MIXTURES OF INVERSE COVARIANCES

Vincent Vanhoucke[†⋆]*, Ananth Sankar*[⋆]

[†] Department of Electrical Engineering, Stanford University, Stanford, CA
[⋆] Nuance Communications, 1380 Willow Road, Menlo Park, CA

ABSTRACT

We introduce a model that approximates full and block-diagonal covariances in a Gaussian mixture, while reducing significantly both the number of parameters to estimate and the computations required to evaluate the Gaussian likelihoods. The inverse covariance of each Gaussian is expressed as a mixture of a small set of prototype matrices. Estimation of both the mixture weights and the prototypes is performed using maximum likelihood estimation. Experiments on a variety of speech recognition tasks show that this model significantly outperforms a diagonal covariance model, while using the *same number* of Gaussian-dependent parameters.

1. INTRODUCTION

When using a Gaussian mixture model (GMM) to represent a distribution, it is common to impose some constraints to the structure of the covariance matrices in order to reduce the number of parameters to estimate, as well as the amount of computations needed to evaluate the GMM. In general, speech recognition systems constrain the covariances to be diagonal, using the assumption that distinct feature components are uncorrelated.

Typical speech input features are weakly correlated because the final stage of the front-end processing is in general some form of whitening of the feature vector. This can be achieved through a Discrete Cosine Transform that approximates the Karhunen-Loève transform [1] in the case of standard Mel filterbank cepstral coefficients (MFCCs), or linear discriminant analysis [2]. Correlations between feature components are also implicitly modeled by the different modes of the mixture itself.

However, explicit modeling of the correlations generally leads to better models [3], both in terms of improving recognition accuracy and reducing the size of the mixtures required to model the acoustics. There is thus a strong interest in making full covariance modeling practical both in terms of the number of parameters to estimate as well as computational efficiency.

2. MIXTURES OF INVERSE COVARIANCES

A GMM for a D-dimensional input vector $\boldsymbol{o}$, composed of N Gaussians with priors π_i, means $\boldsymbol{\mu}_i$ and covariances Σ_i can be expressed as:

$$f(\boldsymbol{o}) = \sum_{i=1}^{N} \pi_i \mathcal{N}(\boldsymbol{o}, \boldsymbol{\mu}_i, \Sigma_i)$$

A Mixture of Inverse Covariances is defined by set of K prototype symmetric matrices Ψ_k, such that for each Gaussian i there is a vector $\boldsymbol{\Lambda}_i$ with components $\lambda_{k,i}$ satisfying:

$$\Sigma_i^{-1} = \sum_{k=1}^{K} \lambda_{k,i} \Psi_k \tag{1}$$

Note that the mixture "weights" $\lambda_{k,i} \in \mathbb{R}$ are not constrained to be positive. Such modeling approaches have been investigated in the past, with various constraints imposed on the structure of the mixture. Semi-tied Covariances [4] and Factored Sparse Inverse Covariances [5] are instances of it with $L = 1, K = D$ and $\text{Rank}(\Psi_k) = 1$. The Semi-tied model was later extended to $K > D$ in [6].

In this paper, we allow arbitrary K, and use full-rank prototypes. This reduces the number of prototypes required to achieve the same modeling power as compared to the rank-one case. A similar model has recently been independently proposed [7]. In the following, we introduce a different mathematical treatment of the full-rank model. We constrain the Ψ_k to be positive definite, which will allow us to derive provably stable algorithms to estimate both the weights and prototype matrices using maximum likelihood.

A block-diagonal mixture can also be expressed in similar fashion by considering L covariance sub-blocks of dimensionality D_l:

$$\Sigma_{i,l}^{-1} = \sum_{k=1}^{K_l} \lambda_{k,i,l} \Psi_{k,l}$$

Since each of the L subspaces is independently modeled with distinct prototypes and weights, this leads to greater model flexibility as shown in Section 5. Without loss of generality, the following developments will focus on the full covariance (single subspace) model.

3. LIKELIHOOD COMPUTATION

The log-likelihood of a Gaussian for observation vector $\boldsymbol{o}$ can be written:

$$\begin{aligned}\mathcal{L}(\boldsymbol{o}) &= c - \frac{1}{2}(\boldsymbol{o}-\boldsymbol{\mu})^\top \Sigma^{-1}(\boldsymbol{o}-\boldsymbol{\mu}) \\ &= c - \frac{1}{2}\boldsymbol{\mu}^\top\Sigma^{-1}\boldsymbol{\mu} - \sum_{k=1}^{K}\frac{\lambda_k}{2}\boldsymbol{o}^\top\Psi_k\boldsymbol{o} + \boldsymbol{\mu}^\top\Sigma^{-1}\boldsymbol{o}\end{aligned}$$

The term $\frac{1}{2}\boldsymbol{\mu}^\top\Sigma^{-1}\boldsymbol{\mu}$ can be absorbed into the constant, $c' \equiv c - \frac{1}{2}\boldsymbol{\mu}^\top\Sigma^{-1}\boldsymbol{\mu}$. The vector $\boldsymbol{\omega} : \omega_k = \frac{1}{2}\boldsymbol{o}^\top\Psi_k\boldsymbol{o}$ is independent of the Gaussian and can be computed as an additional K-dimensional feature vector appended to $\boldsymbol{o}$. $\boldsymbol{\nu} = -\Sigma^{-1}\boldsymbol{\mu}$ is an L-dimensional vector, which leads to:

$$\boldsymbol{o}' = \begin{bmatrix}\boldsymbol{o}\\ \boldsymbol{\omega}\end{bmatrix} \quad \boldsymbol{\nu}' = \begin{bmatrix}\boldsymbol{\nu}\\ \boldsymbol{\Lambda}\end{bmatrix} \quad \mathcal{L}(\boldsymbol{o}) = c' - \boldsymbol{\nu}'^\top\boldsymbol{o}'$$

This computation requires $D+K$ sums and products, to be compared with $2D$ for a diagonal Gaussian. Note that K can be smaller than D, in which case the Gaussians are less expensive to evaluate than in the diagonal case.

The front-end overhead is limited to the computation of $\boldsymbol{\omega}$, which is of the order of $\frac{1}{2}KD^2$ multiplications using the Cholesky decomposition $L_kL_k^\top$ of Ψ_k:

$$\boldsymbol{\alpha}_k(\boldsymbol{o}) = 1/\sqrt{2}L_k^\top\boldsymbol{o} \;\Rightarrow\; \omega_k = \boldsymbol{\alpha}_k(\boldsymbol{o})^\top\boldsymbol{\alpha}_k(\boldsymbol{o})$$

Note that when using a block diagonal model, the front-end overhead is further reduced to at most $\frac{1}{2}K[max_l D_l]^2$.

4. ESTIMATION OF THE MODEL

In the following, we will sometimes represent a symmetric matrix A in vector form — noted $\boldsymbol{A}^\star$, constructed by stacking together the diagonal $\boldsymbol{a}_0$ and the super-diagonals $\boldsymbol{a}_i, i \in [1, D-1]$ multiplied by $\sqrt{2}$:

$$\boldsymbol{A}^\star = [\boldsymbol{a}_0^\top \sqrt{2}\boldsymbol{a}_1^\top \ldots \sqrt{2}\boldsymbol{a}_{D-1}^\top]^\top$$

The $\sqrt{2}$ factor ensures that: $\mathrm{Tr}(AB) = \boldsymbol{A}^{\star\top}\boldsymbol{B}^\star$. Using this convention, and denoting $P = [\boldsymbol{\Psi}_1^\star \ldots \boldsymbol{\Psi}_k^\star]$, we can write Equation 1 as $\boldsymbol{\Sigma}_i^{\star-1} = P\boldsymbol{\Lambda}_i$.

The sample covariance estimated from the observations $\boldsymbol{o}_t$ and priors $\gamma_{i,t}$ is:

$$\bar{\Sigma}_i = \sum_t \gamma_{i,t}(\boldsymbol{o}_t - \boldsymbol{\mu}_i)(\boldsymbol{o}_t - \boldsymbol{\mu}_i)^\top$$

Given the independent parameters $\pi_i, \boldsymbol{\mu}_i$, and the sample covariance $\bar{\Sigma}_i$, the parameters of the model (P, Λ), with $\Lambda = \{\boldsymbol{\Lambda}_1, \ldots, \boldsymbol{\Lambda}_N\}$, can be estimated jointly using the EM algorithm [8]. The maximization step of the algorithm corresponds here to the maximization of the auxiliary function:

$$Q(P,\Lambda) = \sum_{i=1}^{N}\pi_i\left[\log|\Sigma_i^{-1}| - \mathrm{Tr}\left(\Sigma_i^{-1}\bar{\Sigma}_i\right)\right]$$

Conditioned on the Ψ_k being positive definite, the functions $Q(P|\Lambda)$ and $Q(\Lambda|P)$ are both concave. In addition, the domains $\mathcal{L} : \Lambda/\{\forall i, P\boldsymbol{\Lambda}_i \succ 0\}$ and $\mathcal{P} : P/\{\forall i, P\boldsymbol{\Lambda}_i \succ 0\}$ are convex. Thus, the problem of jointly estimating P and Λ can be decomposed into two convex optimization problems [9] to be solved iteratively:

Maximize $Q(\Lambda\|P)$	Maximize $Q(P\|\Lambda)$
Subject to $\Lambda \in \mathcal{L}$	Subject to $P \in \mathcal{P}$

4.1. Maximum Likelihood Estimation of the Weights

The weight estimation given the prototype covariances can be carried out efficiently using a Newton algorithm. The gradient of the auxiliary function can be written:

$$\frac{\partial Q}{\partial\lambda_{k,i}} = \mathrm{Tr}\left[\Psi_k(\Sigma_i - \bar{\Sigma}_i)\right] \text{ or } \frac{\partial Q}{\partial\Lambda_i} = P^\top(\boldsymbol{\Sigma}_i^\star - \bar{\boldsymbol{\Sigma}}_i^\star)$$

The components of the Hessian H are:

$$\frac{\partial^2 Q}{\partial\lambda_{k,i}\partial\lambda_{l,i}} = -\mathrm{Tr}\left[\Psi_k\Sigma_i\Psi_l\Sigma_i\right]$$

The optimization can be noticeably simplified by remarking that for any covariance Σ:

$$\boldsymbol{\Sigma}^{\star\top}\boldsymbol{\Sigma}^{\star-1} = \mathrm{Tr}(\Sigma\Sigma^{-1}) = D \;\; (= \text{dimensionality})$$

Thus, when the gradient equals zero:

$$\boldsymbol{\Sigma}^{\star\top}P\boldsymbol{\Lambda} = (P^\top\bar{\boldsymbol{\Sigma}}^\star)^\top\boldsymbol{\Lambda} = D$$

This relationship defines an affine hyperplane in which $\boldsymbol{\Lambda}$ is constrained to live. Denoting U a basis of the orthogonal of $P^\top\bar{\boldsymbol{\Sigma}}^\star$, we have:

$$\boldsymbol{\Lambda} = \boldsymbol{\Lambda}_0 + U\boldsymbol{\Lambda}' \;\text{ with }\; \boldsymbol{\Lambda}_0 = D\frac{P^\top\bar{\boldsymbol{\Sigma}}^\star}{\|P^\top\bar{\boldsymbol{\Sigma}}^\star\|^2}$$

It is easy to show that if the prototypes are positive definite, then $P\boldsymbol{\Lambda}_0$ is also positive definite. Consequently, $\boldsymbol{\Lambda}_0 \in \mathcal{L}$, and can be used as an approximation of $\boldsymbol{\Lambda}$ in order to initialize the algorithm. The gradient ascent algorithm will be performed on $\boldsymbol{\Lambda}' \in \mathrm{Span}(U)$, which is of dimension $K-1$. By concavity of $Q(\Lambda|P)$, the algorithm will converge to a global maximum.

The projection matrix onto $Span(U) : \Pi = UU^\top$ can be computed once. The Hessian can be computed at each step of the iteration, leading to an update step:

$$\boldsymbol{\Delta} = \Pi H^{-1}P^\top(\boldsymbol{\Sigma}^\star - \bar{\boldsymbol{\Sigma}}^\star)$$

The Newton update $\mathbf{\Lambda} \rightarrow \mathbf{\Lambda} + \gamma\mathbf{\Delta}$ converges after a few iterations. In general $\gamma = 1$, although in the first steps of the iteration it sometimes needs to be reduced to prevent intermediate estimates of $\mathbf{\Lambda}$ to step out of $\mathcal{L}$.

4.2. Maximum Likelihood Estimation of the Prototypes

It is possible to show that maximizing $Q(P|\Lambda)$ is equivalent to maximizing a function Q' whose gradient is:

$$\frac{\partial Q'}{\partial \Psi_k} = \sum_{i=1}^{N} \pi_i \lambda_{k,i} \left(\Sigma_i - \bar{\Sigma}_i\right)$$

Denoting $\boldsymbol{\sigma}_i^p$ the p^{th} column of Σ_i, the Hessian components are:

$$\frac{\partial^2 Q'}{\partial \Psi_k \partial \Psi_k \rfloor_{p,q}} = \frac{-1}{1+\delta_{p,q}} \sum_{i=1}^{N} \pi_i \lambda_{k,i}^2 [\boldsymbol{\sigma}_i^p \boldsymbol{\sigma}_i^{q\top} + \boldsymbol{\sigma}_i^q \boldsymbol{\sigma}_i^{p\top}]$$

The exact Hessian would be expensive to compute using this formula, but it can be well estimated by only adding up the contributions of the few Gaussians with the highest $\pi_i \lambda_{k,i}^2$ weight. This approximation leads to an efficient quasi-Newton algorithm.

An additional speed improvement can be obtained by performing a similar pruning on the gradient based on the magnitude $|\pi_i \lambda_{k,i}|$ of the contribution of each covariance. As an example, in the following experiments, less than 10% of the Gaussians were used to estimate the gradient, and less than 1% were incorporated into the Hessian.

As previously, the step size γ has sometimes to be reduced to a smaller value in the first iterations to avoid stepping out of the domain $\mathcal{P}$.

4.3. Algorithm Initialization

The overall speed of convergence of the algorithm can be much improved by selecting a good initial set of prototypes. In order for these to be representative of the GMM to be modeled, they can be selected using Lloyd clustering applied to all the covariances in the mixture.

A Kullback-Liebler distance criterion is a natural choice of a metric [10]. The distance between the Gaussian means can be ignored, since it is irrelevant to the model. In addition, the variations in the scale of the prototypes — i.e. their determinant – can be normalized for, since it is captured by the weights. This leads to the following symmetric distance measure used for clustering:

$$d(\Psi_k, \Psi_l) = \mathbf{\Psi}_k^{\star\top} \mathbf{\Psi}_l^{\star -1} + \mathbf{\Psi}_l^{\star\top} \mathbf{\Psi}_k^{\star -1}$$

#	Structure	# Classes	# Blocks	# Gaussian Parameters	Error Rate	Relative Improvement
1	Diagonal			27	10.02%	
2	Block	1	2	171+45	8.42%	16.0%
3	Full	1	1	378	8.27%	17.5%
4	MIC	1	1	9	9.99%	0.3%
5	MIC	1	1	27	9.21%	8.1%
6	MIC	30	1	27	8.77%	12.5%
7	MIC	1	2	6+3	9.48%	5.4%
8	MIC	1	2	18+9	8.93%	10.9%

Table 1. Error rates on a set of Italian tasks.

5. APPLICATION TO ACOUSTIC MODELING

In the simplest approach, the model can be used to tie the complete set of covariances in the acoustic model. All the Gaussians are pooled into a single GMM using the priors estimated from the hidden Markov model state occupancy probabilities, and the prototypes are estimated on the entire mixture after being initialized using Lloyd clustering.

A slightly more involved approach uses separate mixtures for distinct state classes. While this leads to a significant increase in the total number of parameters in the system, the additional complexity at run time can be alleviated by only computing the prototype-dependent features for the active states at any given time during the decoding.

In both situations, the model used can be full or block-diagonal. In addition to reducing both the front-end overhead and training time, the block-diagonal model has a combinatorial advantage which it shares with other types of subspace clustering methods (see e.g. [11]): a block-diagonal system with L subspaces and K prototypes per subspace contains implicitly K^L "full" prototypes, while only requiring $K \times L$ weights per Gaussian. As a result, for a given number of Gaussian-dependent parameters, the block diagonal model can "draw" from a larger collection of prototypes than its single-block counterpart.

This block-diagonal approach can only be of interest if the cost of not taking some of the correlations into account remains small. Table 1 (2,3) compares the accuracy of a full-covariance system to a block-diagonal system — both described in Section 6 — for which the feature vector is decomposed into two blocks, one containing the cepstra and Δ features, and the other containing the $\Delta\Delta$ features. Since the accuracy cost is modest, the two blocks can be treated separately with their own sets of weights and prototypes.

6. EXPERIMENTS

The following experiments use context-dependent phonetic hidden Markov models based on Genones [12]: each state cluster shares a common set of Gaussians, while the mixture weights are state-dependent.

The test-set is a collection of 9600 utterances of Italian telephone speech spanning several tasks, including digits, letters, proper names and command lists. The features are 9-dimensional MFCC with Δ and $\Delta\Delta$. The system comprises 3400 triphones and a total of 48000 Gaussians trained using 89000 utterances. The accuracy is evaluated using a sentence understanding error rate, which measures the proportion of utterances in the test-set that were interpreted incorrectly.

Table 1 shows the error rates for three baseline models and the mixture of inverse covariances (MIC) approach. Table 1 (4,5) shows the error rates for the single-class, single-block approach. Without any increase of the number of Gaussian-dependent parameters, the accuracy can be improved by about 8%. Only 9 Gaussian-dependent parameters are required to match the accuracy of the baseline diagonal system in Table 1 (1).

Table 1 (6) uses one distinct set of prototypes for each of 30 phonetic classes. The much larger improvement suggests that a class-based approach is an efficient alternative to using large mixtures to get closer to the performance of a full-covariance model.

Table 1 (7,8) shows the error rate using the 2 block-diagonal model (cepstra + Δ, and $\Delta\Delta$), with the number of Gaussian-dependent parameters being comparable to that of Table 1 (2,3). The results are uniformly better than the single-block model although the total number of parameters in the system is smaller. Following the argument of Section 5, the block-diagonal system with 2 blocks and respectively 6 and 3 prototypes for each block should be similar to a system with about $6 \times 3 = 18$ "full" prototypes. In fact, the single-block model that performs the closest to this configuration has 15 prototypes. This suggests that a block diagonal mixture is the most appropriate when using a limited number of Gaussian-dependent parameters.

7. SUMMARY

A low-complexity approximation to full and block-diagonal covariance Gaussian mixture models was introduced, along with robust maximum likelihood estimation algorithms to compute the parameters of this model. When used in the context of a GMM acoustic model for speech recognition, it leads to significant accuracy gains over a typical diagonal covariance model at little cost in complexity.

8. ACKNOWLEDGMENTS

Many thanks to Michael Schuster and Remco Teunen for their helpful contributions.

9. REFERENCES

[1] R. Clarke, "Relation between the Karhunen Loève and cosine transforms," *IEE Proceedings*, vol. 128, no. 6-F, pp. 359–360, Nov. 1981.

[2] T. Eisele, R. Haeb-Umbach, and D. Langmann, "A comparative study of linear feature transformation techniques for automatic speech recognition," *Proceedings of ICSLP 96*, 1996.

[3] A. Ljolje, "The importance of cepstral parameter correlation in speech recognition," *Computer Speech and Language*, vol. 8, pp. 223–232, 1994.

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

[5] J.A. Bilmes, "Factored sparse inverse covariance matrices," *Proceedings of ICASSP 00*, 2000.

[6] P. Olsen and R. Gopinath, "Modeling inverse covariance matrices by basis expansion," *Proceedings of ICASSP 02*, 2002.

[7] S. Axelrod, R. Gopinath, and P. Olsen, "Modeling with a subspace constraint on inverse covariance matrices," *Proceedings of ICSLP 02*, 2002.

[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*, vol. 39, no. 1, pp. 1–38, 1977.

[9] S. Boyd and L. Vandenberghe, *Convex Optimization*, draft, http://www.stanford.edu/class/ee364/, 2003.

[10] R.M. Gray, "Gauss mixtures quantization: clustering gauss mixtures," *Proceedings of the MSRI Workshop on Nonlinear Estimation and Classification*, 2002.

[11] E. Bocchieri and B. Mak, "Subspace distribution clustering hidden markov model.," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 3, pp. 264–275, March 2001.

[12] V. Digalakis, P. Monaco, and H. Murveit, "Genones: Generalized mixture tying in continuous hidden markov model-based speech recognizers," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 4, pp. 281–289, 1996.