Gaussian Mixture Models[*]

Douglas Reynolds

MIT Lincoln Laboratory, 244 Wood St., Lexington, MA 02140, USA
dar@ll.mit.edu

Synonyms

GMM; Mixture model; Gaussian mixture density

Definition

A Gaussian Mixture Model (GMM) is a parametric probability density function represented as a weighted sum of Gaussian component densities. GMMs are commonly used as a parametric model of the probability distribution of continuous measurements or features in a biometric system, such as vocal-tract related spectral features in a speaker recognition system. GMM parameters are estimated from training data using the iterative Expectation-Maximization (EM) algorithm or Maximum *A Posteriori* (MAP) estimation from a well-trained prior model.

Main Body Text

Introduction

A Gaussian mixture model is a weighted sum of M component Gaussian densities as given by the equation,

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{M} w_i \; g(\mathbf{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i), \tag{1}$$

where $\mathbf{x}$ is a D-dimensional continuous-valued data vector (i.e. measurement or features), $w_i, i = 1, \ldots, M$, are the mixture weights, and $g(\mathbf{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i), i = 1, \ldots, M$, are the component Gaussian densities. Each component density is a D-variate Gaussian function of the form,

$$g(\mathbf{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) = \frac{1}{(2\pi)^{D/2}|\mathit{\Sigma}_i|^{1/2}} \exp\left\{ -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)' \; \mathit{\Sigma}_i^{-1} \; (\mathbf{x}-\boldsymbol{\mu}_i) \right\}, \tag{2}$$

with mean vector $\boldsymbol{\mu}_i$ and covariance matrix $\mathit{\Sigma}_i$. The mixture weights satisfy the constraint that $\sum_{i=1}^{M} w_i = 1$.

The complete Gaussian mixture model is parameterized by the mean vectors, covariance matrices and mixture weights from all component densities. These parameters are collectively represented by the notation,

[*] This work was sponsored by the Department of Defense under Air Force Contract FA8721-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

$$\lambda = \{w_i,\ \boldsymbol{\mu}_i,\ \Sigma_i\} \qquad i = 1, \ldots, M. \tag{3}$$

There are several variants on the GMM shown in Equation (3). The covariance matrices, Σ_i, can be full rank or constrained to be diagonal. Additionally, parameters can be shared, or tied, among the Gaussian components, such as having a common covariance matrix for all components, The choice of model configuration (number of components, full or diagonal covariance matrices, and parameter tying) is often determined by the amount of data available for estimating the GMM parameters and how the GMM is used in a particular biometric application.

It is also important to note that because the component Gaussian are acting together to model the overall feature density, full covariance matrices are not necessary even if the features are not statistically independent. The linear combination of diagonal covariance basis Gaussians is capable of modeling the correlations between feature vector elements. The effect of using a set of M full covariance matrix Gaussians can be equally obtained by using a larger set of diagonal covariance Gaussians.

GMMs are often used in biometric systems, most notably in speaker recognition systems, due to their capability of representing a large class of sample distributions. One of the powerful attributes of the GMM is its ability to form smooth approximations to arbitrarily shaped densities. The classical uni-modal Gaussian model represents feature distributions by a position (mean vector) and a elliptic shape (covariance matrix) and a vector quantizer (VQ) or nearest neighbor model represents a distribution by a discrete set of characteristic templates [1]. A GMM acts as a hybrid between these two models by using a discrete set of Gaussian functions, each with their own mean and covariance matrix, to allow a better modeling capability. Figure 1 compares the densities obtained using a unimodal Gaussian model, a GMM and a VQ model. Plot (a) shows the histogram of a single feature from a speaker recognition system (a single cepstral value from a 25 second utterance by a male speaker); plot (b) shows a uni-modal Gaussian model of this feature distribution; plot (c) shows a GMM and its ten underlying component densities; and plot (d) shows a histogram of the data assigned to the VQ centroid locations of a 10 element codebook. The GMM not only provides a smooth overall distribution fit, its components also clearly detail the multi-modal nature of the density.

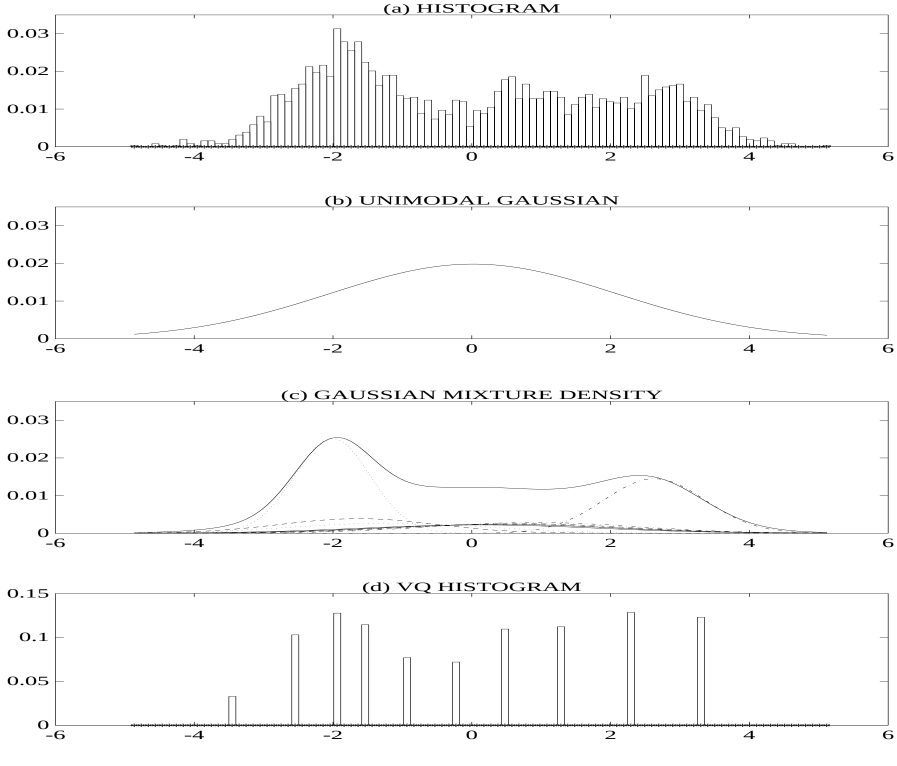

Fig. 1. Comparison of distribution modeling. (a) histogram of a single cepstral coefficient from a 25 second utterance by a male speaker (b) maximum likelihood uni-modal Gaussian model (c) GMM and its 10 underlying component densities (d) histogram of the data assigned to the VQ centroid locations of a 10 element codebook.

The use of a GMM for representing feature distributions in a biometric system may also be motivated by the intuitive notion that the individual component densities may model some underlying set of *hidden* classes. For example, in speaker recognition, it is reasonable to assume the acoustic space of spectral related features corresponding to a speaker's broad phonetic events, such as vowels, nasals or fricatives. These acoustic classes reflect some general speaker dependent vocal tract configurations that are useful for characterizing speaker identity. The spectral shape of the ith acoustic class can in turn be represented by the mean $\boldsymbol{\mu}_i$ of the ith component density, and variations of the average spectral shape can be represented by the covariance matrix Σ_i. Because all the features used to train the GMM are unlabeled, the acoustic classes are hidden in that the class of an observation is unknown. A GMM can also be viewed as a single-state HMM with a Gaussian mixture observation density, or an ergodic Gaussian observation HMM with fixed, equal transition probabilities. Assuming independent feature vectors, the observation density of feature vectors drawn from these hidden acoustic classes is a Gaussian mixture [2, 3].

Maximum Likelihood Parameter Estimation

Given training vectors and a GMM configuration, we wish to estimate the parameters of the GMM, λ, which in some sense best matches the distribution of the training feature vectors. There are several techniques available for estimating the parameters of a GMM [4]. By far the most popular and well-established method is maximum likelihood (ML) estimation.

The aim of ML estimation is to find the model parameters which maximize the likelihood of the GMM given the training data. For a sequence of T training vectors $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$, the GMM likelihood, assuming independence between the vectors[1], can be written as,

$$p(X|\lambda) = \prod_{t=1}^{T} p(\mathbf{x}_t|\lambda). \tag{4}$$

Unfortunately, this expression is a non-linear function of the parameters λ and direct maximization is not possible. However, ML parameter estimates can be obtained iteratively using a special case of the expectation-maximization (EM) algorithm [5].

The basic idea of the EM algorithm is, beginning with an initial model λ, to estimate a new model $\bar{\lambda}$, such that $p(X|\bar{\lambda}) \geq p(X|\lambda)$. The new model then becomes the initial model for the next iteration and the process is repeated until some convergence threshold is reached. The initial model is typically derived by using some form of binary VQ estimation.

On each EM iteration, the following re-estimation formulas are used which guarantee a monotonic increase in the model's likelihood value,

Mixture Weights

$$\bar{w}_i = \frac{1}{T} \sum_{t=1}^{T} \Pr(i|\mathbf{x}_t, \lambda). \tag{5}$$

Means

$$\bar{\boldsymbol{\mu}}_i = \frac{\sum_{t=1}^{T} \Pr(i|\mathbf{x}_t, \lambda)\, \mathbf{x}_t}{\sum_{t=1}^{T} \Pr(i|\mathbf{x}_t, \lambda)}. \tag{6}$$

Variances (diagonal covariance)

$$\bar{\sigma}_i^2 = \frac{\sum_{t=1}^{T} \Pr(i|\mathbf{x}_t, \lambda)\, x_t^2}{\sum_{t=1}^{T} \Pr(i|\mathbf{x}_t, \lambda)} - \bar{\mu_i}^2, \tag{7}$$

where σ_i^2, x_t, and μ_i refer to arbitrary elements of the vectors $\boldsymbol{\sigma_{\mathbf{i}}}^2$, $\mathbf{x}_t$, and $\boldsymbol{\mu}_i$, respectively.

[1] The independence assumption is often incorrect but needed to make the problem tractable.

The *a posteriori* probability for component i is given by

$$\Pr(i|\mathbf{x}_t, \lambda) = \frac{w_i \, g(\mathbf{x}_t|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}{\displaystyle\sum_{k=1}^{M} w_k \, g(\mathbf{x}_t|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)} \qquad (8)$$

Maximum *A Posteriori* (MAP) Parameter Estimation

In addition to estimating GMM parameters via the EM algorithm, the parameters may also be estimated using Maximum *A Posteriori* (MAP) estimation. MAP estimation is used, for example, in speaker recognition applications to derive speaker model by adapting from a universal background model (UBM) [6]. It is also used in other pattern recognition tasks where limited labeled training data is used to adapt a prior, general model.

Like the EM algorithm, the MAP estimation is a two step estimation process. The first step is identical to the "Expectation" step of the EM algorithm, where estimates of the sufficient statistics[2] of the training data are computed for each mixture in the prior model. Unlike the second step of the EM algorithm, for adaptation these "new" sufficient statistic estimates are then combined with the "old" sufficient statistics from the prior mixture parameters using a data-dependent mixing coefficient. The data-dependent mixing coefficient is designed so that mixtures with high counts of new data rely more on the new sufficient statistics for final parameter estimation and mixtures with low counts of new data rely more on the old sufficient statistics for final parameter estimation.

The specifics of the adaptation are as follows. Given a prior model and training vectors from the desired class, $X = \{\mathbf{x}_1 \ldots, \mathbf{x}_T\}$, we first determine the probabilistic alignment of the training vectors into the prior mixture components (Figure 2(a)). That is, for mixture i in the prior model, we compute $\Pr(i|\mathbf{x}_t, \lambda_{\text{prior}})$, as in Equation (8).

We then compute the sufficient statistics for the weight, mean and variance parameters: [3],

$$n_i = \sum_{t=1}^{T} \Pr(i|\mathbf{x}_t, \lambda_{\text{prior}}) \quad \text{weight,} \qquad (9)$$

$$E_i(\mathbf{x}) = \frac{1}{n_i} \sum_{t=1}^{T} \Pr(i|\mathbf{x}_t, \lambda_{\text{prior}}) \mathbf{x}_t \quad \text{mean,} \qquad (10)$$

$$E_i(\mathbf{x}^2) = \frac{1}{n_i} \sum_{t=1}^{T} \Pr(i|\mathbf{x}_t, \lambda_{\text{prior}}) \mathbf{x}_t^2 \quad \text{variance,.} \qquad (11)$$

This is the same as the "Expectation" step in the EM algorithm.

Lastly, these new sufficient statistics from the training data are used to update the prior sufficient statistics for mixture i to create the adapted parameters for mixture i (Figure 2(b)) with the equations:

$$\begin{aligned}
\hat{w}_i &= [\alpha_i^w n_i/T + (1 - \alpha_i^w) w_i] \, \gamma \quad \text{adapted mixture weight,} && (12)\\
\hat{\boldsymbol{\mu}}_i &= \alpha_i^m E_i(\mathbf{x}) + (1 - \alpha_i^m) \boldsymbol{\mu}_i \quad \text{adapted mixture mean,} && (13)\\
\hat{\boldsymbol{\sigma}^2}_i &= \alpha_i^v E_i(\mathbf{x}^2) + (1 - \alpha_i^v)(\boldsymbol{\sigma}_i^2 + \boldsymbol{\mu}_i^2) - \hat{\boldsymbol{\mu}^2}_i \quad \text{adapted mixture variance.} && (14)
\end{aligned}$$

The adaptation coefficients controlling the balance between old and new estimates are $\{\alpha_i^w, \alpha_i^m, \alpha_i^v\}$ for the weights, means and variances, respectively. The scale factor, γ, is computed over all adapted mixture weights to ensure they sum to unity. Note that the sufficient statistics, not the derived parameters, such as the variance, are being adapted.

For each mixture and each parameter, a data-dependent adaptation coefficient α_i^ρ, $\rho \in \{w, m, v\}$, is used in the above equations. This is defined as

$$\alpha_i^\rho = \frac{n_i}{n_i + r^\rho}, \qquad (15)$$

[2] These are the basic statistics needed to be estimated to compute the desired parameters. For a GMM mixture, these are the count, and the first and second moments required to compute the mixture weight, mean and variance.

[3] $\mathbf{x}^2$ is shorthand for diag($\mathbf{x}\mathbf{x}'$)

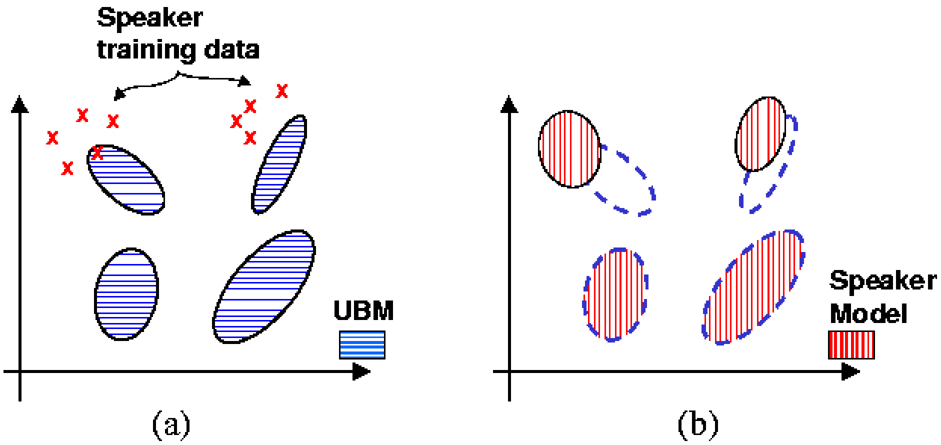

Fig. 2. Pictorial example of two steps in adapting a hypothesized speaker model. (a) The training vectors (x's) are probabilistically mapped into the UBM (prior) mixtures. (b) The adapted mixture parameters are derived using the statistics of the new data and the UBM (prior) mixture parameters. The adaptation is data dependent, so UBM (prior) mixture parameters are adapted by different amounts.

where r^{ρ} is a fixed "relevance" factor for parameter ρ. It is common in speaker recognition applications to use one adaptation coefficient for all parameters ($\alpha_i^w = \alpha_i^m = \alpha_i^v = n_i/(n_i + r)$) and further to only adapt certain GMM parameters, such as only the mean vectors.

Using a data-dependent adaptation coefficient allows mixture dependent adaptation of parameters. If a mixture component has a low probabilistic count, n_i, of new data, then $\alpha_i^{\rho} \rightarrow 0$ causing the de-emphasis of the new (potentially under-trained) parameters and the emphasis of the old (better trained) parameters. For mixture components with high probabilistic counts, $\alpha_i^{\rho} \rightarrow 1$, causing the use of the new class-dependent parameters. The relevance factor is a way of controlling how much new data should be observed in a mixture before the new parameters begin replacing the old parameters. This approach should thus be robust to limited training data.

Related Entries

Speaker Recognition, Speaker Modeling, Speaker Matching, Universal Background Models

References

1. Gray, R.: Vector Quantization. IEEE ASSP Magazine (1984) 4–29
2. Reynolds, D.A.: A Gaussian Mixture Modeling Approach to Text-Independent Speaker Identification. PhD thesis, Georgia Institute of Technology (1992)
3. Reynolds, D.A., Rose, R.C.: Robust Text-Independent Speaker Identification using Gaussian Mixture Speaker Models. IEEE Transactions on Acoustics, Speech, and Signal Processing **3**(1) (1995) 72–83
4. McLachlan, G., ed.: Mixture Models. Marcel Dekker, New York, NY (1988)
5. Dempster, A., Laird, N., Rubin, D.: Maximum Likelihood from Incomplete Data via the EM Algorithm. Journal of the Royal Statistical Society **39**(1) (1977) 1–38
6. Reynolds, D.A., Quatieri, T.F., Dunn, R.B.: Speaker Verification Using Adapted Gaussian Mixture Models. Digital Signal Processing **10**(1) (2000) 19–41