Parameter-based reduction of Gaussian mixture models with a variational-Bayes approach

Pierrick Bruneau, Marc Gelgon, Fabien Picarougne

Parameter-based reduction of Gaussian mixture models with a variational-Bayes approach

Pierrick Bruneau[1,2], Marc Gelgon[1,2] and Fabien Picarougne[1]
(1) Nantes university, LINA (UMR CNRS 6241), Polytech'Nantes
rue C.Pauc, La Chantrerie, 44306 Nantes cedex 3, France (2) INRIA Atlas project-team
firstname.surname@univ-nantes.fr

Abstract

This paper [1] proposes a technique for simplifying a given Gaussian mixture model, i.e. reformulating the density in a more parcimonious manner, if possible (less Gaussian components in the mixture). Numerous applications requiring aggregation of models from various sources, or index structures over sets of mixture models for fast access, may benefit from the technique. Variational Bayesian estimation of mixtures is known to be a powerful technique on punctual data. We derive herein a new version of the Variational-Bayes EM algorithm that operates on Gaussian components of a given mixture and suppresses redundancy, if any, while preserving structure of the underlying generative process. A main feature of the present scheme is that it merely resorts to the parameters of the original mixture, ensuring low computational cost. Experimental results are reported on real data.

1. Introduction

Many current research directions in pattern recognition pertain to applying statistical learning and classification to an ever larger number of classes (e.g. visual objects [9], dynamic video characterisation or speaker recognition). To this aim, sparse class models may be investigated [10]. There is also growing interest for indexing structures that handle sets of probabilistic models, as well as for statistical learning and searching on distributed infrastructures (cluster, peer-to-peer, sensor networks). In this second branch of works, one often faces the need to aggregate models. This is encountered for instance when defining concise parent models from similar leave models in a tree [11], or merging models describing the same class, i.e. the same underlying hidden probability distribution functions (*pdf*s), but estimated from different data sources [3, 8].

[1]This work was funded by the ANR Safimage and the Pays-de-la-Loire MILES project.

A simple weighted sum of Gaussian mixtures is likely to introduce undesirable redundancy, with a view to capturing the underlying density. A straightforward solution, but more expensive in terms of communication and/or computation, would consist in re-estimating a mixture model from the original data or from data sampled from the original mixture. In contrast, a main feature of the present scheme is that it merely resorts to the parameters of the original mixture.

A Gaussian Mixture Model (GMM) is defined by the following *pdf* :

$$p(x) = \sum_{k=1}^{K} \omega_k \mathcal{N}(x \mid \mu_k, \Lambda_k^{-1}) \quad (1)$$

where x is a d-dimensional feature vector and $\mathcal{N}(. \mid \mu_k, \Lambda_k^{-1})$ is a Gaussian *pdf* with mean vector μ_k and precision matrix Λ_k. In the remainder of this paper, we will designate $\mathcal{N}(. \mid \mu_k, \Lambda_k^{-1})$ as the k-th component of the GMM. $\Omega = \{\omega_k\}$ is a weight vector associated to the components, following the constraint $\omega_k \geq 0 \, \forall k, \sum \omega_k = 1$. We introduce a lightweight notation for the GMM parameters : $\theta = \{\Omega, \mu, \Lambda\}$ where $\mu = \{\mu_k\}$ and $\Lambda = \{\Lambda_k\}$.

Estimating a GMM can be decomposed in 2 complementary problems : estimating a correct number of components (K) and correct parameters for the components. This joint estimation is classically known to be difficult. Variational Bayesian estimation of a GMM [2, 4] is an effective way to overcome this issue. Relying on simple hypotheses about the obtained distribution (i.e. variational distribution), and on properly chosen uninformative priors, a simple EM-like algorithm (called VBEM hereafter) computes an effective model by pruning useless components at the end of the process. However, this technique is currently only applicable to punctual data.

We propose to adapt this framework to the model reduction problem evoked previously. We will see that few simple hypotheses about the data that originates from the GMM we want to simplify leads to

978-1-4244-2175-6/08/$25.00 ©2008 IEEE

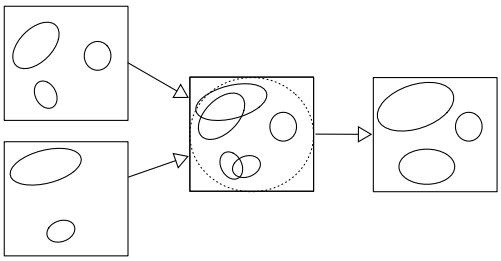

Figure 1. GMMs constitute input data for the algorithm (first step). By adjusting priors (dashed line) on the domain of this global GMM, we obtain a reduced and informative GMM.

a convenient reformulation of the VBEM framework. This reformulation depends only on GMM parameters, while preserving the good properties guaranteed by the VBEM framework (see figure 1 for an illustration of the proposed algorithm).

In section 2 we will decline a reformulation of the VBEM variational *pdf*s that takes parameters rather than punctual data as input. We will see that it leads to coupled update equations, from which we derive an iterative EM-like algorithm (named VBMErge hereafter). In section 3 we provide experimental results obtained by applying this technique to real data and draw concluding remarks.

2 A variational-Bayes technique for model reduction

Introducing virtual sampling in the variational Bayes framework

We follow notations used in [4] for punctual data. Classically, variational mixture estimation considers a set of data $X = \begin{pmatrix} x_1^T \\ \dots \\ x_N^T \end{pmatrix}$ and $Z = \begin{pmatrix} z_1^T \\ \dots \\ z_N^T \end{pmatrix}$ that is assumed to be generated from the mixture. x_i is a d-dimensional feature vector and z_i the associated binary variable indicating from which component x_i was generated (e.g. from k-th component $\equiv z_{ik} = 1, z_{ij} = 0 \, \forall j \neq k$). Z is generally hidden, and the purpose of the procedure is to compute a joint estimate of θ and Z. The associated pdfs are :

$$p(Z \mid \Omega) = \prod_{n=1}^{N} \prod_{k=1}^{K} {\omega_k}^{z_{nk}} \tag{2}$$

$$p(X \mid Z, \mu, \Lambda) = \prod_{n=1}^{N} \prod_{k=1}^{K} \mathcal{N}(x_n, \mu_k, \Lambda_k^{-1})^{z_{nk}} \tag{3}$$

Now consider an arbitrary mixture defining L components, with parameters $\theta' = \{\Omega', \mu', \Lambda'\}$. This model might have redundant components, typically as the result of summing several mixtures. We then assume that X and Z were i.i.d sampled from this distribution. It is therefore possible to regroup X by the component that originated its various items. It leads us to the following formalism : $X = \{\hat{x}_1, \dots, \hat{x}_L\}$ with card$(X) = N, \hat{x}_l = \{x_i \mid z_{il} = 1\}$ and card$(\hat{x}_l) = \omega'_l N$. Let us express the distributions (2) and (3) w.r.t this formalism. To achieve tractability, let us assume : $\forall x_i \in \hat{x}_l, z_{ik} = \text{const} = z_{lk}$. This assumption is equivalent to stating that components from the model we wish to reduce will not be split in the estimated mixture. This conforms to the vision of model reduction we illustrated in figure 1. Thus we can rewrite the expression (3) :

$$p(X \mid Z, \mu, \Lambda) = \prod_{k=1}^{K} \prod_{l=1}^{L} p(\hat{x}_l \mid Z, \mu_k, \Lambda_k)^{z_{lk}} \tag{4}$$

$$p(X \mid Z, \mu, \Lambda) = \prod_{k=1}^{K} \prod_{l=1}^{L} \left[\prod_{i=1}^{\omega'_l N} \mathcal{N}(x_{li} \mid \mu_k, \Lambda_k^{-1}) \right]^{z_{lk}} \tag{5}$$

$$\ln p(X \mid Z, \mu, \Lambda) = \sum_{k=1}^{K} \sum_{l=1}^{L} z_{lk} \left[\sum_{i=1}^{\omega'_l N} \ln \mathcal{N}(x_{li} \mid \mu_k, \Lambda_k^{-1}) \right] \tag{6}$$

For N sufficiently large, we can make the following approximation :

$$\sum_{i=1}^{\omega'_l N} \ln \mathcal{N}(x_{li} \mid \mu_k, \Lambda_k^{-1}) \simeq \omega'_l N E_{\mu'_l, \Lambda'_l} \left[\ln \mathcal{N}(x \mid \mu_k, \Lambda_k^{-1}) \right] \tag{7}$$

This statement is known as *virtual sampling*, and was introduced in [11, 12].

The expectation term can be rewritten as follows :

$$\begin{aligned} &E_{\mu'_l, \Lambda'_l} \left[\ln \mathcal{N}(x \mid \mu_k, \Lambda_k^{-1}) \right] \\ &\quad = \int \mathcal{N}(x \mid \mu'_l, {\Lambda'_l}^{-1}) \ln \mathcal{N}(x \mid \mu_k, \Lambda_k^{-1}) \, \mathrm{dx} \\ &\quad = -KL \left(\mathcal{N}(x \mid \mu'_l, {\Lambda'_l}^{-1}) \parallel \mathcal{N}(x \mid \mu_k, \Lambda_k^{-1}) \right) \\ &\qquad - H(\mathcal{N}(x \mid \mu'_l, {\Lambda'_l}^{-1})) \end{aligned} \tag{8}$$

with $KL(q_0 \parallel q_1)$ the KL divergence of q_1 from q_0 and $H(q_0)$ the entropy of q_0. These two terms have closed-form expressions [5]. Thus by reinjecting (8) into (7), and then (7) into (6), we obtain the following expression :

$$\begin{aligned} \ln p(X \mid Z, \mu, \Lambda) = N \sum_{k=1}^{K} \sum_{l=1}^{L} z_{lk} \omega'_l \\ [- KL \left(\mathcal{N}(x \mid \mu'_l, {\Lambda'_l}^{-1}) \parallel \mathcal{N}(x \mid \mu_k, \Lambda_k^{-1}) \right) \\ - H(\mathcal{N}(x \mid \mu'_l, {\Lambda'_l}^{-1}))] \end{aligned} \tag{9}$$

$$\ln p(X \mid Z, \mu, \Lambda) = N \sum_{k=1}^{K} \sum_{l=1}^{L} z_{lk} \omega'_l \; 0.5[\ln\det\Lambda_k - \mathrm{Tr}(\Lambda_k \Lambda_l^{'-1}) - (\mu'_l - \mu_k)^T \Lambda_k (\mu'_l - \mu_k) - d\ln(2\pi)] \tag{10}$$

Using virtual samples also has consequences on (2) : as we previously stated that $z_{lk} = z_{nk}\ \forall x_n \in \hat{x}_l$, we can write :

$$p(Z \mid \Omega) = \prod_{n=1}^{N} \prod_{k=1}^{K} \omega_k^{z_{nk}} = \prod_{l=1}^{L} \prod_{k=1}^{K} \omega_k^{N\omega'_l z_{lk}} \tag{11}$$

A modified VBEM algorithm

Variational Bayesian estimation of a Gaussian mixture relies on (2) and (3) to model the origination of a data set, but also introduces priors over the parameters occurring in these distributions. Let us recall here expressions defining these priors :

$$p(\Omega) = \mathrm{Dir}(\Omega \mid \alpha_0) \tag{12}$$

where Dir denotes the Dirichlet distribution, and $\alpha = \{\alpha_k\}$ is the hyper-parameter governing it. α_0 designates the initial value.

$$p(\mu, \Lambda) = p(\mu \mid \Lambda) p(\Lambda) = \prod_{k=1}^{K} \mathcal{N}(\mu_k \mid m_0, (\beta_0 \Lambda_k)^{-1})\, \mathcal{W}(\Lambda_k \mid W_0, \nu_0) \tag{13}$$

The Normal-Wishart distribution above uses hyper-parameters m, β, W and ν. The two latter govern the Wishart distribution while m represents the mean vector for the Normal term. β is a normalization term. Variational framework defines a factorized variational distribution $q(Z, \Omega, \mu\Lambda) = q(Z)q(\Omega)q(\mu, \Lambda)$, that is intended to minimize the loss w.r.t. the true, unknown and intractable posterior distribution. Calculations exposed in [2, 4] show that the optimal form for a single factor q_j (denoted q_j^*) is obtained by applying the following general formula :

$$\ln q_j^* = \mathbb{E}_{i \neq j}[\ln p(X, Z, \Omega, \mu, \Lambda)] + \text{const} \tag{14}$$

$p(X, Z, \Omega, \mu, \Lambda)$ is obtained from the product of (2), (3), (12) and (13). Applying this general formula for $q(\Omega)$, $q(\mu \mid \Lambda)$ and $q(\Lambda)$ leads to update expressions for hyper-parameters. Relatively to $q(Z)$, estimates (i.e. $\mathbb{E}[z_{nk}] = r_{nk}$) that depend on moments computed w.r.t. to current hyper-parameters are obtained. Due to conciseness concerns, it is not possible to recall all update equations here, please refer to [4] for details. All these expressions are coupled, and cycling through these equations implements an EM-like algorithm.

Update equations evoked previously are obtained by involving expressions of $p(Z \mid \Omega)$ and $p(X \mid Z, \mu, \Lambda)$. Therefore, using modified versions (10) and (11) causes changes in the standard update equations. In the remainder of this section we review these modifications.

r_{nk} estimates are obtained from normalization of the associated ρ_{nk}. In our modified scheme, we compute these as follows :

$$\ln(\rho_{lk}) = \frac{N\omega'_l}{2}[\left(2\ln\tilde{\omega}_k + \ln\tilde{\Lambda}_k - d\ln(2\pi)\right) - E_{\mu_k,\Lambda_k}\left[\mathrm{Tr}(\Lambda_k \Lambda_l^{'-1}) + (\mu'_l - \mu_k)^T \Lambda_k (\mu'_l - \mu_k)\right]] \tag{15}$$

with $\ln\tilde{\omega}_k = E[\ln\omega_k]$ and $\ln\tilde{\Lambda}_k = E[\ln\det(\Lambda_k)]$.

The moment w.r.t μ_k and Λ_k is easily evaluated to give
$\frac{d}{\beta_k} + \nu_k\left[\mathrm{Tr}(W_k \Lambda_l^{'-1}) + (\mu'_l - m_k)^T W_k (\mu'_l - m_k)\right]$.

Modifications in the hyper-parameter update expressions also emerge from using (10) and (11) :

$$\alpha_k = \alpha_0 + \sum\nolimits_l N\omega'_l r_{lk} \tag{16}$$

$$\beta_k = \beta_0 + \sum\nolimits_l N\omega'_l r_{lk} \tag{17}$$

$$m_k = \beta_k^{-1}(\beta_0 m_0 + \sum\nolimits_l N\omega'_l r_{lk} \mu'_l) \tag{18}$$

$$W_k^{-1} = W_0^{-1} + \beta_0 m_0 m_0^T - \beta_k m_k m_k^T + \sum\nolimits_l N\omega'_l r_{lk} (\mu'_l \mu_l^{'T} + \Lambda_l^{'-1}) \tag{19}$$

$$\nu_k = \nu_0 + \sum\nolimits_l N\omega'_l r_{lk} \tag{20}$$

It is recalled in [4] that the previously described algorithm monotonically decreases the KL distance between the variational pdf and the true posterior. This is equivalent to maximising the lower bound of the complete likelihood. As we can compute this lower bound, and as this bound should never decrease, it supplies our algorithm with a criterion for testing for convergence by comparing two successive values of the bound. As terms needed to compute this bound are numerous, only those that depend on Z or X (i.e. impacted) are listed below :

$$2E[\ln p(X \mid Z, \mu, \Lambda)] = \sum\nolimits_k \sum\nolimits_l N\omega'_l r_{lk}[\ln\tilde{\Lambda}_k - \frac{d}{\beta_k} - \nu_k[\mathrm{Tr}(W_k \Lambda_l^{'-1}) + (\mu'_l - m_k)^T W_k (\mu'_l - m_k)]] \tag{21}$$

$$E[\ln p(Z \mid \Omega)] = \sum_l \sum_k N\omega'_l r_{lk} \ln\tilde{\omega}_k \tag{22}$$

3 Experiments and conclusions

The procedure for our experiments will be closely related to the ones performed by Vasconcelos in [11]. We will use the Columbia object database [7]. This image database contains 100 items, for which we have 72 views (each taken at a different viewpoint, viewpoints are separated by 5° from one to another). We retain 9 40°-separated views for each object. We employ the

same feature extraction process as presented in [11]. Ours differs in that we keep only the 12 first variables as very high-dimensional spaces often have bad properties with regard to density estimation. We also chose to estimate each single GMM with VBEM procedure, instead of classic EM in [11]. It is therefore possible to avoid choosing an arbitrary number of components here. Eventually we obtain a GMM representing each object of the original database. For each object, the first view will be the query object, and the 8 remaining will form a database. This database will be summarized by grouping all its components and computing a reduction on this group using VBMErge. Again, we don't have to choose the final number of components. As mentionned above, this is part of the single-run estimation process. By using this summarized model, only one similarity measurement will be necessary to identify the original group of a query object. To measure similarity between a query object q and a database model p, we traditionnally measure $KL(q \parallel p)$. We can calculate this value directly using a Monte-Carlo integration, or alternatively use the approximation introduced in [6].

Learning time for the summarized model is very small, as usually few iterations are needed (between 5 and 10 in most cases), and the complexity depends only on d and on the number of merged models. These values are usually low (in our case, $d = 12$ and $n_{models} = 8$). In our experiments, only the time to learn each individual GMM (i.e. representing each image) was rather long, but not significantly much compared to classic EM estimation (with equal K). Variational estimation with uninformative priors requires more classes than a classic EM estimation (as some of them will be pruned), but in the same order as the amount used in Vasconcelos' experiments (8 in [11], 50 in our experiments)

We obtain 75% success rate at identifying the correct image given the query and the reduced model. More generally, 95% of the 3 first ranked reduced models sets (i.e. lowest KL divergence) contain the correct answer. Therefore we obtain similar results as obtained in [11] with 8 components per reduced model in a much more parsimonious approach (see figure 2). It is important to notice that all our experiments were carried out with fully random initializations, relying on good uninformative priors. Though doing so helps avoid bad local optima, we might have merged models of inequal relative quality (likelihood w.r.t the best optimum's likelihood). This results in uncertainty that impacts the quality of the model reduction. In many situations this uncertainty might not impact significantly the process, but with high dimensional spaces and high precision requirements it is relevant to use better initialization strategies, such as multiple initial short runs. The value of the lower bound

success rate	0.75
correct image in the 3 1st ranks	0.95
avg. N components / image	6.05
avg. N comp. / reduced model	2.06

Figure 2. Obtained results summary.

(partially defined by equations (21) and (22)) would be a decision criteria between candidate initialisations.

We disclosed a new variational Bayesian approach to model reduction, and an iterative EM-like algorithm that efficiently operates directly on model parameters, which is crucial to scaling up many learning and recognition tasks. Results are promising but could be improved easily with a better initialisation strategy. Besides initialisation issues, t distributions mixture models would lead to a more robust estimation [1], and therefore are an interesting clue for future work. Also, though we didn't exploit that aspect in the present work, prior modelling and ability to reduce models depending only on parameters can be very useful in a distributed context [8].

References

[1] C. Archambeau and M. Verleysen. Robust Bayesian clustering. *Neural Networks*, 20(1):129–138, 2007.

[2] H. Attias. A variational Bayesian framework for graphical models. *NIPS*, 2000.

[3] M. Bechchi, G. Raschia, and N. Mouaddib. Merging distributed database summaries. In *Proc. ACM CIKM '07*, pages 419–428, Lisbon, Nov. 2007.

[4] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, 2006.

[5] R. E. Blahut. *Principles and Practice of Information Theory*. Addison-Wesley, 1987.

[6] J. Goldberger, S. Gordon, and H. Greenspan. An efficient image similarity measure based on approximations of KL-divergence between two Gaussian mixtures. *IEEE ICCV*, 2003.

[7] S. A. Nene, S. K. Nayar, and H. Murase. Columbia object image library (coil-100). Tech. report cucs-006-96, Columbia UCS, 1996.

[8] A. Nikseresht and M. Gelgon. Gossip-based computation of a Gaussian mixture model for distributed multimedia indexing. *IEEE Transactions on Multimedia*, (3):385–392, Mar. 2008.

[9] J. Ponce, M. Hebert, C. Schmid, and A. Zisserman. *Towards category-level object recognition*. Springer, 2006.

[10] M. E. Tipping. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

[11] N. Vasconcelos. Image indexing with mixture hierarchies. *IEEE CVPR*, 2001.

[12] N. Vasconcelos and A. Lippman. Learning mixture hierarchies. *NIPS*, 1998.