MODELING WITH A SUBSPACE CONSTRAINT ON INVERSE COVARIANCE MATRICES

Scott Axelrod, Ramesh Gopinath, Peder Olsen

IBM T.J. Watson Research Center, P.O. Box 218,
Yorktown Heights, NY 10598, USA
{*axelrod,rameshg,pederao*}*@us.ibm.com*

ABSTRACT

We consider a family of Gaussian mixture models for use in HMM based speech recognition system. These "SPAM" models have state independent choices of subspaces to which the precision (inverse covariance) matrices and means are restricted to belong. They provide a flexible tool for robust, compact, and fast acoustic modeling. The focus of this paper is on the case where the means are unconstrained. The models in the case already generalize the recently introduced EMLLT models, which themselves interpolate between MLLT and full covariance models. We describe an algorithm to train both the state-dependent and state-independent parameters. Results are reported on one speech recognition task. The SPAM models are seen to yield significant improvements in accuracy over EMLLT models with comparable model size and runtime speed. We find a 10% relative reduction in error rate over an MLLT model can be obtained while decreasing the acoustic modeling time by 20%.

1. INTRODUCTION

A most popular approach to state of the art speech recognition systems uses continuous parameter Hidden Markov Models (HMMs) with the probability density function for each state represented by a Gaussian mixture model (GMM). In practical systems, the state model must not only provide robust parameter estimation, but also have low memory and computational time overhead. A general approach to achieving this goal is to determine a family of Gaussian models to which each component of the GMM for each state is constrained to belong. Much progress has been made over the past few years as various kinds of constraints on the Gaussian components have been explored. In particular, the Extended Maximum Likelihood Linear Transformation model introduced in [1, 2] constrains the precision matrix (i.e. the inverse covariance matrix) of each Gaussian to be a linear combination of D rank one matrices, where $d \leq D \leq d(d+1)/2$ for when the features are in in $\mathbf{R}^d$. When $D = d$, the EMLLT models reduce to models where the Gaussian are constrained to be diagonal when written in a particular basis (also known as MLLT models [3] or models with global semi-tied covariances [4]). The EMLLT models provide quite a general and flexible class of models, interpolating from MLLT models (when $D = d$) to full covariance models (when $D = d(d+1)/2$).

One can generalize still further and consider the case when the precision matrix is constrained to lie in a subspace of the space of all symmetric matrices and the mean parameters (or else the linear forms in the exponents of the Gaussians) lie in a subspace of $\mathbf{R}^d$. In this case, D is now free to range from 1 to $d(d+1)/2$. These SPAM models (models with a subspace constraint on the precisions and means) subsume both the EMLLT models as well as the LDA and HDA models [5, 6] (as will be described elsewhere [7]). For example, the EMLLT models are obtained as the special case when the means are unconstrained and the precision matrices are constrained to a subspace consisting of rank one matrices.

2. THE SPAM MODEL (FOR UNCONSTRAINED MEANS)

We will consider a system with C states and acoustic vectors in $\mathbf{R}^d$. We assume given the number, M_α, of components of the GMM for each state α, $1 \leq \alpha \leq C$. Let $M = \sum_{\alpha=1}^{C} M_\alpha$ be the total number of Gaussians of the system. We will number the individual Gaussians from 1 to M and let $\mathcal{M}(\alpha)$ be the subset of Gaussians associated with state α, so $\mathcal{M}(\alpha)$ has cardinality $M(\alpha)$. The GMM for state α has the form

$$p(x|\alpha) = \sum_{j \in \mathcal{M}(\alpha)} \pi_j \; p(x|j,\alpha) \tag{1}$$

$$p(x|j,\alpha) = \det\left(\frac{P_j}{2\pi}\right)^{1/2} e^{-\frac{1}{2}(x-\mu_j)^T P_j (x-\mu_j)} \tag{2}$$

The priors π_j are non-negative numbers satisfying $\sum_{j \in \mathcal{M}(\alpha)} \pi_j = 1$. The means μ_j are vectors in $\mathbf{R}^d$. For the general SPAM model the means are constrained to belong to a subspace of $\mathbf{R}^d$, but for simplicity in this paper we shall keep to the case where the means are unconstrained. The precision (inverse covariance) matrices P_j are symmetric $d \times d$ matrices. In addition to the requirement of positive definiteness, the SPAM model constrains the P_j to lie in some D dimensional subspace of the space of symmetric matrices, which is spanned by a collection of matrices $\{S_k\}_{k=1}^{D}$. The matrices S_k are symmetric $d \times d$ matrices and are not required to be positive definite. The precision matrices may be written as

$$P_j = \sum_{k=1}^{D} \lambda_j^k S_k \quad . \tag{3}$$

In summary the SPAM model with unconstrained means and precision matrices belonging to a D dimensional space consists of tied (state independent) parameters and untied (state dependent) parameters:

$$\Theta_{tied} = \{S_k; k = 1, ..., D\} \tag{4}$$

$$\Theta_{untied} = \{\pi_j, \mu_j, \lambda_j^k; j = 1, ..., M, k = 1, ..., D\} \quad . \tag{5}$$

2.1. Model Complexity

The total number of parameters, $nParams$ of our model is

$$nParams = Dd(d+1)/2 + M(1+d+D) \ . \tag{6}$$

When the dimension D of the precision subspace is small compared to both $d(d+1)/2$ and M, we achieve significant compression over the number of parameters $M(1+d+d(d+1)/2)$ required for a full covariance model. Similarly, the computational time required to evaluate all of the probabilities $p(x|\alpha)$ is a significant reduction over that for a full covariance model. The time can be minimized if we rewrite each Gaussian as:

$$\begin{aligned} p(x|j,\alpha) &= e^{b_j + L_j^T x + \sum_{k=1}^D \lambda_j^k f_k} && (7)\\ b_j &= \log\det\left(\frac{P_j}{2\pi}\right)^{1/2} - \frac{1}{2}\mu_j^T P_j \mu_j && (8)\\ L_j &= P_j \mu_j && (9)\\ f_k &= -\frac{1}{2} x^T S_k x \ , && (10) \end{aligned}$$

where $j = 1, ..., M$ and $k = 1, ..., D$. The biases $b_j \in \mathbf{R}$ and the linear weights $L_j \in \mathbf{R}^d$ may be computed offline. The "quadratic features" f_k may be precomputed once for all of the Gaussians. The total number of flops needed to evaluate all the Gaussians for a single input vector (up to terms of order M and dD) is then

$$time = Dd^2 + 2M(d+D) \ . \tag{11}$$

The first term is the time to evaluate $\{f_k\}_{k=1}^D$ and the second term is the time to evaluate the Gaussian exponents. This is a significant savings over the time required to evaluate a full covariance model (which is on the order of Md^2). Further saving can be obtained if the linear weights L_j are constrained to be in a δ dimensional subspace of $\mathbf{R}^d$ (so the means are in a j dependent subspace). In that case the time to evaluate the exponents reduces to $2M(\delta+D)$.

3. PARAMETER ESTIMATION

Given a collection of training data vectors $\{x_{\alpha,i}\}_{i=1}^{N(\alpha)}$ for each state α, our goal is to train all of the parameters $\Theta = (\Theta_{tied}, \Theta_{untied})$ of the model so as to maximize the total data log likelihood $L(\Theta)$, which is just the sum of the $\log\ p(x_{\alpha,i}|\alpha)$ over all α and i. For an arbitrary precision subspace dimension D, this can be a very computationally expensive problem.

To address the problem, we begin by noting that the likelihood of the maximum likelihood model increases monotonically as D increases from 1 to $D = d(d+1)/2$. In the latter case, the SPAM model just reduces to an unconstrained full covariance model (which may be trained by a standard EM algorithm). Although, there are circumstances where the SPAM model provides better accuracy than the full covariance model because there are fewer parameters to overtrain, we shall adopt the view that a SPAM model with $D < d(d+1)/2$ provides a compressed form of a full covariance model with substantially reduced memory and speed costs.

We will train the models for arbitrary D as follows. First we choose a seed full covariance model. Second, as described in section 3.2, we choose a set of tied parameters $\Theta_{tied} = \{S_k\}$ which span a subspace which we expect to capture (in the sense of total likelihood) much of the information contained in the full covariance precision matrices. Finally, we use a generalization of the EM algorithm, described in section 3.1, to train the tied parameters.

We begin by recalling that the EM theorem [8] gives a bound on the difference in likelihood between an "old" set of model parameters $\hat{\Theta}$ and an "updated" parameter set Θ. The bound, which can be derived by a single application of Jensen's inequality (expressing the concavity of the logarithm), and a little manipulation states that:

$$L(\Theta) - L(\hat{\Theta}) \quad \geq \quad \frac{1}{2}Q(\Theta;\hat{\Theta}) - \frac{1}{2}Q(\hat{\Theta};\hat{\Theta}), \text{ where } (12)$$

$$\begin{aligned} Q(\Theta;\hat{\Theta}) &= \sum_{\alpha=1}^{C} N(\alpha) \sum_{j\in\mathcal{M}(\alpha)} \tilde{\pi}_j G(P_j;\ \Sigma_j) + \tilde{\pi}_j \log \pi_j && (13)\\ G(P;\Sigma) &= \log(\det(P)) - \mathrm{Tr}(\Sigma P) \ . && (14) \end{aligned}$$

For a given component j for state α (i.e. $j \in \mathcal{M}(\alpha)$), $\tilde{\pi}_j$ is the average over the samples $x_{\alpha,i}$, $1 \leq i \leq N(\alpha)$ for state α of the posterior probability that the sample is selected from component j:

$$\begin{aligned} \tilde{\pi}_j &= \frac{1}{N(\alpha)} \sum_{i=1}^{N(\alpha)} \gamma_{ij} && (15)\\ \gamma_{ij} &= \gamma_{ij}(\hat{\Theta}) = p(j|x_{\alpha,i},\alpha) = \frac{\hat{\pi}_j p(x_{\alpha,i}|j,\alpha)}{p(x_{\alpha,i}|\alpha)} \ . && (16) \end{aligned}$$

The quantity Σ_j in (13) is given by

$$\Sigma_j \quad = \quad \tilde{\Sigma}_j + (\mu_j - \tilde{\mu}_j)(\mu_j - \tilde{\mu}_j)^T \tag{17}$$

where $\tilde{\mu}_j$ and $\tilde{\Sigma}_j$ are just the (E-step) means and covariances of the distribution ρ^j on $\mathbf{R}^d$ which gives sample $x_{\alpha,i}$ probability

$$\rho_i^j \quad = \quad \frac{1}{N(\alpha)}\left(\frac{\gamma_{ij}}{\tilde{\pi}_j}\right) \ . \tag{18}$$

3.1. Training The Untied Parameters

For a fixed choice of tied parameters $\Theta_{tied} = \{S_k\}_{k=1}^D$, we will train the untied parameters by applying the EM algorithm. To do so, we begin by picking a seed model $\Theta^{(0)}$ (whose tied parameters may differ from Θ_{tied}). Given $\Theta^{(i)}$, we set $\Theta_{tied}^{(i+1)}$ to be Θ_{tied}. The parameters in $\Theta_{untied}^{(i+1)}$ are then chosen (in the M-step) so as to maximize the function $Q(\Theta^{(i+1)};\Theta^{(i)})$ defined by (13). A simple argument using (12) guarantees that the procedure converges to a locally optimal solution.

For the M-step calculation of Θ_{untied} that maximizes $Q(\Theta;\hat{\Theta})$, we proceed as follows. First, note that, us usual, $\pi_j = \tilde{\pi}_j$ is the unique solution that minimizes the contribution of $\sum_j \tilde{\pi}_j \log\ \pi_j$ to (13). Second, note that since we are leaving the means unconstrained, $\mu_j = \tilde{\mu}_j$ is, independent of the value of P_j, the unique solution that minimizes (in fact sets to 0) the quadratic term $(\mu_j - \tilde{\mu}_j)^T P_j (\mu_j - \tilde{\mu}_j)$. We remark that calculations are more difficult when the means are constrained because the optimal choice of μ_j depends on P_j; it is the point in the mean subspace which is closest to μ_j, as measured by the inner product determined by P_j.

So far, everything we have said has been standard EM. The only modification necessary now is that, rather than simply taking $P_j = \Sigma_j^{-1} = \tilde{\Sigma}_j^{-1}$, we need to train the λ_j^k in (3) so as to

maximize (14). To do this, we shall use the Polak-Ribiere conjugate gradient algorithm [9]. One subtlety is that the λ's are constrained by the requirement that P_j be positive definite. We find that this algorithm converges very rapidly and takes little computational time when we perform the line searches rapidly using the following fact. For P, R, and Σ symmetric matrices with P and Σ positive definite, the function

$$f(t) \quad = \quad G(P + tR; \Sigma) - G(P; \Sigma) \tag{19}$$

has the value

$$f(t) \quad = \quad -t\beta + \sum_{p=1}^{d} \log(1 + t\ w_p) \ , \tag{20}$$

where $\beta = \mathrm{Tr}(\Sigma R)$ and the w_i are the eigenvalues of the operator $P^{-1/2} R P^{-1/2}$ (which equal the generalized eigenvalues of the pair (R, P)). Furthermore, the function $f(t)$ is convex and one can determine a priori finite intervals about 0 which must contain the maximum (i.e. the solution of $f'(t) = 0$ for which $P + tR$ is positive definite).

One final note, in order to carry out our optimization, we need to have a valid initial choice of the λ_j^k, i.e. a choice which yields positive definite precision matrices. In the next subsection, we will choose S_D to be a positive definite matrix, so that one valid initial condition is to take $\lambda_j^D = 1$ and all other λ_j^k to be zero.

3.2. Training The Tied Parameters

The goal of this section is to choose tied parameters $\Theta_{tied} = \{S_k\}$ of our SPAM model for which there is a choice of untied parameters Θ_{untied} so that the total model likelihood on the training data is as close to that of the given full covariance model as possible. Fortunately, the bound (12) ensures that we will be doing a pretty good job if we choose our model parameters Θ to maximize the function $Q(\Theta; \hat{\Theta})$ given by (13), where $\hat{\Theta}$ is the set of parameters of the full covariance model.

One approach is to make some initial choice of Θ_{tied}. Then one can iterate the procedure of using the M-step algorithm of subsection 3.1 to find the optimal Θ_{untied}, and then hill climb in the direction of the gradient $\nabla_{\Theta_{tied}} Q(\Theta; \hat{\Theta})$ to find an improved choice of Θ_{tied}. One can, in principle, use (19) and (20) to perform the line search efficiently.

Unfortunately, the above algorithm can be rather compute intensive because the positive definiteness restriction on the precision matrices of all the Gaussians can constrain the line search to making very small updates of the $\{S_k\}$ at each iteration. Therefore, to implement the algorithm in practice, a great deal of computational time may be saved if one finds an initial choice of Θ_{tied} which is a good approximation to the optimal choice. We will now describe a fast algorithm for finding such a good approximation. The experiments of section 4 use this "initial" choice with no further updates. As we will see in section 4, this choice of "initial" Θ_{tied} works well in practice even when foregoing iterative updates.

We begin by performing the Taylor expansion to quadratic order of (14) about $P = \Sigma^{-1}$, the global maximum when P is unconstrained:

$$G(P; \Sigma) \quad \approx \quad G(\Sigma^{-1}; \Sigma) - 0.5 ||P - \Sigma^{-1}||_{\Sigma}^2 \ , \tag{21}$$

where the norm in the quadratic term is the matrix norm on $d \times d$ symmetric matrices coming from the inner product

$$< U, V >_{\Sigma} \quad = \quad \mathrm{Tr}(\Sigma\ U\ \Sigma\ V) \ . \tag{22}$$

Taking π_j and μ_j to have their optimal values ($\tilde{\pi}_j$ and $\tilde{\mu}_j$), keeping only terms up to quadratic order in the approximation for G, and dropping terms constant with respect to Θ, we may write the following approximation to the function $Q(\Theta; \hat{\Theta})$ in (13):

$$Q^{approx,1}(\Theta; \hat{\Theta}) \quad = \quad -0.5 \sum_{j=1}^{M} c_j ||P_j - \Sigma_j^{-1}||_{\Sigma_j} \tag{23}$$

$$c_j \quad = \quad N(\alpha) \tilde{\pi}_j \qquad \text{for } j \in \mathcal{M}(\alpha) \ . \tag{24}$$

Although the $\{S_k, \lambda_j^k\}$ that maximizes (23) may be found by a generalization of one algorithm for calculating singular value decompositions, we will not pursue this further here for lack of space. Instead, we will make one further approximation which will reduce our problem to a genuine singular value decomposition problem. Namely, we will replace the j dependent norms appearing in (23) by the j-independent norm associated to the mean matrix

$$\bar{\Sigma} \quad = \quad \sum_j c_j \Sigma_j \Big/ \sum_j c_j \ . \tag{25}$$

Note that $\bar{\Sigma}$ is similar to the within class covariance matrix W.

With the replacement above, our new approximation to the function $Q(\Theta; \hat{\Theta})$ becomes

$$Q^{approx,2}(\Theta; \hat{\Theta}) \quad = \quad -0.5 \sum_{j=1}^{M} c_j ||P_j - \Sigma_j^{-1}||_{\bar{\Sigma}} \ . \tag{26}$$

The problem of maximizing $Q^{approx,2}$ is just the problem of finding a set $\{S_k\}$ which span a subspace $\mathcal{V}$ of the space of symmetric matrices with the property that the weighted sum of the distances (in the norm $|| \cdot ||_{\bar{\Sigma}}$) of the Σ_j^{-1} to $\mathcal{V}$ is minimized. By mapping the Σ_j^{-1} to vectors $v_j \in \mathbf{R}^{d(d+1)/2}$ in such a way that the inner product $< \cdot, \cdot >_{\bar{\Sigma}}$ on symmetric matrices corresponds to the ordinary inner product on $\mathbf{R}^{d(d+1)/2}$, the problem of finding the $\{S_k\}$ reduces to finding the top D singular vectors of the matrix

$$V = \sum_j c_j v_j v_j^T \ . \tag{27}$$

Note that the λ_j^k are chosen at this stage so that P_j in (3) is the projection of Σ_j^{-1} onto the subspace $\mathcal{V}$. The P_j obtained are not guaranteed to be positive definite at this stage, but this is OK because they will later be trained by the technique of section 3.2.

The choice of the $\{S_k\}$ above seems a natural one since the $\{S_k\}$ are intended to capture as much of the information contained in Σ_j^{-1} as possible. We remark that we have found that the use of the norm on symmetric matrices associated with $\bar{\Sigma}$, which we were lead to by approximating the total likelihood, leads to improvements in error rates over the naive choice of the Frobenius norm (associated with the identity matrix).

For reasons mentioned at the end of section 3.1, we would like to ensure that S_D is positive definite. To do so, we will only take S_k to be the singular vectors described above when $k < D$. We take S_D to be the mean precision matrix:

$$\bar{P} = \sum c_j P_j \Big/ \sum c_j \ . \tag{28}$$

4. EXPERIMENTAL RESULTS

We report on experiments using the same speech recognition system and "car" test set reported on in [10] and [1, 2]. The test set was recorded in a car environment and consisted of a total of 73743 words divided into four tasks. Each task was recorded in a car traveling at speeds ranging from 0 to 60 miles per hour. The system used 39 dimensional feature vectors consisting of 13 dimensional cepstral vectors with corresponding first and second order derivatives. Grammars with a medium sized vocabulary were used. The acoustic model had 89 phonemes and a total of 680 states. All systems reported on here and for the EMLLT models in [1, 2] use a total of 10253 Gaussians. In [1, 2] results were also reported on for MLLT models with larger numbers of total Gaussians in order to verify that the EMLLT models had significantly smaller error rates than MLLT models with comparable numbers of parameters.

SPAM models were trained as described in section 3 with the $\{S_k\}$ taken to be the singular vectors optimizing the approximation (26). Experiments were performed with the weights c_j as in (24) and also with $c_j = \tilde{\pi}_j$ (i.e. with the data vectors reweighted so that the weight of the data vectors for each class is equal). Error rates generally did not depend significantly on the choice of weighting scheme. Results here are reported for the case when $c_j = \tilde{\pi}_j$.

Figure 1 reports word error rates for the SPAM model described above as a function of the dimension D of the precision subspace. For comparison, the circle in the figure plots the error rate obtained for a model with diagonal Gaussians. Also plotted for comparison are the error rates for the EMLLT models of the type describe in [1, 2]. Those EMLLT models are just SPAM models with S_k equal to the rank one matrices $a_k a_k^T$, where the a_k are the rows of the MLLT matrices obtained when various subgroups of HMM states are tied together. The value $D = d = 39$ is indicated by the vertical solid line. The point on the EMLLT curve which is on this line gives the error rate for an ordinary MLLT model. This is the minimum allowed value of D for the EMLLT model (otherwise one can not obtain non-singular precision matrices), but the SPAM models allow D to be as small as 1.

The reader will observe that the SPAM model yields significant error reduction over the EMLLT model with comparable precision subspace dimension (and hence number of parameters, since $M >> d(d+1)/2$). One interesting comparison is that the SPAM model for $D = 21$ has an error rate of 2.53 compared to an error rate of 2.84 for the MLLT model (EMLLT model with $D = 39$), giving a 10.9% reduction in error rate while at the same time achieving an (approximately) 20% reduction in time required to do the decoding.

The error rate in Figure 1 when $D = 1$ is 3.96%, a suprisingly good result. For the case of $D = 1$, one can actually find a (local) maximum of the full likelihood function relatively easily. In this case we found an error rate of 3.86%, showing that the approximation $S_1 = \bar{P}$ is actually quite good.

5. CONCLUSION

The SPAM models are a very general and flexible class of models which subsume many well-known Gaussian modeling techniques. In this paper we have presented algorithms to train SPAM models for the case when the mean subspace is unconstrained. Compared with MLLT and EMLLT models we are able to obtain improvements in performance while at the same time lowering computational cost. It remains to be seen how well these improvements hold up when the SPAM models are further generalized to include other techniques. Further results will be reported on elsewhere [7].

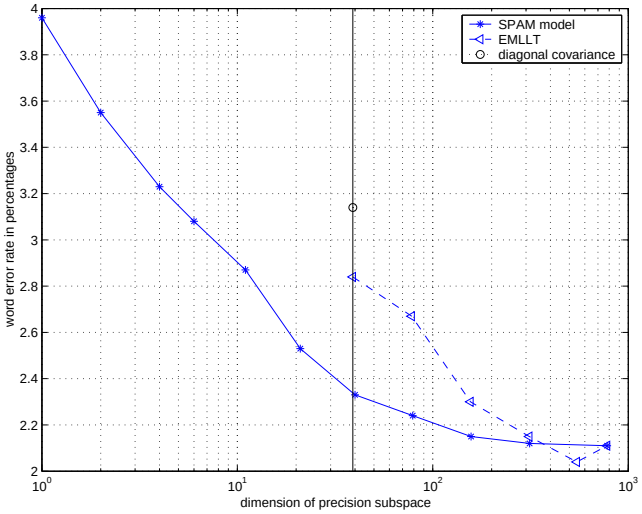

Fig. 1. WER as a function of number of covariance components.

6. REFERENCES

[1] P. Olsen and R. A. Gopinath, "Modeling inverse covariance matrices by basis expansion," in *Proc. ICASSP*, Orlando, Florida, 2002.

[2] P. Olsen and R. A. Gopinath, "Modeling inverse covariance matrices by basis expansion," *Transactions in Speech and Audio Processing*, 2001, submitted.

[3] R. A. Gopinath, "Maximum likelihood modeling with gaussian distributions for classification," in *Proc. ICASSP*, Seattle, USA, 1998, vol. II, pp. 661–664.

[4] M. J. F. Gales, "Semi-tied covariance matrices for hidden Markov models," *IEEE Transactions in Speech and Audio Processing*, 1999.

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

[6] G. Saon, M. Padmanabhan, R. Gopinath, and S. Chen, "Maximum likelihood discriminant feature spaces," in *Proc. ICASSP*, 2000.

[7] S. Axelrod, R.A. Gopinath, and P. Olsen, in preparation.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, 1977.

[9] E. Polak, *Computational Methods in Optimization: A Unified Approach*, Academic Press, 1971.

[10] S. Deligne, E. Eide, R. Gopinath, D. Kanevsky, B. Maison, P. Olsen, H. Printz, and J. Sedivy, "Low–resource speech recognition of 500 word vocabularies," in *Proceedings of the Sixth European Conference on Speech Communication and Technology*, Aarhus, Denmark, September 2001.