## §3.7 GMM与EM

### 一、GMM

略

### 二、EM

通过迭代进行最大似然估计的优化算法，常作为牛顿迭代法的替代，对包含隐变量/缺失数据情况进行参数估计

核心思想：据已有数据来归估计似然函数

设 $X$ 是观察到的样本数据集合，$Y$ 为丢失的数据集合或模型中任何无法直接观测的随机变量，则完整的样本集合为$D=X\cup Y$。

似然函数为 $p(D|\theta)=p(X,Y|\theta)$

由于 $Y$ 未知，在给定参数 $\theta$ 时，似然函数可以看作 $Y$ 的函数：

$$l(\theta)=l(\theta|D)=l(\theta|X,Y)=\ln p(X,Y|\theta)$$

或$\log p(X,Y|\theta)$

由于$Y$未知，因此需要寻找在 $Y$ 的所有可能情况下平均意义上的似然函数最大值，即似然函数对 $Y$ 的期望的最大值：

$$Q(\theta,\theta^{i-1})=E_Y\left(l(\theta|X,Y)\middle|X,\theta^{i-1}\right)=E_Y\left(\ln p(X,Y|\theta)\middle|X,\theta^{i-1}\right)$$

则 $$\theta^i=\arg\max_\theta Q(\theta,\theta^{i-1})$$

1. ***begin initialize*** $\theta^0$，$T$，$i\leftarrow 0$
2. ***do*** $i\leftarrow i+1$
3. **E step**：*compute* $Q(\theta,\theta^{i-1})$
4. M **step**：$\theta^i=\arg\max_\theta Q(\theta,\theta^{i-1})$
5. ***until*** $Q(\theta^{i+1},\theta^i)-Q(\theta^i,\theta^{i-1})\le T$
6. ***return***
7. ***end*** $\hat{\theta}=\theta^{i+1}$

$$Q(\theta,\theta^{i-1})=E_Y\left(l(\theta|X,Y)\middle|X,\theta^{i-1}\right)=E_Y\left(\ln p(X,Y|\theta)\middle|X,\theta^{i-1}\right)$$
$$\theta^i=\arg\max_\theta Q(\theta,\theta^{i-1})$$

**EM—Expectation**

- 观测数据 $X$ 已知，参数 $\theta$ 的当前值 $\theta^{i-1}$ 已知，在完整似然函数中，缺失数据(隐含变量) $Y$ 未知，完整对数似然函数对 $Y$ 求期望。
- 设 $Q(\theta,\theta^{i-1})=E_Y\left(l(\theta|X,Y)\middle|X,\theta^{i-1}\right)=E_Y[\ln p(X,Y|\theta)|X,\theta^{i-1}]=\int_{y\in\gamma}\ln p(X,Y|\theta)p(y|X,\theta^{i-1})dy$
- 通过求期望，去掉了完整似然函数中的变量 $Y$。

**EM—Maximization**

- 对 E 步计算得到的完整似然函数的期望求极大值，得到参数新的估计值，即
$$\theta^i=\arg\max_\theta Q(\theta,\theta^{i-1})$$
- 每次参数更新会增加非完整似然值
- 反复迭代后，会收敛到似然的局部最大值

## §3.8 HMM与Viterbi方法

问题　与时间相关的问题，即过程随着时间而进行，$t$ 时刻发生的事件要受之前时刻发生事件的直接影响，如何识别？

例如：语音识别或手势识别等问题。

隐马尔可夫模型（Hidden Marcov Models, HMMs）

- 数学中具有马尔可夫性质的离散时间随机过程，是用于描述随机过程统计特征的概率模型。
- 具有一组已经设置好的参数，它们可以很好地解释特定类别中的样本。在使用时，一个测试样本被归类为能产生最大后验概率的模型对应的那个类别。

考虑：对于连续时间内的一系列状态，设ω(t)表示t时刻的状态，那么一个长度为$T$的特定状态序列可设为

$$\omega^T=\{\omega(1),\omega(2),\dots,\omega(T)\}$$

- 系统可以在不同的步骤中重新访问一个状态。

例如：$\omega^6=\{\omega_1,\omega_4,\omega_2,\omega_2,\omega_1,\omega_4\}$.

转移概率：即系统在某一时刻 $t$ 处于状态 $\omega_i$的条件下，在时间$t+1$时变为状态$\omega_j$的概率，该概率与具体的时刻无关，记为

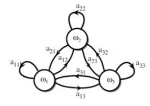
$$P\left(\omega_j(t+1)|\omega_i(t)\right)=a_{ij}$$

$a_{ij}$：$i\to j$

$b_{jk}$：$j$表现$k$

- 没有要求转移概率是对称的（通常$a_{ij}=a_{ji}$）
- 一个特定的状态可能会被连续访问

### 隐马尔可夫模型




基本的马尔可夫模型

- 节点代表离散状态$\omega_i$
- 连线代表转移概率$a_{ij}$

假设 已知一个特定的马尔可夫模型 $\theta$ (转移概率 $a_{ij}$ 的完整集合) 和一个特定状态序列 $\omega^T$。

若要该马尔可夫模型生成特定序列的概率，只需将连续的状态转移概率相乘即可。

例如，求某个特定马尔可夫模型生成序列 $\omega^6=\{\omega_1,\omega_4,\omega_2,\omega_2,\omega_1,\omega_4\}$ 的概率：

$P\left(\omega_j(t+1)|\omega_i(t)\right)=a_{ij}$

$$P(\omega^T|\theta)=a_{14}a_{42}a_{22}a_{21}a_{14}$$

- 如果初始状态$P(\omega(1)=\omega_i)$有一个先验概率，我们也可以包括这个因子。

一阶离散时间的马尔可夫模型：$t+1$时刻的概率只取决于$t$时刻的状态。

在生成语音的马尔可夫模型中，状态代表音素。然而，在语音识别中，接收器只能测量声音的特性，无法直接测量音素。考虑扩充马尔可夫模型：

可见状态：可直接进行外部测量的状态，记为$v(t)$。
隐状态：不能被直接测量得到的内部状态，记为$\omega(t)$。

假设在某一时刻 $t$，系统处于隐状态 $\omega(t)$，同时，系统激发特定可见符号 $v(t)$。

- 马尔可夫模型允许激发的可见状态为连续函数（比如功率谱），本课程只考虑离散符号的情形。

考虑：与状态一样，我们定义了一个特定的可见状态序列，记为 $V^T=\{v(1),v(2),\dots,v(T)\}$。
例如：$V^6=\{v_5,v_1,v_1,v_5,v_2,v_3\}$.

发射概率：在某 $t$ 时刻的状态$\omega(t)$下，可见状态$v_k(t)$激发的概率，该概率同样与具体时刻无关，记为：

$$p\left(v_k(t)|\omega_j(t)\right)=b_{jk}.$$

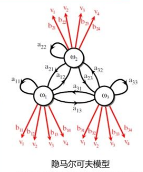
### 隐马尔可夫模型

因为我们只能观测到可见的状态，而不能直接知道$\omega_j$处于什么内部状态，故模型就被称为"隐马尔可夫模型"。

转移概率：$P\left(\omega_j(t+1)|\omega_i(t)\right)=a_{ij}$　（$\sum_j a_{ij}=1$ 对于所有的$i$）

发射概率：$p\left(v_k(t)|\omega_j(t)\right)=b_{jk}$　（$\sum_k b_{jk}=1$ 对于所有的 $j$）

- 每一时刻都必须转移到下一时刻，同时，发出一个可见符号，故有归一化条件。
- 马尔可夫网络是严格因果的，下一时刻状态的概率只取决于上一时刻的状态。
- 最终状态或吸收状态 $\omega_0$ 指系统一旦进入这个状态，就再也无法离开（即：$a_{00}=1$）。
- 该模型展示了任何方式的状态转移都是可能的，然而在一般的隐马尔可夫模型中，这种的任意的状态转移并不能得到保证。




隐马尔可夫模型

黑色连线表示隐状态之间的转移概率，红色的字符表示在每一隐状态产生的可见状态。

估值问题：假设有一个HMM，转移概率 $a_{ij}$ 和 $b_{jk}$ 已知，计算该模型生成的特定的可见状态序列 $V^T$ 的概率。

解码问题：假设有一个HMM及一组观察值 $V^T$，决定最有可能产生这些观察结果的隐状态序列 $\omega^T$。

学习问题：假设已知模型的大致结构（比如隐状态数和可见状态数），但没有给出转移概率 $a_{ij}$和 $b_{jk}$，如何从给定的一组训练样本中确定这些参数。

### 1、估值

#### 估值问题

隐马尔可夫模型产生的可见状态序列 $V^T$ 的概率为：

$$P(V^T)=\sum_{r=1}^{r_{max}}P(V^T|\omega_r^T)P(\omega_r^T)$$

其中，$r$ 是每个特定长度为 $T$ 的隐状态序列的下标：

$$\omega_r^T=\{\omega(1),\omega(2),\dots,\omega(T)\}$$

- 在有$c$个不同隐状态的情况下，$P(V^T)$共有$r_{max}=c^T$项
- 为了计算模型产生特定的可见状态序列$V^T$的概率，应该考虑每一种可能的隐状态序列，计算它们生成 $V^T$ 的概率，然后将这些概率相加。
- 特定可见序列的概率就是对应（隐）转移概率$a_{ij}$和（可见）发射概率$b_{jk}$的乘积结果。

#### 估值问题

$$P(V^T)=\sum_{r=1}^{r_{max}}P(V^T|\omega_r^T)P(\omega_r^T)$$

① 描述隐状态转移概率的第二项 $P(\omega_r^T)$ 可以改写为：

$$P(\omega_r^T)=\prod_{t=1}^{T}P(\omega(t)\mid\omega(t-1))$$

- $P(\omega_r^T)$ 实际上就是$a_{ij}$的乘积。
- $\omega(T)=\omega_0$表示最终的吸收态，它产生唯一独特的可见符号$v_0$。比如在语音识别应用中，$\omega_0$通常表示零状态或者无话语，而 $v_0$ 则表示静音。

② 可设每个时刻发出可见符号的概率仅取决于这个时刻的隐状态，因此，可将第一项写为：

$$P(\mathbf{V}^T\mid\boldsymbol{\omega}_r^T)=\prod_{t=1}^{T}P(v(t)\mid\omega(t))$$

#### 估值问题

$$P(V^T)=\sum_{r=1}^{r_{max}}P(V^T|\omega_r^T)P(\omega_r^T)\ \Rightarrow\ P(\mathbf{V}^T)=\sum_{r=1}^{r_{max}}\prod_{t=1}^{T}P(v(t)\mid\omega(t))P(\omega(t)\mid\omega(t-1))$$

- 我们观察到某一特定可见状态序列$V^T$的概率等于所有可能产生这个可见状态序列的隐状态序列的情况的相加，而每一种可能的隐状态序列的情况发生的概率都是隐状态之间转移概率和产生可见符号发射概率依次相乘得到。
- 估值问题的计算复杂度是$O(c^TT)$，这么大的计算量在实际上是非常不现实的。
  比如，当$c$=10, $T$=20时，我们需要进行$10^{21}$次基本运算！

#### 估值问题的前向算法

$$P(\mathbf{V}^T)=\sum_{r=1}^{r_{max}}\prod_{t=1}^{T}P(v(t)\mid\omega(t))P(\omega(t)\mid\omega(t-1))$$

思路：我们可以递归计算$P(V^T)$，因为每一项 $P(v(t)\mid\omega(t))P(\omega(t)\mid\omega(t-1))$ 均只涉及$v(t)$，$\omega(t)$，$\omega(t-1)$三项。

定义：
$$\alpha_i(t)=\begin{cases}0 & t=0\text{ 且 }j\neq\text{初始状态}\\ 1 & t=0\text{ 且 }j=\text{初始状态}\\ \sum_i\alpha_i(t-1)a_{ij}b_{jk}v(t) & \text{其他}\end{cases}$$

- $b_{jk}v(t)$ 表示由 $t$ 时刻的可见状态 $v(t)$ 确定的发射概率$b_{jk}$。
- $\alpha_j(t)$表示HMM在$t$时刻位于隐状态$\omega_j$且已产生了可见状态序列 $V^T$前$t$个符号的概率。

#### 估值问题的前向算法

$$P(\mathbf{V}^T)=\sum_{r=1}^{r_{max}}\prod_{t=1}^{T}P(v(t)\mid\omega(t))P(\omega(t)\mid\omega(t-1))$$

**HMM Forward Algorithm**

```
1 initialize  ω(1), t = 0, a_ij, b_jk, visible sequence V^T, α(0) = 1
2   for t ← t + 1
3       α_j(t) ← Σ_{i=1}^{c} α_i(t − 1) a_ij b_jk
4   until t = T
5 return P(V^T) ← α_0(T)
6 end
```

$$\alpha_i(t)=\begin{cases}0 & t=0\text{ 且 }j\neq\text{初始状态}\\ 1 & t=0\text{ 且 }j=\text{初始状态}\\ \sum_i\alpha_i(t-1)a_{ij}b_{jk}v(t) & \text{其他}\end{cases}$$

- 在第5行中，$\alpha_0$表示序列的结束。
- 前向算法的计算复杂度为$O(c^2T)$，这比穷举法的效率要高得多。如果同样当$c=10,T=20$的情况，前向算法只需要执行2000次操作，这几乎比穷举法快$10^{17}$倍！

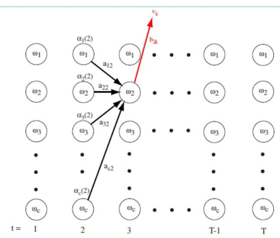
#### 估值问题的前向算法

求$\alpha_2(3)$（$t=3$ 时，系统位于状态$\omega_2$并生成规定可见状态序列的概率）

在$t=2$时且位于状态$\omega_i$的概率为$\alpha_i(2)$，其中$i=1,2,\dots,c$。为了求$\alpha_2(3)$，必须把这些项相加，同时乘以发出字符$v_k$的概率，

即：$$\alpha_2(3)=b_{2k}\sum_i\alpha_i(2)a_{i2}$$

$$\alpha_i(t)=\begin{cases}0 & t=0\text{ 且 }j\neq\text{初始状态}\\ 1 & t=0\text{ 且 }j=\text{初始状态}\\ \sum_i\alpha_i(t-1)a_{ij}b_{jk}v(t) & \text{其他}\end{cases}$$




前向算法网格说明:每一项是按时间对HMM的"展开"。

#### 估值问题

如果把隐马尔可夫模型中的转移概率和发射概率（$a$和$b$）采用参数向量$\theta$表示，那么根据贝叶斯公式，在已知观测序列的情况下，模型的概率为：

$$P(\theta\mid V^T)=\frac{P(V^T\mid\theta)P(\theta)}{P(V^T)}$$

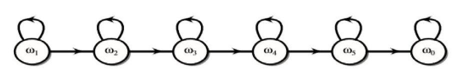
在隐马尔可夫模式识别中，我们可能会有多个HMM，每个模型代表一个类别。对测试样本进行分类，就是计算哪一个模型产生这个测试样本的概率最大。

- 前向算法使我们能够计算$P(V^T|\theta)$
- 模型的先验概率P(θ)由外部的知识确定（比如在语音识别中，可能是一个语言模型），这个先验概率可能依赖于上下文语义，或者是前面的单词等。

#### 估值问题

在语音识别领域，通常使用一个从左向右的隐马尔可夫模型。实际上，几乎所有的隐马尔可夫模型都是从左向右递推的模型。

例如，在隐马尔可夫语音识别中，我们有两个模型，其中一个用来产生发音"stand"，另一个用来产生发音"plate"。现在我们有一个用来测试的未知发音，需要确定哪个模型产生该发音的可能性更大。




这样一个模型可以描述发音"stand"，其中，$\omega_1$代表音素/s/，$\omega_2$代表音素/t/…，直到$\omega_0$代表最终状态。

### 2、解码

#### 维特比（Viterbi）方法

"最有可能"（概率最大）的隐状态序列：$$\omega^{T^*}=\underset{\omega^T}{\text{argmax}}\,P(\omega^T|V^T,\theta)$$

设有Viterbi变量：$$\delta_i(t)=\max_{\omega^t}P(\omega^t=S_i,V^t\mid\theta)$$

思想：利用动态规划求解，复杂性$O(c^2T)$。

递归关系：$$\delta_j(t+1)=\max_i\delta_i(t)a_{ij}b_{jk}\,v(t+1)=\max_i[\delta_i(t)a_{ij}]b_{jk}\,v(t+1)$$
$$\varphi_j(t+1)=\arg\max_i[\delta_i(t)a_{ij}]$$

记忆变量：$\varphi_i(t)$ 记录概率最大路径上当前状态的前一个状态。

目标：找到T时刻最大的$\delta_i(\mathrm{T})$代表的那个隐状态序列。

**HMM Viterbi Algorithm**

1. ***begin initialize*** $\delta_j(1)=\beta_jb_{jk}$，$\varphi_j(1)=0$
2. ***do*** $j\leftarrow j+1$，$t\leftarrow t+1$
3. ***compute*** $\delta_j(t)$，$\varphi_j(t)$
4. ***until*** $j=c$，$t=T$
5. ***Return*** $\omega^{T^*}\leftarrow\arg\max_j[\delta_j(T)]$
6. ***end***

$$\delta_j(t+1)=\max_i\delta_i(t)a_{ij}b_{jk}\,v(t+1)=\max_i[\delta_i(t)a_{ij}]b_{jk}\,v(t+1)$$
$$\varphi_j(t+1)=\arg\max_i[\delta_i(t)a_{ij}]$$

#### 维特比（Viterbi）方法

已知$t=0$时刻，系统的初始隐状态为$\omega_1$，$\delta_j(t)$ 在每个单元内表示。

$$a=\begin{bmatrix}0.5&0.4&0.1\\0&0.6&0.4\\0&0&0\end{bmatrix}$$

$$b=\begin{bmatrix}0.6&0.2&0.2\\0.2&0.5&0.3\\0&0.3&0.7\end{bmatrix}$$

$$\delta_j(t+1)=\max_i\delta_i(t)a_{ij}b_{jk}\,v(t+1)=\max_i[\delta_i(t)a_{ij}]b_{jk}\,v(t+1)$$