Short notes on Local Consistent Gaussian Mixture Model

Jialu Liu

State Key Lab of CAD&CG, College of Computer Science
Zhejiang University
100 Zijinggang Road, 310058, China

Abstract

Gaussian mixture models (GMM) can be viewed as a linear combination of different Gaussian models where each component is a basis function or a "hidden" unit, aiming at offering a comparatively richer model than the single Gaussian. It is among the most statistically generative model for unsupervised learning. In this paper, We take into account the smoothness of the conditional probability distribution along the geodesics of data manifold. I.e., if two observations are "close" in intrinsic geometry, there distribution to different Gaussian components are similar. Therefore, we construct a neighboring graph and adopt Kullback–Leibler divergence as the "distance" measurement to regularize the objective function of GMM. Consequently, new estimating formulae for parameters are obtained. We call this new method *Locally Consistent Gaussian Mixture Model* (LCGMM). Experiments on several real data sets demonstrate the effectiveness of such regularization.

Keywords: Gaussian Mixture Model, regularization, KL–divergence, graph

1. Background

Gaussian mixture model can be viewed as a linear superposition of different Gaussian components in which each is a basis function or a "hidden" unit, aiming at offering a comparatively richer model than the single Gaussian [1]:

$$P(x;\Theta) = \sum_{k=1}^{K} \pi_k p(x;\theta_k)$$

where we assume K clusters altogether and each component prior (π_k) can be viewed as positive weights in an output layer and satisfying $\sum_{k=1}^{K} \pi_k = 1$. Here all parameters are represented by Θ where $\Theta = (\pi_1, \ldots, \pi_K, \theta_1, \ldots, \theta_K)$. Note that each θ_k describes a Gaussian density function p_k, meaning that $p(x;\theta_k) \sim \mathcal{N}(x;\mu_k,\Sigma_k)$.

The optimal parameter Θ is estimated by Maximum Likelihood (ML) principle. Given N observations $\mathcal{X} = (x_1, x_2, \ldots, x_N)$, ML tries to find Θ such that $P(\mathcal{X};\Theta)$ is a maximum. For the sake of efficient optimization, it is typical to introduce the log likelihood function defined as follows:

$$\mathcal{L}(\Theta) = \log P(\mathcal{X};\Theta) = \log \prod_i P(x_i;\Theta)$$

Since the above log likelihood function contains the log of the sum, it is difficult to find the optimal solution. According to the Jensen's Inequality, we know that:

$$\begin{aligned}
\mathcal{L}(\Theta) &= \sum_i \log P(x_i;\Theta) \\
&= \sum_i \log \sum_z P(x_i, z;\Theta) \\
&= \sum_i \log \left(\sum_z Q_i(z) \frac{P(x_i, z;\Theta)}{Q_i(z)} \right) \\
&\geq \sum_i \sum_z Q_i(z) \log \frac{P(x_i, z;\Theta)}{Q_i(z)}
\end{aligned}$$

For each i, let Q_i be some discrete distribution over the latent variable z [1], satisfying that $\sum_z Q_i(z) = 1$ and $Q_i(z) \geq 0$.

*Corresponding author
Email address: remenberl@gmail.com (Jialu Liu)

[1] If z were continuous, the summations over z in the above equation should be replaced with integrals over z.

The equality holds when $Q_i(z) = P(z|x_i;\Theta)$, which here represents the possibility of observation x belonging to the component z. Therefore, after removing the term irrelevant to Θ, the *complete* log likelihood function can be written as [1]:

$$\sum_{i=1}^{N}\sum_{k=1}^{K} P(z_k|x_i;\Theta)\Big(\log\pi_k + \log p(x_i;\theta_k)\Big) \quad (1)$$

With this complete log likelihood, we are able to obtain estimates for Θ under the assumption that $P(z|x;\Theta)$ is fixed. This procedure is known as Expectation-Maximization algorithm [2] , which is a powerful method for finding maximum likelihood solutions for models with latent variables. It is a process of iteration which alternates between an expectation (E) step computing an expectation of the latent variable ($z|x;\Theta^{(t)}$ in the GMM case), and a maximization (M) step computing the new parameters ($\Theta^{(t+1)}$) which maximize the complete log likelihood. Parameters computed either in E or M step are alternatively fixed during the other step as known quantities. Therefore, the EM algorithm can be viewed as coordinate ascent on Q and Θ.

In fact, there is a close similarity between K-means and EM algorithm for Gaussian mixtures [1]. The K-means algorithm does the clustering in a *hard* way, in which each sample is associated directly with only one cluster, while the EM algorithm makes a comparatively *soft* assignment relied on the posterior probabilities. It is noticeable that we can derive the K-means algorithm as a nonprobabilistic limit of EM for GMM. For more information, please see [1].

2. GMM with Local Consistent Regularizer

Gaussian Mixture Model is among the most statistically mature methods for clustering. However, in some cases, owing to ignoring the smoothness of the variation in the probability distribution of samples, GMM might not obtain a comparatively ideal result. In this section, we introduce a novel method LCGMM to extract such kind of intrinsic geometry.

2.1. Model with Graph Regularization

Recall that clustering is the assignment of a set of observations into subsets (called clusters) so that observations in the same cluster are similar in some sense. And in GMM, we tag elements for clusters according to the maximum possibility in different Gaussian models. Unfortunately, such rule will lead to relatively isolated observation compared with its neighbors.

Therefore, we make a specific assumption about the intrinsic connection between the distribution of observations P_X and the conditional distribution $P(z|x;\Theta^{(t)})$, where z represents the components. That is, within some neighboring observations, their distributions $P(z|x;\Theta^{(t)})$ are "similar" to a certain degree. I.e., the variation of $P(z|x;\Theta^{(t)})$ is smooth in the geometry of P_X.

2.1.1. Distance Measurement

After discussing about the idea of improving GMM, we will introduce the detailed model subsequently. The first question is how to measure the "distance".

Here we adopt Kullback–Leibler divergence (KL–divergence) as our "distance" function. Given any two distributions $P_i(z)$ and $P_j(z)$, the KL–divergence between them is defined as below:

$$D\big(P_i(z)\|P_j(z)\big) = \sum_z P_i(z)\log\frac{P_i(z)}{P_j(z)}$$

Since the equation is not symmetric and also for simplicity for the following computations, we modify it a little to represent our distance between conditional distributions of observations like this:

$$\mathcal{D}_{ij} = \frac{1}{2}\Big(D\big(P_i(z)\|P_j(z)\big) + D\big(P_j(z)\|P_i(z)\big)\Big)$$

2.1.2. Structure Description

Now we must consider a model to describe the local geometry structure when data are given. Recent studies on spectral graph theory [3] and manifold learning theory [4] have demonstrated that the local geometric structure can be effectively modeled through a nearest neighbor graph on a scatter of data points. Consider a graph with N vertices where each vertex corresponds to a data point. Define the edge weight matrix W as follows:

$$W_{ij} = \begin{cases} 1 & \text{if } x_i \in N_p(x_j) \text{ or } x_j \in N_p(x_i). \\ 0 & \text{otherwise.} \end{cases}$$

where $N_p(x_i)$ denotes the data sets of p nearing neighbors of x_i. Note that here we make use of Euclidean distance to measure "near".

2.1.3. Model Reconstruction

Once we let $P_i(z) = P(z|x_i;\Theta) = Q_i(z)$ accommodating with the nearest neighbor graph, we are able to

obtain the following term to describe the local smoothness of $P(z|x_i;\Theta)$:

$$\begin{aligned}\mathcal{R} =& \sum_{i,j} \mathcal{D}_{ij} W_{ij} \\ =& \frac{1}{2} \sum_{i,j} \Big(D\big(Q_i(z) \| Q_j(z)\big) + D\big(Q_j(z) \| Q_i(z)\big) \Big) W_{ij}\end{aligned}$$

When x_i and x_j are close to each other, $\mathcal{R}$ should be rather small. Therefore, we can reconstruct our GMM by minimizing $\mathcal{R}$, which will sufficiently help smooth the geometric structure.

We now remodel the log-likelihood of GMM with the regularizer as follows:

$$\begin{aligned}\mathfrak{L} =& \mathcal{L} - \lambda \mathcal{R} \\ =& \sum_i \log P(x_i;\Theta) - \lambda \sum_{i,j} \mathcal{D}_{ij} W_{ij} \\ \geq& \sum_i \sum_z Q_i(z) \log \frac{P(x_i, z; \Theta)}{Q_i(z)} \\ & - \frac{\lambda}{2} \sum_{i,j} \Big(D\big(Q_i(z) \| Q_j(z)\big) + D\big(Q_j(z) \| Q_i(z)\big) \Big) W_{ij}\end{aligned} \tag{2}$$

where λ is the regularization parameter.

2.2. Model Fitting with EM

To find maximum likelihood estimation for this remodelling equation of LCGMM, we also need to make use of the EM algorithm. In our case, the latent variables are the Gaussian components to which the data points belong. Firstly, we need to estimate values to perform the E-step, computing the expectation of $z_k|x_i;\Theta^{(t)}$. Then we use these variables to obtain the parameters $\Theta^{(t+1)}$ which maximize the log likelihood (M-step). These two steps are repeated until a certain stopping criterion is reached.

E-step:

It can be easily verified that the equality in Eq. (2) holds still when $Q_i^{(t)}(z) = P(z|x_i;\Theta^{(t)})$. Consequently, the E-step for LCGMM is exactly the same as that in original GMM. The posterior probabilities for the latent variables can be computed by simply applying Bayes' formula [1]:

$$\begin{aligned}P(z_k|x_i;\Theta^{(t)}) =& E\big(z_k|x_i;\Theta^{(t)}\big) \\ =& \frac{\pi_k^{(t)} p(x_i;\theta_k^{(t)})}{P(x_i;\Theta^{(t)})} \\ =& \frac{\pi_k^{(t)} \mathcal{N}(x_i;\mu_k^{(t)},\Sigma_k^{(t)})}{\sum_{j=1}^K \pi_j^{(t)} \mathcal{N}(x_i;\mu_j^{(t)},\Sigma_j^{(t)})}\end{aligned} \tag{3}$$

M-step:

With simple derivations [1], one can obtain the expected *complete* data log-likelihood for LCGMM:

$$\begin{aligned}&\mathcal{Q}(\Theta^{(t+1)}) = \mathcal{Q}_1(\Theta^{(t+1)}) - \mathcal{Q}_2(\Theta^{(t+1)}) \\ =& \sum_{i=1}^N \sum_{k=1}^K Q_i^{(t)}(z) \Big(\log \pi_k^{(t+1)} + \log p(x_i;\theta_k^{(t+1)}) \Big) \\ & - \frac{\lambda}{2} \sum_{i,j=1}^N \Big(D\big(Q_i^{(t+1)}(z) \| Q_j^{(t+1)}(z)\big) \\ & \qquad + D\big(Q_j^{(t+1)}(z) \| Q_i^{(t+1)}(z)\big) \Big) W_{ij}\end{aligned} \tag{4}$$

Notice that $\mathcal{Q}(\Theta)$ has two parts. The first part $\mathcal{Q}_1(\Theta)$ is exactly the expected complete data log-likelihood for GMM in Eq. (1). And $\mathcal{Q}_2(\Theta)$ is the locally consistent regularizer, which is the part we need to expand.

With the posterior probabilities for the latent variables in Eq. (3) estimated in E-step, we have:

$$\begin{aligned}& D\big(Q_i^{(t+1)}(z) \| Q_j^{(t+1)}(z)\big) \\ =& \sum_{k=1}^K Q_i^{(t+1)}(z_k) \log \frac{Q_i^{(t+1)}(z_k)}{Q_j^{(t+1)}(z_k)} \\ \approx& \sum_{k=1}^K Q_i^{(t)}(z_k) \log \frac{p(x_i;\theta_k^{(t+1)})}{p(x_j;\theta_k^{(t+1)})} + \mathcal{O}(x_i||x_j;\Theta^{(t+1)})\end{aligned} \tag{5}$$

Since $\sum_k Q_i^{(t)}(z_k) = 1$, we get:

$$\mathcal{O}(x_i||x_j;\Theta^{(t+1)}) = \log \frac{P(x_i;\Theta^{(t+1)})}{P(x_i;\Theta^{(t+1)})}$$

It is easy to see that:

$$\mathcal{O}(x_i||x_j;\Theta^{(t+1)}) + \mathcal{O}(x_j||x_i;\Theta^{(t+1)}) = 0$$

meaning that only the former term of the last line in Eq. (5) will be involved in the optimization process.

Remember that the edge weight matrix W is symmetric, consequently, we are able to rewrite the complete data log-likelihood Eq. (4) as follows:

$$\begin{aligned}\mathcal{Q}(\Theta^{(t+1)}) =& \sum_{i=1}^N \sum_{k=1}^K P(z_k|x_i;\Theta^{(t)}) \Big(\log \pi_k^{(t+1)} \\ & + T_{i,k}^{(t)} \log \mathcal{N}(x_i;\mu_k^{(t+1)},\Sigma_k^{(t+1)}) \Big)\end{aligned} \tag{6}$$

where

$$T_{i,k}^{(t)} = 1 - \lambda \sum_{j=1}^N (1 - \frac{P(z_k|x_j;\Theta^{(t)})}{P(z_k|x_i;\Theta^{(t)})}) W_{ij}$$

We now know that the second part $\mathcal{Q}_2(\Theta)$ is the locally consistent regularizer which only involves the parameters $\{\mu_k^{(t+1)}, \Sigma_k^{(t+1)}\}_{k=1}^K$. Thus, the M-step re-estimation equation for $\pi_k^{(t+1)}$ will be exactly the same as that in GMM. It is [1]:

$$\pi_k^{(t+1)} = \frac{\sum_{i=1}^N P(z_k|x_i;\Theta^{(t)})}{N} \tag{7}$$

Next let us derive the re-estimation equations for $\{\mu_k^{(t+1)}, \Sigma_k^{(t+1)}\}_{k=1}^K$.

The relevant part of Eq. (6) is:

$$\sum_{i=1}^N \sum_{k=1}^K P(z_k|x_i;\Theta^{(t)}) \Big(\frac{1}{2} \log |(\Sigma_k^{(t+1)})^{-1}| - \frac{1}{2}(x_i - \mu_k^{(t+1)})^T (\Sigma_k^{(t+1)})^{-1} (x_i - \mu_k^{(t+1)}) \Big) T_{i,k}^{(t)} \tag{8}$$

By taking the derivative of Eq. (8) with respect to ${\mu_k}^{(t+1)}$ and setting it to zero, we get:

$$\sum_{i=1}^N P(z_k|x_i;\Theta^{(t)}) \Big((\Sigma_k^{(t+1)})^{-1} (x_i - \mu_k^{(t+1)}) \Big) T_{i,k}^{(t)} = 0$$

By solving the equation above, one obtains the M–step re-estimation equation for $\mu_k^{(t+1)}$:

$$\mu_k^{(t+1)} = \frac{\sum_{i=1}^N P(z_k|x_i;\Theta^{(t)}) T_{i,k}^{(t)} x_i}{N_k^{(t)}} \tag{9}$$

where

$$N_k^{(t)} = \sum_{i=1}^N P(z_k|x_i;\Theta^{(t)})$$

Let $S_{i,k}^{(t+1)} = (x_i - \mu_k^{(t+1)})(x_i - \mu_k^{(t+1)})^T$, we have:

$$\begin{aligned} &(x_i - \mu_k^{(t+1)})^T (\Sigma_k^{(t+1)})^{-1} (x_i - \mu_k^{(t+1)}) \\ &= \mathrm{Tr}\left(S_{i,k}^{(t+1)} (\Sigma_k^{(t+1)})^{-1} \right) \\ &= \mathrm{Tr}\left((\Sigma_k^{(t+1)})^{-1} S_{i,k}^{(t+1)} \right) \end{aligned}$$

where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix. We can rewrite the Eq. (8) as:

$$\frac{1}{2} \sum_{i=1}^N \sum_{k=1}^K P(z_k|x_i;\Theta^{(t)}) \Big(\log(|(\Sigma_k^{(t+1)})^{-1}|) - \mathrm{Tr}\left((\Sigma_k^{(t+1)})^{-1} S_{i,k}^{(t+1)} \right) \Big) T_{i,k}^{(t)} \tag{10}$$

By taking the derivative of Eq. (10) with respect to $(\Sigma_k^{(t+1)})^{-1}$ and setting it to zero[2], we get:

$$\frac{1}{2} \sum_{i=1}^N P(z_k|x_i;\Theta^{(t)}) \Big(\Sigma_k^{(t+1)} - S_{i,k}^{(t+1)} \Big) T_{i,k}^{(t)} = 0$$

Solving the above equation, we obtain the M-step re-estimation equation for $\Sigma_k^{(t+1)}$:

$$\Sigma_k^{(t+1)} = \frac{\sum_{i=1}^N P(z_k|x_i;\Theta^{(t)}) T_{i,k}^{(t)} S_{i,k}^{(t+1)}}{N_k^{(t)}} \tag{11}$$

When the regularization parameter $\lambda = 0$, we can easily see the above M-step re-estimation equations (Eq. 9 and 11) boil down to the M-step in original GMM. The E-step (Eq. 3) and M-step (Eq. 7, 9 and 11) are alternated until a termination condition is met.

2.3. Flow of LCGMM

The steps can be summarized as follows:

$t \leftarrow 0$
$\Theta^{(0)} \leftarrow$ initial guess values[3]
$Q^{(t)} \leftarrow$ estimation by Eq. (3)
$\Theta^{(t+1)} \leftarrow$ estimation by Eq. (7), (9) and (11)
if (termination condition not satisfied[4])
    $t \leftarrow t + 1$ and go to line. 3
else
    Quit the loop.

Once the algorithm exits the loop, we are able to assign the i^{th} observation to corresponding cluster on the basis of the value of $Q_i(z)$.

References

[1] C. M. Bishop, Pattern Recognition and Machine Learning, 2006.
[2] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the em algorithm, Journal of the Royal Statistical Society. Series B (Methodological).
[3] F. R. K. Chung, Spectral Graph Theory, Vol. 92 of Regional Conference Series in Mathematics, 1997.
[4] M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, in: Advances in Neural Information Processing Systems 14, 2001.

[2] Note that $\partial \log|M|/\partial M = (M^{-1})^T$, $\partial \mathrm{Tr}(MN)/\partial M = N^T$ and both Σ_k and $S_{i,k}$ are symmetric matrices.

[3] We can pick random values or simply make use of other algorithms like K-means to get estimated value.

[4] Termination conditions are like fixed iterations or convergence.