UCSB Spring 2018

ECE 283: Gaussian Mixtures and the EM Algorithm

1 Introduction

Gaussian mixtures are a popular data modeling approach because of their flexibility and analytical tractability. In order to fit such models to the data available to us, we must estimate the model parameters from the data. Maximum likelihood (ML) parameter estimation for Gaussian distributions is straightforward, but direct ML estimation of the parameters of a Gaussian mixture is difficult. However, ML estimation for a single Gaussian distribution is straightforward, which can be leveraged to obtain an efficient iterative algorithm for mixture parameter estimation that is guaranteed to converge to a local optimum of the likelihood function. This algorithm is an application of a general approach called the Expectation Maximization (EM) algorithm.

Our set-up for ML parameter estimation is as follows. Our observations are denoted by $\mathbf{x}_1, ..., \mathbf{x}_N$. These are assumed to be realizations of $\mathbf{X}_1, ..., \mathbf{X}_N$, which are i.i.d. d-dimensional random vectors each with density $p(\mathbf{x}|\theta)$, where θ is a vector of parameters to be estimated. The ML estimate of θ is defined as

$$\hat{\theta}_{ML} = \arg\ \max_\theta p(\mathbf{x}_1, ..., \mathbf{x}_N|\theta) = \arg\ \max_\theta \sum_{i=1}^{N} \log p(\mathbf{x}_i|\theta)$$

where we have used the monotonicity of the log and the conditional independence of the observations to obtain the second equality.

2 Parameter estimation for Gaussian distributions

Suppose that the $\mathbf{X}_i \sim N(\mathbf{m}, \mathbf{C})$, where

$$\mathbf{m} = \mathbb{E}\left[\mathbf{X}_i\right]$$

is a d-dimensional mean vector, and

$$\mathbf{C} = \mathbb{E}\left[(\mathbf{X}_i - \mathbf{m})(\mathbf{X}_i - \mathbf{m})^T\right]$$

The conditional density for any of the $\mathbf{X}_i$ (assuming that $\mathbf{C}$ is invertible) is given by

$$p(\mathbf{x}|\theta) = N(\mathbf{x}|\mathbf{m}, \mathbf{C}) = \frac{1}{(2\pi)^{d/2}|\mathbf{C}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{m})^T \mathbf{C}^{-1}(\mathbf{x} - \mathbf{m})\right)$$

where we have adopted the useful shorthand $N(\mathbf{x}|\mathbf{m}, \mathbf{C})$ in Bishop's book for a Gaussian density. Note that $|\mathbf{C}|$ denotes the determinant of $\mathbf{C}$.

ML estimation requires the maximization of the following cost function over $\mathbf{m}$ and $\mathbf{C}$:

$$J(\theta) = \sum_{i=1}^{N} \log p(\mathbf{x}_i|\theta) = -\frac{1}{2}\sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{m})^T \mathbf{C}^{-1}(\mathbf{x}_i - \mathbf{m}) \ - \frac{d}{2}\log|C| - \frac{d}{2}\log 2\pi$$

It can be shown that the ML estimate of the mean vector is simply the sample mean:

$$\hat{\mathbf{m}}_{ML} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i \tag{1}$$

and that the ML estimate of the covariance matrix is given by averaging the correlations among the variations around the sample mean:

$$\hat{\mathbf{C}}_{ML} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \hat{\mathbf{m}}_{ML})(\mathbf{x}_i - \hat{\mathbf{m}}_{ML})^T \tag{2}$$

(The covariance estimate is biased, and can be made unbiased by multiplying by $\frac{N}{N-1}$.)

3 Parameter estimation for Gaussian mixtures

We now assume that $\mathbf{X}_i$ is modeled as being drawn from a mixture of K Gaussians. For $1 \leq k \leq K$, the kth component of the mixture is a $N(\mathbf{m}_k, \mathbf{C}_k)$ distribution, and π_k denotes the probability of drawing $\mathbf{X}_i$ from the kth component. The parameters characterizing the mixture are therefore given by

$$\theta = \{\mathbf{m}_k, \mathbf{C}_k, \pi_k, 1 \leq k \leq K\}$$

with the constraint that

$$\pi_k \geq 0, \quad \sum_{k=1}^{K} \pi_k = 1$$

The mixture density is given by

$$p(\mathbf{x}|\theta) = \sum_{k=1}^{K} \pi_k N(\mathbf{x}|\mathbf{m}_k, \mathbf{C}_k)$$

For $K > 1$, the ML cost function

$$J(\theta) = \sum_{i=1}^{N} \log p(\mathbf{x}_i|\theta) \tag{3}$$

no longer has a nice quadratic form, hence direct maxiimization becomes difficult. However, if we *knew* which mixture component each $\mathbf{x}_i$ belonged to, then we could simply group all observations drawn from component k together, and estimate $\mathbf{m}_k$, $\mathbf{C}_k$ using ML estimation for a single Gaussian as in (1)-(2). The information on which component $\mathbf{X}_i$ is drawn from can be thought of as a *hidden* variable Z_i, a central concept in the development of the EM algorithm. Since we do not know these hidden variables, we instead try to estimate the probabilities that $\mathbf{X}_i$ belongs to a given component, and then use (1)-(2) to update the corresponding parameters. This leads to the following intuitively reasonable iterative algorithm. (In the following, we drop the "hat" notation for the parameter estimates.)

Assignment step: Assume that we have estimates of the mixture parameters. For each data point $\mathbf{x}_i$ $(1 \leq i \leq N)$ and each cluster $1 \leq k \leq K$, estimate the probabilities that $\mathbf{x}_i$ belongs to mixture component k:

$$p(k|\mathbf{x}_i) = \frac{\pi_k p(\mathbf{x}_i|k]}{p(\mathbf{x}_i)} = \frac{\pi_k N(\mathbf{x}_i|\mathbf{m}_k, \mathbf{C}_k)}{\sum_{j=1}^{K} \pi_j N(\mathbf{x}_i|\mathbf{m}_j, \mathbf{C}_j)} \tag{4}$$

Update step: , We now use these assignment probabilities as weights as to how much each data point counts towards estimating the parameters for a given cluster, and update the parameters for each component k as follows:

$$\mathbf{m}_k = \frac{\sum_{i=1}^{N} p(k|\mathbf{x}_i)\mathbf{x}_i}{\sum_{i=1}^{N} p(k|\mathbf{x}_i)} \tag{5}$$

$$\mathbf{C}_k = \frac{\sum_{i=1}^N p(k|\mathbf{x}_i)(\mathbf{x}_i - \mathbf{m}_k)(\mathbf{x}_i - \mathbf{m}_k)^T}{\sum_{i=1}^N p(k|\mathbf{x}_i)} \tag{6}$$

$$\pi_k = \frac{\sum_{i=1}^N p(k|\mathbf{x}_i)}{N} \tag{7}$$

This algorithm has the desirable property of converging to a local maximum of the log likelihood function. We prove this indirectly by first developing the EM algorithm and its convergence in a more general setting, and then showing that the preceding algorithm represents an application of the EM algorithm.

4 EM Algorithm

We have an observation X and a parameter θ. Both can be vectors, but we are dropping boldface notation in this section. In our Gaussian mixture application, $X = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ and θ is the set of mixture parameters that we previously denoted by θ.

We are interested in settings in which ML cost function $\log p(X|\theta)$ is difficult to optimize. The EM algorithm is based on two ideas. The *first idea* is introduce hidden data Z such that the cost function $\log p(X, Z|\theta)$ is more tractable. In the Gaussian mixture context, the hidden data tells us which component of the mixture each observed data sample belongs to. Of course, we do not see the hidden data Z. Which brings us to the *second idea:* replace the cost function $\log p(X, Z|\theta)$ by its expectation, conditioned on what we know so far, and then optimize over θ. "What we know so far" is the observed data X and the current estimate of the parameter.

The observed data X is often called the "incomplete data," while $X_c = (X, Z)$ is called the "complete data." While our discussion of the EM algorithm focuses on complete data that separates neatly into the observed data X and the hidden data Z, the EM algorithm applies to a more general scenario in which the observed data $X = h(X_c)$, i.e., it is some function of the complete data.

The EM algorithm is an iterative algorithm with the following two steps.

E-step (expectation): Letting $\theta^{(\ell)}$ denote the current parameter estimate, compute the expectation

$$Q\left(\theta|\theta^{(\ell)}\right) = \mathbb{E}\left[\log p(X, Z|\theta)|X, \theta^{(\ell)}\right] \tag{8}$$

Note: Remember that X is the observed data, which is given to us, so the preceding expectation is to average out Z, the hidden data that we do not see. This average uses the current estimate of the parameter $\theta^{(\ell)}$.

M-step (maximization): Update the parameter estimate by maximizing the function computed in the E-step.

$$\theta^{(\ell+1)} = \arg\ \mathrm{max}_\theta Q\left(\theta|\theta^{(\ell)}\right) \tag{9}$$

Let us first apply this to Gaussian mixtures, and then sketch a proof.

4.1 Application to Gaussian mixtures

We now show how applying the EM algorithm to derive the iterative algorithm for Gaussian mixture models described in Section 3. Recall that our observed data $X = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$. Introduce hidden data $Z = \{\mathbf{z}_1, ..., \mathbf{z}_N\}$ such that $\mathbf{z}_i$ tells us which component $\mathbf{x}_i$ is drawn from. In order to simplify the dependence of $\log p(X, Z|\theta)$ on Z, it is convenient to use a "one of K" encoding for the $\mathbf{z}_i = (z_i[1], ..., z_i[K])^T$. Specifically, define $\mathbf{z}_i$ as a K-dimensional vector with $\mathbf{z}_i = \mathbf{e}_k$ (the unit vector with 1 in the kth component and zeros elsewhere) if $\mathbf{x}_i$ is drawn from component k.

Since $\log p(X, Z|\theta) = \sum_{i=1}^{N} \log p(\mathbf{x}_i, \mathbf{z}_i|\theta)$ and expectation is a linear operator, we can separately take the expectations of the terms in the summation. For the ith term, note that

$$P(\mathbf{z}_i = \mathbf{e}_k|\theta) = \pi_k$$

Since $\mathbf{z}_i$ has exactly one nonzero component which equals one, with remaining components equal to zero, we can write

$$P(\mathbf{z}_i|\theta) = \prod_{k=1}^{K} \pi_k^{z_i[k]} \tag{10}$$

Conditioned on $\mathbf{z}_i$, the density of $\mathbf{x}_i$ is a Gaussian:

$$p(\mathbf{x}_i|\mathbf{z}_i = \mathbf{e}_k, \theta) = N\left(\mathbf{x}_i|\mathbf{m}_k, \mathbf{C}_k\right) \tag{11}$$

The "one of K" allows us to write the joint density in a form that has a linear dependence on Z:

$$\log p(\mathbf{x}_i, \mathbf{z}_i|\theta) = \log p(\mathbf{x}_i|\mathbf{z}_i, \theta) + \log p(\mathbf{z}_i|\theta) = \sum_{k=1}^{K} z_i[k] \left(\log N\left(\mathbf{x}_i|\mathbf{m}_k, \mathbf{C}_k\right) + \log \pi_k\right) \tag{12}$$

Conditioning now on the current estimates of the mixture parameters (and recalling that $\mathbf{x}_i$ is fixed–this is the observed data), all we need to do in the E-step is to replace $z_i[k]$ by its conditional expectation for each i and k. Since $z_i[k]$ are Bernoulli random variables, $E[z_i[k]] = P[z_i[k] = 1]$ (where we do not show the conditioning in our notation). We can now use Bayes' rule to show that these conditional expectations are exactly the assignment probabilities $P(k|\mathbf{x}_i)$ computed in (4).

Substituting the variables $\{z_i[k]\}$ by their conditional expectations $P(k|\mathbf{x}_i)$ in (12) and summing over i, we obtain the cost function $Q(\theta|\theta^\ell)$ to be maximized as follows:

$$Q(\theta|\theta^\ell) = \sum_{i=1}^{N} \sum_{k=1}^{K} P[k|\mathbf{x}_i] \left(\log N\left(\mathbf{x}_i|\mathbf{m}_k, \mathbf{C}_k\right) + \log \pi_k\right) \; + \; \lambda \sum_{k=1}^{K} \pi_k \tag{13}$$

where we need a Lagrange multiplier λ to take care of the constraint $\sum_{k=1}^{K} \pi_k = 1$. Since the mixture terms now have separated out, we can now use the ML estimation for Gaussian densities separately for each k, and this leads to the update equations (5) and (6). Taking partial derivatives with respect to π_k leads to the update equation (7).

4.2 Convergence

We now sketch a proof that the EM algorithm converges to a local maximum of $\log p(X|\theta)$, which is the cost function we wish to optimize for computing the ML solution.

Rewriting the joint density of the observed and hidden data, we have

$$p(X, Z|\theta) = p(X|\theta)p(Z|X, \theta)$$

so that

$$\begin{aligned} Q\left(\theta|\theta^{(\ell)}\right) &= \mathbb{E}\left[\log p(X|\theta) + \log p(Z|X, \theta)|X, \theta^{(\ell)}\right] \\ &= \log p(X|\theta) + \mathbb{E}\left[\log p(Z|X, \theta)|X, \theta^{(\ell)}\right] \end{aligned} \tag{14}$$

The second term on the extreme right-hand side can be rewritten as follows:

$$\mathbb{E}\left[\log p(Z|X, \theta)|X, \theta^{(\ell)}\right] = -\mathbb{E}\left[\log \frac{p(Z|X, \theta^{(\ell)})}{p(Z|X, \theta)}|X, \theta^{(\ell)}\right] \; + \; \mathbb{E}\left[\log p(Z|X, \theta^{(\ell)})|X, \theta^{(\ell)}\right]$$

We recognize that the first term on the right-hand side is the negative of an information-theoretic divergence (see appendix for definition and properties), while the second term, which we now denote by a_ℓ is independent of θ. Thus, we can rewrite the preceding as

$$\mathbb{E}\left[\log p(Z|X,\theta)|X,\theta^{(\ell)}\right] = -D\left(p(Z|X,\theta^{(\ell)})||p(Z|X,\theta)\right) + a_\ell$$

Plugging into (14), we can write

$$Q\left(\theta|\theta^{(\ell)}\right) = \log p(X|\theta) - D\left(p(Z|X,\theta^{(\ell)})||p(Z|X,\theta)\right) + a_\ell \tag{15}$$

Thus, when we maximize over θ, we are trying to maximize the ML cost function $\log p(X|\theta)$, but with the divergence term penalizing straying too far away from our current estimate $\theta^{(\ell)}$.

Once we maximize over θ, we obtain a new estimate $\theta^{(\ell+1)}$ that satisfies (by its definition as a maximizer)

$$Q\left(\theta^{(\ell+1)}|\theta^{(\ell)}\right) \geq Q\left(\theta^{(\ell)}|\theta^{(\ell)}\right) \tag{16}$$

Substituting from (15), we obtain

$$\begin{aligned} &\log p(X|\theta^{(\ell+1)}) - D\left(p(Z|X,\theta^{(\ell)})||p(Z|X,\theta^{(\ell+1)})\right) + a_\ell \\ &\quad \geq \log p(X|\theta^{(\ell)}) - D\left(p(Z|X,\theta^{(\ell)})||p(Z|X,\theta^{(\ell)})\right) + a_\ell \end{aligned} \quad l \tag{17}$$

Since $D\left(p(Z|X,\theta^{(\ell)})||p(Z|X,\theta^{(\ell)}\right) = 0$ (divergence of a probability distribution with respect to itself is zero), we obtain upon simplification that

$$\log p(X|\theta^{(\ell+1)}) \geq \log p(X|\theta^{(\ell)}) + D\left(p(Z|X,\theta^{(\ell)})||p(Z|X,\theta^{(\ell+1)})\right) \geq \log p(X|\theta^{(\ell)}) \tag{18}$$

where we have used the fact that the divergence is always nonnegative (it is strictly positive unless the two distributions are identical). Thus, the EM algorithm leads to a monotonic increase in the ML cost function $\log p(X|\theta)$, and therefore converges. Under the appropriate smoothness assumption, it can be shown that the limit is a local maximum of the cost function (i.e., the gradient of the cost function with respect to θ vanishes).

Relaxation of the M-step: In order to ensure convergence, we need only ensure the inequality (16), which only requires that we update the parameter estimate such that $Q\left(\theta|\theta^{(\ell)}\right)$ increases. Thus, if maximization over θ is too difficult, we can relax the M-update so as to just ensure that (16) holds. An algorithm incorporating such a relaxation is called a *generalized EM* algorithm.

Appendix: Divergence

Information-theoretic *divergence*, also termed the *Kullback-Leibler (KL) distance,* is defined as follows.

Divergence: The divergence $D(P||Q)$ between two distributions P and Q (with corresponding densities $p(x)$ and $q(x)$) is defined as

$$D(P||Q) = \mathbb{E}_P\left[\frac{\log p(X)}{\log q(X)}\right] = \sum_x p(x)\frac{\log p(x)}{\log q(x)}$$

where $\mathbb{E}_P$ denotes expectation computed using the distribution P (i.e., X is a random variable with distribution P).

Divergence is nonnegative: The divergence $D(P||Q) \geq 0$, with equality if and only if $P \equiv Q$. The proof is as follows:

$$\begin{aligned} -D(P||Q) &= \mathbb{E}_P\left[\log\left(\tfrac{q(X)}{p(X)}\right)\right] = \textstyle\sum_{x:p(x)>0} p(x)\log\left(\tfrac{q(x)}{p(x)}\right) \\ &\leq \textstyle\sum_{x:p(x)>0} p(x)\left(\tfrac{q(x)}{p(x)} - 1\right) = \left(\textstyle\sum_{x:p(x)>0} q(x)\right) \; - \; 1 \leq 0 \end{aligned}$$

where the first inequality is because $\log x \leq x - 1$. Since equality in the latter inequality occurs if and only if $x = 1$, the first inequality is an equality if and only if $\frac{q(x)}{p(x)} = 1$ wherever $p(x) > 0$. The second inequality follows from the fact that q is a pmf, and is an equality if and only if $q(x) = 0$ wherever $p(x) = 0$. Thus, we get that $D(P||Q) = 0$ if and only if $p(x) = q(x)$ for all x (for continuous random variables, the equalities would only need to hold "almost everywhere.").