

EM 及其特例 GMM

Mia Feng

2018 年 3 月 5 日

1 概述

EM 算法: 无监督, density estimation 模型。用于含有 latent variable 的概率模型的极大似然估计 (MLE)。算法对初始值敏感。因为 latent variable 的不可观测性, 所以似然函数最大化难以处理, 代替的, 我们把难于处理的最大化似然函数问题用一个易于最大化的序列 (对数似然函数序列) 取代, 其极限是原始问题的解。原本的关于参数 θ 的最优化问题被离散到一些 $L(\theta)$ 对数似然序列上求解。虽然最终可以收敛到 $L(\theta)$ 的稳定点, 但是不一定保证能收敛到极大值点 (因为不是关于参数估计序列 θ 的收敛), 即最终结果可能陷入局部最优。算法分为 E (expectation) 步和 M (maximization) 步, E 步求 latent variable 的条件概率分布的期望, M 步通过最大化期望更新参数。迭代直至算法收敛 (参数改动小于某个阈值或者期望变化小于某个阈值)

GMM 假设样本从多个高斯分布中生成, 只是 EM 后验分布取混合高斯分布的一种特例。基于中心极限定理以及 Gaussian Distribution 的良好性质, GMM 算法应用较广。

作为无监督学习算法, GMM 与 k -means 相比是类似的。GMM-E 步中对高斯分量的响应度相当于 k -means 中的距离计算, GMM-M 根据响应度计算高斯分量参数相当于 k -means 中计算分类点的位置。然后都是不断迭代达到最优 [2]。

求解目标: 聚类, 每个样本点给出其可能来自潜在变量 i 的概率。

求解思路: 最大化关于 latent variable 的对数似然函数, 取对数似然函数最大时对应的参数 θ 作为未知变量的解。最终根据 θ^* 求得样本来自 latent variable 的评分 (概率)。

求解方法：MLE (Maximum likelihood estimator)。

Tips: 确定了响应度之后，参数更新有显式的函数表达

Preliminary: Jensen's inequality,

1.1 EM 算法

重点在于写出完全数据的对数似然函数的期望（之所以称为完全，是假定补充了 latent variable 之后的数据），即 Q 函数

Q 函数 [1] 完全数据的对数似然函数 $\log P(Y, Z|\theta)$ 关于在给定观测数据 Y 和当前参数 $\theta^{(i)}$ 下对未观测数据 Z 的条件概率分布 $\log P(Y, Z|\theta^{(i)})$ 的期望

$$Q(\theta, \theta^{(i)}) = \mathbb{E}_Z [\log P(Y, Z|\theta)|Y, \theta^{(i)}] \quad (1)$$

Jensen's Inequality 当函数是 convex function 时，期望的函数值小于等于函数值的期望

$$\mathbb{E}[f(X)] \leq f(\mathbb{E}[X]) \quad (2)$$

推导对 Q 函数利用 Jensen's Inequality，参见 page158-159[1]，收敛性证明参见 page161-162[1]。

9.1.2 EM 算法的导出

上面叙述了 EM 算法。为什么 EM 算法能近似实现对观测数据的极大似然估计呢？下面通过近似求解观测数据的对数似然函数的极大化问题来导出 EM 算法，由此可以清楚地看出 EM 算法的作用。

我们面对一个含有隐变量的概率模型，目标是极大化观测数据（不完全数据） Y 关于参数 θ 的对数似然函数，即极大化

$$\begin{aligned} L(\theta) &= \log P(Y|\theta) = \log \sum_z P(Y, Z|\theta) \\ &= \log \left(\sum_z P(Y|Z, \theta)P(Z|\theta) \right) \end{aligned} \quad (9.12)$$

注意到这一极大化的主要困难是式 (9.12) 中有未观测数据并有包含和（或积分）的对数。

159

事实上，EM 算法是通过迭代逐步近似极大化 $L(\theta)$ 的。假设在第 i 次迭代后 θ 的估计值是 $\theta^{(i)}$ 。我们希望新估计值 θ 能使 $L(\theta)$ 增加，即 $L(\theta) > L(\theta^{(i)})$ ，并逐步达到极大值。为此，考虑两者的差：

$$L(\theta) - L(\theta^{(i)}) = \log \left(\sum_z P(Y|Z, \theta)P(Z|\theta) \right) - \log P(Y|\theta^{(i)})$$

利用 Jensen 不等式 (Jensen inequality)^① 得到其下界：

$$\begin{aligned} L(\theta) - L(\theta^{(i)}) &= \log \left(\sum_z P(Y|Z, \theta^{(i)}) \frac{P(Y|Z, \theta)P(Z|\theta)}{P(Y|Z, \theta^{(i)})} \right) - \log P(Y|\theta^{(i)}) \\ &\geq \sum_z P(Z|Y, \theta^{(i)}) \log \frac{P(Y|Z, \theta)P(Z|\theta)}{P(Z|Y, \theta^{(i)})} - \log P(Y|\theta^{(i)}) \\ &= \sum_z P(Z|Y, \theta^{(i)}) \log \frac{P(Y|Z, \theta)P(Z|\theta)}{P(Z|Y, \theta^{(i)})P(Y|\theta^{(i)})} \quad \checkmark \text{ log 的性质} \end{aligned}$$

令

$$B(\theta, \theta^{(i)}) = L(\theta^{(i)}) + \sum_z P(Z|Y, \theta^{(i)}) \log \frac{P(Y|Z, \theta)P(Z|\theta)}{P(Z|Y, \theta^{(i)})P(Y|\theta^{(i)})} \quad (9.13)$$

则

$$L(\theta) \geq B(\theta, \theta^{(i)}) \quad (9.14)$$

即函数 $B(\theta, \theta^{(i)})$ 是 $L(\theta)$ 的一个下界，而且由式 (9.13) 可知，

$$L(\theta^{(i)}) = B(\theta^{(i)}, \theta^{(i)}) \quad (9.15)$$

因此，任何可以使 $B(\theta, \theta^{(i)})$ 增大的 θ ，也可以使 $L(\theta)$ 增大。为了使 $L(\theta)$ 有尽可能大的增长，选择 $\theta^{(i+1)}$ 使 $B(\theta, \theta^{(i)})$ 达到极大，即

$$\theta^{(i+1)} = \arg \max_{\theta} B(\theta, \theta^{(i)}) \quad (9.16)$$

现在求 $\theta^{(i+1)}$ 的表达式。省去对 θ 的极大化而言是常数的项，由式 (9.16)、式 (9.13) 及式 (9.10)，有

$$\begin{aligned} \theta^{(i+1)} &= \arg \max_{\theta} \left(L(\theta^{(i)}) + \sum_z P(Z|Y, \theta^{(i)}) \log \frac{P(Y|Z, \theta)P(Z|\theta)}{P(Z|Y, \theta^{(i)})P(Y|\theta^{(i)})} \right) \\ &= \arg \max_{\theta} \left(\sum_z P(Z|Y, \theta^{(i)}) \log(P(Y|Z, \theta)P(Z|\theta)) \right) \\ &= \arg \max_{\theta} \left(\sum_z P(Z|Y, \theta^{(i)}) \log P(Y, Z|\theta) \right) \\ &= \arg \max_{\theta} Q(\theta, \theta^{(i)}) \end{aligned} \quad (9.17)$$

式 (9.17) 等价于 EM 算法的一次迭代，即求 Q 函数及其极大化。EM 算法是通过

^① 这里用到的是 $\log \sum_j \lambda_j y_j \geq \sum_j \lambda_j \log y_j$ ，其中 $\lambda_j \geq 0$ ， $\sum_j \lambda_j = 1$ 。

图 1: EM 推导

161

定理 9.1 设 $P(Y|\theta)$ 为观测数据的似然函数, $\theta^{(i)} (i=1,2,\dots)$ 为 EM 算法得到的参数估计序列, $P(Y|\theta^{(i)}) (i=1,2,\dots)$ 为对应的似然函数序列, 则 $P(Y|\theta^{(i)})$ 是单调递增的, 即

$$P(Y|\theta^{(i+1)}) \geq P(Y|\theta^{(i)}) \quad (9.18)$$

证明 由于

$$P(Y|\theta) = \frac{P(Y, Z|\theta)}{P(Z|Y, \theta)}$$

取对数有

$$\log P(Y|\theta) = \log P(Y, Z|\theta) - \log P(Z|Y, \theta)$$

由式 (9.11) $Q(\theta, \theta^{(i)}) = E_{\theta^{(i)}} [\log p(Y, Z|\theta)]$

$$Q(\theta, \theta^{(i)}) = \sum_z \log P(Y, Z|\theta) P(z|Y, \theta^{(i)})$$

令

$$H(\theta, \theta^{(i)}) = \sum_z \log P(Z|Y, \theta) P(z|Y, \theta^{(i)}) \quad (9.19)$$

于是对数似然函数可以写成

$$\log P(Y|\theta) = Q(\theta, \theta^{(i)}) - H(\theta, \theta^{(i)}) \quad (9.20)$$

式 (9.20) 中分别取 θ 为 $\theta^{(i)}$ 和 $\theta^{(i+1)}$ 并相减, 有

$$\begin{aligned} & \log P(Y|\theta^{(i+1)}) - \log P(Y|\theta^{(i)}) \\ &= [Q(\theta^{(i+1)}, \theta^{(i)}) - Q(\theta^{(i)}, \theta^{(i)})] - [H(\theta^{(i+1)}, \theta^{(i)}) - H(\theta^{(i)}, \theta^{(i)})] \end{aligned} \quad (9.21)$$

为证式 (9.18), 只需证式 (9.21) 右端是非负的. 式 (9.21) 右端的第一项, 由于 $\theta^{(i+1)}$ 使 $Q(\theta, \theta^{(i)})$ 达到极大, 所以有

$$Q(\theta^{(i+1)}, \theta^{(i)}) - Q(\theta^{(i)}, \theta^{(i)}) \geq 0 \quad (9.22)$$

其第二项, 由式 (9.19) 可得:

$$\begin{aligned} H(\theta^{(i+1)}, \theta^{(i)}) - H(\theta^{(i)}, \theta^{(i)}) &= \sum_z \log p(z|Y, \theta^{(i+1)}) P(z|Y, \theta^{(i)}) \\ &\quad - \sum_z \log p(z|Y, \theta^{(i)}) P(z|Y, \theta^{(i)}) \\ &= \sum_z \left(\log \frac{P(z|Y, \theta^{(i+1)})}{P(z|Y, \theta^{(i)})} \right) P(z|Y, \theta^{(i)}) \\ &\leq \log \left(\sum_z \frac{P(z|Y, \theta^{(i+1)})}{P(z|Y, \theta^{(i)})} P(z|Y, \theta^{(i)}) \right) \\ &= \log \left(\sum_z P(z|Y, \theta^{(i+1)}) \right) = 0 \end{aligned} \quad (9.23)$$

这里的不等号由 Jensen 不等式得到.

由式 (9.22) 和式 (9.23) 即知式 (9.21) 右端是非负的.

定理 9.2 设 $L(\theta) = \log P(Y|\theta)$ 为观测数据的对数似然函数, $\theta^{(i)} (i=1,2,\dots)$ 为 EM 算法得到的参数估计序列, $L(\theta^{(i)}) (i=1,2,\dots)$ 为对应的对数似然函数序列.

- (1) 如果 $P(Y|\theta)$ 有上界, 则 $L(\theta^{(i)}) = \log P(Y|\theta^{(i)})$ 收敛到某一值 L ;
- (2) 在函数 $Q(\theta, \theta')$ 与 $L(\theta)$ 满足一定条件下, 由 EM 算法得到的参数估计序列 $\theta^{(i)}$ 的收敛值 θ^* 是 $L(\theta)$ 的稳定点. 非极大值点.

证明 (1) 由 $L(\theta) = \log P(Y|\theta)$ 的单调性及 $P(Y|\theta)$ 的有界性立即得到.

(2) 证明从略, 参阅文献 [6].

定理 9.2 关于函数 $Q(\theta, \theta')$ 与 $L(\theta)$ 的条件在大多数情况下都是满足的. EM 算法的收敛性包含关于对数似然函数序列 $L(\theta^{(i)})$ 的收敛性和关于参数估计序列 $\theta^{(i)}$ 的收敛性两层意思, 前者并不蕴含后者. 此外, 定理只能保证参数估计序列收敛到对数似然函数序列的稳定点, 不能保证收敛到极大值点. 所以在应用中, 初值的选择变得非常重要, 常用的办法是选取几个不同的初值进行迭代, 然后对得到的各个估计值加以比较, 从中选择最好的.

图 2: EM 收敛性证明

1.2 GMM 算法

EM 的特例，假设样本来自混合高斯分布（多个高斯分布），latent variable 服从 multinomial 分布（这是自然的，是为了标志数据从哪个分布产生，自然只有 0,1 两种可能性，多个高斯成分均匀混合对应 multinomial）。算法其实也可以用别的方法求极值，EM 只是一种求解方法。

高斯混合分布 k 个高斯分布均匀混合得到的概率分布模型。

$$P(y|\theta) = \sum_{k=1}^K \alpha_k \phi(y|\theta_k) \quad (3)$$

其中，系数 $\alpha_k \geq 0$, $\sum_{k=1}^K \alpha_k = 1$ 注意这里 P 是 PDF (probability distribution function) 而非 CDF (cumulative distribution function)。高斯分布的钟形图是 PDF，混合高斯分布的 PDF 是由各高斯成分线性叠加形成的。这也解释了系数 α_k 为什么一定小于 1 且加和为 1，因为 GMM 整体的 PDF 积分才为 1。

李航的书上没有具体求各项的导数，这里用了 CS229-notes 中的推导，并补充了讲义中关于方差的更新公式的推导。

3 Mixture of Gaussians

Armed with our general definition of the EM algorithm, let's go back to our old example of fitting the parameters ϕ, μ and Σ in a mixture of Gaussians. For the sake of brevity, we carry out the derivations for the M-step updates only for ϕ and μ_j , and leave the updates for Σ_j as an exercise for the reader.

The E-step is easy. Following our algorithm derivation above, we simply calculate

$$w_j^{(i)} = Q_i(z^{(i)} = j) = P(z^{(i)} = j | x^{(i)}; \phi, \mu, \Sigma)$$

Here, " $Q_i(z^{(i)} = j)$ " denotes the probability of $z^{(i)}$ taking the value j under the distribution Q_i .

Next, in the M-step, we need to maximize, with respect to our parameters ϕ, μ, Σ , the quantity

$$\begin{aligned} & \sum_{i=1}^m \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \phi, \mu, \Sigma)}{Q_i(z^{(i)})} \\ &= \sum_{i=1}^m \sum_{j=1}^k Q_i(z^{(i)} = j) \log \frac{p(x^{(i)} | z^{(i)} = j; \mu, \Sigma) p(z^{(i)} = j; \phi)}{Q_i(z^{(i)} = j)} \\ &= \sum_{i=1}^m \sum_{j=1}^k w_j^{(i)} \log \frac{\frac{1}{(2\pi)^{n/2} |\Sigma_j|^{1/2}} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j)\right) \cdot \phi_j}{w_j^{(i)}} \end{aligned}$$

图 3: GMM 算法推导-1

D _{μ_i} $\sum_{k=1}^K \sum_{j=1}^m w_j^{(i)} \log \frac{A \exp(E) C}{D}$ $A \propto D, C$ 为常数
 $A \propto \phi_j$

$= \sum_{k=1}^K \sum_{j=1}^m w_j^{(i)} [\nabla_{\mu_i} (\phi_j A + E + \phi_j C - \phi_j D)]$

$= \sum_{k=1}^K [\nabla_{\mu_i} (\phi_j A) + \nabla_{\mu_i} (C)]$

Let's maximize this with respect to μ_i . If we take the derivative with respect to μ_i , we find

$$\begin{aligned} & \nabla_{\mu_i} \sum_{i=1}^m \sum_{j=1}^k w_j^{(i)} \log \frac{1}{(2\pi)^{n/2} |\Sigma_j|^{1/2}} \exp \left(-\frac{1}{2} (x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j) \right) \cdot \phi_j \\ & = -\nabla_{\mu_i} \sum_{i=1}^m \sum_{j=1}^k w_j^{(i)} \frac{1}{2} (x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j) \\ & = \frac{1}{2} \sum_{i=1}^m w_i^{(i)} \left(\nabla_{\mu_i} (2\mu_i^T \Sigma_i^{-1} x^{(i)} - \mu_i^T \Sigma_i^{-1} \mu_i) \right) \\ & = \sum_{i=1}^m w_i^{(i)} (\Sigma_i^{-1} x^{(i)} - \Sigma_i^{-1} \mu_i) \quad \sum_{i=1}^m w_i^{(i)} (x^{(i)} - \mu_i) = 0 \end{aligned}$$

Setting this to zero and solving for μ_i therefore yields the update rule

$$\mu_i := \frac{\sum_{i=1}^m w_i^{(i)} x^{(i)}}{\sum_{i=1}^m w_i^{(i)}}$$

which was what we had in the previous set of notes.

Let's do one more example, and derive the M-step update for the parameters ϕ_j . Grouping together only the terms that depend on ϕ_j , we find that we need to maximize

$$\sum_{i=1}^m \sum_{j=1}^k w_j^{(i)} \log \phi_j.$$

However, there is an additional constraint that the ϕ_j 's sum to 1, since they represent the probabilities $\phi_j = p(z^{(i)} = j; \phi)$. To deal with the constraint that $\sum_{j=1}^k \phi_j = 1$, we construct the Lagrangian

$$\mathcal{L}(\phi) = \sum_{i=1}^m \sum_{j=1}^k w_j^{(i)} \log \phi_j + \beta \left(\sum_{j=1}^k \phi_j - 1 \right),$$

where β is the Lagrange multiplier.² Taking derivatives, we find

$$\frac{\partial}{\partial \phi_j} \mathcal{L}(\phi) = \sum_{i=1}^m \frac{w_i^{(i)}}{\phi_j} + \beta$$

$\sum_{j=1}^k \frac{w_j^{(i)}}{\phi_j} = \frac{1}{\beta}$

$\therefore \phi_j = \frac{\sum_{i=1}^m w_i^{(i)}}{-\beta}$

²We don't need to worry about the constraint that $\phi_j \geq 0$, because as we'll shortly see, the solution we'll find from this derivation will automatically satisfy that anyway.

图 4: GMM 算法推导-2

Setting this to zero and solving, we get

$$\phi_j = \frac{\sum_{i=1}^m w_j^{(i)}}{-\beta}$$

I.e., $\phi_j \propto \sum_{i=1}^m w_j^{(i)}$. Using the constraint that $\sum_j \phi_j = 1$, we easily find $\phi_j := \frac{w_j^{(i)}}{\sum_{j=1}^m w_j^{(i)}}$ that $-\beta = \sum_{i=1}^m \sum_{j=1}^k w_j^{(i)} = \sum_{i=1}^m 1 = m$. (This used the fact that $w_j^{(i)} = Q_i(z^{(i)} = j)$, and since probabilities sum to 1, $\sum_j w_j^{(i)} = 1$.) We therefore have our M-step updates for the parameters ϕ_j :

$$\boxed{\phi_j := \frac{1}{m} \sum_{i=1}^m w_j^{(i)}}.$$

The derivation for the M-step updates to Σ_j are also entirely straightforward.

$$\begin{aligned} \nabla_{\Sigma_j} &= -\frac{\sum_{i=1}^m w_i^{(i)} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_i)^2}{\sum_{i=1}^m w_i^{(i)}} \\ &\quad + \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_j|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x}^{(i)} - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_j)\right) \cdot \phi_j \\ &\text{其中, } \phi_j \text{ 与 } \boldsymbol{\mu}_i, \Sigma_j \text{ 无关, } w_i^{(i)} \text{ 为常数.} \\ &= \nabla_{\Sigma_j} \sum_{i=1}^m \sum_{j=1}^k w_j^{(i)} \left[\log \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_j|^{\frac{1}{2}}} - \frac{1}{2} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_j) \right] \\ &\text{注意是关于 } \Sigma_j \text{ 的导数, 故 } j \neq l \text{ 项可略} \\ &= \sum_{i=1}^m w_i^{(i)} \left[\nabla_{\Sigma_j} \left(\log \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_j|^{\frac{1}{2}}} \right) - \nabla_{\Sigma_j} \left((\mathbf{x}^{(i)} - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_j) \right) \right] \\ &= \sum_{i=1}^m w_i^{(i)} \left[\nabla_{\Sigma_j} - (\mathbf{x}^{(i)} - \boldsymbol{\mu}_j)^2 \right] \\ \therefore \Sigma_j &= \frac{\sum_{i=1}^m w_i^{(i)} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_j)^2}{\sum_{i=1}^m w_i^{(i)}} \end{aligned}$$

图 5: GMM 算法推导-3

2 算法实现

1. Take initial guesses for $\hat{\mu}_1, \hat{\sigma}_1^2, \hat{\mu}_2, \hat{\sigma}_2^2, \hat{\pi}$

2. E-step: compute responsibilities

$$\hat{\gamma}_i = \frac{\hat{\pi}\phi_{\hat{\theta}_2}(y_i)}{(1 - \hat{\pi})\phi_{\hat{\theta}_1}(y_i) + \hat{\pi}\phi_{\hat{\theta}_2}(y_i)} \quad \text{for } i = 1, \dots, N$$

3. M-step: compute weighted means and variances

$$\begin{aligned} \hat{\mu}_1 &= \frac{\sum_{i=1}^N (1 - \hat{\gamma}_i)y_i}{\sum_{i=1}^N (1 - \hat{\gamma}_i)} \\ \hat{\sigma}_1^2 &= \frac{\sum_{i=1}^N (1 - \hat{\gamma}_i)(y_i - \hat{\mu}_1)^2}{\sum_{i=1}^N (1 - \hat{\gamma}_i)} \end{aligned}$$

and similarly for $\hat{\mu}_2, \hat{\sigma}_2^2$ using $\hat{\gamma}_i$ in place of $(1 - \hat{\gamma}_i)$. The mixing probability is given by $\hat{\pi} = \sum_{i=1}^N \hat{\gamma}_i / N$.

4. Iterate steps 2 and 3 until convergence.

图 6: GMM 算法步骤

3 Implementation

聚类测试：数据用四个二元高斯分布混合生成。但是在更新方差的时候总会出现奇异矩阵，这样的话没法更新。还不知道原因是什么。暂时将方差固定，但是结果还不是很好，留待解决

参考文献

- [1] 李航. 统计学习方法. 清华大学出版社, 2012.
- [2] Orange 先生. 高斯混合模型的终极理解. http://blog.csdn.net/xmu_jupiter/article/details/50889023, 2016. Blog, 2016.