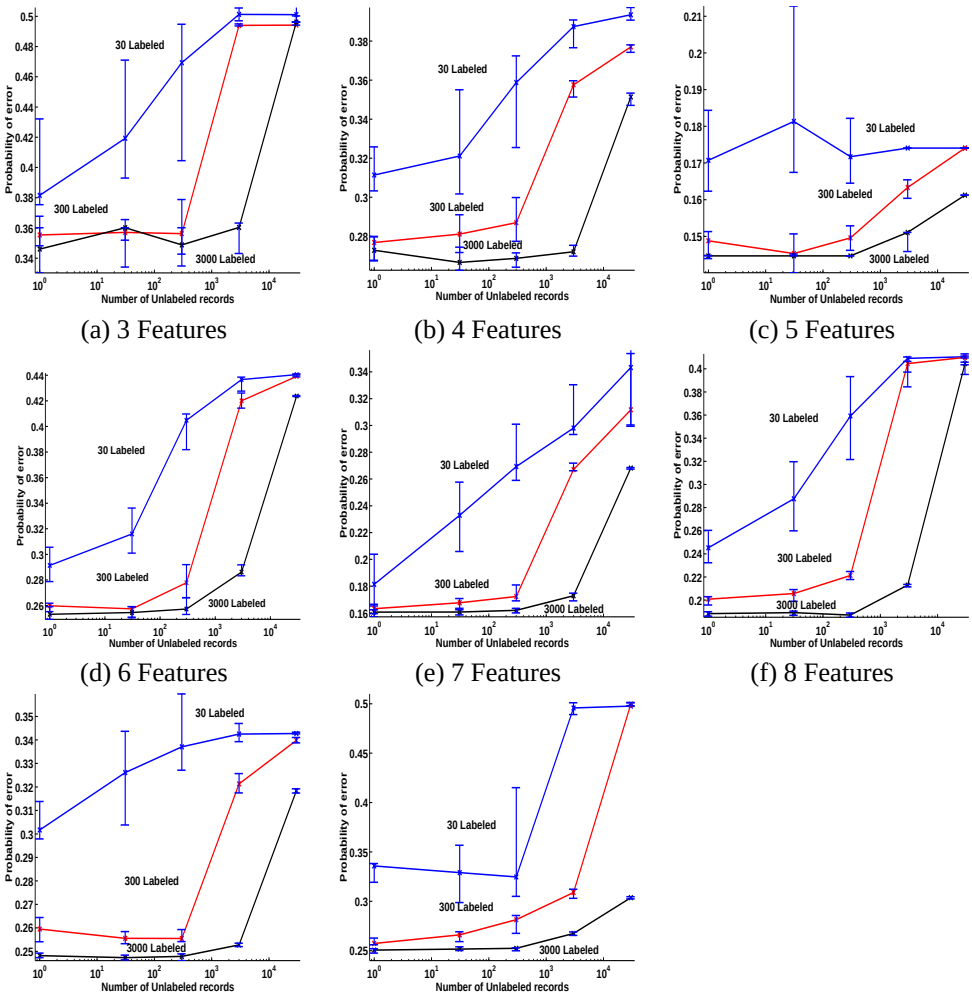


(a) 3 Features (b) 4 Features (c) 5 Features

(d) 6 Features (e) 7 Features (f) 8 Features

(g) 9 Features (h) 10 Features

Figure 8: Classification error in the case of learning with the correct structure. The correct structure is TAN.



(a) 3 Features (b) 4 Features (c) 5 Features

(d) 6 Features (e) 7 Features (f) 8 Features

(g) 9 Features (h) 10 Features

Figure 9: Classification error in the case of learning with an incorrect structure. The correct structure is TAN and assumed structure is NB.