理解隐马尔可夫模型

隐马尔可夫模型（Hidden Markov Model，简称 HMM）由 Baum 等人在 1966 年提出[1]，是一种概率图模型，用于解决序列预测问题，可以对序列数据中的上下文信息建模。所谓概率图模型，指用图为相互依赖的一组随机变量进行建模，图的顶点为随机变量，边为变量之间的概率关系。

在隐马尔可夫模型中，有两种类型的节点，分别为观测值序列与状态值序列，后者是不可见的，它们的值需要通过从观测值序列进行推断而得到。很多现实应用可以抽象为此类问题，如语音识别，自然语言处理中的分词、词性标注，计算机视觉中的动作识别。隐马尔可夫模型在这些问题中得到了成功的应用。本文作为已经出版的《机器学习与应用》，清华大学出版社，雷明著第 16 章“循环神经网络”中隐马尔可夫模型一节的扩充，已经被独立成一章，在第二版中出版。为降低阅读与理解难度，本文尽量不过多涉及概率图模型的概念，而是从序列建模的角度对 HMM 进行解释。

马尔可夫过程与马尔可夫模型

马尔可夫过程是随机过程的典型代表。所谓随机过程，是指一个系统的状态随着时间线随机的演化。这种模型可以计算出系统每一时刻处于各种状态的概率以及这些状态之间的转移概率。首先定义状态的概念，在t时刻系统的状态为z_t，在这里是一个离散型随机变量，取值来自一个有限集

$$S = \{s_1, ..., s_n\}$$

例如我们要为天气进行建模，需观察每一天的天气，则状态集为

$$\{\text{晴天，阴天，雨天}\}$$

为简化表示，将状态用整数编号，可以写成

$$\{1,2,3\}$$

从 1 时刻开始到T 时刻为止，系统所有时刻的状态值构成一个随机变量序列

$$\mathbf{z}=\{z_1,...,z_T\}$$

系统在不同时刻可以处于同一种状态，但在任一时刻只能有一种状态。不同时刻的状态之间是有关系的。例如，如果今天是阴天，明天下雨的可能性会更大，在时刻t 的状态由它之前时刻的状态决定，可以表示为如下的条件概率

$$p\left(z_t|z_{t-1},...,z_1\right)$$

即在从 1 到$t-1$时刻系统的状态值分别为$z_1,...,z_{t-1}$的前提下，时刻t 系统的状态为z_t 的概率。如果要考虑之前所有时刻的状态计算太复杂。为此进行简化，假设t 时刻的状态只与$t-1$时刻的状态有关，与更早的时刻无关，即忘记了更早的信息。上面的概率可以简化为

$$p\left(z_t|z_{t-1},...,z_1\right)=p\left(z_t|z_{t-1}\right)$$

该假设称为一阶马尔可夫假设，满足这一假设的马尔可夫模型称为一阶马尔可夫模型。如果状态有n 种取值，在t 时刻取任何一个值与$t-1$时刻取任何一个值的条件概率构成一个$n\times n$的矩阵$\mathbf{A}$，称为状态转移概率矩阵，其元素为

$$a_{ij}=p\left(z_t=j|z_{t-1}=i\right)$$

该值表示$t-1$时刻的状态为i，t 时刻的状态为 j，即从状态i 转移到状态 j 的概率。如果知道了状态转移矩阵，就可以计算出任意时刻系统状态取每个值的概率。状态转移概率矩阵的元素必须满足如下约束：

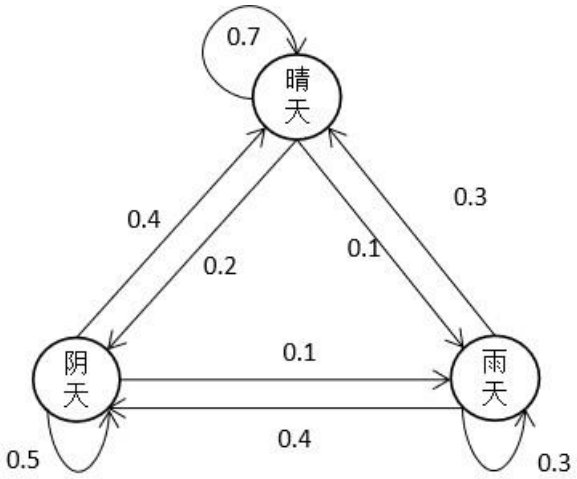
$$a_{ij}\geq 0$$

$$\sum_{j=1}^{n}a_{ij}=1$$

第一条是因为概率值必须在$[0,1]$之间，第二条是因为无论t 时刻的状态值是什么，在下一个时刻一定会转向n 个状态中的一个，因此它们的转移概率和必须为 1。以天气为例，假设状态转移矩阵为

$$\begin{bmatrix}0.7 & 0.2 & 0.1\\ 0.4 & 0.5 & 0.1\\ 0.3 & 0.4 & 0.3\end{bmatrix}$$

其对应的状态转移图（状态机）如下图所示，图中每个顶点表示状态，边表示状态转移概率，是有向图



有一个需要考虑的问题是系统初始时刻处于何种状态，这同样是随机的，可以用向量$\boldsymbol{\pi}$表示。以天气为例，假设初始时处于晴天的概率是 0.5，处于阴天的概率是 0.4，处于雨天的概率是 0.1，则$\boldsymbol{\pi}$为

$$\begin{bmatrix}0.5 & 0.4 & 0.1\end{bmatrix}$$

为简化表述，引入一个特殊的状态s_0消掉$\boldsymbol{\pi}$，该状态的编号为 0。它是系统初始时所处的状态，即$z_0 = s_0$，在接下来的时刻从它转向其他状态，但在后续任何时刻都不会再进入此状态。加入初始状态之后，对状态转移矩阵也进行扩充，行和列的下标变为从 0 开始。以天气问题为例，扩充后的状态转移矩阵为

$$\begin{bmatrix}0 & 0.5 & 0.4 & 0.1\\ 0 & 0.7 & 0.2 & 0.1\\ 0 & 0.4 & 0.5 & 0.1\\ 0 & 0.3 & 0.4 & 0.3\end{bmatrix}$$

给定一阶马尔可夫过程的参数，由该模型产生一个状态序列$z_1,...,z_T$的概率为

$$\begin{aligned}p(z_1,...,z_T) &= p(z_t|z_1,...,z_{t-1})\,p(z_{t-1}|z_1,...,z_{t-2})...\\ &= p(z_t|z_{t-1})\,p(z_{t-1}|z_{t-2})...\\ &= \prod_{t=1}^{T} a_{z_t z_{t-1}}\end{aligned}$$

结果就是状态转移矩阵的元素乘积。在这里假设任何一个时刻的状态转移矩阵都是相同的，即状态转移矩阵与时刻无关。

对于上面的天气问题，连续 3 天全部为晴天的概率为

$$
\begin{aligned}
&p(z_1=1,z_2=1,z_3=1)\\
&=p(z_1=1|z_0)\,p(z_2=1|z_1=1)\,p(z_3=1|z_2=1)\\
&=a_{01}\times a_{11}\times a_{11}\\
&=0.5\times 0.7\times 0.7\\
&=0.245
\end{aligned}
$$

状态转移矩阵通过训练样本学习得到，采用最大似然估计。给定一个状态序列$\mathbf{z}$，马尔可夫过程的对数似然函数为

$$
\begin{aligned}
L(\mathbf{A})&=\ln p(\mathbf{z};\mathbf{A})\\
&=\ln\prod_{t=1}^{T}a_{z_{t-1}z_t}\\
&=\sum_{t=1}^{T}\ln a_{z_{t-1}z_t}\\
&=\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}\ln a_{ij}
\end{aligned}
$$

这里使用了指示变量来方便表述。因为状态转移矩阵要满足上面的两条约束，因此要求解的是如下带约束的最优化问题

$$
\begin{aligned}
&\max_{\mathrm{A}} L(\mathbf{A})\\
&\sum_{j=1}^{n}a_{ij}=1,i=1,...,n\\
&a_{ij}\geq 0,i,j=1,...,n
\end{aligned}
$$

由于对数函数的定义域要求自变量大于0，因此可以去掉不等式约束，上面的最优化问题变成带等式约束的优化问题，可以用拉格朗日乘数法求解。构造拉格朗日乘子函数

$$
L(\mathbf{A},\boldsymbol{\alpha})=\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}\ln a_{ij}+\sum_{i=1}^{n}\alpha_i\left(1-\sum_{j=1}^{n}a_{ij}\right)
$$

对a_{ij}求偏导数并令导数为0，可以到得

$$
\frac{\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}}{a_{ij}}=\alpha_i
$$

解得

$$
a_{ij}=\frac{1}{\alpha_i}\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}
$$

对α_i求偏导数并令导数为0，可以得到

$$
1-\sum_{j=1}^{n}a_{ij}=0
$$

将 a_{ij} 代入上式可以得到

$$1-\sum_{j=1}^{n}\left(\frac{1}{\alpha_i}\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}\right)=0$$

解得

$$\alpha_i=\sum_{j=1}^{n}\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}=\sum_{t=1}^{T}1\{z_{t-1}=i\}$$

合并后得到下面的结果

$$a_{ij}=\frac{\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}}{\sum_{t=1}^{T}1\{z_{t-1}=i\}}$$

这一结果也符合我们的直观认识：从状态 i 转移到状态 j 的概率估计值就是在训练样本中，从状态 i 转移到状态 j 的次数除以从状态 i 转移到下一个状态的总次数。对于多个状态序列，方法与单个状态序列相同。

隐马尔可夫模型

在实际应用中，有些时候我们不能直接观察到状态的值，即状态的值是隐含的，只能得到观测的值。为此对马尔可夫模型进行扩充，得到隐马尔可夫模型。

隐马尔可夫模型描述了观测变量和状态变量之间的概率关系。与马尔可夫模型相比，隐马尔可夫模型不仅对状态建模，而且对观测值建模。不同时刻的状态值之间，同一时刻的状态值和观测值之间，都存在概率关系。

首先定义观测序列

$$\mathbf{x}=\{x_1,...,x_T\}$$

这是直接能观察或者计算得到的值。任一时刻的观测值来自有限的观测集

$$V=\{v_1,...,v_m\}$$

接下来定义状态序列

$$\mathbf{z}=\{z_1,...,z_T\}$$

任一时刻的状态值也来自有限的状态集

$$S=\{s_1,...,s_n\}$$

这与马尔可夫模型中的状态定义相同。在这里，状态是因，观测是果，即因为处于某种状态所以才有某一观测值。

例如，如果我们要识别视频中的动作，状态就是要识别的动作，有站立、坐下、行走等取值，在进行识别之前无法得到其值。观测是能直接得到的值如人体各个关节点的坐标，隐马尔可夫模型的作用是通过观测值推断出状态值，即识别出动作。

除之前已定义的状态转移矩阵之外，再定义观测矩阵$\mathbf{B}$，其元素为

$$b_{ij} = p\left(v_j | s_i\right)$$

该值表示t时刻状态值为s_i时观测值为v_j的概率。显然该矩阵也要满足和状态转移矩阵同样的约束条件：

$$b_{ij} \geq 0$$

$$\sum_{j=1}^{n} b_{ij} = 1$$

另外还要给出初始时状态取每种值的概率$\boldsymbol{\pi}$。隐马尔可夫模型可以表示为一个五元组

$$\left\{S, V, \boldsymbol{\pi}, \mathbf{A}, \mathbf{B}\right\}$$

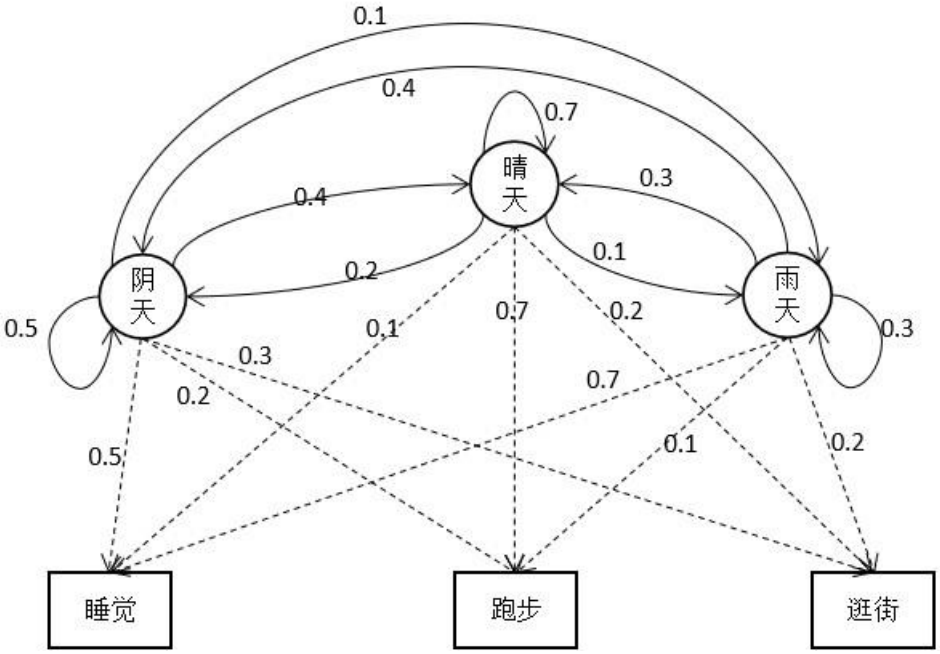
如果加上初始状态则可以消掉参数$\boldsymbol{\pi}$，只剩下$\mathbf{A}$和$\mathbf{B}$。在实际应用中，一般假设矩阵$\mathbf{A}$和$\mathbf{B}$在任何时刻都是相同的即与时间无关，这样简化了问题的计算。

任意一个状态序列可以看做是这样产生的：系统在 1 时刻处于状态z_1，在该状态下得到第观测值x_1。接下来从z_1转移到z_2，并在此状态下得到观测值x_2。以此类推，得到整个观测序列。由于每一时刻的观测值只依赖于本时刻的状态值，因此在状态序列$\mathbf{z}$下出现观测序列$\mathbf{x}$的概率为

$$\begin{aligned} p\left(\mathbf{z}, \mathbf{x}\right) &= p\left(\mathbf{z}\right) p\left(\mathbf{x}|\mathbf{z}\right) \\ &= p\left(z_t | z_{t-1}\right) p\left(z_{t-1} | z_{t-2}\right) \ldots p\left(z_1 | z_0\right) p\left(x_t | z_t\right) p\left(x_{t-1} | z_{t-1}\right) \ldots p\left(x_1 | z_1\right) \\ &= \left(\prod_{t=1}^{T} a_{z_t z_{t-1}}\right) \prod_{t=1}^{T} b_{z_t x_t} \end{aligned}$$

这就是所有时刻的状态转移概率，观测概率的乘积。

以天气问题为例，假设我们不知道每天的天气，但能观察到一个人在各种天气下的活动，根据这一现象来推断天气。这里的活动有 3 种情况，睡觉，跑步，逛街。对于这个问题，天气是状态值，活动是观测值。该隐马尔可夫模型如下图所示



这一问题的观测矩阵为

$$\begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.1 & 0.7 & 0.2 \\ 0.7 & 0.1 & 0.2 \end{bmatrix}$$

在隐马尔可夫模型中，隐藏状态和观测值的数量是根据实际问题人工设定的；状态转移矩阵和混淆矩阵通过样本学习得到。隐马尔可夫模型需要解决以下三个问题：

1.估值问题，给定隐马尔可夫模型的参数 $\mathbf{A}$ 和 $\mathbf{B}$，计算一个观测序列 $\mathbf{x}$ 出现的概率值 $p(\mathbf{x})$。

2.解码问题，给定隐马尔可夫模型的参数 $\mathbf{A}$ 和 $\mathbf{B}$ 以及一个观测序列 $\mathbf{x}$，计算最有可能产生此观测序列的状态序列 $\mathbf{z}$。

3.学习问题，给定隐马尔可夫模型的结构，但参数未知，给定一组训练样本，确定隐马尔可夫模型的参数 $\mathbf{A}$ 和 $\mathbf{B}$。

按照定义，隐马尔可夫模型对条件概率 $p(\mathbf{x}|\mathbf{z})$ 建模，因此是一种生成模型。

中文分词问题

下面以中文分词问题为例，介绍隐马尔可夫模型如何用于实际问题，这是典型的序列标注问题。中文分词即断句，是自然语言处理中的核心、基础问题。因为中文和英文不同，各个词之间没有空格隔开。对于下面的句子

我是中国人

正确的分词结果为

我 是 中国人

在这里观测序列是输入的语句，每个字为每个时刻的观测值。状态序列为分词的结果，每个时刻的状态值有如下几种情况

$$\{\mathrm{B},\mathrm{M},\mathrm{E},\mathrm{S}\}$$

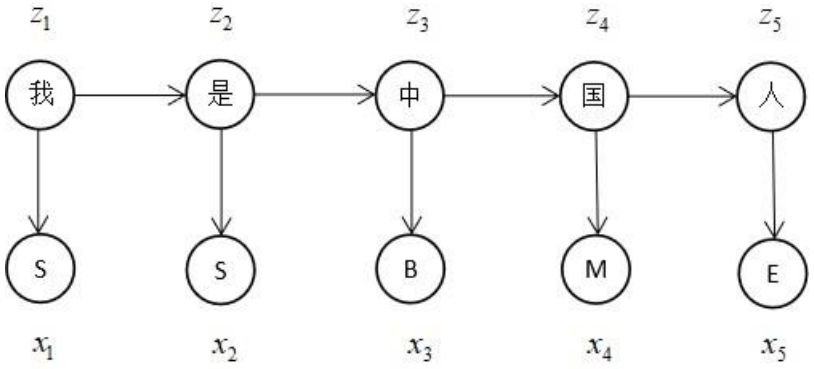
其中 B 表示当前字为一个词的开始，M 表示当前字是一个词的中间位置，E 表示当前字是一个词的结尾，S 表示单字词。则上面这个句子的分词标注结果为

我/S 是/S 中/B 国/M 人/E

显然，得到了这个标注结果，我们就可以得到分词结果，做法很简单：

遇到 S，则为一个单字词；遇到 B，则为一个词的开始，直到遇到下一个 E，则为一个词的结尾。

分词问题为给定观测序列，计算出概率最大的状态序列，对应的就是分词的结果。这通过解码算法实现。隐马尔可夫模型的参数则通过用语料库训练得到。下图是分词的隐马尔可夫模型按时间线展开后的结果



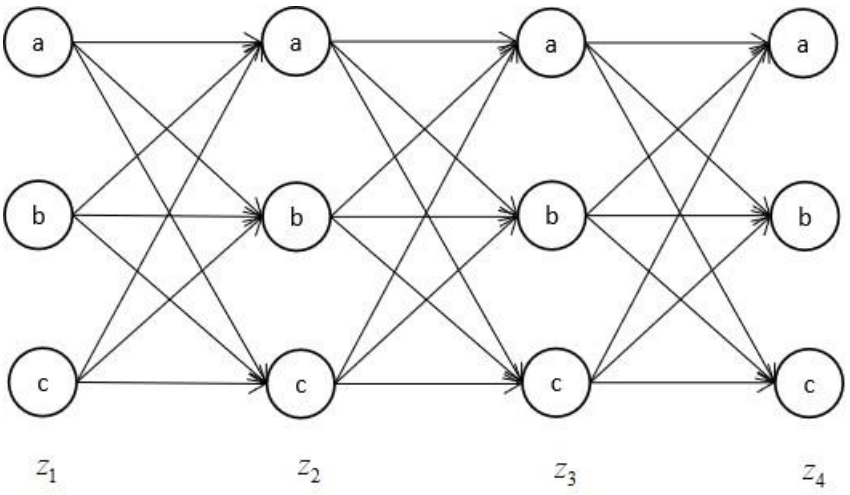
对于中文分词，词性标注等问题，在《机器学习与应用》中有详细的讲解，包括如何用循环神经网络解决此问题，感兴趣的读者可以进一步阅读。

估值问题

估值问题需要计算隐马尔可夫模型产生一个观测序列$\mathbf{x}=\{x_1,...,x_T\}$的概率。因为任意一种状态序列取值都可能会导致出现此观测序列，根据全概率公式，其值为

$$p(\mathbf{x})=\sum_{\mathbf{z}} p(\mathbf{x}|\mathbf{z})\,p(\mathbf{z})$$

上式列举所有可能的状态序列，以及该状态序列产生此观测序列的概率，要对n^T项求和。因为每一时刻的状态取值有n种可能，因此长度为T的状态序列总共有n^T种可能。下图展示了这一过程



已经推导过，任意一个状态序列出现的概率为

$$p(\mathbf{z})=\prod_{t=1}^{T}p(z_t|z_{t-1})$$

由于每一时刻的观测值只依赖于本时刻的状态值，因此有

$$p(\mathbf{x}|\mathbf{z})=\prod_{t=1}^{T}p(x_t|z_t)$$

产生一个观测序列的概率为

$$p(\mathbf{x})=\sum_{\mathbf{z}}\prod_{t=1}^{T}p(z_t|z_{t-1})p(x_t|z_t)=\sum_{\mathbf{z}}\prod_{t=1}^{T}b_{z_t x_t}a_{z_t z_{t-1}}$$

直接计算这个值的复杂度是$O(n^T T)$。显然上面的公式有很多重复计算。例如要计算产生观测序列$(x_1,...,x_5)$的概率，产生它的状态序列为$(z_1,...,z_5)$，假设状态取值有 3 种情况。无论z_5取什么值，为了计算整个序列出现的概率，任何一个长度为 4 的子序列$(z_1,...,z_4)$产生观测子序列$(x_1,...,x_4)$的概率都要被重复计算 3 次。利用这一特点可以使用动态规划算法高效求解。

假设已经计算出了长度为$t-1$的观测序列的概率，现在要计算长度为t的观测序列的概率。如果状态的取值有n种可能，则z_t的取值有n种可能。定义变量

$$\alpha_i(t)=p(x_1,...,x_t,z_t=i)$$

这个变量是到时刻t为止的观测序列，产生它的状态序列中，最后一个状态为i，即$z_t=i$的概率。因此有

$$p(\mathbf{x})=p(x_1,...,x_T)=\sum_{i=1}^{n}p(x_1,...,x_T,z_T=i)=\sum_{i=1}^{n}\alpha_i(T)$$

根据定义可以得到这个变量的递归计算公式

$$\alpha_j(t)=\sum_{i=1}^{n}\alpha_i(t-1)a_{ij}b_{jx_t},\ j=1,...,n,t=1,...,T$$

由此得到计算观测序列概率的高效算法。

初始化 $\alpha_i(0)=a_{0i},i=1,...,n$

循环，对 $t=1,...,T$

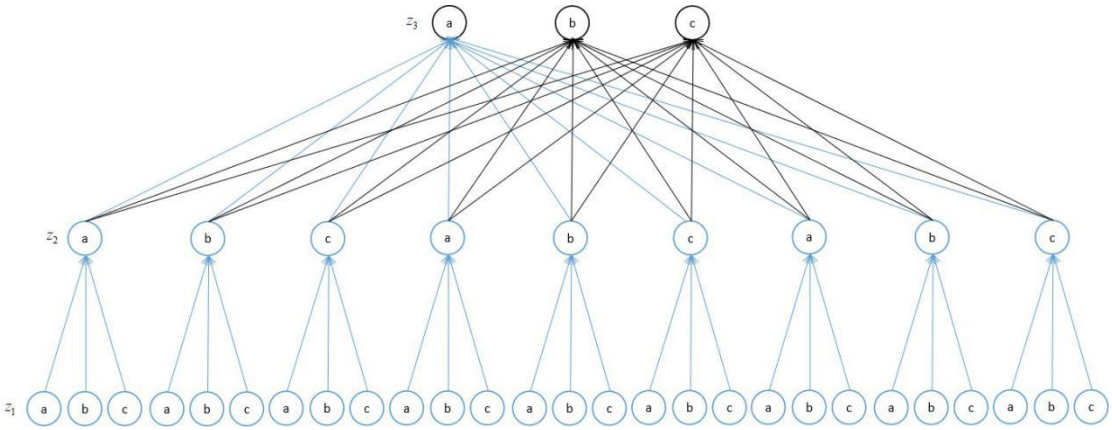
循环，对 $j=1,...,n$

递归计算 $\alpha_j(t)=\sum_{i=1}^{n}\alpha_i(t-1)a_{ij}b_{jx_t}$

输出： $p(\mathbf{x})=\sum_{i=1}^{n}\alpha_i(T)$

上面算法的时间复杂度为 $O(n^2T)$，这比之前大为减少。此算法称为前向算法，也可以实现后向算法，即从后向前计算。这需要定义变量 β 然后反向递推计算，原理与前向算法相同。

下面给出前向算法的直观解释。如果将状态序列所有时刻的路径展开，可以形成如下图所示的树结构



前向变量是对上图中以某一节点为根的子树中所有路径求和的结果。在上图中在 3 时刻的值 z_3 经过值 a 的所有路径构成的子树以蓝色表示，这一子树求和的结果即为 $\alpha_a(3)$。只要得到所有子树的求和结果，通过递推可以得到以它们的父节点为根的子树的结果。

解码问题

解码问题指已知一个观测序列，寻找出最有可能产生它的状态序列，这是实际应用时最常见的问题。根据贝叶斯公式，解码问题可以形式化的定义为如下最大后验概率问题

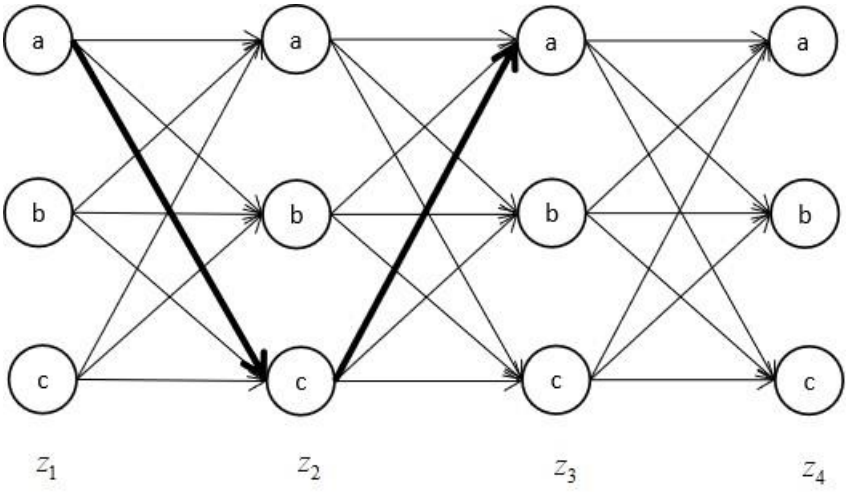
$$\begin{aligned}\arg\max_{\mathbf{z}} p(\mathbf{z}|\mathbf{x}) &= \arg\max_{\mathbf{z}} \frac{p(\mathbf{x},\mathbf{z})}{p(\mathbf{x})} \\ &= \arg\max_{\mathbf{z}} \frac{p(\mathbf{x},\mathbf{z})}{\sum_{\mathbf{z}} p(\mathbf{x},\mathbf{z})} \\ &= \arg\max_{\mathbf{z}} p(\mathbf{x},\mathbf{z})\end{aligned}$$

和贝叶斯分类器相同，忽略掉分母，因为它对所有状态序列是相同的。贝叶斯分类器是已知特征向量计算类后验概率，这里是已知观测序列反算状态序列的条件概率。

最简单的方法是列举所有可能的状态序列，然后计算它们产生该观测序列的概率，找出概率最大的那个。但这是没有必要的，通过使用动态规划算法，可以高效的解决此问题。动态规划求解最优路径时的核心结论是：要保证一个解是全局最优解，其部分解也必须是最优的。根据这一结论，可以得到经典的维特比（Viterbi）算法。

要保证 $p(x_1,...,x_T,z_1,...,z_T)$ 的概率最大，就需要保证 $p(x_1,...,x_{T-1},z_1,...,z_{T-1})$ 的概率最大，这相当于寻找一条产生最大概率的路径，这条路径对应着一个状态序列。这和前面的前向算法类似，只要把求和换成求最大值即可。

如果整体路径是最优的，那么子路径也是最优的。假设概率最大的路径是 $(z_1,...,z_T)$，在 t 时刻经过的节点为 z_t，路径序列 $z_t,...,z_T$ 必须是最优的。假设它不是最优的，则存在另外一个序列 $z_t^{'},...,z_T$ 的概率值更大，这与 $(z_1,...,z_T)$ 是最优解矛盾。下图是维特比算法求解的示意图



上图中最优路径用加粗线表示。如果得到了 1 时刻到 3 时刻的最优路径，根据递推公式可以得到更长的序列的最优路径。

基于这个思想，从 1 时刻开始，递推的计算 t 时刻状态 $z_t=i$ 的子序列的最大概率路径，

最后就可以得到整个问题的最优解。这一过程与前向算法、后向算法类似，区别在于是求极大值而不是求和。定义如下变量

$$\alpha_t(i)=\max_{z_1,...,z_{t-1}} p(z_t=i,z_{t-1},...,z_1,x_t,...,x_1),i=1,2...,T$$

即产生观测序列$(x_1,...,x_t)$的所有状态序列$(z_1,...,z_t)$中，t时刻的状态$z_t=i$的概率的最大值。根据它的定义，可以得到递推计算公式

$$\alpha_t(i)=\max_j\left(\alpha_{t-1}(j)a_{ji}b_{ix_t}\right),j=1,...,n,t=1,...,T$$

最后可以得到产生观测序列的最大概率为

$$\max_i \alpha_T(i)$$

上面的定义只能得到最大概率，但要求解的得到这个最大概率的状态序列，为此定义下面的变量记住这个最优路径

$$\beta_t(i)=\arg\max_j \alpha_{t-1}(j)a_{ji},i=1,...,n,j=1,...,n$$

即t时刻的状态$z_t=i$的概率最大的状态序列中，$t-1$时刻的状态值。有了这两个变量，就可以得到维特比算法。

初始化，$\alpha_1(i)=a_{0i}b_{ix_1},i=1,..,n,\beta_1(i)=0,i=1,...,n$

循环，对$t=2,...,T$

循环，对$i=1,...,n$

计算$\alpha_t(i)=\max_j\left(\alpha_{t-1}(j)a_{ji}b_{ix_t}\right)$

计算$\beta_t(i)=\arg\max_j \alpha_{t-1}(j)\mathrm{A}_{ji}$

得到最大概率，$p_{\max}=\max_i \alpha_T(i),z_T=\arg\max_i \alpha_T(i)$

反向回溯计算最优路径，循环，对$t=T-1,...,1$

计算$z_t=\beta_{t+1}(z_{t+1})$

结束

在算法实现时，需要存储所有的$\beta_t(i)$，而只用存储当前步的$\alpha_t(i)$。这个算法的时间复杂度为$O(nT)$。

训练算法

训练时给定一组样本，确定状态转移矩阵和观测矩阵。目标是状态转移矩阵和观测矩阵能很好的解释这组样本，通过最大似然估计实现。如果已知训练样本集中每个观测序列对应的状态序列，则可以直接根据最大似然估计得到模型参数，具体方法已经介绍，不同的是增加了观测矩阵。

下面考虑第二种情况，训练样本集只有观测值而没有状态值。假设有l个训练样本，第i个样本的观测序列为$\mathbf{x}_i$，其对应的状态序列为$\mathbf{z}_i$，序列长度为T，$\mathbf{z}_i$未知，计算$\mathbf{x}_i$的边缘概率时要对其所有可能的取值求和。假设状态集的大小为n，观测集的大小为m。为简化表述，考虑对单个样本的情况，对数似然函数为

$$\begin{aligned}
L(\mathbf{A},\mathbf{B}) &= \ln p(\mathbf{x};\mathbf{A},\mathbf{B}) = \sum_{\mathbf{z}} \ln p(\mathbf{x},\mathbf{z};\mathbf{A},\mathbf{B}) \\
&= \sum_{\mathbf{z}} \ln\left(p(\mathbf{z};\mathbf{A},\mathbf{B})\, p(\mathbf{x}|\mathbf{z};\mathbf{A},\mathbf{B})\right) \\
&= \sum_{\mathbf{z}} \ln\left(\left(\prod_{t=1}^{T} p(z_t|z_{t-1})\right)\left(\prod_{t=1}^{T} p(x_t|z_t)\right)\right) \\
&= \sum_{\mathbf{z}} \ln\left(\left(\prod_{t=1}^{T} a_{z_{t-1}z_t}\right)\left(\prod_{t=1}^{T} b_{z_t x_t}\right)\right) \\
&= \sum_{\mathbf{z}} \sum_{t=1}^{T} \left(\ln a_{z_{t-1}z_t} + \ln b_{z_t x_t}\right) \\
&= \sum_{\mathbf{z}} \left(\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T}\left(1\{z_{t-1}=i \wedge z_t=j\}\ln a_{ij}\right) + \sum_{j=1}^{n}\sum_{k=1}^{m}\sum_{t=1}^{T}\left(1\{z_t=j \wedge x_t=k\}\ln b_{jk}\right)\right)
\end{aligned}$$

这里含有隐变量（状态变量），因此需要用 EM 算法求解。EM 算法的详细原理在 SIGAI 之前的公众号文章“理解 EM 算法”以及《机器学习与应用》一书中有详细的讲解。

按照 EM 算法框架，在 E 步根据参数$\mathbf{A}$和$\mathbf{B}$的当前估计值计算隐变量$\mathbf{z}$的条件概率

$$Q(\mathbf{z}) = p(\mathbf{z}|\mathbf{x};\mathbf{A},\mathbf{B})$$

在 M 步计算数学期望，构造下界函数

$$\begin{aligned}
&\sum_{\mathbf{z}} Q(\mathbf{z}) \ln \frac{p(\mathbf{x},\mathbf{z};\mathbf{A},\mathbf{B})}{Q(\mathbf{z})} \\
&= \sum_{\mathbf{z}} Q(\mathbf{z}) \left(\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T}\left(1\{z_{t-1}=i \wedge z_t=j\}\ln a_{ij}\right) + \sum_{j=1}^{n}\sum_{k=1}^{m}\sum_{t=1}^{T}\left(1\{z_t=j \wedge x_t=k\}\ln b_{jk}\right) - \ln Q(\mathbf{z})\right)
\end{aligned}$$

在这里$\ln Q(\mathbf{z})$是与$\mathbf{A}$和$\mathbf{B}$无关的常数，可以忽略。由于状态转移矩阵和观测矩阵满足等式约束，构造拉格朗日乘子函数

$$L(\mathbf{A},\mathbf{B},\boldsymbol{\mu},\boldsymbol{\nu})=$$

$$\sum_{\mathbf{z}}Q(\mathbf{z})\left(\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T}\left(1\{z_{t-1}=i\wedge z_t=j\}\ln a_{ij}\right)+\sum_{j=1}^{n}\sum_{k=1}^{m}\sum_{t=1}^{T}\left(1\{z_t=j\wedge x_t=k\}\ln b_{jk}\right)\right)+$$

$$\sum_{i=1}^{n}\mu_i\left(1-\sum_{j=1}^{n}a_{ij}\right)+\sum_{j=1}^{n}\nu_j\left(1-\sum_{k=1}^{m}b_{jk}\right)$$

对 a_{ij} 求偏导数并令其为 0，可以得到

$$\frac{\partial L(\mathbf{A},\mathbf{B},\boldsymbol{\mu},\boldsymbol{\nu})}{\partial a_{ij}}=\sum_{\mathbf{z}}Q(\mathbf{z})\frac{1}{a_{ij}}\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}-\mu_i=0$$

解得

$$a_{ij}=\frac{1}{\mu_i}\sum_{\mathbf{z}}Q(\mathbf{z})\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}$$

对 b_{jk} 求偏导数并令其为 0，可以得到

$$\frac{\partial L(\mathbf{A},\mathbf{B},\boldsymbol{\mu},\boldsymbol{\nu})}{\partial b_{jk}}=\sum_{\mathbf{z}}Q(\mathbf{z})\frac{1}{b_{jk}}\sum_{t=1}^{T}1\{z_t=j\wedge x_t=k\}-\nu_j=0$$

解得

$$b_{jk}=\frac{1}{\nu_j}\sum_{\mathbf{z}}Q(\mathbf{z})\sum_{t=1}^{T}1\{z_t=j\wedge x_t=k\}$$

对 μ_i 求偏导数，并令其为 0，可以得到

$$\begin{aligned}\frac{\partial L(\mathbf{A},\mathbf{B},\boldsymbol{\mu},\boldsymbol{\nu})}{\partial \mu_i}&=1-\sum_{j=1}^{n}a_{ij}\\&=1-\sum_{j=1}^{n}\frac{1}{\mu_i}\sum_{\mathbf{z}}Q(\mathbf{z})\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}=0\end{aligned}$$

解得

$$\begin{aligned}\mu_i&=\sum_{j=1}^{n}\sum_{\mathbf{z}}Q(\mathbf{z})\sum_{t=1}^{T}1\{z_{t-1}=i\wedge z_t=j\}\\&=\sum_{\mathbf{z}}Q(\mathbf{z})\sum_{t=1}^{T}1\{z_{t-1}=i\}\end{aligned}$$

对 ν_j 求偏导数，并令其为 0，可以得到

$$\frac{\partial L(\mathbf{A},\mathbf{B},\boldsymbol{\mu},\boldsymbol{\nu})}{\partial \nu_j} = 1-\sum_{k=1}^{m} b_{jk}$$

$$= 1-\sum_{k=1}^{m}\frac{1}{\nu_j}\sum_{\mathbf{z}} Q(\mathbf{z})\sum_{t=1}^{T} 1\{z_t = j \wedge x_t = k\} = 0$$

解得

$$\nu_j = \sum_{k=1}^{m}\sum_{\mathbf{z}} Q(\mathbf{z})\sum_{t=1}^{T} 1\{z_t = j \wedge x_t = k\}$$

$$= \sum_{\mathbf{z}} Q(\mathbf{z})\sum_{t=1}^{T} 1\{z_t = j\}$$

将 μ_i 和 ν_j 的值分别代入 a_{ij} 和 b_{jk} 的解，可以得到

$$a_{ij} = \frac{\sum_{\mathbf{z}} Q(\mathbf{z})\sum_{t=1}^{T} 1\{z_{t-1} = i \wedge z_t = j\}}{\sum_{\mathbf{z}} Q(\mathbf{z})\sum_{t=1}^{T} 1\{z_{t-1} = i\}}$$

$$b_{jk} = \frac{\sum_{\mathbf{z}} Q(\mathbf{z})\sum_{t=1}^{T} 1\{z_t = j \wedge x_t = k\}}{\sum_{\mathbf{z}} Q(\mathbf{z})\sum_{t=1}^{T} 1\{z_t = j\}}$$

但上面两个值直接计算的成本太高，状态序列 $\mathbf{z}$ 的所有可能取值有 n^T 种。这一问题可用估值问题中使用的技巧解决，递推的计算这两个值。

$$\sum_{\mathbf{z}} Q(\mathbf{z})\sum_{t=1}^{T} 1\{z_{t-1} = i \wedge z_t = j\} = \sum_{t=1}^{T}\sum_{\mathbf{z}} 1\{z_{t-1} = i \wedge z_t = j\} Q(\mathbf{z})$$

$$= \sum_{t=1}^{T}\sum_{\mathbf{z}} 1\{z_{t-1} = i \wedge z_t = j\}\, p(\mathbf{z}|\mathbf{x};\mathbf{A},\mathbf{B})$$

$$= \frac{1}{p(\mathbf{x};\mathbf{A},\mathbf{B})}\sum_{t=1}^{T}\sum_{\mathbf{z}} 1\{z_{t-1} = i \wedge z_t = j\}\, p(\mathbf{z},\mathbf{x};\mathbf{A},\mathbf{B})$$

$$= \frac{1}{p(\mathbf{x};\mathbf{A},\mathbf{B})}\sum_{t=1}^{T} \alpha_i(t) a_{ij} b_{jx_t} \beta_j(t+1)$$

类似的有

$$\sum_{\mathbf{z}} Q(\mathbf{z})\sum_{t=1}^{T} 1\{z_{t-1} = i\} = \sum_{j=1}^{n}\sum_{\mathbf{z}} Q(\mathbf{z})\sum_{t=1}^{T} 1\{z_{t-1} = i \wedge z_t = j\}$$

$$= \frac{1}{p(\mathbf{x};\mathbf{A},\mathbf{B})}\sum_{j=1}^{n}\sum_{t=1}^{T} \alpha_i(t) a_{ij} b_{jx_t} \beta_j(t+1)$$

因此有

$$a_{ij}=\frac{\sum_{t=1}^{T}\alpha_i(t)a_{ij}b_{jx_t}\beta_j(t+1)}{\sum_{j=1}^{n}\sum_{t=1}^{T}\alpha_i(t)a_{ij}b_{jx_t}\beta_j(t+1)}$$

用同样的方法可以计算出b_{jk}。由此得到求解隐马尔可夫模型训练问题的 Baum-Welch 算法。

用随机数初始化矩阵$\mathbf{A}$和$\mathbf{B}$的元素，矩阵元素要满足等式约束条件

循环，直到收敛：

E 步：循环，根据当前参数值用前向算法和后向算法计算α和β，然后计算

$$\gamma_t(i,j)=\alpha_i(t)a_{ij}b_{jx_t}\beta_j(t+1)$$

M 步：更新参数的值

$$a_{ij}=\frac{\sum_{t=1}^{T}\gamma_t(i.j)}{\sum_{j=1}^{n}\sum_{t=1}^{T}\gamma_t(i,j)}$$

$$b_{ik}=\frac{\sum_{i=1}^{n}\sum_{t=1}^{T}1\{x_t=k\}\gamma_t(i,j)}{\sum_{i=1}^{n}\sum_{t=1}^{T}\gamma_t(i,j)}$$

结束循环

参考文献

[1] Baum, L. E., Petrie, T. Statistical Inference for Probabilistic Functions of Finite State Markov Chains. The Annals of Mathematical Statistics. 37 (6): 1554 – 1563. 1966.

[2] Baum, L. E., Eagon, J. A. An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. Bulletin of the American Mathematical Society. 73 (3): 360. 1967.

[3] Baum, L. E., Petrie, T., Soules, G., Weiss, N. A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains. The Annals of Mathematical Statistics. 41: 164. 1970

[4] Baum, L.E. An Inequality and Associated Maximization Technique in Statistical Estimation of Probabilistic Functions of a Markov Process. Inequalities. 3: 1 – 8. 1972.

[5] Lawrence R. Rabiner. A tutorial on Hidden Markov Models and selected applications in speech recognition. Proceedings of the IEEE. 77 (2): 257 – 286. 1989.