人工智能之机器学习

隐马尔可夫模型

上海育创网络科技股份有限公司

主讲人：刘老师(GerryLiu)

课程要求

- 课上课下“九字”真言
  - 认真听，**善摘录，勤思考**
  - **多温故，乐实践**，再发散
- 四不原则
  - 不懒散惰性，不迟到早退
  - 不请假旷课，不拖延作业
- 一点注意事项
  - 违反“四不原则”，不推荐就业

课程内容

- 隐马尔可夫模型
- Viterbi算法(必须掌握)

马尔可夫性质

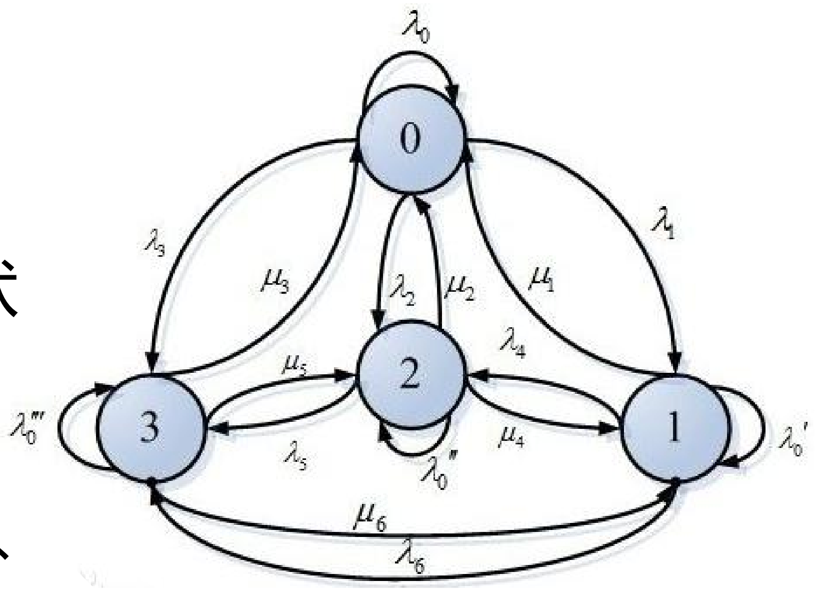
- 设{X(t), t ∈ T}是一个随机过程，E为其状态空间，若对于任意的$t_1<t_2< \dots<t_n<t$，任意的$x_1,x_2,\dots,x_n,x\in E$，随机变量X(t)在已知变量$X(t_1)=x_1,\dots,X(t_n)=x_n$之下的条件分布函数只与$X(t_n)=x_n$有关，而与$X(t_1)=x_1,\dots,X(t_{n-1})=x_{n-1}$无关，即条件分布函数满足下列等式，此性质称为马尔可夫性；如果随机过程满足马尔可夫性，则该过程称为马尔可夫过程。

$$p\left(X(t)\le x \middle| X(t_1)=x_1,\dots,X(t_n)=x_n\right)=p\left(X(t)\le x \middle| X(t_n)=x_n\right)$$

$$p\left(X_{n+1}=x \mid X_1=x_1,\dots,X_n=x_n\right)=p\left(X_{n+1}=x \mid X_n=x_n\right)$$

马尔可夫链

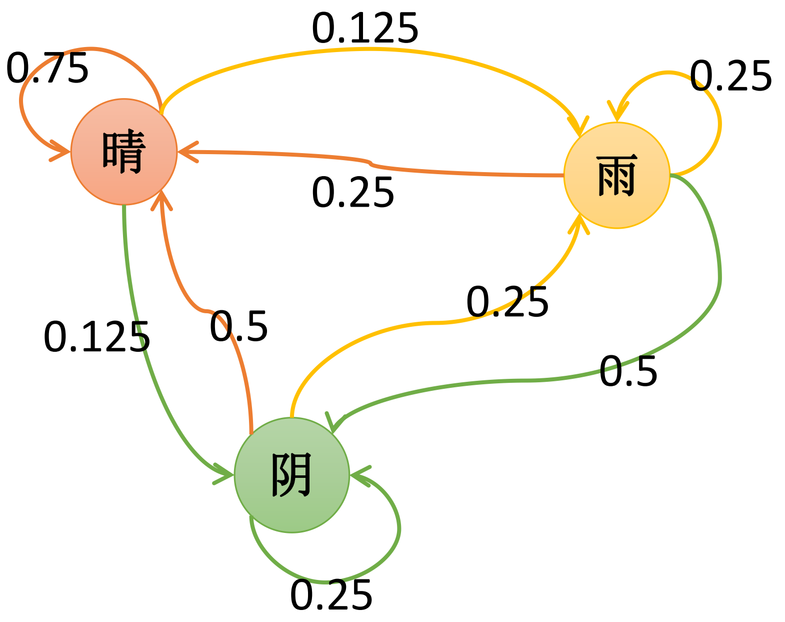
- 马尔可夫链是指具有马尔可夫性质的随机过程。在过程中，在给定当前信息的情况下，过去的信息状态对于预测将来状态是无关的。
- 在马尔可夫链的每一步，系统根据概率分布，可以从一个状态变成另外一个状态，也可以保持当前状态不变。状态的改变叫做转移，状态改变的相关概率叫做转移概率。
- 马尔可夫链中的三元素是：状态空间S、转移概率矩阵P、初始概率分布π。



马尔可夫链案例

- 设将天气状态分为晴、阴、雨三种状态，假定某天的天气状态只和上一天的天气状态有关，状态使用1(晴)、2(阴)、3(雨)表示，转移概率矩阵P如下：

今/明	晴	阴	雨
晴	0.75	0.125	0.125
阴	0.5	0.25	0.25
雨	0.25	0.5	0.25



马尔可夫链案例

- 第n+1天天气状态为j的概率为：

$$\pi\left(X_{n+1}=j\right)=\sum_{i=1}^{K}\pi\left(X_n=i\right)\cdot P\left(X_{n+1}=j \mid X_n=i\right)$$

$$\Rightarrow \pi^{n+1}=\pi^{n}\cdot P$$

- 因此，矩阵P即为条件概率转移矩阵。
  - 矩阵P的第i行元素表示，在上一个状态为i的时候的分布概率，即每行元素的和必须为1

马尔可夫链案例

初始概率π[0.5,0.3,0.2]

$$P=\begin{bmatrix}0.75 & 0.125 & 0.125\\ 0.5 & 0.25 & 0.25\\ 0.25 & 0.5 & 0.25\end{bmatrix}$$

第n天	晴	阴	雨
0	0.5	0.3	0.2
1	0.575	0.2375	0.1875
2	0.5969	0.225	0.1781
3	0.6047	0.2199	0.1754
4	0.6073	0.2183	0.1744
5	0.6082	0.2177	0.1741
6	0.6085	0.2175	0.174
7	0.6086	0.2174	0.1739
8	0.6087	0.2174	0.1739
9	0.6087	0.2174	0.1739
10	0.6087	0.2174	0.1739
11	0.6087	0.2174	0.1739

马尔可夫链案例

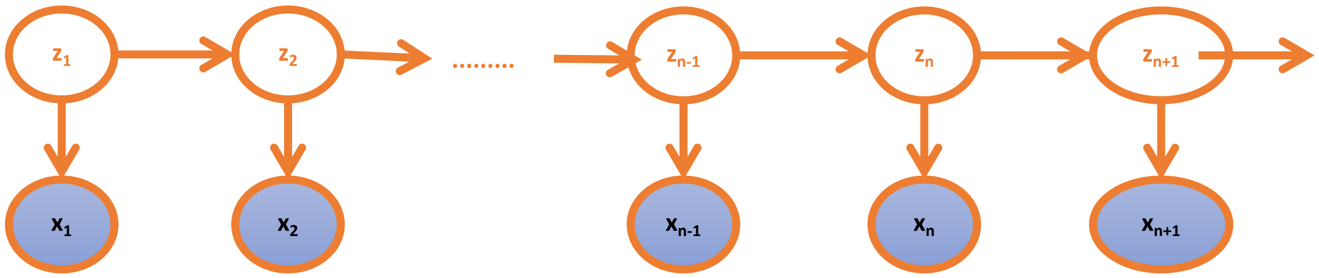
初始概率π[0.1,0.6,0.3]

$$P=\begin{bmatrix}0.75 & 0.125 & 0.125\\ 0.5 & 0.25 & 0.25\\ 0.25 & 0.5 & 0.25\end{bmatrix}$$

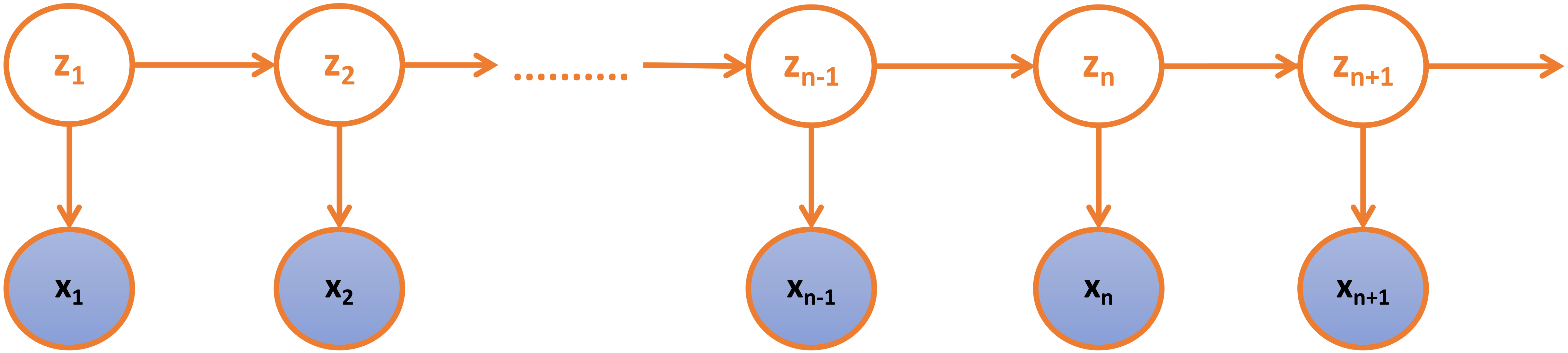
第n天	晴	阴	雨
0	0.1	0.6	0.3
1	0.45	0.3125	0.2375
2	0.5531	0.2531	0.1937
3	0.5898	0.2293	0.1809
4	0.6022	0.2215	0.1763
5	0.6065	0.2188	0.1747
6	0.6079	0.2179	0.1742
7	0.6084	0.2176	0.174
8	0.6086	0.2174	0.1739
9	0.6087	0.2174	0.1739
10	0.6087	0.2174	0.1739
11	0.6087	0.2174	0.1739

HMM

- 隐马尔可夫模型(Hidden Markov Model, HMM)是一种统计模型，在语音识别、行为识别、NLP、故障诊断等领域具有高效的性能。
- HMM是关于时序的概率模型，描述一个含有未知参数的马尔可夫链所生成的不可观测的状态随机序列，再由各个状态生成观测随机序列的过程。HMM是一个双重随机过程---具有一定状态的隐马尔可夫链和随机的观测序列。
- HMM随机生成的状态随机序列被称为状态序列；每个状态生成一个观测，由此产生的观测随机序列，被称为观测序列。



HMM



- $z_1,z_2...,z_n$是不可观测的状态，$x_1,x_2,...x_n$是可观测到的序列；不可观测的状态决定可观测序列的值(z的取值决定x的取值)。
- 在z_1、z_2不可观测的情况下，x_1和z_2独立吗？x_1和x_2独立吗？

HMM

- HMM由隐含状态S、可观测状态O、初始状态概率矩阵/向量π、隐含状态转移概率矩阵A、可观测值转移矩阵B(又称为混淆矩阵，Confusion Matrix)；
- π和A决定了状态序列，B决定观测序列，因此HMM可以使用三元符号表示，称为HMM的三元素：

$$\lambda = \left(A, B, \pi\right)$$

HMM参数说明

$$\lambda = (A, B, \pi)$$

- S是所有可能的状态集合：

$$S = \{s_1, s_2, \ldots, s_n\}$$

- O是所有可能的观测集合：

$$O = \{o_1, o_2, \ldots, o_m\}$$

- I是长度为T的状态序列，Q是对应的观测序列

$$I = \{i_1, i_2, \ldots, i_T\} \qquad Q = \{q_1, q_2, \ldots, q_T\}$$

HMM参数

$$\lambda = (A, B, \pi)$$

- A是隐含状态转移概率矩阵:

$$A = \left[a_{ij}\right]_{n*n} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

- 其中
  - a_{ij}是在时刻t处于状态s_i的条件下时刻t+1转移到状态s_j的概率。

$$a_{ij} = p\left(i_{t+1} = s_j \middle| i_t = s_i\right)$$

HMM参数

$$\lambda = (A, B, \pi)$$

- B是可观测值转移概率矩阵:

$$B = [b_{ij}]_{n*m} = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1m} \\ b_{21} & b_{22} & \dots & b_{2m} \\ \dots & \dots & \dots & \dots \\ b_{n1} & b_{n2} & \dots & b_{nm} \end{bmatrix}$$

- 其中
  - b_{ij}是在时刻t处于状态s_i的条件下生成观测值o_j的概率。

$$b_{ij} = p(q_t = o_j | i_t = s_i)$$

HMM参数

$$\lambda = (A, B, \pi)$$

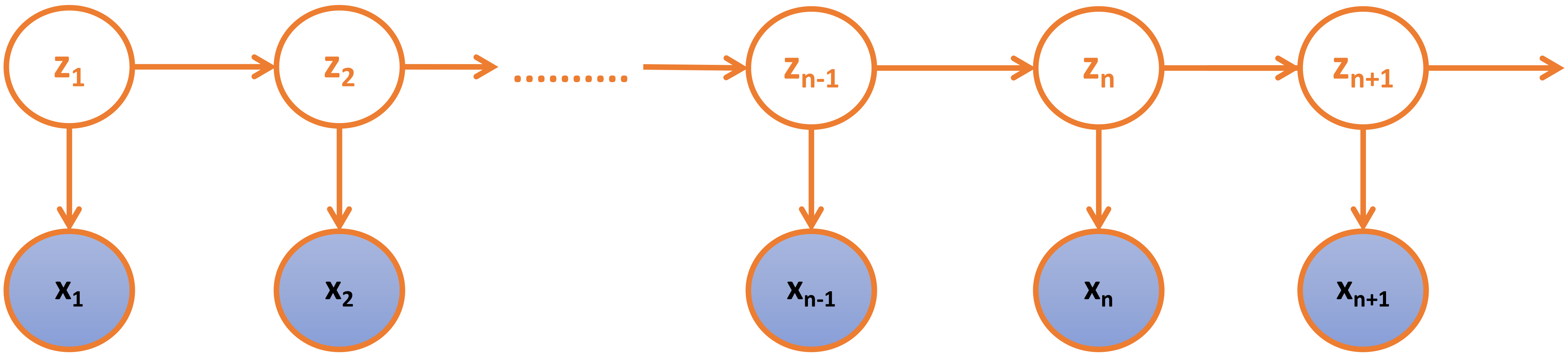
- π是初始状态概率向量:

$$\pi = (\pi_i)_{1*n} = (\pi_1, \pi_2, \ldots, \pi_n)$$

- 其中
  - π_i是在时刻t=1处于状态s_i的概率。

$$\pi_i = p(i_1 = s_i)$$

HMM的两个基本性质



$$p\left(i_t \middle| i_{t-1}, q_{t-1}, i_{t-2}, q_{t-2}, \ldots, i_1, q_1\right) = p\left(i_t \middle| i_{t-1}\right)$$

$$p\left(q_t \middle| i_t, i_{t-1}, q_{t-1}, i_{t-2}, q_{t-2}, \ldots, i_1, q_1\right) = p\left(q_t \middle| i_t\right)$$

HMM案例

- 假设有三个盒子，编号为0,1,2；每个盒子都装有黑白两种颜色的小球，球的比例如下：

编号	白球	黑球
0	4	6
1	8	2
2	5	5

- 按照下列规则的方式进行有放回的抽取小球，得到球颜色的观测序列：
  - 按照π的概率选择第一个盒子，从盒子中随机抽取出一个小球，记录颜色后，放回盒子中；
  - 按照某种条件概率选择新的盒子，重复该操作；
  - 最终得到观测序列：“白黑白白黑”

HMM案例

- 状态集合：S={盒子1，盒子2，盒子3}
- 观测集合：O={白，黑}
- 状态序列和观测序列的长度T=5
- 初始概率分布π
- 状态转移概率矩阵A
- 观测概率矩阵B

$$\pi = \begin{pmatrix} 0.2 \\ 0.5 \\ 0.3 \end{pmatrix}$$

$$A = \begin{bmatrix} 0.5 & 0.4 & 0.1 \\ 0.2 & 0.2 & 0.6 \\ 0.2 & 0.5 & 0.3 \end{bmatrix} \quad B = \begin{bmatrix} 0.4 & 0.6 \\ 0.8 & 0.2 \\ 0.5 & 0.5 \end{bmatrix}$$

HMM案例思考

- 在给定参数π、A、B的时候，得到观测序列为“白黑白白黑”的概率是多少??
- 给定多组观测序列，那么最有可能的模型参数π、A、B是多少?
- 在给定参数π、A、B以及观测序列为“白黑白白黑”的时候，得到状态序列为“盒子1,盒子2,盒子1,盒子3,盒子2”的概率是多少??
- 在给定参数π、A、B以及观测序列为“白黑白白黑”的时候，最大可能的状态序列是什么??

HMM的三个问题

- 概率计算问题：**前向-后向算法**
  - 给定模型λ=(A,B,π)，计算模型λ下观测到序列$Q=\{q_1,q_2,...,q_T\}$出现的概率P(Q|λ)
- 学习问题：**Baum-Welch算法**(状态未知)
  - 已知观测序列$Q=\{q_1,q_2,...,q_T\}$，估计模型λ=(A,B,π)的参数，使得在该模型下观测序列P(Q|λ)最大。
- 预测问题：**Viterbi算法**
  - 给定模型λ=(A,B,π)和观测序列$Q=\{q_1,q_2,...,q_T\}$，求给定观测序列条件概率P(I|Q，λ)最大的状态序列I

概率计算问题

- 直接计算法
  - 暴力算法
- 前向算法
- 后向算法

直接计算法

- 按照概率公式，列举所有可能的长度为T的状态序列$I=\{i_1,i_2,...,i_T\}$，求各个状态序列I与观测序列$Q=\{q_1,q_2,...,q_T\}$的联合概率$P(Q,I;\lambda)$，然后对所有可能的状态序列求和，从而得到最终的概率$P(Q;\lambda)$

$$I=\{i_1,i_2,\dots,i_T\} \qquad p(I;\lambda)=\pi_{i_1}a_{i_1i_2}a_{i_2i_3}\dots a_{i_{T-1}i_T}$$

$$Q=\{q_1,q_2,\dots,q_T\} \qquad p(Q|I;\lambda)=b_{i_1q_1}b_{i_2q_2}\dots b_{i_Tq_T}$$

$$P(Q,I;\lambda)=p(Q|I;\lambda)p(I;\lambda)=\pi_{i_1}a_{i_1i_2}a_{i_2i_3}\dots a_{i_{T-1}i_T}b_{i_1q_1}b_{i_2q_2}\dots b_{i_Tq_T}$$

$$p(Q;\lambda)=\sum_I p(Q,I;\lambda)=\sum_{i_1,i_2,\dots,i_T}\pi_{i_1}a_{i_1i_2}a_{i_2i_3}\dots a_{i_{T-1}i_T}b_{i_1q_1}b_{i_2q_2}\dots b_{i_Tq_T}$$

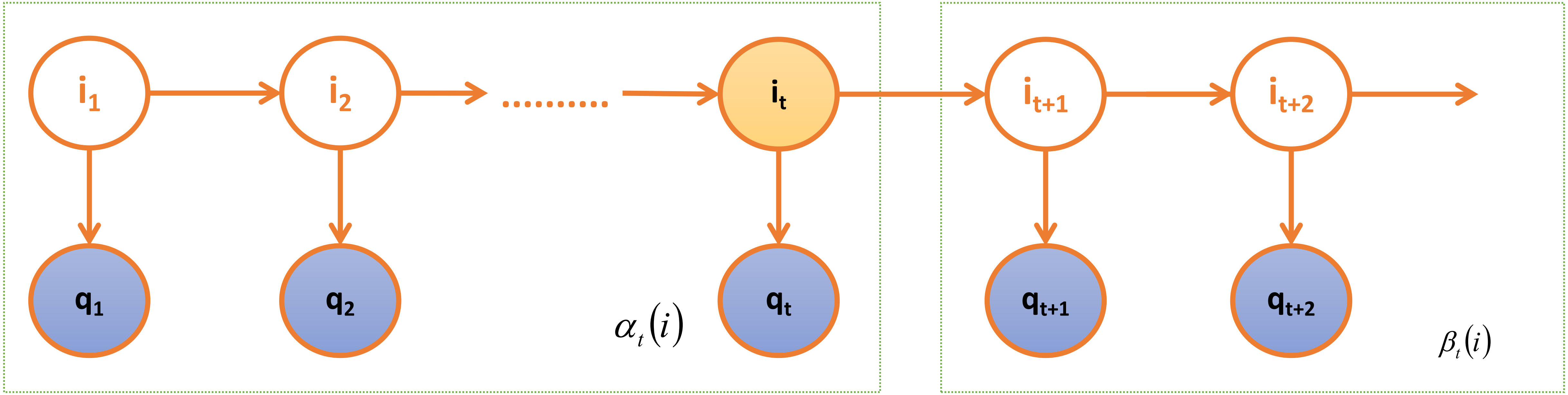
前向概率-后向概率

- 前向概率-后向概率指的其实是在一个观测序列中，时刻t对应的状态为s_i，并且观测序列已知情况下的联合概率值转换过来的值。

$$
\begin{aligned}
p&\left(q_1, q_2 \ldots q_T, i_t = s_i\right) \\
&= p\left(q_1, q_2 \ldots q_t, i_t = s_i\right) p\left(q_{t+1}, q_{t+2} \ldots q_T, \middle| q_1, q_2 \ldots q_t, i_t = s_i\right) \\
&= p\left(q_1, q_2 \ldots q_t, i_t = s_i\right) p\left(q_{t+1}, q_{t+2} \ldots q_T, \middle| i_t = s_i\right) \\
&= \alpha_t(i) * \beta_t(i)
\end{aligned}
$$

前向概率-后向概率

$$\alpha_t(i) = p(q_1, q_2 \dots q_t, i_t = s_i) \quad \beta_t(i) = p(q_{t+1}, q_{t+2} \dots q_T | i_t = s_i)$$



前向算法

- 定义：给定λ，定义到时刻t部分观测序列为$q_1,q_2,\ldots,q_t$且状态为s_i的概率为**前向概率**。记做：

$$\alpha_t(i)=p(q_1,q_2,\ldots,q_t,i_t=s_i;\lambda)$$

前向算法

$$
\begin{aligned}
\alpha_t(i) &= p(q_1, q_2, \ldots, q_t, i_t = s_i) \\
&= p(q_1, q_2, \ldots, q_{t-1}, i_t = s_i) * p(q_t \mid q_1, q_2, \ldots, q_{t-1}, i_t = s_i) \\
&= p(q_1, q_2, \ldots, q_{t-1}, i_t = s_i) * p(q_t \mid i_t = s_i) \\
&= \left[\sum_{j=1}^{n} p(q_1, q_2, \ldots, q_{t-1}, i_{t-1} = s_j, i_t = s_i) \right] * p(q_t \mid i_t = s_i) \\
&= \left[\sum_{j=1}^{n} p(q_1, q_2, \ldots, q_{t-1}, i_{t-1} = s_j) p(i_t = s_i \mid i_{t-1} = s_j) \right] * p(q_t \mid i_t = s_i) \\
&= \left(\sum_{j=1}^{n} \alpha_{t-1}(j) a_{ji} \right) b_{iq_t}
\end{aligned}
$$

前向算法

$$\alpha_t(i) = p(q_1, q_2, \ldots, q_t, i_t = s_i; \lambda)$$

- 初值: $\alpha_1(i) = p(q_1, i_1 = s_i; \lambda) = \pi_i b_{iq_1}$

- 递推：对于t=1,2,...,T-1

$$\alpha_{t+1}(i) = \left(\sum_{j=1}^{n} \alpha_t(j) a_{ji} \right) b_{iq_{t+1}}$$

- 最终:

$$P(Q;\lambda) = \sum_{i=1}^{n} \alpha_T(i) \beta_T(i) = \sum_{i=1}^{n} \alpha_T(i)$$

HMM案例-前向算法

- 在给定参数π、A、B的时候，得到观测序列为“白黑白白黑”的概率是多少??

$$\pi = \begin{pmatrix} 0.2 \\ 0.5 \\ 0.3 \end{pmatrix} \qquad A = \begin{bmatrix} 0.5 & 0.4 & 0.1 \\ 0.2 & 0.2 & 0.6 \\ 0.2 & 0.5 & 0.3 \end{bmatrix} \qquad B = \begin{bmatrix} 0.4 & 0.6 \\ 0.8 & 0.2 \\ 0.5 & 0.5 \end{bmatrix}$$

HMM案例-前向算法

$$\pi=\begin{pmatrix}0.2\\0.5\\0.3\end{pmatrix}\qquad A=\begin{bmatrix}0.5&0.4&0.1\\0.2&0.2&0.6\\0.2&0.5&0.3\end{bmatrix}\qquad B=\begin{bmatrix}0.4&0.6\\0.8&0.2\\0.5&0.5\end{bmatrix}$$

$$\alpha_1(1)=\pi_1 b_{1q_1}=0.08$$

$$\alpha_1(2)=\pi_2 b_{2q_1}=0.4$$

$$\alpha_1(3)=\pi_3 b_{3q_1}=0.15$$

HMM案例-前向算法

$$\pi=\begin{pmatrix}0.2\\0.5\\0.3\end{pmatrix}\quad A=\begin{bmatrix}0.5&0.4&0.1\\0.2&0.2&0.6\\0.2&0.5&0.3\end{bmatrix}\quad B=\begin{bmatrix}0.4&0.6\\0.8&0.2\\0.5&0.5\end{bmatrix}$$

$$\alpha_1(1)=\pi_1 b_{1q_1}=0.08$$

$$\alpha_1(2)=\pi_2 b_{2q_1}=0.4$$

$$\alpha_1(3)=\pi_3 b_{3q_1}=0.15$$

$$\alpha_2(1)=\left(\sum_{j=1}^{3}\alpha_1(j)a_{j1}\right)b_{1q_2}$$
$$=(0.08*0.5+0.4*0.2+0.15*0.2)*0.6$$
$$=0.09$$

$$\alpha_2(2)=0.0374$$

$$\alpha_2(3)=0.1465$$

$$\alpha_3(1)=0.032712\qquad \alpha_4(1)=0.01702872\qquad \alpha_5(1)=0.0142036956$$

$$\alpha_3(2)=0.093384\qquad \alpha_4(2)=0.04048728\qquad \alpha_5(2)=0.0065122938$$

$$\alpha_3(3)=0.037695\qquad \alpha_4(3)=0.03530505\qquad \alpha_5(3)=0.0182933775$$

HMM案例-前向算法

$$\alpha_5(1) = 0.0142036956$$

$$\alpha_5(2) = 0.0065122938$$

$$\alpha_5(3) = 0.0182933775$$

$$p(Q;\lambda) = \sum_{i=1}^{3} \alpha_5(i)$$

$$= 0.0142036956 + 0.0065122938 + 0.0182933775$$

$$= 0.0390093669$$

后向算法

- 定义：给定λ，定义到时刻t状态为s_i的前提下，从t+1到T部分观测序列为$q_{t+1},q_{t+2},\ldots,q_T$的概率为**后向概率**。记做：

$$\beta_t(i) = p\left(q_{t+1}, q_{t+2}, \ldots, q_T \middle| i_t = s_i; \lambda\right)$$

后向算法

$$
\begin{aligned}
\beta_t(i) &= p\left(q_{t+1}, q_{t+2}, \ldots, q_T \middle| i_t = s_i\right) \\
&= \sum_{j=1}^{n} p\left(i_{t+1} = s_j, q_{t+1}, q_{t+2}, \ldots, q_T \middle| i_t = s_i\right) \\
&= \sum_{j=1}^{n} p\left(q_{t+1}, q_{t+2}, \ldots, q_T \middle| i_{t+1} = s_j\right) p\left(i_{t+1} = s_j \middle| i_t = s_i\right) \\
&= \sum_{j=1}^{n} p\left(q_{t+2}, \ldots, q_T \middle| i_{t+1} = s_j\right) p\left(q_{t+1} \middle| i_{t+1} = s_j\right) p\left(i_{t+1} = s_j \middle| i_t = s_i\right) \\
&= \sum_{j=1}^{n} \left(a_{ij} b_{jq_{t+1}} \beta_{t+1}(j)\right)
\end{aligned}
$$

后向算法

$$\beta_t(i) = p(q_{t+1}, q_{t+2}, \ldots, q_T | i_t = s_i; \lambda)$$

- 初值： $\beta_T(i) = 1$

- 递推：对于t=T-1,T-2,...,1

$$\beta_t(i) = \sum_{j=1}^{n}\left(a_{ij} b_{jq_{t+1}} \beta_{t+1}(j)\right)$$

- 最终：

$$P(Q;\lambda) = \sum_{i=1}^{n} \alpha_1(i)\beta_1(i) = \sum_{i=1}^{n} \pi_i b_{iq_1} \beta_1(i)$$

单个状态的概率

- 求给定模型λ和观测序列Q的情况下，在时刻t处于状态s_i的概率，记做：

$$\gamma_t(i) = p(i_t = s_i | Q; \lambda)$$

- 单个状态概率的意义主要是用于判断在每个时刻最可能存在的状态，从而可以得到一个状态序列作为最终的预测结果。

单个状态的概率

$$p\left(i_t = s_i, Q; \lambda\right) = \alpha_t(i)\beta_t(i)$$

$$\gamma_t(i) = p\left(i_t = s_i \middle| Q; \lambda\right) = \frac{p\left(i_t = s_i, Q; \lambda\right)}{p(Q; \lambda)}$$

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(Q; \lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{n} \alpha_t(j)\beta_t(j)}$$

两个状态的联合概率

- 求给定模型λ和观测序列Q的情况下，在时刻t处于状态s_i并时刻t+1处于状态s_j概率，记做：

$$\xi_t(i,j) = p(i_t = s_i, i_{t+1} = s_j \mid Q; \lambda)$$

两个状态的联合概率

$$\xi_t(i,j) = p\left(i_t = s_i, i_{t+1} = s_j \middle| Q; \lambda\right)$$

$$= \frac{p\left(i_t = s_i, i_{t+1} = s_j, Q; \lambda\right)}{p(Q; \lambda)}$$

$$= \frac{p\left(i_t = s_i, i_{t+1} = s_j, Q; \lambda\right)}{\sum_{i=1}^{n}\sum_{j=1}^{n} p\left(i_t = s_i, i_{t+1} = s_j, Q; \lambda\right)}$$

$$p\left(i_t = s_i, i_{t+1} = s_j, Q; \lambda\right) = \alpha_t(i) a_{ij} b_{j q_{t+1}} \beta_{t+1}(j)$$

学习问题

- 若训练数据包含观测序列和状态序列，则HMM的学习问题非常简单，是监督学习算法。
- 若训练数据只包含观测序列，则HMM的学习问题需要使用EM算法求解，是无监督学习算法。

学习问题_监督学习

- 直接利用大数定理的结论“频率的极限是概率”，直接给出HMM的参数估计；
  - 备注：
    - St表示所有序列数目。Sti当在所有序列中，时刻t对应状态为i的样本序列数目；Stij表示时刻t为状态i，时刻t+1为状态j的的样本序列数目；qtij表示t时刻对应状态为i，时刻t对应的观测值为j

$$\hat{\pi}_i = \frac{\left|s_{t=1,i}\right|}{\left|s_{t=1}\right|} \qquad \hat{a}_{ij} = \frac{\sum_{t=1}^{T-1}\left|s_{tij}\right|}{\sum_{t=1}^{T-1}\left|s_{ti}\right|} \qquad \hat{b}_{ij} = \frac{\sum_{t}\left|q_{tij}\right|}{\sum_{t}\left|s_{ti}\right|}$$

学习问题_非监督学习

- 若训练数据中只有观测序列，则HMM的学习问题需要使用EM算法，属于非监督算法；此时一般使用Baum-Welch算法。
- 所有的观测数据为$Q=\{q_1,q_2,\ldots,q_T\}$，所有的隐状态为$I=\{i_1,i_2,\ldots,i_T\}$，则完整的数据为(O,I)，完整数据的对数似然函数为$\ln(p(Q,I;\lambda))$；然后直接使用EM算法的方式来进行参数估计。

Baum-Welch算法

$$p(O,I;\lambda)=\pi_{i_1}b_{i_1q_1}a_{i_1i_2}b_{i_2q_2}\ldots a_{i_{T-1}i_T}b_{i_Tq_T}$$

$$\begin{aligned}L(\lambda,\ \overline{\lambda})&=\sum_I \ln\left(p(Q,I;\lambda)\right)p\left(I\middle|Q;\overline{\lambda}\right)\\&=\sum_I \ln\left(p(Q,I;\lambda)\right)\frac{p\left(I,Q;\overline{\lambda}\right)}{p\left(Q;\overline{\lambda}\right)}\\&\propto\sum_I \ln\left(p(Q,I;\lambda)\right)p\left(I,Q;\overline{\lambda}\right)\end{aligned}$$

$$L(\lambda,\overline{\lambda})=\sum_I \ln\left(\pi_{i_1}\right)p\left(I,Q;\overline{\lambda}\right)+\sum_I\left(\sum_{t=1}^{T-1}\ln a_{i_ti_{t+1}}\right)p\left(I,Q;\overline{\lambda}\right)+\sum_I\left(\sum_{t=1}^{T}\ln b_{i_tq_t}\right)p\left(I,Q;\overline{\lambda}\right)$$

Baum-Welch算法

π求解

$$\gamma_t(i)=\frac{\alpha_t(i)\beta_t(i)}{P(Q;\lambda)}=\frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{n}\alpha_t(j)\beta_t(j)}$$

- 极大化L，使用拉格朗日乘子法，求解π的值：

$$\sum_{I}\ln\left(\pi_{I_1}\right)p\left(I,Q;\overline{\lambda}\right)=\sum_{i=1}^{n}\ln\left(\pi_i\right)p\left(Q,i_1=i;\overline{\lambda}\right) \qquad \sum_{i=1}^{n}\ln\left(\pi_i\right)p\left(Q,i_1=i;\overline{\lambda}\right)+\beta\left(\sum_{i=1}^{n}\pi_i-1\right)$$

$$p\left(Q,i_1=i;\overline{\lambda}\right)+\beta\pi_i=0 \qquad \beta=-p\left(Q;\overline{\lambda}\right)$$

$$\pi_i=\frac{p\left(Q,i_1=i;\overline{\lambda}\right)}{p\left(Q;\overline{\lambda}\right)}=p\left(i_1=i\middle|Q;\overline{\lambda}\right)=\gamma_1(i)$$

Baum-Welch算法

A求解

- 极大化L，使用拉格朗日乘子法，求解a_{ij}的值：

$$\sum_{I}\left(\sum_{t=1}^{T-1}\ln a_{i_t i_{t+1}}\right)p(I,Q;\overline{\lambda})=\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T-1}\ln a_{ij}\,p(Q,i_t=i,i_{t+1}=j;\overline{\lambda})$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T-1}\ln a_{ij}\,p(Q,i_t=i,i_{t+1}=j;\overline{\lambda})+\beta\left(\sum_{i=1}^{n}\sum_{j=1}^{n}a_{ij}-n\right)$$

$$\sum_{t=1}^{T-1}p(Q,i_t=i,i_{t+1}=j;\overline{\lambda})+\beta a_{ij}=0 \qquad \beta=-\sum_{t=1}^{T-1}p(Q,i_t=i;\overline{\lambda})$$

$$a_{ij}=\frac{\sum_{t=1}^{T-1}p(Q,i_t=i,i_{t+1}=j;\overline{\lambda})}{\sum_{t=1}^{T-1}p(Q,i_t=i;\overline{\lambda})}=\frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)}$$

Baum-Welch算法

B求解

- 极大化L，使用拉格朗日乘子法，求解b_{ij}的值：

$$\sum_{I}\left(\sum_{t=1}^{T}\ln b_{i_t q_t}\right)p\left(I,Q;\bar{\lambda}\right)=\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{T}\ln b_{ij}\,p\left(Q,i_t=i,q_t=j;\bar{\lambda}\right)$$

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{T}\ln b_{ij}\,p\left(Q,i_t=i,q_t=j;\bar{\lambda}\right)+\beta\left(\sum_{i=1}^{n}\sum_{j=1}^{n}b_{ij}-n\right)$$

$$\sum_{t=1}^{T}p\left(Q,i_t=i,q_t=j;\bar{\lambda}\right)+\beta b_{ij}=0 \qquad \beta=-\sum_{t=1}^{T}p\left(Q,i_t=i;\bar{\lambda}\right)$$

$$b_{ij}=\frac{\sum_{t=1}^{T}p\left(Q,i_t=i,q_t=j;\bar{\lambda}\right)}{\sum_{t=1}^{T}p\left(Q,i_t=i;\bar{\lambda}\right)}=\frac{\sum_{t=1,q_t=j}^{T}p\left(Q,i_t=i;\bar{\lambda}\right)}{\sum_{t=1}^{T}p\left(Q,i_t=i;\bar{\lambda}\right)}=\frac{\sum_{t=1,q_t=j}^{T}\gamma_t(i)}{\sum_{t=1}^{T}\gamma_t(i)}$$

Baum-Welch算法

- 极大化L函数，分别可以求得π、a、b的值。

$$\pi_i = \gamma_1(i) \qquad a_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \qquad b_{ij} = \frac{\sum_{t=1, q_t = o_j}^{T} \gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(i)}$$

预测问题

- 近似算法
- Viterbi算法

近似算法

- 直接在每个时刻t时候最优可能的状态作为最终的预测状态，使用下列公式计算概率值：

$$\gamma_t(i)=\frac{\alpha_t(i)\beta_t(i)}{P(Q;\lambda)}=\frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{n}\alpha_t(j)\beta_t(j)}$$

Viterbi算法

- Viterbi算法实际是用动态规划的思路求解HMM预测问题，求出概率最大的“路径”，每条“路径”对应一个状态序列。

$$\delta_t(i) = \max_{i_1, i_2, \ldots, i_{t-1}} p(i_t = i, i_1, i_2, \ldots, i_{t-1}, q_t, q_{t-1}, \ldots, q_1; \lambda)$$

$$\delta_1(i) = \pi_i b_{iq_1} \quad \delta_{t+1}(i) = \max_{1 \le j \le n} \left(\delta_t(j) a_{ji} \right) b_{iq_{t+1}}$$

$$P^* = \max_{1 \le i \le n} \delta_T(i)$$

HMM案例-Viterbi

- 在给定参数π、A、B的时候，得到观测序列为“白黑白白黑”，求出最优的隐藏状态序列。

$$\pi = \begin{pmatrix} 0.2 \\ 0.5 \\ 0.3 \end{pmatrix} \qquad A = \begin{bmatrix} 0.5 & 0.4 & 0.1 \\ 0.2 & 0.2 & 0.6 \\ 0.2 & 0.5 & 0.3 \end{bmatrix} \qquad B = \begin{bmatrix} 0.4 & 0.6 \\ 0.8 & 0.2 \\ 0.5 & 0.5 \end{bmatrix}$$

HMM案例-Viterbi

$$\delta_1(i) = \pi_i b_{iq_1} = \pi_i b_{i白}$$

$$\delta_1(1) = 0.08$$
$$\delta_1(2) = 0.4$$
$$\delta_1(3) = 0.15$$

$$\delta_2(i) = \max_{1 \le j \le 3}\left(\delta_1(j) a_{ji}\right) b_{iq_2} = \max_{1 \le j \le 3}\left(\delta_1(j) a_{ji}\right) b_{i黑}$$

$$\delta_2(1) = \max\{0.08*0.5, 0.4*0.2, 0.15*0.2\}*0.6 = 0.048$$
$$\delta_2(2) = \max\{0.08*0.4, 0.4*0.2, 0.15*0.5\}*0.2 = 0.016$$
$$\delta_2(3) = \max\{0.08*0.1, \boxed{0.4*0.6}, 0.15*0.3\}*0.5 = \boxed{0.12}$$

HMM案例-Viterbi

$$\delta_1(1)=0.08 \qquad \delta_2(1)=0.048$$
$$\delta_1(2)=0.4 \qquad \delta_2(2)=0.016$$
$$\delta_1(3)=0.15 \qquad \delta_2(3)=0.12$$

$$\delta_3(i)=\max_{1\le j\le 3}\left(\delta_2(j)a_{ji}\right)b_{iq_3}=\max_{1\le j\le 3}\left(\delta_2(j)a_{ji}\right)b_{i白}$$

$$\delta_3(1)=\max\{0.048*0.5,0.016*0.2,0.12*0.2\}*0.4=0.024*0.4=0.0096$$
$$\delta_3(2)=\max\{0.048*0.4,0.016*0.2,\boxed{0.12*0.5}\}*0.8=0.06*0.8=\boxed{0.048}$$
$$\delta_3(3)=\max\{0.048*0.1,0.016*0.6,0.12*0.3\}*0.5=0.036*0.5=0.018$$

HMM案例-Viterbi

$$\delta_1(1)=0.08 \qquad \delta_2(1)=0.048 \qquad \delta_3(1)=0.0096$$
$$\delta_1(2)=0.4 \qquad \delta_2(2)=0.016 \qquad \delta_3(2)=0.048$$
$$\delta_1(3)=0.15 \qquad \delta_2(3)=0.12 \qquad \delta_3(3)=0.018$$

$$\delta_4(i)=\max_{1\le j\le 3}\left(\delta_3(j)a_{ji}\right)b_{iq_4}=\max_{1\le j\le 3}\left(\delta_3(j)a_{ji}\right)b_{i白}$$

$$\delta_4(1)=\max\{0.0096*0.5,0.048*0.2,0.018*0.2\}*0.4=0.0096*0.4=0.0096$$
$$\delta_4(2)=\max\{0.0096*0.4,\boxed{0.048*0.2},0.018*0.5\}*0.8=0.0096*0.8=\boxed{0.00768}$$
$$\delta_4(3)=\max\{0.0096*0.1,0.048*0.6,0.018*0.3\}*0.5=0.0288*0.5=0.0144$$

HMM案例-Viterbi

$\delta_1(1)=0.08$ $\delta_2(1)=0.048$ $\delta_3(1)=0.0096$ $\delta_4(1)=0.00384$

$\delta_1(2)=0.4$ $\delta_2(2)=0.016$ $\delta_3(2)=0.048$ $\delta_4(2)=0.00768$

$\delta_1(3)=0.15$ $\delta_2(3)=0.12$ $\delta_3(3)=0.018$ $\delta_4(3)=0.0144$

$$\delta_5(i)=\max_{1\le j\le 3}\left(\delta_4(j)a_{ji}\right)b_{iq_5}=\max_{1\le j\le 3}\left(\delta_4(j)a_{ji}\right)b_{i黑}$$

$$\delta_5(1)=\max\{0.00384*0.5,0.00768*0.2,0.0144*0.2\}*0.6=0.00288*0.6=0.001728$$

$$\delta_5(2)=\max\{0.00384*0.4,0.00768*0.2,0.0144*0.5\}*0.2=0.0072*0.2=0.00144$$

$$\delta_5(3)=\max\{0.00384*0.1,\boxed{0.00768*0.6},0.0144*0.3\}*0.5=0.004608*0.5=\boxed{0.002304}$$

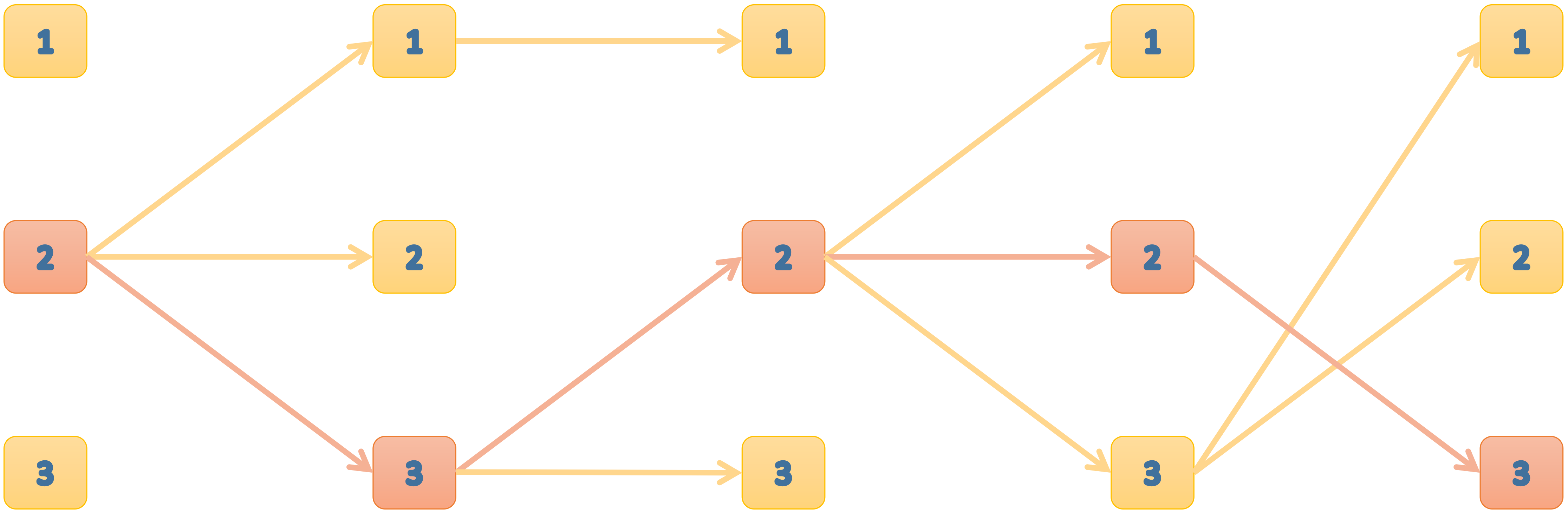
HMM案例-Viterbi

$$\delta_1(1)=0.08 \qquad \delta_2(1)=0.048$$
$$\delta_1(2)=0.4 \qquad \delta_2(2)=0.016$$
$$\delta_1(3)=0.15 \qquad \delta_2(3)=0.12$$

$$\delta_3(1)=0.0096 \qquad \delta_4(1)=0.00384 \qquad \delta_5(1)=0.001728$$
$$\delta_3(2)=0.048 \qquad \delta_4(2)=0.00768 \qquad \delta_5(2)=0.00144$$
$$\delta_3(3)=0.018 \qquad \delta_4(3)=0.0144 \qquad \delta_5(3)=0.002304$$

- 最终盒子序列为: (2, 3, 2, 2, 3)， 备注：盒子序号从1开始

HMM案例-Viterbi



HMM应用

- 安装pip install hmmlearn或者conda install -c omnia hmmlearn

```
C:\Users\ibf>pip install hmmlearn
Collecting hmmlearn
  Downloading https://files.pythonhosted.org/packages/94/1d/af4071cbe561f62d744946211414beb22ea7c72d4bb2f42636b6820b5ae5
/hmmlearn-0.2.1.tar.gz (150kB)
    100% |████████████████████████████████| 153kB 76kB/s
Requirement already satisfied: numpy in c:\anaconda3\lib\site-packages (from hmmlearn) (1.15.4)
Requirement already satisfied: scikit-learn>=0.16 in c:\anaconda3\lib\site-packages (from hmmlearn) (0.20.1)
Requirement already satisfied: scipy>=0.13.3 in c:\anaconda3\lib\site-packages (from scikit-learn>=0.16->hmmlearn) (1.1.
0)
Building wheels for collected packages: hmmlearn
  Running setup.py bdist_wheel for hmmlearn ... done
  Stored in directory: C:\Users\ibf\AppData\Local\pip\Cache\wheels\9a\a4\ee\917f0de81626b684fd2139ef5df47744c35ebeacc9e9
50487b
Successfully built hmmlearn
Installing collected packages: hmmlearn
Successfully installed hmmlearn-0.2.1
You are using pip version 10.0.1, however version 18.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
```

```
C:\Users\ibf>python
Python 3.6.6 |Anaconda 4.3.1 (64-
Type "help", "copyright", "credit
>>> import hmmlearn
>>> hmmlearn.__version__
'0.2.1'
```

- HMM的常见应用主要用于进行特征提取的场景中或者数据标注的场景中；