EM 算法简介

吴天阳　张卓立

XJTU

强基数学

2022 年 11 月 13 日

目录

前置知识

定义 1 (极大似然估计)

设 $X_1, \cdots, X_n$ 是来自密度函数为 $f(x;\theta)$ 的独立随机样本，称 $\mathcal{L}(\theta) = f(x_1, \cdots, x_n; \theta) = \prod_{i=1}^{n} f(x_i;\theta)$ 为关于 θ 的似然函数，则 θ 的极大似然估计为

$$\hat{\theta} = \arg\max_{\theta \in \Theta} L(\theta)$$

其中 Θ 为参数空间.

前置知识

定义 1 (极大似然估计)

设 $X_1, \cdots, X_n$ 是来自密度函数为 $f(x;\theta)$ 的独立随机样本，称 $\mathcal{L}(\theta) = f(x_1, \cdots, x_n; \theta) = \prod_{i=1}^{n} f(x_i;\theta)$ 为关于 θ 的**似然函数**，则 θ 的极大似然估计为

$$\hat{\theta} = \arg\max_{\theta \in \Theta} L(\theta)$$

其中 Θ 为参数空间.

注 1

(1) θ 可以是一个参数向量 $(\theta_1, \cdots, \theta_n)$，包含多个参数.

(2) 计算时一般采用**对数似然函数** $l(\theta) = \log \mathcal{L}(\theta) = \sum_{i=1}^{n} \log f(x_i;\theta)$.

前置知识

定理 2 (Jensen 不等式)

设 $f(x)$ 是 $\mathbb{R}$ 上的实值凸函数，X 为随机变量，若 $\mathbb{E}X$ 存在，则

$$\mathbb{E}[f(X)] \geqslant f(\mathbb{E}X)$$

上式取到等号，当且仅当，X 为常量.

前置知识

定理 2 (Jensen 不等式)

设 $f(x)$ 是 $\mathbb{R}$ 上的实值凸函数，X 为随机变量，若 $\mathbb{E}X$ 存在，则

$$\mathbb{E}[f(X)] \geqslant f(\mathbb{E}X)$$

上式取到等号，当且仅当，X 为常量.

注 2

(1) 当 $f(x)$ 为凹函数时，上述不等号反向，即 $\mathbb{E}[f(X)] \leqslant f(\mathbb{E}X)$.

问题引入

假设有 A 和 B 两个硬币，每轮选择一个硬币，进行 N 次投掷得到一个样本集，包含 N 个样本 $X_1, \cdots, X_n$，X_i 表示每次的投掷结果（正面或反面），设硬币 A,B 分别服从不同的 Bernoulli 分布，记为 $B(1, p_1)$, $B(1, p_2)$. 则样本中每个样本 $X_1, \cdots, X_n$ 来自这两个分布中的一个，但无法确定具体是哪一个分布，即不知道是投掷硬币 A 还是硬币 B 得到的 x_i.

问题引入

假设有 A 和 B 两个硬币，每轮选择一个硬币，进行 N 次投掷得到一个样本集，包含 N 个样本 $X_1, \cdots, X_n$，X_i 表示每次的投掷结果（正面或反面），设硬币 A,B 分别服从不同的 Bernoulli 分布，记为 $B(1, p_1)$, $B(1, p_2)$. 则样本中每个样本 $X_1, \cdots, X_n$ 来自这两个分布中的一个，但无法确定具体是哪一个分布，即不知道是投掷硬币 A 还是硬币 B 得到的 x_i.

- 可以将该问题分为以下两个：
  1. 这轮投掷是使用硬币 A 还是硬币 B ？
  2. 硬币 A 正面的概率 p_1 和硬币 B 正面的概率 p_2 分别是多少 ？

问题引入

假设有 A 和 B 两个硬币，每轮选择一个硬币，进行 N 次投掷得到一个样本集，包含 N 个样本 $X_1, \cdots, X_n$，X_i 表示每次的投掷结果（正面或反面），设硬币 A,B 分别服从不同的 Bernoulli 分布，记为 $B(1, p_1)$, $B(1, p_2)$. 则样本中每个样本 $X_1, \cdots, X_n$ 来自这两个分布中的一个，但无法确定具体是哪一个分布，即不知道是投掷硬币 A 还是硬币 B 得到的 x_i.

- 可以将该问题分为以下两个：
  1. 这轮投掷是使用硬币 A 还是硬币 B？
  2. 硬币 A 正面的概率 p_1 和硬币 B 正面的概率 p_2 分别是多少？
- 已知：
  1. 模型的分布（均满足二项分布）
  2. 观察到的样本（投掷结果）
- 未知：
  1. 每个个体来自于哪个分布（投掷 A 还是 B）
  2. 模型参数（p_1, p_2）

算法思路

- 通过引入隐变量 $Z = (Z_1, \cdots, Z_n)$ 来将描述未被观测到的隐含数据，表示上文中每个个体来自于哪个分布，是由隐变量 Z 控制的.

算法思路

- 通过引入隐变量 $Z = (Z_1, \cdots, Z_n)$ 来将描述未被观测到的隐含数据，表示上文中每个个体来自于哪个分布，是由隐变量 Z 控制的.
- 举一个例子：我们假设还有第三个硬币 C，也服从 Bernoulli 分布 $B(1, p_3)$，每次试验时，先投掷硬币 C，若硬币 C 为正面，则投掷硬币 A，反之，则投掷硬币 B. 于是，隐变量就是硬币 C 的投掷结果，记为随机变量 Z，且 $Z \sim B(1, p_3)$.

算法思路

- 通过引入隐变量 $Z = (Z_1, \cdots, Z_n)$ 来将描述未被观测到的隐含数据，表示上文中每个个体来自于哪个分布，是由隐变量 Z 控制的.
- 举一个例子：我们假设还有第三个硬币 C，也服从 Bernoulli 分布 $B(1, p_3)$，每次试验时，先投掷硬币 C，若硬币 C 为正面，则投掷硬币 A，反之，则投掷硬币 B. 于是，隐变量就是硬币 C 的投掷结果，记为随机变量 Z，且 $Z \sim B(1, p_3)$.
- 通过这个例子，我们可以假设隐变量 $Z_1, \cdots, Z_n$ 是来自密度函数为 $g(z)$ 的分布.
- 由于隐变量是人为假定的，我们假设**隐变量独立同分布于某一种分布**，从而能进一步对其研究. 在实际应用中，试验数据应该满足这一假设，才能使用 EM 算法.
- 这一假设分布也称为先验分布.

算法大致流程

以上文抛硬币问题为例

1. 初始化参数：投掷硬币 A,B 正面的概率分别为 p_1, p_2.

算法大致流程

以上文抛硬币问题为例

1. 初始化参数：投掷硬币 A,B 正面的概率分别为 p_1, p_2.
2. 计算每个样本是来自于哪个分布：由 A 投掷出来的概率大，还是由 B 投掷出来的概率大.

算法大致流程

以上文抛硬币问题为例

1. 初始化参数：投掷硬币 A,B 正面的概率分别为 p_1, p_2.
2. 计算每个样本是来自于哪个分布：由 A 投掷出来的概率大，还是由 B 投掷出来的概率大.
3. 重新估计参数：通过每个样本属于 A 的概率，从而得到硬币 A 正面的期望次数和反面的期望次数，通过极大似然得到对 p_1 的估计. 同理可以得到 p_2 的估计. 对它们进行更新.

算法大致流程

以上文抛硬币问题为例

1. 初始化参数：投掷硬币 A,B 正面的概率分别为 p_1, p_2.
2. 计算每个样本是来自于哪个分布：由 A 投掷出来的概率大，还是由 B 投掷出来的概率大.
3. 重新估计参数：通过每个样本属于 A 的概率，从而得到硬币 A 正面的期望次数和反面的期望次数，通过极大似然得到对 p_1 的估计. 同理可以得到 p_2 的估计. 对它们进行更新.
4. 若参数变化小于阈值，退出循环；否则返回步骤 2.

算法大致流程

以上文抛硬币问题为例

1. 初始化参数：投掷硬币 A,B 正面的概率分别为 p_1, p_2.
2. 计算每个样本是来自于哪个分布：由 A 投掷出来的概率大，还是由 B 投掷出来的概率大.
3. 重新估计参数：通过每个样本属于 A 的概率，从而得到硬币 A 正面的期望次数和反面的期望次数，通过极大似然得到对 p_1 的估计. 同理可以得到 p_2 的估计. 对它们进行更新.
4. 若参数变化小于阈值，退出循环；否则返回步骤 2.

上述问题算法步骤 2 最难得到，这里就需要通过引入隐变量方法对其计算（利用 Bayes 公式）.

EM 算法推导

我们直接从最大似然估计开始推导

$$\log \mathcal{L}(\theta) = \sum_{i=1}^{n} \log f(x_i; \theta) = \sum_{i=1}^{n} \log \sum_{z_i} f(x_i, z_i; \theta) \tag{2.1}$$

$$= \sum_{i=1}^{n} \log \sum_{z_i} g(z_i) \frac{f(x_i, z_i; \theta)}{g(z_i)}$$

$$\geqslant \sum_{i=1}^{n} \sum_{z_i} g(z_i) \log \frac{f(x_i, z_i; \theta)}{g(z_i)} \tag{2.2}$$

EM 算法推导

我们直接从最大似然估计开始推导

$$\log \mathcal{L}(\theta) = \sum_{i=1}^{n} \log f(x_i;\theta) = \sum_{i=1}^{n} \log \sum_{z_i} f(x_i, z_i;\theta) \tag{2.1}$$

$$= \sum_{i=1}^{n} \log \sum_{z_i} g(z_i) \frac{f(x_i, z_i;\theta)}{g(z_i)}$$

$$\geqslant \sum_{i=1}^{n} \sum_{z_i} g(z_i) \log \frac{f(x_i, z_i;\theta)}{g(z_i)} \tag{2.2}$$

- (2.1) 处引入了 z_i，从而将边缘分布转化为联合分布的形式.

EM 算法推导

我们直接从最大似然估计开始推导

$$\log \mathcal{L}(\theta) = \sum_{i=1}^{n} \log f(x_i;\theta) = \sum_{i=1}^{n} \log \sum_{z_i} f(x_i, z_i;\theta) \tag{2.1}$$

$$= \sum_{i=1}^{n} \log \sum_{z_i} g(z_i) \frac{f(x_i, z_i;\theta)}{g(z_i)}$$

$$\geqslant \sum_{i=1}^{n} \sum_{z_i} g(z_i) \log \frac{f(x_i, z_i;\theta)}{g(z_i)} \tag{2.2}$$

- (2.1) 处引入了 z_i，从而将边缘分布转化为联合分布的形式.
- (2.2) 处，由于 $\log(x)$ 为凹函数，且 $\sum_{z_i} g(z_i) \frac{f(x_i, z_i;\theta)}{g(z_i)} = \mathbb{E}\left[\frac{f(x_i, z_i;\theta)}{g(z_i)}\right]$，由 Jensen 不等式可得.

EM 算法推导

$$\log \mathcal{L}(\theta) \geqslant \sum_{i=1}^{n} \sum_{z_i} g(z_i) \log \frac{f(x_i, z_i; \theta)}{g(z_i)}$$

于是我们得到了对数似然函数的一个下界，这个下界是由期望 $\mathbb{E}\left[\log \frac{f(x_i, z_i; \theta)}{g(z_i)}\right]$ 构成，这里就是 EM 算法中 Exception 部分. 我们希望将最大化问题转化为对右式最大化，也就是 EM 中的 Maximum 部分，先将 θ 固定（使用上一次迭代的 θ_t），考虑当 $g(z_i)$ 满足什么条件时，Jensen 不等式取到等号.

EM 算法推导

$$\log \mathcal{L}(\theta) \geqslant \sum_{i=1}^{n} \sum_{z_i} g(z_i) \log \frac{f(x_i, z_i; \theta)}{g(z_i)}$$

于是我们得到了对数似然函数的一个下界，这个下界是由期望 $\mathbb{E}\left[\log \frac{f(x_i, z_i; \theta)}{g(z_i)}\right]$ 构成，这里就是 EM 算法中 Exception 部分. 我们希望将最大化问题转化为对右式最大化，也就是 EM 中的 Maximum 部分，先将 θ 固定（使用上一次迭代的 θ_t），考虑当 $g(z_i)$ 满足什么条件时，Jensen 不等式取到等号.

$$\frac{f(x_i, z_i; \theta)}{g(z_i)} = c \Rightarrow f(x_i, z_i; \theta) = c g(z_i)$$

$$\text{（对}z_i\text{求和）} \Rightarrow f(x_i; \theta) = \sum_{z_i} f(x_i, z_i; \theta) = c \sum_{z_i} g(z_i) = c.$$

于是 $g(z_i) = \frac{f(x_i, z_i; \theta)}{c} = \frac{f(x_i, z_i; \theta)}{f(x_i; \theta)} = f(z_i | x_i; \theta).$

EM 算法流程

1. 初始化参数 θ_0.

EM 算法流程

1. 初始化参数 θ_0.
2. - E 步：计算条件概率期望 $g_i(z_i) = f(z_i|x_i;\theta_t)$，

     对数似然函数 $\log \mathcal{L}(\theta) = \sum_{i=1}^{n} \sum_{z_i} g_i(z_i) \log \frac{f(x_i, z_i; \theta)}{g_i(z_i)}$
   - M 步：极大化对数似然函数，更新参数 θ：

$$\theta_{t+1} = \arg\max_{\theta \in \Theta} \log \mathcal{L}(\theta) = \sum_{i=1}^{n} \sum_{z_i} g_i(z_i) \log f(x_i, z_i; \theta)$$

EM 算法流程

1. 初始化参数 θ_0.
2.
   - E 步：计算条件概率期望 $g_i(z_i) = f(z_i|x_i;\theta_t)$，

     对数似然函数 $\log \mathcal{L}(\theta) = \sum_{i=1}^{n} \sum_{z_i} g_i(z_i) \log \frac{f(x_i, z_i; \theta)}{g_i(z_i)}$
   - M 步：极大化对数似然函数，更新参数 θ：

$$\theta_{t+1} = \arg\max_{\theta \in \Theta} \log \mathcal{L}(\theta) = \sum_{i=1}^{n} \sum_{z_i} g_i(z_i) \log f(x_i, z_i; \theta)$$

3. 返回第 2 步，直到 θ_{t+1} 收敛.

EM 算法流程

1. 初始化参数 θ_0.
2. - E 步：计算条件概率期望 $g_i(z_i) = f(z_i|x_i;\theta_t)$，

     对数似然函数 $\log \mathcal{L}(\theta) = \sum_{i=1}^{n} \sum_{z_i} g_i(z_i) \log \frac{f(x_i, z_i; \theta)}{g_i(z_i)}$
   - M 步：极大化对数似然函数，更新参数 θ：

$$\theta_{t+1} = \arg\max_{\theta \in \Theta} \log \mathcal{L}(\theta) = \sum_{i=1}^{n} \sum_{z_i} g_i(z_i) \log f(x_i, z_i; \theta)$$

3. 返回第 2 步，直到 θ_{t+1} 收敛.

- M 步中，由于对数似然函数中 $g_i(z_i)$ 与 θ 无关，所以可以去掉.

一个例子

```
prA, prB = 0.3, 0.7  # 初始化参数, 硬币 A,B 正面朝上的概率
samples = [4, 6, 0, 9, 5]  # 每个样本中正面朝上的个数
for _ in range(10):
    expectA, expectB = np.zeros(2), np.zeros(2)  # 硬币 A,B 的
    ↪ 期望
    for i in range(len(samples)):
        tmp1 = np.power(prA, samples[i]) * np.power(1 - prA,
        ↪ 10 - samples[i])
        tmp2 = np.power(prB, samples[i]) * np.power(1 - prB,
        ↪ 10 - samples[i])
        chooseA = tmp1 / (tmp1 + tmp2)  # 选择硬币 A 的概率
        chooseB = 1 - chooseA  # 选择硬币 B 的概率
        expectA += np.array([samples[i] * chooseA, (10 -
        ↪ samples[i]) * chooseA])  # E 步
        expectB += np.array([samples[i] * chooseB, (10 -
        ↪ samples[i]) * chooseB])
    prA = expectA[0] / np.sum(expectA)  # M 步
    prB = expectB[0] / np.sum(expectB)
```

迭代结果

迭代次数	A 正面期望	A 背面期望	B 正面期望	B 背面期望	A 正面概率	B 正面概率
1	6.82	18.19	17.18	7.81	0.27	0.69
2	5.82	17.2	18.18	8.8	0.25	0.67
3	4.99	16.32	19.01	9.68	0.23	0.66
4	4.27	15.53	19.73	10.47	0.22	0.65
5	3.59	14.75	20.41	11.25	0.2	0.64
6	2.92	13.95	21.08	12.05	0.17	0.64
7	2.21	13.04	21.79	12.96	0.14	0.63
8	1.39	11.96	22.61	14.04	0.1	0.62
9	0.52	10.75	23.48	15.25	0.05	0.61
10	0.03	10.04	23.97	15.96	0	0.6