Two Entropy-Based Methods for Learning Unsupervised Gaussian Mixture Models

Antonio Peñalver, Francisco Escolano, and Juan M. Sáez

Robot Vision Group
Alicante University, Spain
a.penalver@umh.es, {sco, jmsaez}@dccia.ua.es

Abstract. In this paper we address the problem of estimating the parameters of a Gaussian mixture model. Although the EM (Expectation-Maximization) algorithm yields the maximum-likelihood solution it requires a careful initialization of the parameters and the optimal number of kernels in the mixture may be unknown beforehand. We propose a criterion based on the entropy of the pdf (probability density function) associated to each kernel to measure the quality of a given mixture model. Two different methods for estimating Shannon entropy are proposed and a modification of the classical EM algorithm to find the optimal number of kernels in the mixture is presented. We test our algorithm in probability density estimation, pattern recognition and color image segmentation.

1 Introduction

Gaussian Mixture models have been widely used for density estimation, pattern recognition and function approximation. One of the most common methods for fitting mixtures to data is the EM algorithm [6]. However, this algorithm is prone to initialization errors and it may converge to local maxima of the log-likelihood function. In addition, the algorithm requires that the number of elements (kernels) in the mixture is known beforehand (model-selection).

A d-dimensional random variable $\mathbf{y}$ follows a finite-mixture distribution when its pdf $p(\mathbf{y}|\Theta)$ can be described by a weighted sum of known pdf's named kernels. When all these kernels are Gaussian, the mixture is named in the same way:

$$p(\mathbf{y}|\Theta) = \sum_{i=1}^{K} \pi_i p(\mathbf{y}|\Theta_i) \tag{1}$$

where $0 \leq \pi_i \leq 1, i = 1, ..., K$, and $\sum_{i=1}^{K} \pi_i = 1$, being K the number of kernels, $\pi_1, ..., \pi_k$ the *a priori* probabilities of each kernel, and Θ_i the parameters describing the kernel. In Gaussian mixtures, $\Theta_i = \{\mu_i, \Sigma_i\}$, that is, the average vector and the covariance matrix. The set of parameters of a given mixture is $\Theta \equiv \{\Theta_1, ..., \Theta_k, \pi_1, ..., \pi_k\}$. Obtaining the optimal set of parameters Θ^* is usually posed in terms of maximizing the log-likelihood of the pdf to be estimated:

D.-Y. Yeung et al. (Eds.): SSPR&SPR 2006, LNCS 4109, pp. 649–657, 2006.

$$\ell(Y|\Theta) = \log p(Y|\Theta) = \log \prod_{n=1}^{N} p(y_n|\Theta) = \sum_{n=1}^{N} \log \sum_{k=1}^{K} \pi_k p(y_k|\Theta_k). \quad (2)$$

With $\Theta^* = \arg\max_\Theta \ell(\Theta)$ and $Y = \{y_1, ...y_N\}$ is a set of N i.i.d. samples of the variable Y. The EM (Expectation-Maximization) algorithm [6][12] generates a sequence of estimations of the set of parameters $\{\Theta^*(t), t = 1, 2, ...\}$ by alternating an expectation step and the maximization one until convergence. The equations are:

$$p(k|\mathbf{y}_n) = \frac{\pi_k p(\mathbf{y}^{(n)}|k)}{\Sigma_{j=1}^{K} \pi_j p(\mathbf{y}^{(n)}|k)} \quad (3)$$

$$\begin{aligned} \pi_k = \tfrac{1}{N} \textstyle\sum_{n=1}^{N} p(k|\mathbf{y}_n), \;\; \mu_k = \frac{\sum_{n=1}^{N} p(k|\mathbf{y}_n)\mathbf{y}_n}{\sum_{n=1}^{N} p(k|\mathbf{y}_n)}, \\ \Sigma_k = \frac{\sum_{n=1}^{N} p(k|\mathbf{y}_n)(\mathbf{y}_n - \mu_k)(\mathbf{y}_n - \mu_k)^T}{\sum_{n=1}^{N} p(k|\mathbf{y}_n)}, \end{aligned} \quad (4)$$

A detailed description of this classic algorithm is given in [12]. Here we focus on the fact that if K is unknown beforehand it cannot be estimated through maximizing the log-likelihood because $\ell(\Theta)$ grows with K.

In a classical EM algorithm with a fixed number of kernels density can be underestimated giving a poor description of the data. The so called model-selection problem has been addressed in many ways [16][17][8][7][14]. In this paper we propose a method that starting with only one kernel, finds the maximum-likelihood solution. In order to do so, it tests whether the underlying pdf of each kernel is Gaussian and otherwise it replaces that kernel with two kernels adequately separated from each other. In order to detect non-Gaussianity we compare the entropy of the underlying pdf with the theoretical entropy of a Gaussian. After the kernel with worse degree of Gaussianity has been splited in two, new EM steps are performed in order to obtain a new maximum-likelihood solution. In the next sections we describe two different entropy estimation techniques to test whether a given kernel describes properly the underlying data.

2 Entropy Estimation

Entropy is a basic concept in information theory [4]. For a discrete variable Y with $y_1, ..., y_N$ a the set of values, we have:

$$H(Y) = -E_y[\log(P(Y))] = -\sum_{i=1}^{N} P(Y = y_i) \log P(Y = y_i). \quad (5)$$

A fundamental result of information theory is that Gaussian variables have the maximum entropy among all the variables with equal variance. Consequently the entropy of the underlying distribution of a kernel should reach a maximum when such a distribution is Gaussian. This theoretical maximum entropy is given by:

$$H_{max}(Y) = \frac{1}{2} \log[(2\pi e)^d |\Sigma|]. \quad (6)$$

Then, in order to decide whether a given kernel is truly Gaussian or must be replaced by two other kernels, we compare the estimated entropy of the underlying data with the entropy of a Gaussian.

The estimation of the Shannon entropy of a probability density given a set of samples has been studied widely in the past [1]. In this paper we present results with two different methods: "plug-in" and "non plug-in".

2.1 Entropy Estimation with Parzen's Windows

The Parzen's windows approach [11] is a non-parametric method for estimating pdf's for a finite set of patterns. The general form of these pdf's using a Gaussian kernel and assuming diagonal covariance matrix $\psi = Diag(\sigma_1^2, ...\sigma_{N_a}^2)$ is:

$$P^*(Y, a) \equiv \frac{1}{N_a} \sum_{y_a \in a} K_\psi(y - y_a), \tag{7}$$

where $K_\psi(y - y_a)$ is a gaussian kernel centered y y_a, a is a sample of the variable Y and N_a is the size of the sample. In [15] a method for adjusting the widths of the kernels using maximum likelihood is proposed. Given the definition of entropy in Equation 5, we have:

$$H_b(Y) \equiv -E_b[\log(P(Y))] = -\frac{1}{N_b} \sum_{y_b \in b} \log(P(y_b)) \tag{8}$$

where b is a sample of the variable Y and N_b is the size of the sample. If expression in Equation 7 is plugged into Equation 8 then the entropy is estimated by:

$$H^*(Y) = \frac{1}{N_b} \sum_{y_b \in b} \log \left(\frac{1}{N_a} \sum_{y_a \in a} K_\psi(y_b - y_a)) \right) \tag{9}$$

2.2 Renyi's Entropy and Entropic Spanning Graphs

Entropic Spanning Graphs obtained from data to estimate Renyi's α-entropy[10] belong to the "non plug-in" methods of entropy estimation. Renyi's α-entropy of a probability density function f is defined as:

$$H_\alpha(f) = \frac{1}{1 - \alpha} \ln \int_z f^\alpha(z) dz \tag{10}$$

for $\alpha \in (0, 1)$. The α entropy converges to the Shannon entropy $-\int f(z) \ln f(z) dz$ as $\alpha \to 1$, so it is possible to obtain the second one from the first one.

A graph G consists of a set of vertices $X_n = \{x_1, ..., x_n\}$, with $x_n \in R^d$ and edges $\{e\}$ that connect vertices in graph: $e_{ij} = (x_i, x_j)$. If we denote by $M(X_n)$ the possible sets of edges in the class of acyclic graphs spanning X_n (spanning trees), the total edge length functional of the Euclidean power weighted Minimal Spanning Tree is:

$$L_\gamma^{MST}(X_n) = \min_{M(X_n)} \sum_{e \in M(X_n)} \mid e \mid^\gamma \tag{11}$$

with $\gamma \in (0, d)$ y $\mid e \mid$ the euclidean distance between graph vertices.

The MST has been used as a way to test for randomness of a set of points. In [9] it was showed that in d-dimensional feature space, with $d \geq 2$:

$$H_\alpha(X_n) = \frac{d}{\gamma}\left[\ln \frac{L_\gamma(X_n)}{n^\alpha} - \ln \beta_{L_\gamma,d}\right] \tag{12}$$

is an asymptotically unbiased, and almost surely consistent, estimator of the α-entropy of f where $\alpha = (d - \gamma)$ and $\beta_{L_\gamma,d}$ is a constant bias correction depending on the graph minimization criterion, but independent of f. Closed form expressions are not available for $\beta_{L_\gamma,d}$, only known approximations and bounds: (i) Monte Carlo simulation of uniform random samples on unit cube $[0,1]^d$; (ii) Large d approximation: $(\gamma/2)\ln(d/(2\pi e))$ in [2].

We can estimate $H_\alpha(f)$ for different values of $\alpha = (d-\gamma)/d$ by changing the edge weight exponent γ. As γ modifies the edge weights monotonically, the graph is the same for different values of γ, and only the total length in expression 12 needs to be recomputed.

Entropic spanning graphs are suitable for estimating α-entropy with $\alpha \in [0,1[$, so Shannon entropy can not be directly estimated with this method. Figure 1 on the left hand shows that the shape of the function does not depend neither on the nature of data nor on their size.

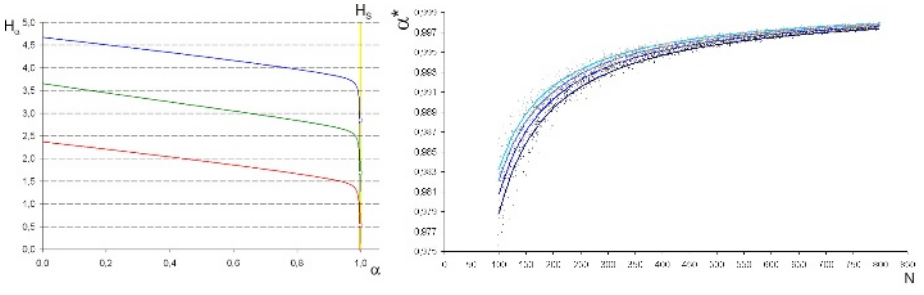

Fig. 1. Left: H_α for gaussian distributions with different covariance matrices. Right: α^* for dimensions between 2 and 5 and different number of samples.

We will approximate the value of H_α for $\alpha = 1$ by means of a continuous function that captures the tendency of H_α in the environment of 1. From a value of $\alpha \in [0,1[$, we can calculate the tangent line $y = mx + b$ to H_α in this point, using $m = H_\alpha^{'}$, $x = \alpha$ and $y = H_\alpha$. In any case, this line will be continuous and we will be able to calculate its value for $x = 1$.

From now on, we will call α^* to the α value that generates the correct entropy value in $\alpha = 1$, following the described procedure.

As H_α is a monotonous decreasing function, we can estimate α^* value in the Gaussian case by means of a dichotomic search between two well separated α values for a constant number of samples, problem dimension and different covariance matrices. Experimentally, we have verified that α^* is almost constant for diagonal covariance matrices with variance value greater than 0.5.

In order to appreciate the effects of the dimension and the number of samples on the problem, we calculated α^* for a set of 1000 distributions with random $2 \leq d \leq 5$ and number of samples. Experimentally we have verified that the shape of the underlying curve adjusts suitably to a function of the type: $\alpha^* = 1 - \frac{a+b\exp^{cD}}{N}$, where N is the number of samples, D is the problem dimension and a, b, c are three constants to estimate. In order to estimate these values, we used Monte Carlo Simulation, minimizing the mean square error between expression and data. We obtained $a = 1.271, b = 1.3912$ and $c = -0.2488$. Figure 1 on the right hand shows α_* for different dimension an number of samples.

3 Entropy-Based EM Algorithm

Comparing the estimations given for Equations 6 with 9 and 12, we have a way of quantifying the degree of Gaussianity of a given kernel. Given a set of kernels for the mixture (initially one kernel) we evaluate the real global entropy $H(y)$ and the theoretical maximum entropy $H_{max}(y)$ of the mixture by considering the individual pairs of entropies for each kernel, and their prior probabilities:

$$H(Y) = \sum_{k=1}^{K} \pi_k H_k(Y) \quad \text{and} \quad H_{max}(Y) = \sum_{k=1}^{K} \pi_k H_{max_k}(Y)\,. \tag{13}$$

If the ratio $H(y)/H_{max}(y)$ is above a given threshold we consider that all kernels are well fitted. Otherwise, we select the kernel with the lowest individual ratio and it is replaced by two other kernels that are conveniently placed and initialized. Then, a new EM with $K + 1$ kernels starts.

A low $H(y)/H_{max}(y)$ local ratio indicates that multi-modality arises and thus the kernel must be replaced by two other kernels. In the split step the original covariance matrix needs to generate two new matrices with two restrictions: overall dispersion must remain almost constant and the new matrices must be positive definite. This is an ill-posed problem because the number of equations is less than the number of unknowns [13][18].

From definition of mixture in equation 1, considering that the $K*$ component is the one with lowest Gaussianity threshold, it must be decomposed into the K_1 and K_2 components with parameters $\Theta_{k_1} = (\mu_{k_1}, \Sigma_{k_1})$ and $\Theta_{k_2} = (\mu_{k_2}, \Sigma_{k_2})$. The corresponding priors, the mean vectors and the covariance matrices should satisfy the following split equations:

$$\begin{gathered} \pi_* = \pi_1 + \pi_2 \\ \pi_*\mu_* = \pi_1\mu_1 + \pi_2\mu_2 \\ \pi_*(\Sigma_* + \mu_*\mu_*^T) = \pi_1(\Sigma_1 + \mu_1\mu_1^T) + \pi_2(\Sigma_2 + \mu_2\mu_2^T) \end{gathered} \tag{14}$$

Recently, in [5] a spectral decomposition of the actual covariance matrix is performed and the original problem is replaced by estimating the new eigenvalues and eigenvectors of new covariance matrices.

Let $\sum_* = V_* \Lambda_* V_*^T$ be the spectral decomposition of the covariance matrix $\sum_*$, with $\Lambda_* = diag(\lambda j*^1, ..., \lambda j*^d)$ a diagonal matrix containing the eigenvalues of $\sum_*$ with increasing order, $*$ the component with the lowest entropy ratio,

π_*, π_1, π_2 the priors of both original and new components, μ_*, μ_1, μ_2 the means and $\sum_*, \sum_1, \sum_2$ the covariance matrices. Let also be D a dxd rotation matrix with columns orthonormal unit vectors. D is constructed by generating its lower triangular matrix independently from $d(d-1)/2$ different uniform $U(0,1)$ densities. The proposed split operation is given by:

$$\begin{gathered}
\pi_1 = u_1\pi_*, \;\; \pi_2 = (1-u_1)\pi_* \\
\mu_1 = \mu_* - (\sum_{i=1}^{d} u_2^i \sqrt{\lambda_*^i} V_*^i)\sqrt{\frac{\pi_2}{\pi_1}}, \quad \mu_2 = \mu_* - (\sum_{i=1}^{d} u_2^i \sqrt{\lambda_*^i} V_*^i)\sqrt{\frac{\pi_1}{\pi_2}} \\
\Lambda_1 = diag(u_3)diag(\iota - u_2)diag(\iota + u_2)\Lambda_* \frac{\pi_*}{\pi_1} \\
\Lambda_2 = diag(\iota - u_3)diag(\iota - u_2)diag(\iota + u_2)\Lambda_* \frac{\pi_*}{\pi_2} \\
V_1 = DV_*, \;\; V_2 = D^T V_*
\end{gathered} \tag{15}$$

where, ι is a d x 1 vector of ones, $u_1, u_2 = (u_2^1, u_2^2, ..., u_2^d)^T$ and $u_3 = (u_3^1, u_3^2, ..., u_3^d)^T$ are $2d+1$ random variables needed to construct priors, means and eigenvalues for the new component in the mixture. They are calculated as

$$\begin{gathered}
u_1 \sim be(2,2), u_2^1 \sim be(1,2d), \\
u_2^j \sim U(-1,1), u_3^1 \sim be(1,d), u_3^j \sim U(0,1) \quad \text{and} \quad j = 2, ..., d
\end{gathered} \tag{16}$$

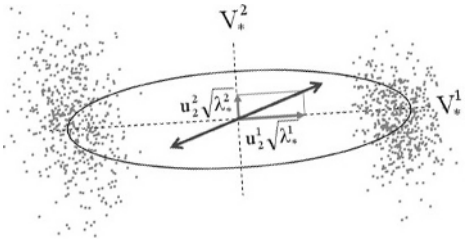

Fig. 2. 2-D Example of splitting one kernel into two new kernels

A graphical description of the splitting process in the 2-D case is showed in Fig.2. Directions and magnitudes of variability are defined by eigenvectors and eigenvalues of the covariance matrix. Otherwise, a completed algorithmic description of the process is showed in Fig. 3.

4 Experiments and Discussion

In order to test our approach we have performed several experiments with synthetic, real and image data. In the first one we have generated 2500 samples from 5 bi-dimensional Gaussians with different prior probabilities, averages and covariance matrices. We have used a Gaussianity threshold of 0.95, and a convergence threshold of 0.001 for the EM algorithm. In both, "plug-in" and "non plug-in" entropy estimation approaches our algorithm converges after 30 iterations finding correctly $k = 5$. In Figure 4 we show the evolution of the algorithm.

ENTROPY BASED EM ALGORITHM
Initialization: Start with a unique kernel.
$K \leftarrow 1$. $\Theta_1 \leftarrow \{\mu_1, \Sigma_1\}$ with $\mu_1 =$ data average and $\Sigma_1 =$ data covariance.
repeat: *//Main loop*
  repeat: *//E, M Steps*
    Estimate log-likelihood in iteration i: ℓ_i
  until: $|\ell_i - \ell_{i-1}| <$ CONVERGENCE_TH
  Evaluate $H(Y)$ and $H_{max}(Y)$ globally
  if $(H(Y)/H_{max} <$ ENTROPY_TH)
    Select kernel K_* with the lowest ratio and decompose into K_1 and K_2
    Initialize parameters Θ_1 and Θ_2(Eq.15)
      Initialize new averages: μ_1 and μ_2
      Initialize new eigenvalues and eigenvector matrices: Λ_1, Λ_2, V_1 and V_2
      Set new priors: π_1 and π_2
  else Final $\leftarrow$ True
until: Final = True

Fig. 3. Entropy Based EM algorithm

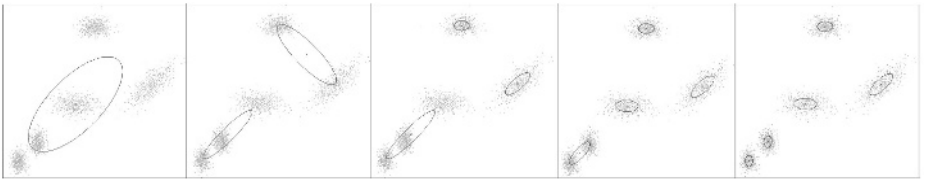

Fig. 4. Evolution of our algorithm from 1 to 5 final kernels

We have also tested our algorithm in unsupervised color image segmentation. At each pixel i in the image we compute a 3-dimensional feature vector x_i with the components in the RGB color space. We obtain the number of components (classes) M and $y_i \in [1, 2, ..., M]$ to indicate from which class the pixel i_{th} came. Therefore our image model sets that each pixel is generated by one of the Gaussian densities in the Gaussian mixture model. We have used different entropy thresholds and a convergence threshold of 0.1 for the EM algorithm. In Fig. 5 we show some results obtained from three different images. The greater it is the demanded threshold the higher is the number of kernels (colors) generated. In the "non plug-in" approach, a random selection of 1000 points has been made to estimate the MST due to memory problems. The results obtained with both methods are identical.

Finally, we have applied the proposed method to the well known *Iris* [3] data set, that contains 3 classes of 50 (4-dimensional) instances referred to a type of iris plant: *Versicolor*, *Virginica* and *Setosa*. 50 samples are insufficient to construct the pdf using Parzen. In order to test our method, we have generated 300 training samples from the averages and covariances of the original classes

Fig. 5. Color image segmentation with increasing gaussianity thresholds

and we have checked the performance in a classification problem with the original 150 samples. Starting with $K = 1$, the method correctly selected $K = 3$. Then, a maximum a posteriori classifier was built, with classification performance of 98%. With the MST approach, with no pdf estimation required, the algorithm can be executed with the original data set with the same classification performance.

5 Conclusions and Future Work

In this paper we propose a method for finding the optimal number of kernels in a Gaussian mixture based on maximum entropy. The algorithm starts with only one kernel overcoming the local convergence of the usual EM algorithm. The "plug-in" entropy estimation approach is suitable for low-dimensional problems with large data, while the "non plug-in" approach is appropriate for high-dimensional settings with a reduced data set. The algorithm is efficient for density estimation, pattern recognition and unsupervised color image segmentation. We are currently exploring methods to remove noisy features from data.

References

1. E. Beirlant, E. Dudewicz, L. Gyorfi, and E. Van der Meulen. Nonparametric entropy estimation. *International Journal on Mathematical and Statistical Sciences*, 6(1):17–39, 1996.
2. D.J. Bertsimas and G. Van Ryzin. An asymptotic determination of the minimum spanning tree and minimum matching constants in geometrical probability. *Operations Research Letters*, 9(1):223–231, 1990.

3. C.L Blake and C.J. Merz. Uci repository of machine learning databases. *University of California, Irvine, Dept. of Information and Computer Sciences*, 1998.
4. T. Cover and J. Thomas. *Elements of Information Theory*. J. Wiley and Sons, 1991.
5. P. Dellaportas and I. Papageorgiou. Multivariate mixtures of normals with unknown number of components. *Statistics and Computing*, To appear.
6. A. Dempster, N. Laird, and D. Rubin. Maximum likelihood estimation from incomplete data via the em algorithm. *Journal of The Royal Statistical Society*, 39(1):1–38, 1977.
7. M.A.T Figueiredo and A.K. Jain. Unsupervised selection and estimation of finite mixture models. In *International Conference on Pattern Recognition. ICPR2000*, Barcelona, Spain, 2000. IEEE.
8. M.A.T Figueiredo, J.M.N Leitao, and A.K. Jain. On fitting mixture models. *Energy Minimization Methods in Computer Vision and Pattern Recognition. Lecture Notes in Computer Science*, 1654(1):54–69, 1999.
9. A.O. Hero and O. Michel. Asymptotic theory of greedy aproximations to minnimal k-point random graphs. *IEEE Trans. on Infor. Theory*, 45(6):1921–1939, 1999.
10. A.O. Hero and O. Michel. Applications of entropic graphs. *IEEE Signal Processing Magazine*, 19(5):85–95, 2002.
11. E. Parzen. On estimation of a probability density function and mode. *Annals of Mathematical Statistics*, 33(1):1065–1076, 1962.
12. R.A. Redner and H.F. Walker. Mixture densities, maximum likelihood, and the em algorithm. *SIAM Review*, 26(2):195–239, 1984.
13. S. Richardson and P.J. Green. On bayesian analysis of mixtures with unknown number of components (with discussion). *Journal of the Royal Statistical Society B*, (1), 1997.
14. N. Ueda, R. Nakano, Z. Ghahramani, and G. E. Hinton. Smem algorithm for mixture models. *Neural Computation*, 12(1):2109–2128, 2000.
15. P. Viola, N. N. Schraudolph, and T. J. Sejnowski. Empirical entropy manipulation for real-world problems. *Adv. in Neural Infor. Proces. Systems*, 8(1), 1996.
16. N. Vlassis and A. Likas. A kurtosis-based dynamic approach to gaussian mixture modeling. *IEEE Trans. Systems, Man, and Cybernetics*, 29(4):393–399, 1999.
17. N. Vlassis and A. Likas. A greedy em algorithm for gaussian mixture learning. *Neural Processing Letters*, 15(1):77–87, 2000.
18. Z. Zhang, K.L. Chan, Y. Wu, and C. Chen. Learning a multivariate gaussian mixture models with the reversible jump mcmc algorithm. *Statistics and Computing*, (1), 2004.