Notes on Unsupervised Learning

Jens Behley

May 12, 2021

1 Derivation EM-Algorithm for Gaussian Mixture Models

Let's say our model $P(\mathbf{x})$ decomposes as follows:

$$P(\mathbf{x}) = \sum_k P(h = k, \mathbf{x}) \tag{1}$$

$$= \sum_k P(h = k) P(\mathbf{x}|h = k), \tag{2}$$

where h is a latent variable. For the Gaussian mixture model, we assume

$$P(h = k) = \lambda_k \tag{3}$$

$$P(\mathbf{x}|h = k) = \mathcal{N}(\mathbf{x}|\mu_k, \Sigma_k), \tag{4}$$

where $P(h)$ is a categorical distribution, with $\sum_k \lambda_k = 1$. Therefore, we have our parameters given by $\theta_h = \{\lambda_1, \ldots, \lambda_K\}$ and $\theta_k = \{\mu_k, \Sigma_k\}, k = 1, \ldots, K$.

We want to determine the maximum likelihood parameters for our given data $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ and therefore want to maximize:

$$\theta^* = \arg\max_\theta \prod_{i=1}^N P(\mathbf{x}_i) \tag{5}$$

$$= \arg\max_\theta \prod_{i=1}^N \sum_k P(h = k, \mathbf{x}_i) \tag{6}$$

$$= \arg\max_\theta \prod_{i=1}^N \sum_k P(h = k) P(\mathbf{x}_i|h = k) \tag{7}$$

As always, we can instead maximize the log-likelihood as the logarithm does

not change the location of the maximum, but simplifies the derivation:

$$\theta^* = \arg\max_\theta \prod_{i=1}^{N} \sum_k P(h=k, \mathbf{x}_i) \tag{8}$$

$$= \arg\max_\theta \log \prod_{i=1}^{N} \sum_k P(h=k, \mathbf{x}_i) \tag{9}$$

$$= \arg\max_\theta \sum_{i=1}^{N} \log \sum_k P(h=k, \mathbf{x}_i) \tag{10}$$

However, following our usual receipt for maximizing the log-likelihood, we want to compute the gradient, but due to the sum we cannot easily separate the parameters θ_h and θ_k, $k = 1, \ldots, K$.

For latent variable models, we can use a method called Expectation Maximization. But first we will derive a lower bound that is easier to optimize by introducing distributions $q_i(h)$, where $\sum_k q_i(h_i = k) = 1$:

$$\sum_{i=1}^{N} \log \sum_k \frac{q_i(h=k)}{q_i(h_i=k)} P(\mathbf{x}_i, h=k) \tag{11}$$

$$= \sum_{i=1}^{N} \log \sum_k q_i(h=k) \frac{P(\mathbf{x}_i, h=k)}{q_i(h=k)} \tag{12}$$

$$\geq \underbrace{\sum_{i=1}^{N} \sum_k q_i(h=k) \log \frac{P(\mathbf{x}_i, h=k)}{q_i(h=k)}}_{\mathcal{B}(q_i, \theta)}. \tag{13}$$

We used Jensen' inequality in the last line, which is for a convex function f, where $\sum_i \pi_i = 1, \pi_i \geq 0$:

$$f\left(\sum_i \pi_i x_i\right) \leq \sum_i \pi_i f(x_i) \tag{14}$$

Since log is a concave function, we have $\log\left(\sum_i \pi_i x_i\right) \geq \sum_i \pi_i \log(x_i)$, as the sign switches.

We have a sum of N terms, where we now have to determine the $q_i(h)$, where we can now rewrite $P(\mathbf{x}, h=k)$ as $P(h=k|\mathbf{x})P(\mathbf{x})$, resulting in:

$$\sum_k q_i(h=k) \log \frac{P(\mathbf{x}_i, h=k)}{q_i(h=k)} \tag{15}$$

$$= \sum_k q_i(h=k) \log \frac{P(h=k|\mathbf{x}_i)P(\mathbf{x}_i)}{q_i(h=k)} \tag{16}$$

$$= \underbrace{\sum_k q_i(h=k) \log \frac{P(h=k|\mathbf{x}_i)}{q_i(h=k)}}_{-\mathbb{KL}(q_i || P(h=k|\mathbf{x}_i))} + \sum_k q_i(h=k) \log P(\mathbf{x}_i), \tag{17}$$

where $\mathbb{KL}(Q||P)$ is the Kullback-Leibler divergence (short KL divergence) that measures the difference between the probability distributions Q and P. The KL-divergence is always positive and zero if $P = Q$. Therefore, to minimize the KL divergence and maximize the lower bound $\mathcal{B}(q_i, \theta)$, we set $q_i(h = k) = P(h = k|\mathbf{x}_i)$, where we can compute $P(h|\mathbf{x}_i)$ as follows:

$$P(h = k|\mathbf{x}_i) = \frac{P(h = k, \mathbf{x}_i)}{P(\mathbf{x}_i)} \tag{18}$$

$$= \frac{P(h = k, \mathbf{x}_i)}{\sum_j P(h = j, \mathbf{x}_i)} \tag{19}$$

$$= \frac{P(h = k)P(\mathbf{x}_i|h = k)}{\sum_k P(h = j)P(h = j, \mathbf{x}_i)} \tag{20}$$

In the E-Step, we determine $P(h = k|\mathbf{x}_i)$ for each training example and this will make the lower bound $\mathcal{B}(q_i, \theta)$ tight for the last parameters $\theta^{[t-1]}$.

In the M-Step, we now determine the parameters $\theta^{[t]}$, which maximize the bound, let now say $r_{ik}^{[t-1]} = P(h = k|\mathbf{x}_i)$, which results in the following optimization:

$$\theta^{[t]} = \arg\max_\theta \sum_{i=1}^{N} \sum_k q_i(h = k) \log \frac{P(\mathbf{x}_i, h = k)}{q_i(h = k)} \tag{21}$$

$$= \arg\max_\theta \sum_{i=1}^{N} \sum_k r_{ik}^{[t-1]} \log P(h = k)P(\mathbf{x}_i|h = k) - \underbrace{\log r_{ik}^{[t-1]}}_{\text{const}} \tag{22}$$

$$= \arg\max_\theta \sum_{i=1}^{N} \sum_k r_{ik}^{[t-1]} \log P(h = k)P(\mathbf{x}_i|h = k) \tag{23}$$

$$= \arg\max_\theta \sum_{i=1}^{N} \sum_k r_{ik}^{[t-1]} \left(\log \lambda_k^{[t]} + \log \mathcal{N}(\mathbf{x}|\mu_k^{[t]}, \Sigma_k^{[t]}) \right) \tag{24}$$

We can now find the parameters $\lambda_k^{[t]}, \mu_k^{[t]}, \Sigma_k^{[t]}$ by rearranging the parameters.

For the normal distributions, we already know that the maximum likelihood parameters are given by the mean and variance over the data. But here, we have to account for the responsibilities $r_{ik}^{[t-1]}$:

$$\mu_k^{[t]} = \frac{\sum_{i=1}^N r_{ik}^{[t-1]} \mathbf{x}_i}{\sum_{i=1}^N r_{ik}^{[t-1]}} \tag{25}$$

$$\Sigma_k^{[t]} = \frac{\sum_{i=1}^N r_{ik}^{[t-1]} \left(\mathbf{x}_i - \mu_k^{[t]} \right) \left(\mathbf{x}_i - \mu_k^{[t]} \right)^T}{\sum_{i=1}^N r_{ik}^{[t-1]}}. \tag{26}$$

For the $\lambda_k^{[t]}$, we have to ensure that $\sum_k \lambda_k = 1$, which we achieve by a

lagrange multiplier:

$$\theta_h^{[t]} = \arg\max_{\theta} \underbrace{\sum_{i=1}^{N} \sum_{k} r_{ik}^{[t-1]} \log \lambda_k + \rho \left(\sum_k \lambda_k - 1 \right)}_{\mathcal{L}}, \tag{27}$$

where we dropped that parts from Eq. 24 that are not dependent on $\theta_h^{[t]}$ and added the lagrange multiplier $\rho \left(\sum_k \lambda_k - 1 \right)$. Now the partial derivative with respect to λ_j and the Lagrange multiplier ρ given by:

$$\frac{\partial \mathcal{L}}{\partial \lambda_j} = \sum_{i=1}^{N} r_{ij}^{[t-1]} \frac{1}{\lambda_j} + \rho \tag{28}$$

$$\frac{\partial \mathcal{L}}{\partial \rho} = \sum_k \lambda_k - 1 \tag{29}$$

Setting both to 0, we get:

$$\lambda_j = -\frac{\sum_{i=1}^{N} r_{ij}^{[t-1]}}{\rho} \tag{30}$$

$$\sum_k \lambda_k = 1 \tag{31}$$

We now solve for ρ by inserting Eq. 30 into Eq. 31:

$$\sum_k \lambda_k = 1 \Leftrightarrow \sum_k -\frac{\sum_{i=1}^{N} r_{ik}^{[t-1]}}{\rho} = 1 \Leftrightarrow -\sum_{i=1}^{N} \underbrace{\sum_k r_{ik}^{[t-1]}}_{=1} = \rho \Leftrightarrow -N = \rho \tag{32}$$

Using this, we finally arrive at:

$$\lambda_j^{[t]} = \frac{1}{N} \sum_{i=1}^{N} r_{ij}^{[t-1]} \tag{33}$$

To summarize: Using the EM-Algorithm that optimizes a tight lower bound of the log-likelihood, we arrive at the following procedure to determine the parameter of the Gaussian Mixture Model:

1. Initialize $\mu_k^{[0]}$ by selecting K random examples $\mathbf{x}_i \in \mathcal{X}$, $\Sigma_k^{[0]} = \mathbf{I}$, and $\lambda_k^{[0]} = 1/K$.

2. **E-Step:** Determine $r_{ik}^{[t-1]}$ by evaluating $P(h = k|\mathbf{x}_i)$ with last parameters from timestep $t-1$, i.e.,

$$P(h = k|\mathbf{x}_i) = \frac{P(h = k)P(\mathbf{x}_i|h = k)}{\sum_k P(h = j)P(h = j, \mathbf{x}_i)} \tag{34}$$

$$= \frac{\lambda_k^{[t-1]} \mathcal{N}\left(\mathbf{x}_i \mid \mu_k^{[t-1]}, \Sigma_k^{[t-1]}\right)}{\sum_k \lambda_k^{[t-1]} \mathcal{N}\left(\mathbf{x}_i \mid \mu_k^{[t-1]}, \Sigma_k^{[t-1]}\right)} \tag{35}$$

3. M-Step: Update parameters given the $r_{ik}^{[t-1]}$, i.e.,

$$\lambda_k^{[t]} = \frac{1}{N} \sum_{i=1}^{N} r_{ik}^{[t-1]} \tag{36}$$

$$\mu_k^{[t]} = \frac{\sum_{i=1}^{N} r_{ik}^{[t-1]} \mathbf{x}_i}{\sum_{i=1}^{N} r_{ik}^{[t-1]}} \tag{37}$$

$$\Sigma_k^{[t]} = \frac{\sum_{i=1}^{N} r_{ik}^{[t-1]} \left(\mathbf{x}_i - \mu_k^{[t]}\right) \left(\mathbf{x}_i - \mu_k^{[t]}\right)^T}{\sum_{i=1}^{N} r_{ik}^{[t-1]}}. \tag{38}$$

4. Repeat E-Step and M-Step until convergence.

Note that different initial μ can lead to different results as we can get stuck in a local maximum. Therefore, some care has to be taken to get good initial parameters for the mean. Also notable, we have to set the number of mixtures K in advance.