INTERSPEECH 2009 BRIGHTON

A Semi-supervised Version of Heteroscedastic Linear Discriminant Analysis

Haolang Zhou[1], *Damianos Karakos*[12], *Andreas G. Andreou*[1]

[1]Center for Language and Speech Processing
[2]Center of Excellence in Human Language Technology
[1]Department of Electrical and Computer Engineering
Johns Hopkins University
Baltimore, MD 21218

haolangzhou@jhu.edu, damianos@jhu.edu, andreou@jhu.edu

Abstract

Heteroscedastic Linear Discriminant Analysis (HLDA) was introduced in [1] as an extension of Linear Discriminant Analysis to the case where the class-conditional distributions have unequal covariances. The HLDA transform is computed such that the likelihood of the training (labeled) data is maximized, under the constraint that the projected distributions are orthogonal to a nuisance space that does not offer any discrimination. In this paper we consider the case of semi-supervised learning, where a large amount of unlabeled data is also available. We derive update equations for the parameters of the projected distributions, which are estimated *jointly* with the HLDA transform, and we empirically compare it with the case where no unlabeled data are available. Experimental results with synthetic data and real data from a vowel recognition task show that, in most cases, semi-supervised HLDA results in improved performance over HLDA.

Index Terms: heteroscedastic linear discriminant analysis, nuisance space, semi-supervised learning.

1. Introduction

Heteroscedastic Linear Discriminant Analysis (HLDA) was introduced in [1] as a linear method for dimensionality reduction, and has been used with success in complex classification tasks which assume Gaussian parametric families, such as in speech recognition. HLDA is a suitable technique for several reasons: (i) it preserves the Gaussianity of the original data (by virtue of being a linear transformation); (ii) it does not have constraining assumptions about the class-conditional covariances; and (iii) it can be easily estimated from very large amounts of data (for example, several millions of samples resulting from just a few hours of speech); only first and second moment statistics of the labeled data need to be computed. For these reasons, many state-of-the-art speech recognition systems use it routinely as part of a two-pass training procedure [2], with substantial gains in word-error-rate.

HLDA is a *maximum-likelihood* estimation technique: its goal is to jointly estimate Gaussian models in a reduced dimensional space and a transform that will maximize the likelihood of the observed, high-dimensional, data. (Another similar maximum-likelihood approach appears in [3].) This fits nicely with standard procedures (such as the EM-algorithm [4]) which are used to estimate generative models with some hidden structure. It is thus not surprising that HLDA was seamlessly integrated with tools that train maximum likelihood models, such as HTK [5]. Furthermore, the fact that LDA is a special case of HLDA [6, 7], justifies the superiority of HLDA on experiments with synthetic data [7] as well as real speech data [1].

As mentioned above, HLDA is a supervised technique, which requires class-conditional first and second-order statistics to be computed. In the absence of enough labeled data, though, the second-order statistics can be very noisy, thus resulting in a noisy estimate of the transform. This, in turn, can exacerbate the modeling mismatch and degrade performance much more than one would expect, because the transformation will introduce a multiplicative noise component.

Semi-supervised learning has been proposed as a framework for tackling this problem; its goal is to improve statistical models using both labeled and unlabeled data. It has attracted a significant amount of research lately (see, for instance, [8] and references therein), mainly due to the fact that unlabeled data can be easily and cheaply collected. The unlabeled data can be used in a variety of ways: to compute global statistics that are independent of label information, to extract regions of high density that should share the same label, or to better estimate a manifold where data naturally lie.

This paper describes a semi-supervised version of HLDA. Our formulation is similar to [9], which shows how to use the maximum likelihood criterion when both labeled and unlabeled data are available. The idea is to use the EM algorithm, which at each iteration estimates a class posterior distribution ("soft" count) for each unlabeled data sample using the conditional distributions estimated in the previous iteration, then updates the class-conditional parameters (Gaussian in our case) based on these "soft" counts. In our formulation, estimation of the HLDA transformation matrix is part of the optimization of the EM objective function (expected log-likelihood of training data).

There has been previous work on a semi-supervised version of LDA, which is very different from ours. For instance, [10] try to maximize the Rayleigh quotient of LDA, but with a regularizer which imposes a constraint on the "closeness" between unlabeled data points, so that unlabeled data which are very close in the original space remain close in the projected space as well. The graph Laplacian plays a central role in this line of work, as well as in other semi-supervised learning research which focuses on manifold regularization [11].

The paper is organized as follows: Section 2 contains mathematical preliminaries and a review of HLDA. In Section 3 we present the semi-supervised formulation and give the EM update equations. Next, Section 4 contains experimental results with synthetic data as well as real vowel data which show that semi-supervised HLDA gives improvements over supervised HLDA. Finally, concluding remarks appear in Section 5.

2. Mathematical Preliminaries and Review of HLDA

In what follows, it is assumed that a labeled training corpus exists, consisting of L observations $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_L$, each of dimen-

Copyright © 2009 ISCA

sionality n. Their associated labels are denoted by $c_1, \ldots, c_L$, with $c_i \in \{1, \ldots, C\}$. Furthermore, an unlabeled corpus is also available, consisting of U observations $\boldsymbol{x}_{L+1}, \ldots, \boldsymbol{x}_{L+U}$.

The number of observations of class c is N_c, and hence $N_1 + \cdots + N_C = L$. The sample mean of class c is denoted by $\boldsymbol{\mu}_c$, while the sample covariance of class c (the "within-class" covariance) is denoted by Σ_c. Specifically,

$$\boldsymbol{\mu}_c \triangleq \frac{1}{N_c} \sum_{j:c_j=c} \boldsymbol{x}_j, \;\; \Sigma_c \triangleq \frac{1}{N_c} \sum_{j:c_j=c} (\boldsymbol{x}_j - \boldsymbol{\mu}_c)(\boldsymbol{x}_j - \boldsymbol{\mu}_c)^{\mathrm{T}},$$

where a^{T} is the transpose of matrix (or vector) a. The global mean and variance are denoted by $\boldsymbol{\mu}$ and Σ, respectively. The k-dimensional Gaussian density with mean $\boldsymbol{\mu}$ and covariance matrix Σ is denoted by $\phi_{(k)}(\cdot; \boldsymbol{\mu}, \Sigma)$, or just $\phi(\cdot; \boldsymbol{\mu}, \Sigma)$ when the dimensionality is clear from the context.

Projecting a vector $\boldsymbol{x}$ into $\mathbb{R}^p$ is done by multiplying it with a $p \times n$ matrix Θ ($p < n$).

The HLDA formulation of [1] seeks a transformation matrix Θ with the property that the p dimensions of the transformed data are dependent on the class label, while the rest $n-p$ dimensions are independent of the class label (nuisance dimensions). Thus, the dimensionality reduction which results from the "removal" of the $n-p$ dimensions aims at keeping only the class-discriminative information. The maximum-likelihood criterion arises from computing the likelihood of the *original data* as a function of the "transformed" models and Θ.

A summary of the maximum-likelihood approach appears below.

- Each class-conditional distribution in the original space is assumed to be Gaussian.
- The data are transformed as $\boldsymbol{y}_j = \Theta \boldsymbol{x}_j, \;\; j = 1, \ldots, L$, where Θ is an $n \times n$ invertible matrix.
- The class-conditional distributions in the transformed space are Gaussians with parameters

$$\begin{aligned} \widetilde{\boldsymbol{\mu}}_c &= (\widetilde{\boldsymbol{\mu}}_c^{(p)}, \widetilde{\boldsymbol{\mu}}^{(n-p)}) \triangleq (\widetilde{\mu}_{c,1}, \ldots, \widetilde{\mu}_{c,p}, \widetilde{\mu}_{p+1}, \ldots, \widetilde{\mu}_n)^{\mathrm{T}}, \\ \widetilde{\Sigma}_c &= \begin{pmatrix} \widetilde{\Sigma}_c^{(p)} & \mathbf{0} \\ \mathbf{0} & \widetilde{\Sigma}^{(n-p)} \end{pmatrix}. \end{aligned}$$

  Note that only the first p dimensions are useful in discriminating between the classes.

- The objective in the estimation of Θ is the maximization of the log-likelihood of the *original* data:

$$\mathcal{L}(\{(\boldsymbol{x}_1, c_1), \ldots, (\boldsymbol{x}_N, c_N)\}) = \sum_c \left(\sum_{j:c_j=c} \log \phi(\boldsymbol{x}_j; \boldsymbol{\mu}_c, \Sigma_c) + N_c \log p(c) \right), \quad (1)$$

  where $p(c)$ is the (known) prior on the classes.

  Conditioned on class c, the relationship between the pdfs of $\boldsymbol{x}$ and $\boldsymbol{y} = \Theta \boldsymbol{x}$ can be easily established

$$\begin{aligned} \phi(\boldsymbol{x}; \boldsymbol{\mu}_c, \Sigma_c) &= |\Theta| \, \phi(\Theta \boldsymbol{x}; \widetilde{\boldsymbol{\mu}}_c, \widetilde{\Sigma}_c) \\ &= |\Theta| (\phi_{(p)}(\Theta^{(p)} \boldsymbol{x}; \widetilde{\boldsymbol{\mu}}_c^{(p)}, \widetilde{\Sigma}_c^{(p)}) \times \\ &\quad \phi_{(n-p)}(\Theta^{(n-p)} \boldsymbol{x}; \widetilde{\boldsymbol{\mu}}^{(n-p)}, \widetilde{\Sigma}^{(n-p)})) \end{aligned}$$

  The log-likelihood of the (labeled) training data is given by

$$\begin{aligned} & L \log |\Theta| + \sum_c N_c \Big[\log p(c) - \frac{1}{2} \log |\widetilde{\Sigma}_c^{(p)}| \\ & - \frac{1}{2} \sum_{j:c_j=c} (\Theta^{(p)} \boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}_c^{(p)})^{\mathrm{T}} (\widetilde{\Sigma}_c^{(p)})^{-1} (\Theta^{(p)} \boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}_c^{(p)}) \Big] \\ & - \frac{1}{2} \sum_{j=1}^{L} (\Theta^{(n-p)} \boldsymbol{x}_j - \widetilde{\mu}^{(n-p)})^{\mathrm{T}} (\widetilde{\Sigma}^{(n-p)})^{-1} (\Theta^{(n-p)} \boldsymbol{x}_j - \widetilde{\mu}_c^{(n-p)}) \\ & - \frac{L}{2} \log(2\pi)^n - \frac{L}{2} \log |\widetilde{\Sigma}^{(n-p)}| \end{aligned} \quad (2)$$

  which is maximized when

$$\widetilde{\boldsymbol{\mu}}_c^{(p)} = \frac{1}{N_c} \sum_{j:c_j=c} \Theta^{(p)} \boldsymbol{x}_j = \Theta^{(p)} \boldsymbol{\mu}_c \quad (3)$$

$$\begin{aligned} \widetilde{\Sigma}_c^{(p)} &= \frac{1}{N_c} \sum_{j:c_j=c} (\Theta^{(p)} \boldsymbol{x}_j - \widetilde{\mu}_c^{(p)})(\Theta^{(p)} \boldsymbol{x}_j - \widetilde{\mu}_c^{(p)})^{\mathrm{T}} \\ &= \Theta^{(p)} \Sigma_c (\Theta^{(p)})^{\mathrm{T}} \end{aligned} \quad (4)$$

$$\widetilde{\boldsymbol{\mu}}^{(n-p)} = \frac{1}{N} \sum_{j=1}^{N} \Theta^{(n-p)} \boldsymbol{x}_j = \Theta^{(n-p)} \boldsymbol{\mu} \quad (5)$$

$$\begin{aligned} \widetilde{\Sigma}^{(n-p)} &= \\ & \frac{1}{N} \sum_{j=1}^{N} (\Theta^{(n-p)} \boldsymbol{x}_j - \widetilde{\mu}^{(n-p)})(\Theta^{(n-p)} \boldsymbol{x}_j - \widetilde{\mu}^{(n-p)})^{\mathrm{T}} \\ &= \Theta^{(n-p)} \Sigma (\Theta^{(n-p)})^{\mathrm{T}}, \end{aligned} \quad (6)$$

  where $\boldsymbol{\mu}, \Sigma$ are the global mean and covariance of the data, respectively. Substituting these values in the expression for the log-likelihood of the training data, it becomes

$$\begin{aligned} & \mathcal{L}^*(\{(\boldsymbol{x}_1, c_1), \ldots, (\boldsymbol{x}_L, c_L)\}) = \sum_c N_c \log p(c) \\ & + L \log |\Theta| - \frac{L}{2} \log(2\pi)^n - \sum_c \frac{N_c}{2} \log |\Theta^{(p)} \Sigma_c (\Theta^{(p)})^{\mathrm{T}}| \\ & - \sum_c \frac{p N_c}{2} - \frac{L}{2} \log |\Theta^{(n-p)} \Sigma (\Theta^{(n-p)})^{\mathrm{T}}| - \frac{(n-p)L}{2} \\ &= \sum_c N_c \log p(c) + L \log |\Theta| - \sum_c \frac{N_c}{2} \log |\Theta^{(p)} \Sigma_c (\Theta^{(p)})^{\mathrm{T}}| \\ & - \frac{L}{2} \log |\Theta^{(n-p)} \Sigma (\Theta^{(n-p)})^{\mathrm{T}}| - \frac{nL}{2} \log(2\pi e) \end{aligned} \quad (7)$$

  Expression (7) is the objective function of HLDA. The maximizing Θ cannot be given in closed form, and a gradient descent algorithm is needed for its computation, except for the special case where the projected per-class Gaussians are constrained to have diagonal covariance matrices; Gales [12] gives an efficient iterative algorithm for this case.

3. Semi-supervised HLDA

The goal of semi-supervised HLDA can be simply stated as follows:

Find a transformation matrix Θ and parameters $\widetilde{\boldsymbol{\mu}}_c^{(p)}, \widetilde{\Sigma}_c^{(p)}$, $c = 1, \ldots, C$ and $\widetilde{\mu}^{(n-p)}, \widetilde{\Sigma}^{(n-p)}$ such that the total likelihood of the labeled and unlabeled data is as high as possible.

In mathematical terms, the objective is to maximize

$$\mathcal{L} = \sum_{j=1}^{L} \log(p(\boldsymbol{x}_j, c_j)) + \sum_{j=1}^{U} \log(p(\boldsymbol{x}_{L+j})), \quad (8)$$

The second term on the right-hand-side of (8) corresponds to the marginal distribution of the data, which can be obtained from the joint distribution by summing over the class labels. Direct optimization of such a function through computation of derivatives is very hard; one solution is to resort to the EM algorithm [4] which aims at maximizing the expected log-likelihood

of the *complete* data. The EM theorem guarantees that the log-likelihood of the data increases when this objective function increases (also see [13]). Here, the initial model is obtained by applying HLDA on only labeled data.

Thus, the objective function of EM is

$$Q(M, M_0) = E_{M_0}[\log p_M(\Psi)|\boldsymbol{X}] \tag{9}$$

where M, M_0 correspond to the "new" and "old" model parameters, respectively, Ψ is a random quantity that corresponds to the "complete" training data, and $\boldsymbol{X}$ corresponds to the observed "incomplete" data. Thus,

- $\Psi = ((\boldsymbol{x}_1, c_1), \ldots, (\boldsymbol{x}_L, c_L), (\boldsymbol{x}_{L+1}, C_{L+1}), \ldots, (\boldsymbol{x}_{L+U}, C_{L+U}))$ for random labels $C_{L+1}, \ldots, C_{L+U}$,
- $\boldsymbol{X} = ((\boldsymbol{x}_1, c_1), \ldots, (\boldsymbol{x}_L, c_L), \boldsymbol{x}_{L+1}, \ldots, \boldsymbol{x}_{L+U})$,
- $M = (\Theta, \boldsymbol{\pi}, \widetilde{\boldsymbol{\mu}}_c^{(p)}, \widetilde{\Sigma}_c^{(p)}, \widetilde{\mu}^{(n-p)}, \widetilde{\Sigma}^{(n-p)})$ where $\boldsymbol{\pi}$ is the estimated prior distribution on the classes.

After some algebra, it can be shown that (9) is equal to

$$\begin{aligned}
&\sum_{j=1}^{L} \Big(\log(p(c_j)) + \log|\Theta| - \frac{1}{2}\log(2\pi e)^p |\widetilde{\Sigma}_{c_j}^{(p)}| \\
&-\frac{1}{2}\log(2\pi e)^{n-p}|\widetilde{\Sigma}^{(n-p)}| \\
&-\frac{1}{2}(\Theta^{(p)}\boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}_{c_j}^{(p)})^{\mathrm{T}}(\widetilde{\Sigma}_{c_j}^{(p)})^{-1}(\Theta^{(p)}\boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}_{c_j}^{(p)}) \\
&-\frac{1}{2}(\Theta^{(n-p)}\boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}^{(n-p)})^{\mathrm{T}}(\widetilde{\Sigma}^{(n-p)})^{-1}(\Theta^{(n-p)}\boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}^{(n-p)})\Big) \\
&+\sum_{j=L+1}^{L+U}\sum_{c} \gamma_j(c)\Big(\log(p(c)) + \log|\Theta| - \frac{1}{2}\log(2\pi e)^p |\widetilde{\Sigma}_{c}^{(p)}| \\
&-\frac{1}{2}\log(2\pi e)^{n-p}|\widetilde{\Sigma}^{(n-p)}| \\
&-\frac{1}{2}(\Theta^{(p)}\boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}_{c}^{(p)})^{\mathrm{T}}(\widetilde{\Sigma}_{c}^{(p)})^{-1}(\Theta^{(p)}\boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}_{c}^{(p)}) \\
&-\frac{1}{2}(\Theta^{(n-p)}\boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}^{(n-p)})^{\mathrm{T}}(\widetilde{\Sigma}^{(n-p)})^{-1}(\Theta^{(n-p)}\boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}^{(n-p)})\Big)
\end{aligned}$$

where all the parameters correspond to the "new" model M, except for $\gamma_j(c) = p(c_j = c|\boldsymbol{x}_j)$, the posterior distribution of the j-th sample on the unlabeled data, which is computed with the "old" model M_0. After re-arranging terms, we come up with the objective function

$$\begin{aligned}
&(L+U)\log|\Theta| + \sum_{c}\hat{N}_c\Big[\log p(c) - \frac{1}{2}\log|\widetilde{\Sigma}_c^{(p)}| \\
&-\frac{1}{2}\sum_{j=1}^{L+U} z_j(c)(\Theta^{(p)}\boldsymbol{x}_j - \widetilde{\mu}_c^{(p)})^{\mathrm{T}}(\widetilde{\Sigma}_c^{(p)})^{-1}(\Theta^{(p)}\boldsymbol{x}_j - \widetilde{\mu}_c^{(p)})\Big] \\
&-\frac{L+U}{2}\log(2\pi)^n - \frac{L+U}{2}\log|\widetilde{\Sigma}^{(n-p)}| \\
&-\frac{1}{2}\sum_{j=1}^{L+U}(\Theta^{(n-p)}\boldsymbol{x}_j - \widetilde{\mu}^{(n-p)})^{\mathrm{T}}(\widetilde{\Sigma}^{(n-p)})^{-1}(\Theta^{(n-p)}\boldsymbol{x}_j - \widetilde{\mu}_c^{(n-p)})
\end{aligned} \tag{10}$$

where $z_j(c) = \mathbf{I}(j \le L \wedge c_j = c) + \mathbf{I}(j > L)\gamma_j(c)$ is the class posterior of the j-th sample given the observation (which, in the case of labeled examples, contains the true class), and $\hat{N}_c = \sum_j z_j(c)$ is the total "soft" count for class c. As can be easily seen, (10) has the same form as the objective function (2) of supervised HLDA, with the only difference that the "hard" count N_c got replaced by the "soft" count $\hat{N}_c$, computed using the models in the previous EM iteration. Thus, the EM update equations have the same form as (3)-(6), with the only difference that the empirical averages in the computation of $\widetilde{\boldsymbol{\mu}}_c^{(p)}, \widetilde{\Sigma}_c^{(p)}$ should involve $z_j(c)$ (note that $\widetilde{\boldsymbol{\mu}}^{(n-p)}, \widetilde{\Sigma}^{(n-p)}$ remain as before). Hence,

$$\widetilde{\boldsymbol{\mu}}_c^{(p)} = \Theta^{(p)}\frac{1}{\hat{N}_c}\sum_{j:c_j=c} z_j(c)\boldsymbol{x}_j \triangleq \Theta^{(p)}\hat{\boldsymbol{\mu}}_c \tag{11}$$

$$\begin{aligned}
\widetilde{\Sigma}_c^{(p)} &= \frac{1}{\hat{N}_c}\sum_{j:c_j=c} z_j(c)(\Theta^{(p)}\boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}_c^{(p)})(\Theta^{(p)}\boldsymbol{x}_j - \widetilde{\boldsymbol{\mu}}_c^{(p)})^{\mathrm{T}} \\
&\triangleq \Theta^{(p)}\hat{\Sigma}_c(\Theta^{(p)})^{\mathrm{T}}
\end{aligned} \tag{12}$$

Furthermore, since prior $p(c)$ sum to 1, differentiating (10) with respect to $p(c)$ gives its update equation:

$$p(c) = \frac{\hat{N}_c}{L+U} \tag{13}$$

Finally, by plugging-in the updated parameters into the objective function, we obtain a function of Θ, whose maximization has to involve a gradient-descent method, as before:

$$\begin{aligned}
&\mathcal{L}^* = \\
&(L+U)\log|\Theta| + \sum_{c}\hat{N}_c\Big[\log p(c) - \frac{1}{2}\log|\Theta^{(p)}\hat{\Sigma}_c(\Theta^{(p)})^{\mathrm{T}}|\Big] \\
&-\frac{L+U}{2}\log|\Theta^{(n-p)}\Sigma(\Theta^{(n-p)})^{\mathrm{T}}| - \frac{n(L+U)}{2}\log(2\pi e)
\end{aligned}$$

4. Experiments

First we aim to test the semi-supervised version of HLDA under different degrees of class "overlap" in the original space, which is reflected by the Bayes error (which we approximate by computing the classification error rate in the original space). Three conditions are designated as "Hard", "Medium" and "Easy", depending on the degree of overlap between the classes. In terms of classification difficulty, the approximate Bayes error rates are 55.48%, 19.57% and 1.66%, respectively.

Artificial data are generated under their respective conditions. For each condition, 100 data sets of 15-dimensional full covariance Gaussian data are generated for 5 classes, with each data set containing 1000 training (labeled) samples per class and 2000 test samples per class.

For each data set, an HLDA transform is trained to project the original 15-dimensional data into a 4-dimensional space; Gaussian modeling is then done in this lower-dimensional space. The average error rate over the 100 data sets is then reported for each condition.

We have 3 experimental setups for each condition. For each experiment a different amount of training data (50, 100 or 250 points per class) is used as labeled data, while the rest of the training data is stripped of its labels to be used as unlabeled data. The semi-supervised version of HLDA is then used to compute a transform into 4-dimensional space and the average error rate (over the 100 data sets of each condition) is then reported for each different amount of labeled and unlabeled data used.

In Figure 1, we plot the average error rate for the 3 experiments of condition "Easy". The HLDA average error rate using only labeled data is shown in the figure as the leftmost point of each curve. The average error rates can also be plotted in three dimensional space, with the z-axis showing the average error rate and the x,y axes showing the number of the labeled and unlabeled data respectively.

As the figure shows for condition "Easy", semi-supervised HLDA provides the largest gain when there is only a small amount of labeled data: semi-supervised HLDA using 4750

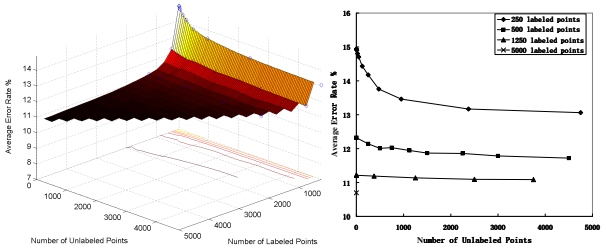

Figure 1: Experiments on condition "Easy".

points of unlabeled data gives an average error rate of 13.06%, compared to 14.93% of HLDA using 250 points of labeled data (50 labeled points per class). The figure also shows that (i) having more labeled data is always better than having the same amount of unlabeled data (as expected), and (ii) as more unlabeled data become available the improvement slows down.

In Figure 2, we show the results of condition "Medium" (left side) and condition "Hard" (right side). While for condition "Medium" we see a similar trend as condition "Easy" with less relative improvement, for condition "Hard" we discover that semi-supervised HLDA does not give any significant gain, and in some cases it deteriorates performance. This may be due to the fact that, when the class-conditional distributions have significant overlap, the "cluster assumption" [8] does not hold any more.

We now proceed to test the semi-supervised HLDA on a real data set. The data set we used is the Deterding vowel data [14]. The speech data of eleven vowels uttered by multiple speakers was collected at a 10-kHz sampling rate and lowpass filtered at 4.7 kHz, then the signal was transformed to a 10-dimensional vector. Training data consist of 528 vectors from eight speakers, and test data consist of 462 vectors from seven speakers.

# of Unlabeled Points	# of Labeled Points					
	198	264	330	396	462	528
0 (HLDA)	**67.32**	57.36	54.98	53.68	50.65	50.87
33	75.97	**56.28**	**52.16**	51.08	**49.57**	-
66	79.87	57.79	54.55	51.95	**49.57**	-
99	79.00	58.66	53.25	**50.87**	-	-
132	79.87	58.44	53.25	52.16	-	-
165	77.92	59.96	54.55	-	-	-
198	79.87	58.87	57.14	-	-	-
231	79.00	59.31	-	-	-	-
264	78.79	59.52	-	-	-	-
297	79.00	-	-	-	-	-
330	79.44	-	-	-	-	-

Table 1: Error rates on the Deterding vowel data.

For each experiment, we remove the labels of a certain amount of training data, then use all or part of it as unlabeled data for semi-supervised HLDA training of a projection transform into 9 dimensional space. The resulting error rates are shown in Table 1. One thing to note from this table is that only when the labeled data have reached a certain amount, the semi-supervised HLDA uses the unlabeled data efficiently and provides a gain. We plan to analyze these results in more detail in a subsequent publication.

5. Concluding Remarks

From the above experiments we observe that, under certain conditions, semi-supervised HLDA gives improvements over supervised HLDA by efficiently utilizing unlabeled data. Future work will focus on applying semi-supervised HLDA to a large vocabulary speech recognition task.

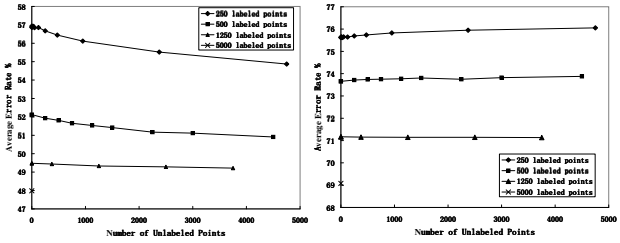

Figure 2: Experiments on condition "Medium" and "Hard".

6. Acknowledgments

We would like to thank Professor Sanjeev Khudanpur for all the insightful discussions.

7. References

[1] N. Kumar and A. G. Andreou, "Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition," *Speech Comm.*, vol. 26, pp. 283–297, 1998.

[2] M. Gales and S. Young, *The Application of Hidden Markov Models in Speech Recognition*, NOW Pub., 2008.

[3] G. Saon, M. Padmanabhan, R. Gopinath, and S. Chen, "Maximum likelihood discriminant feature spaces," in *Proc. of the Acoustics, Speech, and Signal Processing*, 2000, pp. 1129–1132.

[4] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. of the Royal Stat. Soc., Series B (Methodological)*, vol. 39, pp. 1–38, 1977.

[5] S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland, *The HTK Book*, University of Cambridge, 2006.

[6] N. Campbell, "Canonical variate analysis – a general formulation," *Australian J. of Stat.*, vol. 26, pp. 86–96, 1984.

[7] H. Zhou, D. Karakos, S. Khudanpur, A. G. Andreou, and C. E. Priebe, "On projections of Gaussian distributions using maximum likelihood criteria," in *Proc. of the Information Theory and Applications Workshop*, La Jolla, California, February 2009.

[8] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning*, MIT Press, 2006.

[9] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," *Mach. Learning*, vol. 39, no. 2, pp. 103–134, 2000.

[10] D. Cai, X. He, and J. Han, "Semi-supervised discriminant analysis," in *Proc. of the IEEE Int'l Conf. on Comp. Vision (ICCV)*, Rio De Janeiro, Brazil, October 2007.

[11] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. of Mach. Learning Research*, vol. 7, no. Nov, pp. 2399–2434, 2006.

[12] M. Gales, "Semi-tied covariance matrices for hidden Markov models," *IEEE Trans. on Speech and Audio Proc.*, vol. 7, no. 3, pp. 272–281, May 1999.

[13] F. Jelinek, *Statistical Methods for Speech Recognition*, MIT Press, 1997.

[14] D. Deterding, M. Niranjan, and A. J. Robinson, "Vowel recognition (deterding data), berkeley, ca. [online]. http://ftp.ics.uci.edu/pub/machinemented/connectionist-bench/vowel/," 2004.