EM 算法的问题和出路

鲁晨光

摘要: 求解混合模型需要最大化预测的分布和样本分布之间的似然度或最小化相对熵。流行的 EM 算法收敛证明中有两个结论：1)只增不减 Q(负的联合交叉熵)可以最大化似然度；2)每个 E-step 不会减少 Q. 然而，有反例证明上面两个结论都错了，并且第二个错误掩盖了第一个错误。EM 算法经常导致收敛困难，原因是模型混合比例没有匹配样本分布. 一个改进的算法——信道匹配算法即 CM 算法——可以提高混合模型迭代收敛的正确性. 文中得到一个重要公式：相对熵的极小值等于 Shannon 互信息 R 减去语义互信息 G(即平均 log(normalized likelihood))的极小值. 重复增大 G(而不是 Q)和减小 R 就可以最小化相对熵。新算法的收敛可以得到严格证明——通过 Shannon 等人分析信息率失真函数用到的变分方法和迭代方法. 使用交叉熵方法和语义信息方法分析 CM 算法和 EM 算法及 MM 算法（Neal 和 Hinton 提出）的区别和联系，可以加深我们对几种算法的理解. CM 算法不仅可用于混合模型，也可以用于半监督学习和多标签学习。

关键词：EM 算法; CM 算法；混合模型; Shannon 信道; 语义信道; 信息论失真函数; Shannon 互信息；预测互信息

1 引言

EM 算法就是 Expectation-Maximization 算法. 其最典型的应用是求解混合模型(一种聚类方法)，而混合模型是无监督学习的典型. EM 算法最早由 Dempster 等人于 1977 年明确提出[1]，后来有很多改进的 EM 算法和改进的收敛证明[2]. 比较著名的收敛证明是 Wu 的证明[3]. 其中比较著名的改进版本是 Neal 和 Hinton 于 1999 年提出的 Maximization-Maximization(即 MM)算法[4].

虽然 EM 算法及改进的 EM 算法有不少成功的例子，但是也有很多人抱怨，EM 算法经常会局部收敛或收敛不对. 笔者研究发现, 流行的 EM 算法收敛证明存在两个严重问题：1) 通过 Q 函数(即后面的负的联合交叉熵 $H'(X,Y|\theta)$)只增不减证明混合模型收敛(即预测的分布和样本分布一致) 是不对的，因为存在反例； 2) 认为 E-step 不会减小 Q 也是不对的；3)第二个错误掩盖了第一个错误.

笔者还发现：1) EM 算法在很多情况下收敛，不是因为 E-step 也增大 Q, 而是因为 E-step 在所有情况下减小预测分布和样本分布之间的相对熵(或 Kullback-Leibler 离散度) [5]； 2) 存在一种改进的算法——信道匹配算法(CM 算法) ——求解混合模型更可靠. 3) EM 算法缺少优化模型比例步骤,

所以收敛很慢, 甚至失败. 如果在 M-step 之前优化模型比例, EM 算法和 CM 算法就等价.

CM 算法可以说是改进的 MM 算法, 但是其基本思想是不同的. CM 算法通过语义信道和 Shannon 信道相互匹配实现混合模型收敛, 也没有用到 Jensen 不等式.

最近 20 年, 交叉熵方法在机器学习领域取得显著成功[6,7]. 作者早在 1991 年就推广 Shannon 互信息公式[8]提出交互交叉熵[9], 随后提出一个语义信息理论[10-12]. 根据这个理论, 一组真值函数构成一个语义信道, 真值函数和似然函数可以相互转换. 信道匹配就是语义信道和 Shannon 信道相互匹配. 笔者用平均对数标准似然度即 log(normalized likelihood)定义语义互信息, 所以语义互信息也就是交互交叉熵. 作者先前的两篇英文文章介绍了 CM 算法在最大似然估计(半监督学习)[13]和混合模型(无监督学习)[14]中的应用. 但是[14]中关于混合模型的收敛证明并不完善. 笔者现在发现, Shannon 及后来者分析信息率失真函数 $R(D)$用到的变分方法[15]和迭代方法[16]可以用来严格证明 CM 算法收敛.

本文主要贡献:

1)从交叉熵的角度分析 EM 算法(包括 MM 算法) , 并通过和它们 CM 算法的联系, 澄清流行的 EM 算法收敛证明存在的问题, 以及算法本身存在的问题;

2)提供 CM 算法使混合模型收敛的严格证明.

一个演示 CM 算法的文件包（含有 Excel 迭代文件和 Word 说明文件），包括迭代过程数据和 Q 函数检验，可以从这里得到：
http://survivor99.com/lcg/CM-iteration.zip .

2 从交叉熵的角度看 EM 算法

2.1 数学定义

定义 1. 1. 1 我们用 U 表示实例空间, 用 X 表示取值于 U 中元素的随机变量, 即 $X\in U=\{x_1, x_2, \ldots, x_m\}$. 为便于理论探讨, 且假设 U 是一维的. 再用 V 表示标签空间, 用 Y 表示取值于 V 中元素的随机变量, 即 $Y\in V=\{y_1, y_2, \ldots, y_n\}$.

定义 1. 1. 2 一个样本由若干样例组成, 即 $D=\{(x(t); y(t)|t=1, 2, \ldots, N; x(t)\in U; y(t)\in V\}$, 其中 t 是样例序号, N 是样例总数. 一个条件样本是 $D_j=\{x(1), x(2), \ldots, x(N_j)|Y=y_j\}$, N_j 是含有 y_j 的样例总数.

定义 1. 1. 3 我们用 θ 表示预测模型(或模型参数) . 相对每个 y_j, 存在一个预测子模型 θ_j. 给定 y_j 时预测的 X 的概率分布是 $P(X|\theta_j)$, 它也就是模型 θ_j 和 D_j 之间的似然函数.

定义 1. 1. 4 设 $P(X)$是样本分布, $P(Y)$是标签概率分布, $P(X|y_j)$是条件样本分布(三者都来自 D 的统计) , 则 θ 和 X 之间, 以及 θ_j 和 X 之间的预测熵(即交叉熵) 分别是:

$$H_\theta(X) = -\sum_i P(x_i)\log P_\theta(x_i), \quad \text{其中} P_\theta(x_i) = \sum_j P(y_j)P(x_i\,|\,\theta_j) \tag{1}$$

$$H(X|\theta_j) = -\sum_i P(x_i|y_j)\log P(x_i|\theta_j) \qquad (2)$$

定理 1.1.1 如果在定义 1.1.2 中假设 D_j 中实例是由 N_j 个独立同分布随机变量产生的(后面都这样假设)；则 θ_j 和 X 之间的平均对数似然度就等于负的交叉熵.

证明：设 D_j 中 x_i 的个数是 N_{ij}, 当 N_j 很大时, $P(x_i|y_j)= N_{ij}/N_j$. 因为独立同分布假设, 于是有对数似然度

$$\begin{aligned}&\log L_X(\theta_j) = \log P(x(1),x(2),...,x(N_j)|\theta_j) = \log\prod_i P(x_i|\theta_j)^{N_{ji}}\\ &=N_j\sum_i P(x_i|y_j)\log P(x_i|\theta_j)= - N_j H(X|\theta_j)\end{aligned} \qquad (3)$$

所以似然度对数的平均是 $L_X(\theta_j)/N_j=-H(X|\theta_j)$. **证毕**.

容易证明, 在 $P(X|\theta_j)= P(X|y_j)$(对所有 j)时, 交叉熵最小, 负的交叉熵或似然度最大.

同理 θ 和 D 之间的似然度 $\log L_X(\theta)= -NH_\theta(X)$. EM 算法中的目标函数 Q 也就是联合似然度, 它和联合交叉熵之间的关系是：

$$\begin{aligned}&\log L_{X,Y}(\theta)\\ &= \log\prod_j [P(y_j)P(x(1),x(2),...,x(N_j)|\theta_j)]\\ &= N\sum_j P(y_j)\sum_i P(x_i|y_j)\log P(x_i,y_j|\theta_j)\\ &=-NH(X,Y|\theta_j)\end{aligned} \qquad (4)$$

2.2 用交叉熵解释 EM 算法用于混合模型

假设 n 个高斯分布函数(真模型)是：

$$P^*(X|y_j) =P(X|\theta_j^*)=K_j\exp[-(X-c_j)^2/(2d_j^2)],\ j=1,2,\ldots,n \qquad (5)$$

其中 K_j 是归一化系数, c_j 是中心, d_j 是标准差. 假设一个样本分布 $P(X)$是两个高斯分布的混合

$$P(X)=P^*(y_1)P^*(X|y_1)+P^*(y_2)P^*(X|y_2) \qquad (6)$$

其中 $P^*(y_1)$, $P^*(y_2)$是真的混合比例. 我们只知道模型是高斯分布且 n=2, 并不知道这些参数. 现在我们猜测 5 个参数 $P(y_1)$, c_1, c_2, d_1, d_2 ($P(y_2)$=1-$P(y_1)$) . 根据猜测得到的分布是

$$P_\theta(X)=P(y_1)P(X|\theta_1)+P(y_2)P(X|\theta_2) \qquad (7)$$

$P_\theta(X)$相对 $P(X)$的相对熵或 Kullback-leibler 距离是

$$H(P||P_\theta) = \sum_i P(x_i)\log\frac{P(x_i)}{P_\theta(x_i)} \qquad (8)$$

如果两者很接近, 相对熵就接近 0, 比如小于 0.001 比特, 那么就算我们猜对了.

EM 算法的基本思想是：目的是改变 $P(Y)$和 θ 求 $L_X(\theta)$达最大(等价于相对熵 $H(P||P_\theta)$达最小). 用交叉熵的语言描述 EM 算法的基本公式如下：

$$
\begin{aligned}
&\log L_X(\theta)=N\sum_i P(x_i)\log P_\theta(x_i)\\
&=N\sum_i P(x_i)\log\sum_j P(x_i\mid\theta_j)P(y_j)\\
&\geq L=N\sum_i\sum_j P(x_i)P(y_i\mid x_i)\log\frac{P(x_i,y_j\mid\theta)}{P(y_i\mid x_i)}\\
&=N\sum_i\sum_j P(x_i)P(y_i\mid x_i)\log P(x_i,y_j\mid\theta)\\
&\quad -N\sum_i\sum_j P(x_i)P(y_i\mid x_i)\log P(y_i\mid x_i)\\
&=Q(\theta\mid\theta^t)-H
\end{aligned}
\tag{9}
$$

其中 θ^t 表示 M-step 优化前的 θ. 不等号是因为 Jensen 不等式. $Q(\theta|\theta^t)$简记为 Q, 它就是联合似然度 $L_{X,Y}(\theta)$—— 可理解为负的联合交叉熵, 即 $H'(X,Y|\theta)=-H(X,Y|\theta)$ (忽略 N, 下同) . 这样, 混合模型问题就变为求 $L=Q-H$ 达最大的问题. Dempster 和 Wu 等人认为求 L 达最大就变成求 Q 达最大——笔者认为这是有问题的.

EM 算法步骤：

E-step： 写出 y_j 的条件概率

$$
\begin{aligned}
&P(y_j\mid X)=P(y_j)P(X\mid\theta_j)/P_\theta(X)\\
&P_\theta(X)=\sum_j P(y_j)P(X\mid\theta_j)
\end{aligned}
\tag{10}
$$

其中 $P(X|\theta_j)$就是流行的方法中的 $P(X|y_j, \theta)$. 本文写法可以使各种交叉熵公式和各种 Shannon 熵公式之间的联系更加清楚.

M-step: 改变 $P(Y)$和 θ, 最大化 $Q(\theta|\theta^t)=NH'(X,Y|\theta)= -NH(X,Y|\theta)$.
如果改变不了, 迭代结束, 否则转到 E-step.

Dempster 和 Wu 等人认为 M-step 可以增大 $Q(\theta|\theta^t)$, E-step 也不减少 Q, 所以反复迭代就能收敛.

MM 把目标函数由 Q 改为 $F=Q+H(Y)$ ($H(Y)$是 Y 的 Shannon 熵), 并且声称在 E-step 也最大化 F, 这样收敛更快.

2.3 EM 算法收敛证明存在的问题

真的模型比例 $P^*(Y)$和真模型产生的条件概率分布 $P^*(X|Y)$(参看式(6))决定了联合概率分布 $P^*(X,Y)$和 $H^*(X,Y)$. 下面说明，Q 可能大于 $Q^*\approx -NH^*(X,Y)$.

真模型和真比例产生的联合 Shannon 熵是：

$$
\begin{aligned}
H^*(X,Y) &= -\sum_j\sum_i P^*(x_i|y_j^*)P^*(y_j)\log[P^*(x_i|y_j)P^*(y_j)]\\
&= H^*(Y)+H^*(X\mid Y)
\end{aligned}
\tag{11}
$$

它有确定值， 可大可小. 我们以 n=2 为例讨论. $P^*(Y)$越均匀, 比如 $P(y_1)=P(y_2)=0.5$, $H^*(Y)$就越大(0.5 bit). d_1 和 d_2 越大, $H^*(X|Y)$就越大. 如果

$H^*(X,Y)$ 很大， 而初始参数使得 $H(X,Y|\theta)$不够大(也就是说联合似然度 $Q=L_{X,Y}(\theta)$比真模型的联合似然度 $L_{X,Y}(\theta^*)=-NH^*(X,Y)$还大)，那么不断增大 Q 方向就反了.

比如 $U=\{1,2,3,\ldots,100\}$, 真模型比例是 $P^*(y_1)=P^*(y_2)=0.5$；真模型参数是 $c_1^*=35$, $c_2^*=65$, $d_1^*=d_2^*=d^*=15$. 假设的比例和参数中只有 $d_1=d_2=d$, 和 d^* 不同，d 分 log 左边的 d 和 log 右边的 d。Q 的变化如表 1 所示。

Table 1 A Counterexample against Popular Convergence Proof of EM Algorithm

表 1 流行的 EM 算法收敛证明存在反例

	left d	right d	Q/N (bits)=$-H(X,Y\|\theta_i)$
True Q^*	15	15	-6.89
Larger Q	10	10	-6.75, counterexample
Larger Q	5	12	-6.59, counterexample
Less Q	5	5	-8.11

可见 $H^*(X,Y)$ 较大, 是 6.89 比特. 如果初始比例和参数是 $P(y_1)=0.5$; $d_1=d_2=d=10$； $c_1=c_1^*$, $c_2=c_2^*$. 则预测的联合交叉熵 $H(X,Y|\theta)$较小, 是 6.75 比特。 如果 left $d=5$, right $d=12$, 则 $H(X, Y|\theta)=6.59$，更小。Q 比 Q^*更大时再增加 Q 方向就反了.

如果 $d_1=d_2=d=5$, 左右一样，而其他不变, 则 $H(X,Y|\theta)=8.11$ 比特, 这说明预测和事实相差太远, 联合交叉熵也会增大. 这时似然度 $Q= -8.11N$ 比特, 比较小. 这时增大 Q 是对的.

我们再看流行的 EM 算法收敛证明的第二个错误. 该证明认为 E-step 只增大 Q 不会减小 Q， 即只会减小 $H(X,Y|\theta)$， 而事实上也存在反例. 后面例 2 说明(参看图 4)，在 $H(X,Y|\theta)<H^*(X,Y)$的情况下， 正是因为 E-step 可以增大 $H(X,Y|\theta)$, 即减小 Q, 才使得迭代可能收敛. 所以, 流行的 EM 算法收敛证明的第二个错误掩盖了第一个错误.

MM 算法看来可以加快迭代收敛, 但是问题是类似的, 它要求 F 在 E-step 只增不减——这也不是不必要的. 下面介绍 CM 算法, 它继承了 EM 算法和 MM 算法的某些优点，但是有重要不同之处.

3 CM 算法及其收敛证明

3.1 信道匹配算法(CM 算法)用于混合模型

CM 算法或信道匹配算法——即语义信道和 Shannon 信道相互匹配的迭代算法. 笔者曾推广 Shannon 信息率失真函数 $R(D)$到 $R(G)$用于数据压缩和语义通信优化[10-12]， G 是语义互信息，R 是 Shannon 互信息，$R(G)$是给定 G 时 R 的最小值. 可以证明，任何一个 $R(G)$函数都是碗状的，参看图 1.

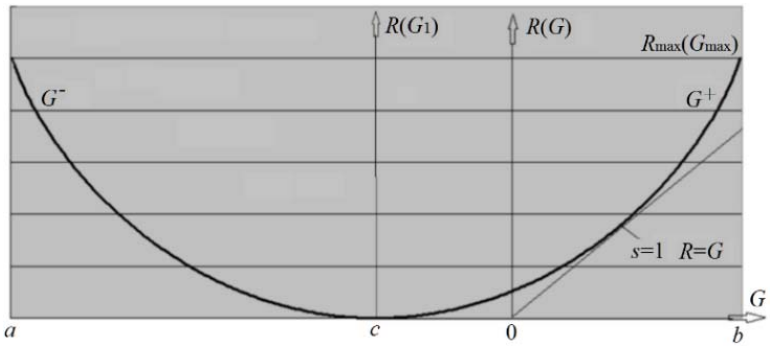

Fig. 1 Any $R(G)$ function is bowl like. It has a point where $R=G$, which implies $P(X|y_j)=P(X|\theta_j)(j=1,2,\ldots)$.

图 1 任一 R(G) 函数都是碗状的，其中存在 $R=G$ 的点，这时 $P(X|y_j)=P(X|\theta_j)(j=1,2,\ldots)$.

要使 $P(X|y_j)=P(X|\theta_j)(j=1,2,\ldots)$, 就要最小化 $R(G)$-G. 语义互信息公式是：

$$G=I(X;\theta)=\sum_j P(y_j)\sum_i P(x_i\mid y_j)\log\frac{P(x_i|\theta_j)}{P(x_i)}$$
$$=\sum_i P(x_i)\sum_j P(y_j\mid x_i)\log\frac{T(\theta_j\mid x_i)}{T(\theta_j)} \tag{11}$$

CM 迭代算法就是轮流优化 log 右边的预测模型(Right-step)和左边的 Shannon 信道(Left-step). G 和其他几种交叉熵之间的关系是：

$$\begin{aligned}G=I(X;\theta)&=H(X)-H(X\mid\theta)\\&=H(X)+H_\theta(Y)-H(Y,X|\theta)\\H(X\mid\theta)&=-\sum_j\sum_i P(x_i,y_j)\log P(x_i|\theta_j)\\H_\theta(Y)&=-\sum_i P^{+1}(y_j)\log P(y_j)\\H(X,Y|\theta)&=-\sum_j\sum_i P(x_i,y_j)\log[P(X\mid\theta_j)P(y_j)]\end{aligned} \tag{12}$$

其中 $H(X)$是 X 的 Shannon 熵. $P^{+1}(y_j)$是调整后的 y_j 的概率. $H(X,Y|\theta)=-Q/N$. 其他几种交叉熵定义如公式(12) . CM 算法迭代分三步：

Left-step a ：构造 Shannon 信道——和 EM 算法中的 E-step 相同, 参看式(10) 或式(13) 中后一个公式.

Left-step b : 调整 $P(Y)$, 即轮流用下面两个公式做局部迭代：

$$P(y_j)=\sum_i P(x_i)P(y_j\mid x_i)=\sum_i P(x_i)P(y_j\mid x_i),\ j=1,2,...$$
$$P(y_j\mid x_i)=P(x_i\mid\theta_j)P(y_j)/\sum_k P(y_k)P(x_i\mid\theta_k), \tag{13}$$
$$i=1,2,...;\ j=1,2,...$$

直至子模型比例 $P(Y)$不变， 记为 $P^{+1}(Y)$. 这样做的原因是, 由(10) 产生的 Shannon 信道 $P(Y|X)$同 $P(X)$和 θ 很可能不相匹配, 即$\sum_i P(x_i)P(Y|x_i)\neq P(Y)$. 重复(13) 可以保证相对熵 $H(P||P_\theta)$减小, 而不是 Q 增大.

如果相对熵 $H(P||P\theta)$小于一个极小值， 比如 0.0001 比特, 则迭代结束. 否则继续.

Right-step: 在下面 log 左边不变情况下, 我们优化 log 后面的模型参数 θ, 最大化语义互信息：

$$G=I(X;\theta)=\sum_i\sum_j P(x_i)\frac{P(x_i\mid\theta_j)}{P_\theta(x_i)}P(y_j)\log\frac{P(x_i\mid\theta_j)}{P(x_i)} \tag{14}$$

转到 Step a.

3.2 用于混合模型的 CM 算法收敛证明

该证明利用 $R(G)$函数的性质：$R(G)$函数是凹的，R-G 存在唯一极小值 0， 这时 $R=G$。这时因为相对熵 $H(P||P_\theta)$接近 R-G.

在 Left-step a(或 EM 算法的 E-step)之后，Shannon 互信息 $I(X;Y)$变成

$$R=\sum_i\sum_j P(x_i)\frac{P(x_i\mid\theta_j)}{P_\theta(x_i)}P(y_j)\log\frac{P(y_j\mid x_i)}{P^{+1}(y_j)}$$

其中

$$P^{+1}(y_j)=\sum_i P(x_i)P(y_j\mid x_i)=\sum_i P(x_i)\frac{P(x_i\mid\theta_j)}{P_\theta(x_i)}P(y_j)$$

我们定义

$$R''=\sum_i\sum_j P(x_i)\frac{P(x_i\mid\theta_j)}{P_\theta(x_i)}P(y_j)\log\frac{P(x_i\mid\theta_j)}{P_\theta(x_i)} \tag{15}$$

容易证明 R''-G=$H(P||P_\theta)$. 于是有

$$R=\sum_i\sum_j P(x_i)\frac{P(x_i\mid\theta_j)}{P_\theta(x_i)}P(y_j)\log\left[\frac{P(x_i\mid\theta_j)P(y_j)}{P_\theta(x_i)P^{+1}(y_j)}\right]$$
$$=R''-H(Y^{+1}||Y) \tag{16}$$
$$H(Y^{+1}||Y)=\sum_j P^{+1}(y_j)\log[P^{+1}(y_j)/P(y_j)]$$
$$H(P\,||\,P_\theta)=R''-G=R-G+H(Y^{+1}\,||\,Y) \tag{17}$$

CM 算法中的三步似乎正好分别改进 R, $H(Y^{+1}||Y)$和 G. 然而，收敛证明难在：当我们最小化 R 或 $H(Y^{+1}||Y)$时， 其他两项也会改变。比如，当我们最小化 $H(Y^{+1}||Y)$时，R-G 似乎有可能增大，使得 R"-G 不会减小。可幸的是，我们可以采用求解 Shannon 信息率失真函数 $R(D)$用到的变分方法和迭代方法(参看[16], p. 316) 证明 R-G 在每一步都是减小的。,

CM 算法收敛证明：证明 $P_\theta(X)$ 收敛到 $P(X)$也就是证明 $H(P||P_\theta)$收敛到 0. 首先我们证明 CM 算法的每一步最小化 $I(X;Y)$-$I(X;\theta)$——它在第一个 Left-step a 变为式(19)中的 R-G. 显然，Right-step 最小化 R-G, 因为它在最大化 G 的时候不改变 R。剩下的问题就是证明 Left-step a 和 Left-step b 如何最小化 R-G.

我们仿照求解 Shannon 信息率失真函数 $R(D)$用到的变分方法和迭代方法(参看[16], p. 316)。现在, 失真量 d_{ij} 变成语义信息量 $I(x_i;\theta_j)=\log[P(x_i|\theta_j)/P(x_i)]$；$R(D)$函数的参数 s 变成 1。为了最小化 $I(X;Y)$-$I(X;\theta)$, 我们用拉格朗日乘子法分别优化 $P(Y|X)$ 和 $P(Y)$。优化 $P(Y|X)$的限制条件是

$$\sum_j P(y_j \mid x_i)=1,\ i=1,2,...,n \tag{18}$$

优化 P(Y)的限制条件是

$$P(y_j)=\sum_i P(x_j)P(y_j \mid x_i), j=1,2 \tag{19}$$

所以拉格朗日函数是

$$F=I(X;Y)-I(X;\theta)-\mu_i\sum_j P(y_j \mid x_i)-\alpha\sum_j P(y_j) \tag{22}$$

为了优化 $P(Y|X)$，我们固定 F 中的 $P(y_j)$并且令 $\partial F/\partial P(y_j \mid x_i)=0$. 于是得到优化的 $P(Y|X)$ (详见附录 I):

$$P^*(y_j \mid x_i)=P(y_j)P(x_i \mid \theta_j)/\sum_k P(y_k)P(x_i \mid \theta_k),\ i=1,2,\ldots,n;\ j=1,2 \tag{23}$$

这正是 EM 算法中 E-step 和 CM 算法中 Left-step a 用到的公式.

为了优化 $P(Y)$，我们固定 F 中的 $P(y_j|x_i)$并且令 $\partial F/\partial P(y_j)=0$. 于是得到优化的 $P(Y)$ (详见附录 I):

$$P^*(y_j)=\sum_i P(x_j)P(y_j \mid x_i),\ j=1,2,\ldots,n \tag{24}$$

这正是 Left-step b 用到的公式。它和式(23)一起在 Left-step b 中不仅最小化 R-G, 也最小化 $H(Y^{+1}||Y)$。

为了优化 $P(Y|X)$，我们固定 F 中的 $P(y_j)$并且令 $\partial F/\partial P(y_j|x_i)=0$. 于是得到优化的 $P(Y|X)$(参看[16], p. 316)：

$$P(y_j \mid x_i)=P(y_j)P(X \mid \theta_j)\mu_i/P(x_i),\ i=1,2,\ldots,n;\ j=1,2$$

因为 $P(Y|x_i)$是归一化的, 即(18) 成立, 所以有

$$P(y_j \mid X) = P(y_j)P(X \mid \theta_j) / \sum_j P(y_j)P(X \mid \theta_j) \qquad (20)$$

这正是 E-step 和 Left-step a 用到的公式. 所以给定 $P(Y)$和参数 θ时, R''是 Shannon 互信息的极小值. Lesft-step b 把 $P(Y)$改为 $P^{+1}(Y)$(利用(13) 迭代) , 这也正是求 $R(D)$函数参数形式用到的局部迭代([16], p.326) , 使得 $P(Y|X)$既满足(20), 也匹配 $P(X)$, 即使得$\Sigma_i P(x_i)P(Y|x_i)=P(Y)$. 所以 Left-step (a 和 b) 减小 R''和 R, 从而减小相对熵 $H(P||P_\theta)$. **证毕**.

3.3 用 CM 算法求解混合模型的两个例子

下面是用 CM 算法求解混合模型的两个例子. 为了检验理论我们直接采用真模型参数产生的概率分布 $P(X)$, 而不是从样本统计得到 $P(X)$. 当样本很大时, 两者结果应该一样. 当样本不大时, 两种结果应该非常接近. 设 n=2, 真实 Shannon 互信息 $R^*=G^*=H(X)-H^*(X|Y)$.

例 1 初始 Shannon 互信息是 R, $R<R^*$, 迭代过程中 R 增大. 有关数据见表 2 . 迭代过程中, 各种信息量和相对熵(即 $R''-G$) 的变化如图 2 所示.

Table 2 Real and guessed model parameters and iterative results for Example 1 ($R<R^*$)

表 2 真实和猜测的参数及迭代结果 ($R<R^*$)

	Real parameters			Start parameters $H(P\|\|P_\theta)$=0.680 bit			Parameters after 5 Right-steps $H(P\|\|P_\theta)$=0.00092 bit		
	c	d	$P^*(Y)$	c	d	$P(Y)$	c	d	$P(Y)$
y_1	35	8	0.7	30	15	0.5	35.4	8.3	0.720
y_2	65	12	0.3	70	15	0.5	65.2	11.4	0.280

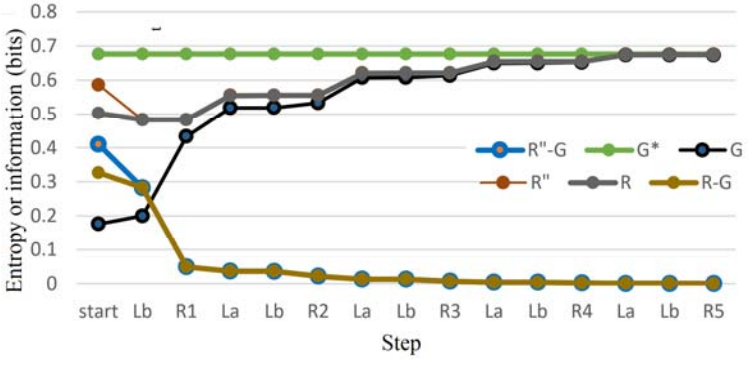

Fig. 2 The iterative process as $R<R^*$. $H(P||P_\theta)$= $R''-G$ decreases in all steps. G is monotonically increasing with R except in the first Left-step b. G and R gradually approach $G^*=R^*$ so that $H(P||P_\theta)$= $R''-G$ is close to 0. Five iterations are needed for convergence.

图 2 $R<R^*$时迭代过程. $H(P||P_\theta)= R''-G$ 在每一步减少. G 单调增大，R 也单调增大(除了在 Left-step b)； G 和 R 逐渐接近 $G^*=R^*$ 使得 $H(P||P_\theta)= R''-G$ 接近 0. 迭代 5 次后收敛.

迭代收敛时，预测分布 $P_\theta(X)$和实际分布 $P(X)$重合情况如图 3 所示.

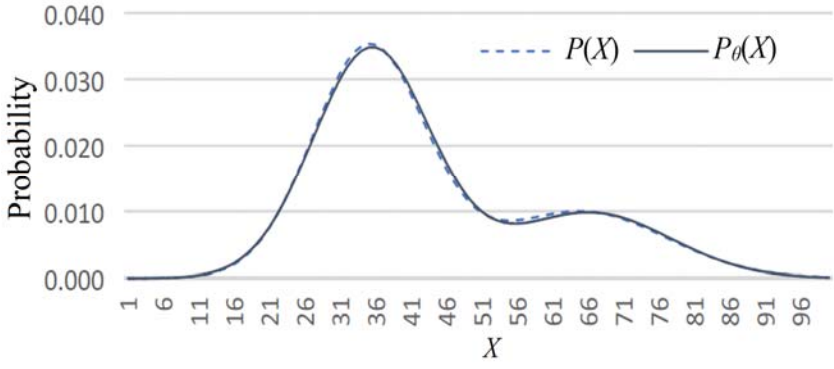

Fig. 3 After 5 iterations, predictive distribution $P_\theta(X)$ is close to sampling distribution $P(X)$

图 3 迭代收敛时, 预测分布 $P_\theta(X)$和样本分布 $P(X)$接近

这个例子符合 EM 算法收敛证明者预期，迭代过程中信息增大，Q 或负熵 $H'(X,Y|\theta)$也增大. 但是下面例子不同.

例 2 初始 Shannon 互信息是 R, $R>R^*$; 迭代过程中 R 是减小的. 这个例子是对 EM 算法的挑战, 因为 Q 是可能下降的. 正是因为在 E-step 中 Q 会下降, R 才可能接近 R^*. 具体数据和结果参看表 3 和图 4.

表 3 **真实和猜测的参数及迭代结果** ($R<R^*$)

Table 3 Real and guessed model parameters and iterative results for Example 2 ($R>R^*$)

	Real parameters			Starting parameters $H(P\|\|P_\theta)$=0.680 bit			Parameters after 5 Right-steps $H(P\|\|P_\theta)$=0.00092 bit		
	c	d	$P^*(Y)$	c	d	$P(Y)$	c	d	$P(Y)$
y_1	35	8	0.1	30	8	0.5	38	9.3	0.134
y_2	65	12	0.9	70	8	0.5	65.8	11.5	0.866

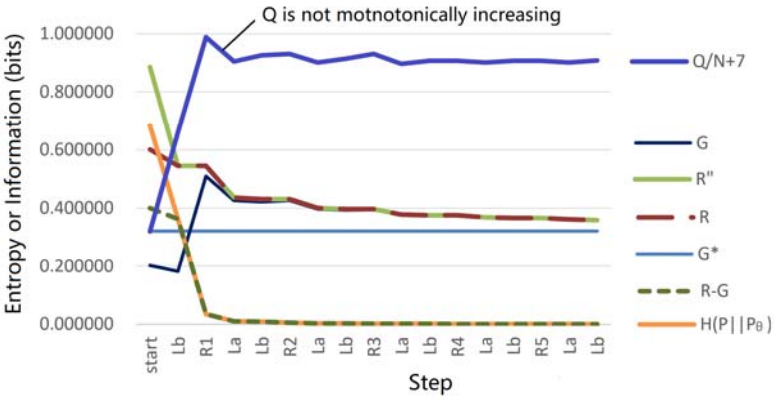

Fig. 4 The iterative process as $R>R^*$. $H(P||P_\theta)=R''-G$ decreases in all steps. R is monotonically decreasing. G increases in all Right-steps and decreases in all Left-steps. G and R gradually approach $G^*=R^*$ so that $H(P||P_\theta)=R''-G$ is close to 0. Five iterations are needed for convergence.

图 4 $R>R^*$时的迭代过程. $H(P||P_\theta)=R''-G$ 在每一步减小. R 单调减小, 而 G 在所有 Right-step 增大, 在所有 Left-step 减小. G 和 R 逐渐接近 $G^*=R^*$, 使得 $H(P||P_\theta)=R''-G$ 接近 0. 迭代 5 次后收敛.

注意, 在这个例子中, Q 在第二个 Left-step a(也就是 E-step)是下降的, 如果不下降, 也无法收敛. 因为 Q 比真模型参数产生的 Q^*还大. EM 算法收敛证明的要害就在这里. 真模型参数是: $P^*(y_1)=0.1$; $c_1^*=35$, $c_2^*=35$; $d_1^*=8$, $d_2^*=12$. Q^*/N=-6.03 bits. 迭代前几步数据如下：

Left a: $P(y_1)=0.5$; log 左右参数都是: $c_1=30$, $c_2=70$; $d_1=d_2=8$. Q/N= -6.68

Left b: $P(y_1)$改为 0.1617. Q/N= -6.34.

Right: log 右边参数改为 $c_1=37.8$, $c_2=66.6$; $d_1=8.5$, $d_2=10.5$. Q/N=-6.01>Q^*/N=-6.03.

Left a: 令左边参数和右边相同; Q/N 变为-6.077. 这一步中 Q 是减小的。

由图 4 可见，Q 接近 Q^*不等于 $H(P||P_\theta)$接近 0.

4 CM 算法和 EM 算法及 MM 之间的联系

EM 算法基本公式可以写成

$$L_X(\theta)>L=Q+H \tag{21}$$

如果写成交叉熵的形式(两边除以 N) 就是

$$-H_\theta(X)\geq -H(X,Y|\theta)+H_\theta(Y|X) \tag{22}$$

关于上面几种熵的定义, 参看(1) 和(12) . 如果上面两边加上 Shannon 熵 $H(X)$, 上式可以变为

$$-H(P||P_\theta)\geq H(X)-H(X,Y|\theta)+H_\theta(Y|X)$$

$=[H(X)+H_\theta(Y)-H(X,Y|\theta)]-[H_\theta(Y)- H_\theta(Y|X)]= G-R''$

即 $H(P||P_\theta) \le R''-G$. 可见 EM 算法和 CM 算法是相通的. 但是 CM 算法没有用到 Jensen 不等式.

如果我们在 EM 算法中的 M-step 先调整 $P(Y)$(调整目的不是最大化 Q, 而是要使 Shannon 信道合理, 如 CM 算法中 Left-step b, 那么两种算法就等价. 两者关系是：

EM 算法的 E-step = CM 算法的 Left-step a

EM 的 M-step ≈ CM 算法的 Left-step b + Right-step

为什么 EM 算法大多情况下收敛？可能的原因是：1)因为没有遇到反例；2)因为调整 $P(Y)$在某些情况下不重要； 3)应用者可能调整了 $P(Y)$.

我们再来看 MM 算法. 如果忽略 N, 目标函数 F 可以写成：

$$F=Q+H(Y)=-H(X,Y|\theta_j)+H(Y)\approx -H_\theta(X|Y) \qquad (23)$$

可见 F 约等于 X 的后验交叉熵. 用≈是因为可能 $P^{+1}(Y)\neq P(Y)$, 用 $H_\theta(Y)$ 才能得到$-H_\theta(X|Y)$. 因为 $G=H(X) -H_\theta(X|Y)$, 最大化 F 就近似于最大化预测互信息 G. 这样优化 F 就可以像 CM 算法的 Right-step 一样, 不改变 $P(Y)$, 而只改变 θ. 这也是为什么 MM 算法能加快收敛. 但是 Neal 和 Hinton 声称 E-step 也最大化 F, 这就和 CM 算法不同了. CM 算法在 Left-step b 只调整 $P(Y)$, 而不最大化 G. 理由是真实 G^*可能并不大, G 需要下降——见上面图 4.

从收敛需要的迭代次数看, 几乎所有的 EM 算法或改进的 EM 算法, 迭代次数大多在 10 次以上[4,17-19]. 而 CM 算法需要的迭代次数大多在 10 次以下[1].

Neal 和 Hinton 用一个例子比较了 EM 算法和 MM 算法。这个例子中一个高斯分布被另一个高斯分布所覆盖. 现在我们用相同的例子看看 CM 算法的迭代次数。

例 3. 真实和开始参数及混合比例如表 4 所示(来自 Neal 和 Hinton [4])，从原始数据 X'转换到表中使用的 X 的转换公式是 $X=20(X'-50)$. 用 CM 算法, 只需要 9 个 left-steps 和 8 个 right-steps, 相对熵达到 $H(P||P_\theta)=0.00072$ bit.

Table 4 例 3 中真实和猜测的模型参数和迭代结果

	Real parameters			Starting parameters $H(P\|\|P_\theta)$=0.680 bit			Parameters after 9 Left-steps $H(P\|\|P_\theta)$=0.00072 bit		
	μ^*	σ^*	$P^*(Y)$	μ	σ	$P(Y)$	μ	σ	$P(Y)$
y_1	46	2	0.7	30	20	0.5	46.001	2.032	0.6990
y_2	50	20	0.3	70	20	0.5	50.08	19.17	0.3010

[1] 检验 CM 算法的 EXCEL 文件下载： http://survivor99.com/lcg/CM-iteration.zip

表 5 显示了三种算法收敛所需要的迭代数。其中 CM 算法的收敛参数大多数都更接近真实参数时，CM 算法的迭代数是 MM 算法的一半，是 EM 算法的 1/4.

Table 5 The iteration numbers and final parameters and ratios for different algorithms

Algorithm	Sample size	Iteration number	Convergent parameters				
			μ_1	μ_2	σ_1	σ_2	$P(y_1)$
EM	1000	about 36	46.14	49.68	1.90	19.18	0.731
MM	1000	about 18	46.14	49.68	1.90	19.18	0.731
CM	∞	9	46.001	50.08	2.03	19.17	0.699
Real parameters			46	50	2	20	0.7

CM 算法的迭代过程如图 5 所示。

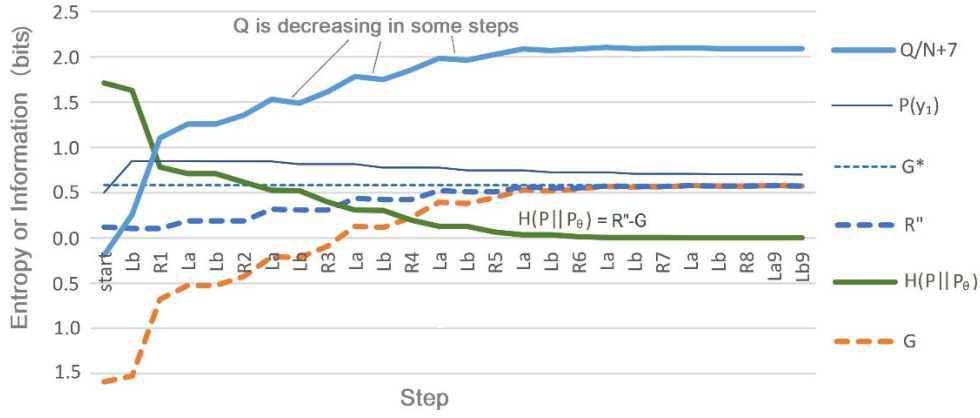

Fig. 4 The iterative process of the CM algorithm for Example 3 used by Neal and Hinton [4].

例 3 也是流行的 EM 算法收敛证明的反例，因为存在 $H(P||P_\theta)$变小或似然度增大时 Q 减小的情况。在例 2 中, Q 在第二个 Left-step a 或 E-step 减小；而在例 3 种，Q 在好几个 Left-step-b 减小， 即使把 Left-step b 并入 M-step， 也不能保证 Q 增大和似然度增大一致。

因为 Left-step b 只调整 $P(Y)$, 其计算实践应该好 MM 算法的 E-step 差不多。所以，CM 算法需要的时间大概是 MM 算法所需要的时间的一半。.

5 结论和讨论

流行的 EM 算法中的 Q 函数可以理解为负的联合交叉熵，流行的 EM 算法收敛证明忽视了初始的 Q 可能大于真实的 Q^*，误以为：1)Q 只增不减可以使模型收敛；2)E-step 只增不减 Q. 第二个错误正好掩盖了第一个错误，所以一直没有被发现. EM 算法本身忽略了模型比例 $P(Y)$的调整, 所以经常收敛困难. 本文利用语义信息方法得到一种改进的迭代算法——信道匹配算法, 即 CM 算法. 重要的是, 我们得到公式：相对熵=$R(G)$-G. 据此可以严格证明 CM 算法收敛——用 Shannon 和后来者求解信息率失真函数用到的变分方法和迭代方法.

实际迭代例子表明 CM 算法在大多数情况下收敛很快, 包括子模型分布重叠的情况(如上面两个例子). 马近文, 徐雷和 Jordna[20]证明了 EM 算法在子模型分布不太重合的情况下有较好的渐进收敛率, 而在重合较多的情况下不然. 这个结论是可信的, 因为重合少时真模型的后验联合熵 $H^*(X,Y)$就小, 相应的联合似然度 Q^*就大, 不断增大 Q 就可以收敛.

用变分方法求解混合模型, 前人已经用到, 但是都没有用到语义互信息或预测互信息. $R(G)$函数作为 $R(D)$函数的一个特殊版本, 其重要性应该会引起更多学者关注. 混合模型问题就是最大预测信息要求下的数据压缩问题，这个问题有很广泛的意义， 在图像压缩和概率预测领域都将用到. 信道匹配算法也可以用于不可见实例分类(半监督学习)[21]和多标签分类[22]. 它在别处表现稍有不同(在 Left-step 可能最大化语义信息或似然度). 可以预期, 信道匹配算法将在各种类型的学习中有较好表现.

附录 I：为最小化 R-G 优化 $P(Y|X)$和 $P(Y)$

为了优化 $P(Y|X)$, 令

$$\frac{\partial F}{\partial P(y_j \mid x_i)} = \frac{\partial}{\partial P(y_j \mid x_i)}\{\sum_j \sum_i P(x_i)P(y_j \mid x_i)\log\frac{P(y_j \mid x_i)}{P(y_j)} - \sum_j \sum_i P(x_i)P(y_j \mid x_i)\log\frac{P(x_i \mid \theta_j)}{P(x_i)} - \mu_i \sum_j P(y_j \mid x_i)\} = 0 \tag{1}$$

于是

$$\begin{aligned} &P(x_i)[1+\log P(y_j \mid x_i)] - P(x_i)\log[P(x_i \mid \theta_j)/P(x_i)] - \mu_i = 0 \\ &\log[P(y_j \mid x_i)/P(y_j)] = \log[P(x_i \mid \theta_j)/P(x_i)] + (\mu_i\text{-}1)/P(x_i) \end{aligned} \tag{2}$$

令 $\log(1/\lambda_i)=(\mu_i-1)/P(x_i)$, 于是有

$$P(y_j \mid x_i) = P(y_j)P(x_i \mid \theta_j)/\lambda_i,\ i=1,2,\ldots,n;\ j=1,2 \tag{3}$$

因为 F 对 $P(y_j|x_i)$的二阶偏导数大于 0，所以上面 上式使 $I(X;Y)$-$I(X;\theta)$达极小. 又因为 $P(Y|x_i)$是归一化的，所以优化的 $P(Y|X)$是

$$P^*(y_j \mid x_i) = P(y_j)P(x_i \mid \theta_j) / \sum_k P(y_k)P(x_i \mid \theta_k), \; i=1,2,\ldots,n; \; j=1,2 \quad (4)$$

为了优化 $P(Y)$, 令

$$\frac{\partial F}{\partial P(y_j)} = \frac{\partial}{\partial P(y_j)}[\sum_j \sum_i P(x_i)P(y_j \mid x_i)\log\frac{P(y_j \mid x_i)}{P(y_j)} + \alpha\sum_j P(y_j)] = 0 \quad (5)$$

于是

$$-\sum_i P(x_i)P(y_j \mid x_i) / P(y_j) + \alpha = 0 \quad (6)$$

$$P(y_j) = \frac{1}{\alpha}\sum_i P(x_j)P(y_j \mid x_i) \quad (7)$$

因为 F 对 $P(y_j)$的二阶偏导数大于 0，所以上式使 $I(X;Y)$-$I(X;\theta)$达极小。又因为$\sum_j P^*(y_j)=1$，所以 $\alpha=1$. 因此, 优化的 $P(Y)$是

$$P^*(y_j) = \sum_i P(x_i)P(y_j \mid x_i), j=1, 2 \quad (8)$$

QED.

References

[1] Dempster A P, Laird N M, Rubin D B. Maximum Likelihood from Incomplete Data via the EM Algorithm[J]. Journal of the Royal Statistical Society, Series B, 1977, 39(1): 1–38

[2] Anonymity, Expectation–maximization algorithm[OL], Wikipedia, [2018-1-26] https://en.wikipedia.org/wiki/Expectation–maximization_algorithm.

[3] Wu C F J. On the convergence properties of the EM algorithm[J]. Annals of Statistics, 1983, 11(1): 95–10

[4] Neal R, Hinton G. A view of the EM algorithm that justifies incremental, sparse, and other variants[C]// Learning in Graphical Models. Michael I. Jordan (ed.). Cambridge: MIT Press, 1999: 355–368

[5] Kullback S, Leibler, R. On information and sufficiency[J]. Annals of Mathematical Statistics, 1951, 22(1): 79–86

[6] Goodfellow I, Bengio Y, Courville A. Deep Learning[M]. Cambridge: MIT Press, 2016. http://www.deeplearningbook.org/

[7] Kevin M. Machine Learning: A Probabilistic Perspective[M]. Cambridge: MIT Press, 2012

[8] Shannon C E. A mathematical theory of communication[J]. Bell System Technical Journal, 1948, 27(3-4): 379–429 and 623–656

[9] Lu C. B-fuzzy set quai-Boolean algebra and generalized mutual entropy formula[J]. Fuzzy Systems and Mathematics, 1991, 5(1): 76-80 (in Chinese)
(鲁晨光, B-模糊准布尔代数和广义交互熵公式. 模糊系统和数学,1991,5(1): 76-80

[10] Lu C. A Generalized Information Theory, Hefei: China Science and Technology University Press, 1993 (in Chinese)

(鲁晨光,广义信息论,合肥：中国科技大学出版社，1993)
[11] Lu C. Coding Meanings of Generalized Entropy and Generalized Mutual Information[J], J. of China Institute of Communication, 1994, 15(6): 37-44 (in Chinese)
（鲁晨光, 广义熵和广义互信息的编码意义，通信学报，1994，15(6)：37-44)
[12] Lu C. A generalization of Shannon's information theory[J]. Int. J. of General Systems, 1999, 28(6): 453-490
[13] Lu C. Semantic Channel and Shannon Channel Mutually Match and Iterate for Tests and Estimations with Maximum Mutual Information and Maximum Likelihood[C]//2018 IEEE International Conference on Big Data and Smart Computing, Piscataway: IEEE Conference Publishing Services, 2018: 227-234
[14] Lu C. Channels' matching algorithm for mixture models[C]//IFIP International Federation for Information Processing, Shi et al. (eds.). Switzerland: Springer International Publishing, 2017: 321–332
[15] Shannon, C. E.: Coding theorems for a discrete source with a fidelity criterion[J]. IRE Nat. Conv. Rec., Part 4, 1959: 142–163.
[16] Zhou J P. Foundation of Information Theory, Beijing: Posts and Telecom Press, 1983
（周炯槃，信息理论基础，北京：人民邮电出版社, 1983)
[17] Kok M, Dahlin J B, Schon T B, Wills A. Newton-based maximum likelihood estimation in nonlinear state space models. IFAC-PapersOnLine 2015, 48(28): 398–403
[18] Springer T, Urban K. Comparison of the EM algorithm and alternatives[J], Numerical Algorithms, 2014, 67(2): 335–364
[19] Huang W H, Chen Y G. The multiset EM algorithm[J]. Statistics & Probability Letters, 2017, 126(C): 41–48
[20] Lu C. Semantic Channel and Shannon Channel Mutually Match and Iterate for Tests and Estimations with Maximum Mutual Information and Maximum Likelihood[C]//2018 IEEE International Conference on Big Data and Smart Computing, Piscataway: IEEE Conference Publishing Services, 2018: 227-234
[21] Ma J, Xu L, Jordan M. Asymptotic convergence rate of the EM algorithm for gaussian mixtures[J], Neural Computation, 2000, 12(12): 2881-907
[22] Yu J, Chaomu C, Yang M S. On convergence and parameter selection of the EM and DA-EM algorithms for Gaussian mixtures[J]. Pattern Recognition, 2018，77(5): 188-203
[22] Lu C. Semantic Channel and Shannon Channel Mutually Match for Multi-label Classification[OL]. [2018-3-16] http://survivor99.com/lcg/CM/recent.html

Problems with the EM Algorithm and the Way Out

Abstract To solve a mixture model, we need to maximize likelihood or minimize relative entropy between predictive distribution and sample distribution. The popular convergence proof of the EM (Expectation-Maximization) algorithm uses two conclusions: 1) We may achieve the maximum likelihood by increasing Q (a negative cross entropy) continuously；2) Q is non-decreasing in every E-step. However, some counterexamples prove that the both conclusions are wrong and that the second error covers up the first error. The EM algorithm often meets with convergent difficulties because component mixture ratios do not match the sampling distribution. An improved algorithm——Channel Matching (CM) algorithm——may raise the convergent validity of mixture models. It is proved that the minimum of the relative entropy is equal to the minimum of Shannon's mutual information R minus semantic mutual information (or average log(normalized likelihood)) G. Increasing G and

decreasing R repeatedly can minimize the relative entropy. The convergence of the CM algorithm can be strictly proved by the variational method and iterative method that Shannon and others use to analyze rate distortion function. Using predictive information and cross entropy as analytical tools, we can understand distinctions and relationships between CM, EM, and MM algorithms better. The CM algorithm can be used not only for mixture models (non-supervised learning), but also for semi-supervised learning and multi-label learning.

Key words: EM algorithm; CM algorithm； mixture models; Shannon channel； semantic channel; rate distortion function； Shannon's mutual information； predictive mutual information