

一、隐马尔可夫模型 00:25

1. 切比雪夫 04:30

- 师生关系：马尔可夫是著名数学家切比雪夫的学生，切比雪夫提出了大数定律和切比雪夫不等式等重要统计学理论。

2. 马尔可夫性 07:14

1) 马尔可夫性的定义 07:18

- 核心概念：随机过程 X_t 中， S_{t+1} 的状态仅依赖于 S_t ，而与 S_1 到 S_{t-1} 无关，即 $P(S_{t+1}|S_t, S_{t-1}, \dots, S_1) = P(S_{t+1}|S_t)$
- 数学表示：满足该性质的随机变量序列 $S_1, S_2, \dots$ 称为马尔可夫链(MC)

2) 马尔可夫性的通俗解释 09:10

- 时间关系：未来状态只与当前状态有关，与过去状态无关
- 生活启示：该性质可形象理解为"要想放眼未来，就要珍惜当下"

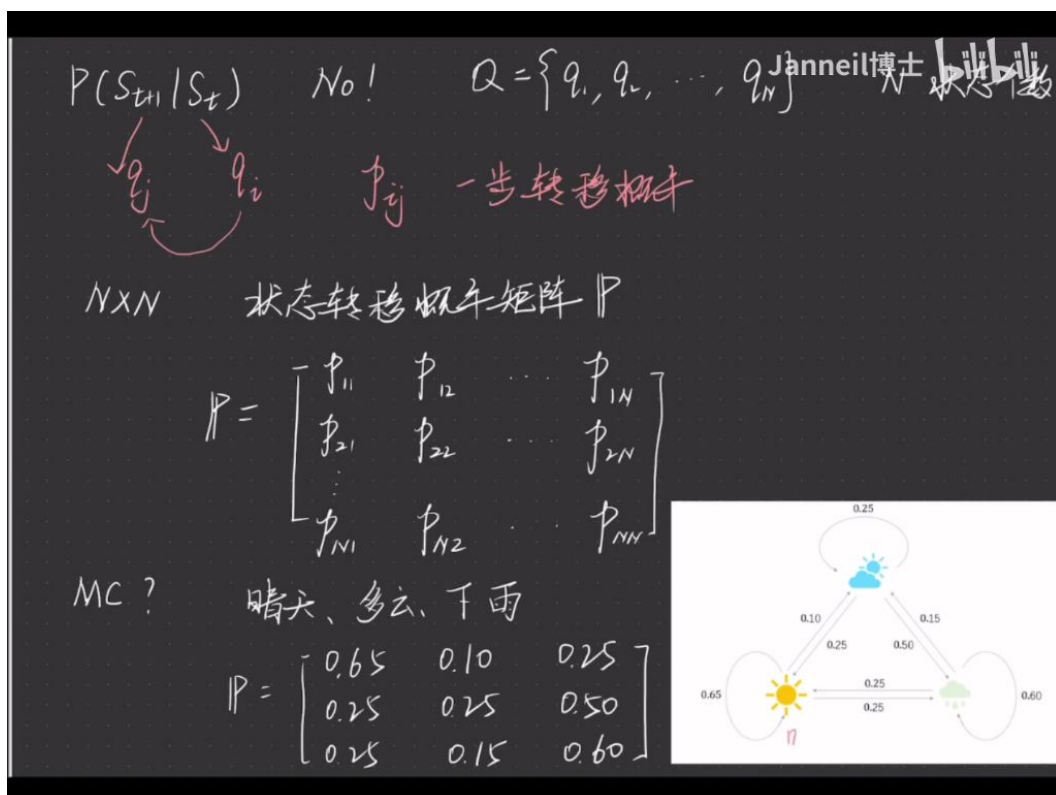
3) 状态空间与一步转移概率 11:35

- 状态空间：类似随机变量的样本空间，包含所有可能状态 $Q = \{q_1, q_2, \dots, q_N\}$
- 转移概率：从状态 q_i 转移到 q_j 的概率记为 p_{ij} ，称为一步转移概率

4) 状态转移概率矩阵 13:38

- 矩阵形式：所有转移概率组成的 $N \times N$ 矩阵 P ，其中 p_{ij} 表示从 q_i 到 q_j 的转移概率
- 性质要求：
 - 每行元素和为1： $\sum_{j=1}^N p_{ij} = 1$
 - 每个元素 $0 \leq p_{ij} \leq 1$

5) 例题1:天气状态转移概率矩阵 15:40



- 状态设置：晴天、多云、下雨三种天气状态

转移矩阵：

$$\begin{bmatrix}
 0.65 & 0.10 & 0.25 \\
 0.25 & 0.25 & 0.50 \\
 0.25 & 0.15 & 0.60
 \end{bmatrix}$$

- 矩阵解读：

- 对角线元素表示保持原状态的概率
- 非对角线元素表示状态间转换概率
- 每行概率和为1，符合概率分布要求

6) 初始状态分布向量 20:41

- 定义： $\pi = (\pi_1, \pi_2, \dots, \pi_N)$ 表示初始时刻各状态的概率分布

- 性质要求：

- $\sum_{i=1}^N \pi_i = 1$
- $0 \leq \pi_i \leq 1$

7) 马尔可夫链两大要素 21:55

- 构成要素：

- 状态转移概率矩阵P
- 初始状态分布向量 π

- 决定原理：已知当前状态和转移矩阵即可确定下一时刻状态

8) 例题2:天气预报的马尔可夫链原理 22:45

初始状态分布 $\pi = (\pi_1, \pi_2, \dots, \pi_N)^T$ Janneil博士 bilibili

$$\sum_{i=1}^N \pi_i = 1, \quad 0 \leq \pi_i \leq 1$$

马尔可夫链 $\left\{ \begin{array}{l} \text{初始状态分布 } \pi \\ \text{状态转移概率矩阵 } P \end{array} \right.$

晴天、多云、下雨

$$P = \begin{bmatrix} 0.65 & 0.10 & 0.25 \\ 0.25 & 0.25 & 0.50 \\ 0.25 & 0.15 & 0.60 \end{bmatrix}$$

- 初始条件： $\pi = (0.9, 0.05, 0.05)$ 表示当前大概率是晴天

- 预测方法：通过连续乘以转移矩阵P预测未来天气

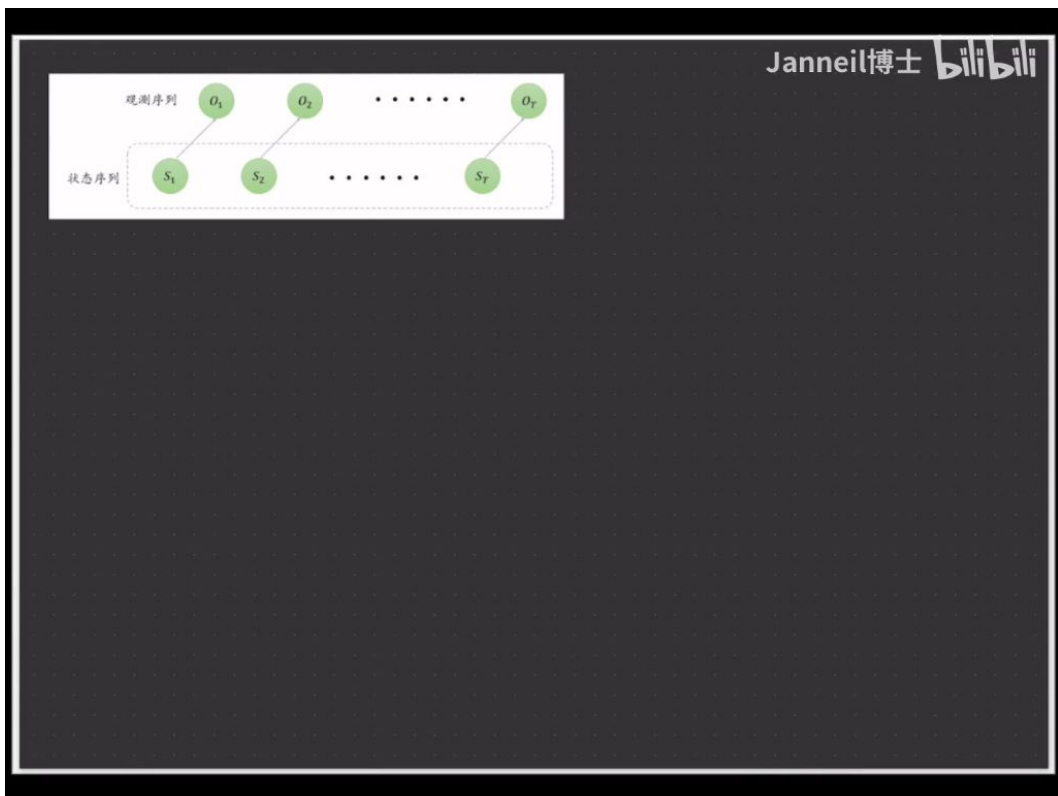
- 1天后： P
- 2天后： P^2

- 3天后: P^3
 - 计算结果: 三天后天气概率为(0.43,0.38,0.38), 仍以晴天概率最大但需带伞防范
3. 马尔可夫模型 24:40

- 要素扩展: 从马尔可夫链的两个要素(初始分布和转移矩阵)扩展到三个要素
- 隐藏特性: 引入观测序列的概念, 状态序列变为隐藏的

4. 隐马尔可夫模型 25:16

1) 初始状态概率分布 26:00



- 状态空间定义: 记作 Q , 包含 N 个状态 $\{q_1, q_2, \dots, q_N\}$
- 分布元素: $\pi = \{\pi_1, \pi_2, \dots, \pi_N\}$, 其中 π_i 表示初始处于第 i 个状态的概率
- 数学表示: $\pi_i = P(I_1 = q_i)$, I_1 代表初始时刻的状态变量

2) 状态转移概率矩阵 26:28

- 状态序列表示 27:28
 - 序列符号: $I = \{I_1, I_2, \dots, I_T\}$, I_t 表示第 t 时刻的状态
 - 下标说明: 字母 I 表示状态变量 (替代 S 符号), 数字下标表示时刻序号
- 矩阵定义与结构 30:07
 - 矩阵阶数: $N \times N$ 方阵, 记为 A

元素排列:

$$\begin{bmatrix} a_{11} & a_{12} & \cdots \\ \vdots & \vdots & \ddots \\ a_{N1} & a_{N2} & \cdots \end{bmatrix}$$

- 元素含义 30:55
 - 转移概率: $a_{ij} = P(I_{t+1} = q_j | I_t = q_i)$
 - 双重下标: 第一个下标 i 表示当前状态, 第二个下标 j 表示转移目标状态
- 约束条件 32:06

- 行和归一: $\sum_{j=1}^N a_{ij} = 1$ (每行构成概率分布)
- 值域限制: $\forall a_{ij} \in [0,1]$

3) 观测概率矩阵 33:04

● 观测空间定义 33:05

- 符号表示: $V = \{v_1, v_2, \dots, v_M\}$, 包含M种可能观测值
- 观测序列: $O = \{O_1, O_2, \dots, O_T\}$, O_t 表示第t时刻的观测

● 矩阵结构与含义 34:00

- 矩阵阶数: $N \times M$ 矩阵, 记为B
- 元素定义: $b_j(k) = P(O_t = v_k | I_t = q_j)$
- 特殊说明: 表示在状态 q_j 时观测到 v_k 的概率

● 矩阵约束条件 35:50

- 行和归一: $\sum_{k=1}^M b_j(k) = 1$ (每行构成概率分布)
- 值域限制: $\forall b_j(k) \in [0,1]$

● 三要素总结 38:03

- 完整定义: 隐马尔可夫模型由 (π, A, B) 三元组确定
 - π : 初始状态概率分布
 - A: 状态转移概率矩阵
 - B: 观测概率矩阵

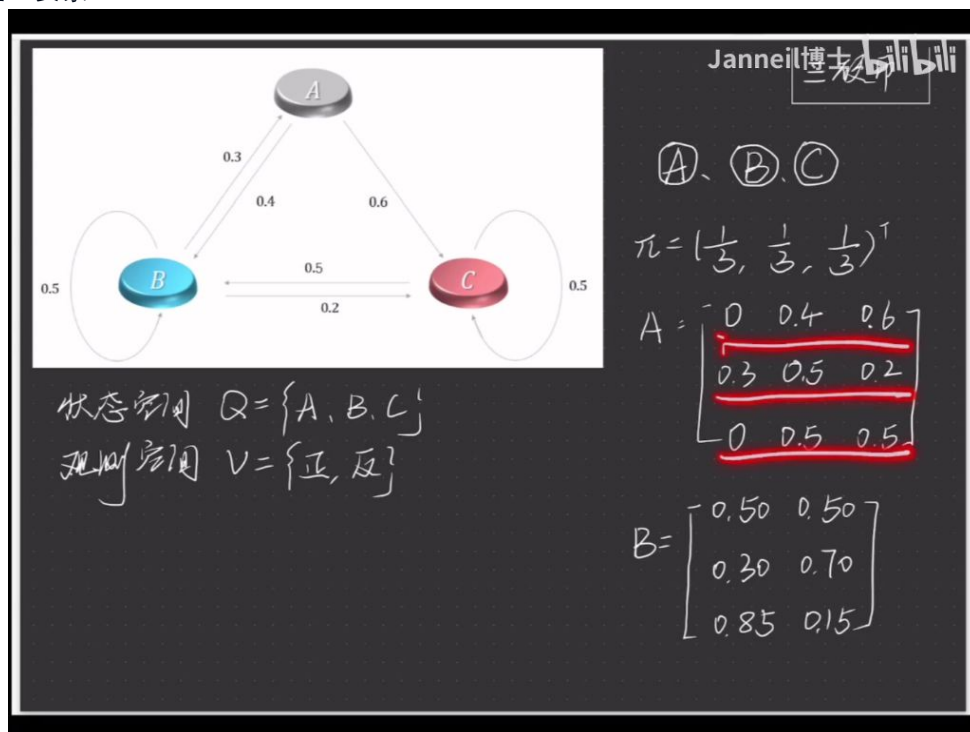
核心特征: 状态序列不可观测 (隐藏), 只能观测到依赖状态的输出序列

- 注: 所有时间戳均根据课程记录中的ppt_name和讲解进度标注, 图片插入位置与对应知识点严格匹配。符号表示采用LaTeX格式, 如状态转移概率 a_{ij} 写作 a_{ij} 。

4) 应用案例 38:38

● 例题:三硬币模型观测序列生成

- 模型三要素



- 初始状态分布: $\pi = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$, 表示等概率从A、B、C三枚硬币中随机选取
- 状态空间: (圆圈标记用于区分矩阵符号)

- 观测空间: $V = \{\text{正}, \text{反}\}$
- 转移概率矩阵

矩阵结构:

- $a = \begin{bmatrix} 0 & 0.4 & 0.6 \\ 0.3 & 0.5 & 0.2 \\ 0 & 0.5 & 0.5 \end{bmatrix}$

- 行规范: 每行概率和为1
- 具体转移规则:
 - $A \rightarrow B: 0.4, A \rightarrow C: 0.6$ (无自转移)
 - $B \rightarrow A: 0.3, B \text{自持}: 0.5, B \rightarrow C: 0.2$
 - $C \rightarrow B: 0.5, C \text{自持}: 0.5$ (无A转移)

◆ 观测概率矩阵

矩阵结构:

- $b = \begin{bmatrix} 0.5 & 0.5 \\ 0.3 & 0.7 \\ 0.85 & 0.15 \end{bmatrix}$

- 具体规则:
 - A硬币: 正反均匀 (0.5/0.5)
 - B硬币: 偏反 (正面0.3, 反面0.7)
 - C硬币: 偏正 (正面0.85, 反面0.15)

■ 单次观测概率计算

- 问题: 求第一次观测为正面的概率 $P(O_1 = \text{正})$

计算方法: 全概率公式分解

- $P(O_1 = \text{正}) = \sum_{i \in \{A, B, C\}} P(i_1) P(O_1 = \text{正} | i_1)$

■ 具体计算:

- $P(A) \times 0.5 = \frac{1}{3} \times 0.5$
- $P(B) \times 0.3 = \frac{1}{3} \times 0.3$
- $P(C) \times 0.85 = \frac{1}{3} \times 0.85$

- 结果: $0.1667 + 0.1 + 0.2833 \approx 0.55$

■ 连续观测概率计算

Janneil博士 bilibili

Handwritten notes on a chalkboard showing HMM calculations. It includes a state transition diagram with states A, B, C and transitions with probabilities. Calculations for forward and backward probabilities are shown, including the formula $P(O) = \sum_{i_1, i_2, \dots, i_T} \pi(i_1) \prod_{t=1}^{T-1} a_{i_t i_{t+1}} b_{i_{t+1}}(o_{t+1})$. The final result for $P(O)$ is 0.163.

- 正向计算：按时间顺序展开所有可能路径（绿色/紫色/橙色箭头）
- 逆向计算：可考虑从终态反推（ABC汇聚计算）
- 计算优化：后续将介绍更高效的前向-后向算法

5) 观测序列生成算法 01:01:22

Janneil博士 bilibili

Handwritten notes on a chalkboard showing HMM calculations and a diagram of the generation process. The diagram shows three states A, B, C at time 1 and time 2, with arrows indicating transitions between them. The calculations for the forward and backward probabilities are repeated, showing the same result for $P(O) = 0.163$.

- 生成过程：根据隐马尔可夫模型 $\lambda = (A, B, \pi)$ 生成观测序列 $O = (o_1, o_2, \dots, o_T)$ 的步骤如下：
 - 按初始状态分布 π 产生状态 i_1
 - 令 $t = 1$

- 按状态 i_t 的观测概率分布 $b_{i_t}(k)$ 生成 o_t
- 按状态转移概率分布 $a_{i_t, i_{t+1}}$ 产生状态 i_{t+1}
- 令 $t = t + 1$, 若 $t < T$ 则转步骤3, 否则终止
- 核心要素:
 - 初始分布: 决定第一个隐藏状态的选取概率
 - 状态转移: 通过矩阵A控制状态间的转移规律
 - 观测概率: 矩阵B决定从隐藏状态生成观测值的概率

6) 应用案例 01:04:14

- 例题: 盒子和球模型观测序列生成

例 10.1 (盒子和球模型) 假设有 4 个盒子, 每个盒子里都装有红、白两种颜色的球, 盒子里的红、白球数由表 10.1 列出。

	盒 子			
	1	2	3	4
红球数	5	3	6	8
白球数	5	7	4	2

按照下面的方法抽球, 产生一个球的颜色的观测序列:

- 开始, 从 4 个盒子里以等概率随机选取 1 个盒子, 从这个盒子里随机抽出 1 个球, 记录其颜色后, 放回;
- 然后, 从当前盒子随机转移到下一个盒子, 规则是: 如果当前盒子是盒子 1, 那么下一盒子一定是盒子 2; 如果当前是盒子 2 或 3, 那么分别以概率 0.4 和 0.6 转移到左边或右边的盒子; 如果当前是盒子 4, 那么各以 0.5 的概率停留在盒子 4 或转移到盒子 3;
- 确定转移的盒子后, 再从这个盒子里随机抽出 1 个球, 记录其颜色, 放回;
- 如此下去, 重复进行 5 次, 得到一个球的颜色的观测序列:

$O = (\text{红}, \text{红}, \text{白}, \text{白}, \text{红})$

Janneil博士 bilibili

-
- 模型设定:
 - 状态集合: $Q = \{\text{盒子1}, \text{盒子2}, \text{盒子3}, \text{盒子4}\}$, $N = 4$
 - 观测集合: $V = \{\text{红}, \text{白}\}$, $M = 2$
 - 初始分布: $\pi = [0.25, 0.25, 0.25, 0.25]$

转移矩阵:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0.4 & 0 & 0.6 & 0 \\ 0 & 0.4 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 \end{bmatrix}$$

观测矩阵:

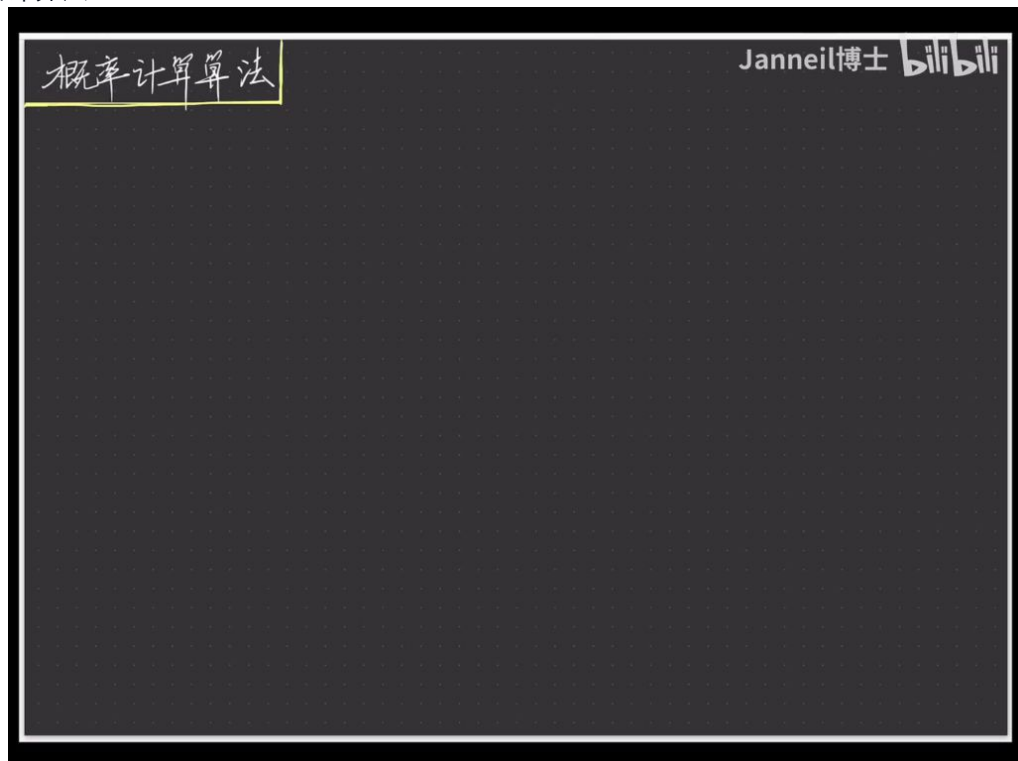
$$B = \begin{bmatrix} 0.5 & 0.5 \\ 0.3 & 0.7 \\ 0.6 & 0.4 \\ 0.8 & 0.2 \end{bmatrix}$$

- 序列生成:
 - 随机选择初始盒子 (等概率1/4)
 - 根据当前盒子颜色分布抽取球 (如盒子1红球概率0.5)
 - 按转移规则确定下一个盒子

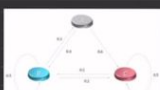
- 重复5次得到观测序列如 $O = (\text{红}, \text{红}, \text{白}, \text{白}, \text{红})$
- 关键区别：
 - 相比三硬币模型，盒子数量增加到4个
 - 转移规则更复杂（如盒子4有0.5自环概率）
 - 观测概率计算方式相同（红球数/总数）
- 隐马尔可夫模型三大问题
 - 概率计算：
 - 直接算法：列举所有可能状态序列组合（计算复杂度高）
 - 前向算法：按时间顺序递推计算概率
 - 后向算法：逆向递推计算概率
 - 学习问题：通过观测序列估计模型参数 $\lambda = (A, B, \pi)$
 - 预测问题：根据模型和观测序列预测未来状态
 - 算法选择：
 - 前向/后向算法比直接计算更高效
 - 前向算法适合实时计算
 - 后向算法常与前向算法结合使用

7) 概率计算算法 01:12:17

- 直接算法 01:13:51



-
- 隐马尔可夫模型的三要素 01:13:56
 - 模型组成：隐马尔可夫模型 λ 由三要素组成：
 - π ：初始状态概率分布
 - A ：状态转移概率矩阵
 - B ：观测概率矩阵
 - 参数表示：模型参数记作 $\lambda = (\pi, A, B)$
- 观测序列的生成过程 01:16:18



硬币例子: $\pi = \{\frac{1}{2}, \frac{1}{2}, \frac{1}{2}\}$
 $A: \begin{bmatrix} 0.5 & 0.4 \\ 0.5 & 0.5 \end{bmatrix}$
 $B: \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}$
 $C: \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}$

观测序列 $O = \{A, B, A\}$
 $P(O=A) = P(A=A)P(A=B)P(A=A) + P(B=A)P(B=B)P(B=A) + P(C=A)P(C=B)P(C=A)$
 $= 0.5 \times 0.5 \times 0.5 + 0.5 \times 0.5 \times 0.5 + 0.5 \times 0.5 \times 0.5 = 0.375$

根据隐马尔可夫模型定义, 可以将一个长度为 T 的观测序列 $O = (o_1, o_2, \dots, o_T)$ 的生成过程描述如下。

算法 10.1 (观测序列的生成)
 输入: 隐马尔可夫模型 $\lambda = (A, B, \pi)$, 观测序列长度 T ;
 输出: 观测序列 $O = (o_1, o_2, \dots, o_T)$ 。

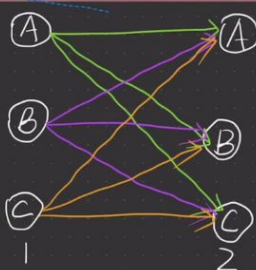
- 按照初始状态分布 π 产生状态 i_1 ;
- 令 $t = 1$;
- 按照状态 i_t 的观测概率分布 $b_{i_t}(k)$ 生成 o_t ;
- 按照状态 i_t 的状态转移概率分布 $\{a_{i_t i_{t+1}}\}$ 产生状态 i_{t+1} , $i_{t+1} = 1, 2, \dots, N$;
- 令 $t = t + 1$; 如果 $t < T$, 转步 (3); 否则, 终止。

Janneil博士 bilibili

① π (A) (B) (C)
 ② B $P(O_1=? | i_1=?)$
 ③ π, A, i_2, B

计算: $\frac{1}{3} \times 0.5 \times (0 \times 0.5 + 0.4 \times 0.3 + 0.6 \times 0.85) = 0.105$
 $\frac{1}{3} \times 0.3 = 0.1$
 0.047

计算: $\frac{1}{3} \times 0.5 \times (0.5 \times 0.5 + 0.5 \times 0.5 + 0.5 \times 0.5) = 0.25$
 $\frac{1}{3} \times 0.3 = 0.1$
 0.163



- 输入输出:
 - 输入: 模型 λ 和观测序列长度 T
 - 输出: 观测序列 $O = (o_1, o_2, \dots, o_T)$
- 生成步骤:
 - 按 π 产生初始状态 i_1
 - 令 $t = 1$
 - 按 $b_{i_t}(k)$ 生成 o_t
 - 按 $a_{i_t i_{t+1}}$ 产生 i_{t+1}
 - $t = t + 1$, 若 $t < T$ 返回步骤3
- 计算每一种状态组合的概率 01:18:01
 - 状态组合数量: 对于 T 个时刻和 N 个状态, 共有 N^T 种可能的状态序列
 - 计算复杂度: 直接计算法需要计算 $T \times N^T$ 次运算
 - 三硬币例子: 3^T 种组合
 - 四状态例子: 4^T 种组合
- 计算状态概率和观测概率 01:20:51

概率计算算法

模型 $\lambda = (\pi, A, B)$

观测 $O = (o_1, o_2, \dots, o_T)$

① 直接计算法 ✓

② 前向算法

③ 后向算法

乘法公式 $T \times N^T$

$P(O, I | \lambda) = P(I | \lambda) P(O | I, \lambda)$

3
3×3
3×3×3
⋮
3×3×3×3 ⋯ ×3 = 3^T

- 状态概率计算：
 - 初始状态： $P(i_1 | \lambda) = \pi_{i_1}$
 - 转移概率： $P(i_{t+1} | i_t) = a_{i_t i_{t+1}}$
- 观测概率计算：在状态 i_t 下观测到 o_t 的概率为 $b_{i_t}(o_t)$

- 联合概率的计算方法 01:23:06
 - 乘法公式： $P(O, I | \lambda) = P(I | \lambda) P(O | I, \lambda)$

展开形式：

- $P(O, I | \lambda) = \pi_{i_1} b_{i_1}(o_1) \prod_{t=2}^T a_{i_{t-1} i_t} b_{i_t}(o_t)$
- 两种理解方式：
 - 先计算状态序列概率，再计算观测概率
 - 交替计算状态转移和观测概率

- 全概率公式的应用 01:24:20

观测序列概率：对所有可能状态序列求和

- $P(O | \lambda) = \sum_I P(O, I | \lambda)$
- 具体实现：需要枚举所有 N^T 种状态序列组合
- 直接计算法的复杂度 01:25:27

概率计算算法

模型 $\lambda = (\pi, A, B)$

观测 $O = (O_1, O_2, \dots, O_T)$

① 直接计算法 ✓

② 前向算法

③ 后向算法

3
3×3
3×3×3
⋮
3×3×3 ⋯ ×3 = 3^T

乘法公式 $P(O, I | \lambda) = P(I) P(O | I)$

- 时间复杂度: $O(T \times N^T)$
- 计算瓶颈:
 - 状态数量N增加会指数级增大计算量
 - 序列长度T增加会线性增大计算量
- 适用场景: 仅适用于小规模问题, 需要更高效的算法 (如前向算法)
- 前向算法 01:26:18
 - 三硬币模型观测概率计算 01:30:17
 - 模型基本设定

三硬币

状态空间 $Q = \{A, B, C\}$ 全概率公式

观测空间 $V = \{正, 反\}$

$P(O=正)$

$= P(i_1=A)P(O=正|i_1=A)$
 $+ P(i_1=B)P(O=正|i_1=B)$
 $+ P(i_1=C)P(O=正|i_1=C)$

0.55

$\pi = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})^T$

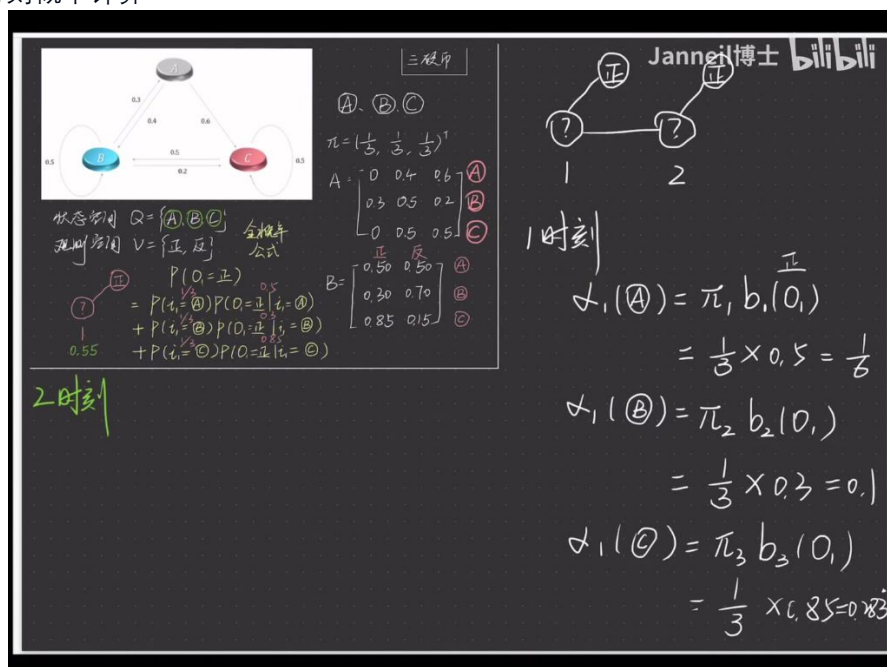
$A = \begin{bmatrix} 0 & 0.4 & 0.6 \\ 0.3 & 0.5 & 0.2 \\ 0 & 0.5 & 0.5 \end{bmatrix}$

$B = \begin{bmatrix} 0.5 & 0.5 \\ 0.3 & 0.7 \\ 0.85 & 0.15 \end{bmatrix}$

1时刻

- 初始概率分布: $\pi = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, 表示三个硬币被选中的初始概率相等
- 状态空间: $Q = \{A, B, C\}$, 对应三种硬币

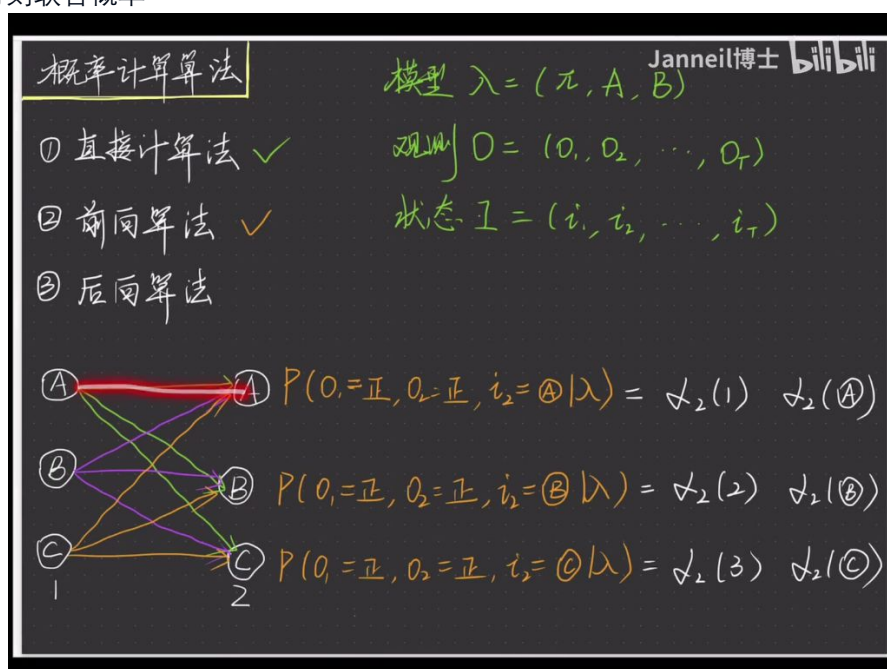
- 观测空间: $V=\{\text{正}, \text{反}\}$, 表示硬币的两种可能结果
- 直接算法
 - 单时刻概率计算



- 计算方法:

第一个时刻观测到正面的概率由全概率公式计算:

- $P(O_1 = \text{正}) = \sum_{i=A,B,C} P(i_1 = i)P(O_1 = \text{正} | i_1 = i)$
- 具体计算示例:
 - 硬币A: $\frac{1}{3} \times 0.5 = \frac{1}{6}$
 - 硬币B: $\frac{1}{3} \times 0.3 = 0.1$
 - 硬币C: $\frac{1}{3} \times 0.85 \approx 0.283$
- 多时刻联合概率



- 计算复杂度: 高达 $T \times N^T$, 其中 T 为时刻数, N 为状态数
- 缺点: 需要计算所有可能状态序列的组合, 计算量随时刻数指数增长

◆ 前向算法

- 算法原理

三硬币

状态空间 $Q = \{A, B, C\}$ 全概率公式
观测空间 $V = \{正, 反\}$

1时刻

2时刻

核心思想: 通过汇聚计算减少重复运算

递推公式:

- 核心思想: 通过汇聚计算减少重复运算
- 递推公式:
- 计算步骤:
 - 初始化: $\alpha_1(i) = \pi_i b_i(o_1)$
 - 递推计算各时刻各状态的 α 值
 - 终止: $P(O|\lambda) = \sum_{i=1}^N \alpha_T(i)$

● 计算示例

三硬币

状态空间 $Q = \{A, B, C\}$ 全概率公式
观测空间 $V = \{正, 反\}$

1时刻

2时刻

核心思想: 通过汇聚计算减少重复运算

○ 第二时刻计算:

■ 汇聚到A: $\alpha_2(A) = (\alpha_1(A) \times 0 + \alpha_1(B) \times 0.3 + \alpha_1(C) \times 0) \times 0.5 = 0.015$

■ 汇聚到B: $\alpha_2(B) = (\alpha_1(A) \times 0.4 + \alpha_1(B) \times 0.5 + \alpha_1(C) \times 0.5) \times 0.3 = 0.0775$

■ 汇聚到C: $\alpha_2(C) = (\alpha_1(A) \times 0 + \alpha_1(B) \times 0 + \alpha_1(C) \times 0.5) \times 0.85 \approx 0.2224$

○ 最终概率: $0.015 + 0.0775 + 0.2224 = 0.315$

● 算法优势

The image shows a chalkboard with handwritten mathematical derivations for the forward algorithm. It includes a small diagram of a robot in a hallway, a transition matrix, and detailed calculations for the forward probabilities α_1 and α_2 . The diagram on the right shows the state transitions between time steps 1 and 2.

Janneil博士 Bilibili

○

○ 计算复杂度: 降低为 $T \times N^2$

○ 效率提升: 相比直接法的指数级复杂度, 前向算法只需多项式时间

○ 物理意义: 每个时刻只需计算N个状态的汇聚情况, 避免重复计算

■ 后向算法简介

概率计算算法

模型 $\lambda = (\pi, A, B)$

观测 $O = (o_1, o_2, \dots, o_T)$

状态 $I = (i_1, i_2, \dots, i_T)$

① 直接计算法 ✓

② 前向算法 ✓

③ 后向算法

$P(o_1=\text{正}, o_2=\text{正}, i_2=\text{A}|\lambda) = \alpha_2(1) \alpha_2(\text{A})$
 $P(o_1=\text{正}, o_2=\text{正}, i_2=\text{B}|\lambda) = \alpha_2(2) \alpha_2(\text{B})$
 $P(o_1=\text{正}, o_2=\text{正}, i_2=\text{C}|\lambda) = \alpha_2(3) \alpha_2(\text{C})$

- 基本思想: 与前向算法方向相反, 从后向前计算
- 概率定义: $\beta_t(i)$ 表示在 t 时刻处于状态 i 时, 后续观测序列的概率
- 与前向算法的区别:
 - 前向算法包含当前及之前所有观测
 - 后向算法只包含之后所有观测
- 后向算法 01:50:02
 - 例题: 盒子和球模型观测概率计算 01:53:25

例 10.2 考虑盒子和球模型 $\lambda = (A, B, \pi)$, 状态集合 $Q = \{1, 2, 3\}$, 观测集合 $V = \{\text{红}, \text{白}\}$,

$$A = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \\ 0.7 & 0.3 \end{bmatrix}, \quad \pi = \begin{bmatrix} 0.2 \\ 0.4 \\ 0.4 \end{bmatrix}$$

设 $T = 3$, $O = (\text{红}, \text{白}, \text{红})$, 试用前向算法计算 $P(O|\lambda)$ 。

模型参数定义

- 模型结构: 隐马尔可夫模型 $\lambda = (A, B, \pi)$, 包含状态转移矩阵 A 、观测概率矩阵 B 和初始状态分布 π
- 状态集合: $Q = \{1, 2, 3\}$ 对应三个盒子

- 观测集合: $V = \{\text{红, 白}\}$ 对应球的颜色
- 参数矩阵:

$$\begin{aligned} \circ A &= \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} \\ \circ B &= \begin{bmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \\ 0.7 & 0.3 \end{bmatrix} \\ \circ \pi &= \begin{bmatrix} 0.2 & 0.4 \\ 0.4 & s \\ 0.4 & s \end{bmatrix} \end{aligned}$$

■ 向后算法计算过程

- 第三时刻初始化
 - 终止条件: $\beta_3(i) = 1$ ($i = 1, 2, 3$), 因为第三时刻后没有观测需要计算
 - 物理意义: 表示在第三时刻处于各状态时, 后续观测序列的概率 (无后续观测时为1)
- 第二时刻计算
 - 计算原理: $\beta_2(i) = \sum_{j=1}^3 a_{ij} b_j(o_3) \beta_3(j)$
 - 具体计算:
 - $\beta_2(1) = 0.5 \times 0.5 + 0.2 \times 0.4 + 0.3 \times 0.7 = 0.54$
 - $\beta_2(2) = 0.3 \times 0.5 + 0.5 \times 0.4 + 0.2 \times 0.7 = 0.49$
 - $\beta_2(3) = 0.2 \times 0.5 + 0.3 \times 0.4 + 0.5 \times 0.7 = 0.57$
 - 物理意义: 表示在第二时刻处于各状态时, 后续观测到 $o_3 = \text{红的}$ 的概率
- 第一时刻计算
 - 递推公式: $\beta_1(i) = \sum_{j=1}^3 a_{ij} b_j(o_2) \beta_2(j)$
 - 具体计算:
 - $\beta_1(1) = 0.5 \times 0.5 \times 0.54 + 0.2 \times 0.6 \times 0.49 + 0.3 \times 0.3 \times 0.57 \approx 0.2451$
 - $\beta_1(2) = 0.3 \times 0.5 \times 0.54 + 0.5 \times 0.6 \times 0.49 + 0.2 \times 0.3 \times 0.57 \approx 0.2622$
 - $\beta_1(3) = 0.2 \times 0.5 \times 0.54 + 0.3 \times 0.6 \times 0.49 + 0.5 \times 0.3 \times 0.57 \approx 0.2277$

■ 最终概率计算

- 计算公式: $P(O|\lambda) = \sum_{i=1}^3 \pi_i b_i(o_1) \beta_1(i)$
- 具体计算:
 - $0.2 \times 0.5 \times 0.2451 + 0.4 \times 0.4 \times 0.2622 + 0.4 \times 0.7 \times 0.2277 \approx 0.13022$
- 验证: 计算结果与前向算法一致, 验证了向后算法的正确性

■ 算法推广

- 一般形式: 对于T时刻的HMM, 向后算法从T时刻开始逆向递推
- 递推公式: $\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)$
- 终止条件: $\beta_T(i) = 1$ ($i = 1, \dots, N$)
- 初始计算: $P(O|\lambda) = \sum_{i=1}^N \pi_i b_i(o_1) \beta_1(i)$
- 计算图示: 每个时刻的状态通过转移概率与下一时刻所有状态相连, 形成逆向传播网络

8) 概率计算算法的小结 02:10:33

- 概率计算的基础概念

- **状态转移与观测概率**: 在隐马尔可夫模型中, t 时刻的状态可以转移到 $t+1$ 时刻的 N 种状态, 需要考虑 N 种可能的转移路径。观测概率 $b_j(o_t)$ 表示在状态 j 观测到 o_t 的概率, 从观测矩阵 B 中提取。
- **矩阵元素标识**: 转移概率 a_{ij} 来自矩阵 A 的第 i 行第 j 列, 观测概率 $b_j(o_t)$ 来自矩阵 B 的第 j 行对应观测值的概率。

● 例题10.2: 盒子和球模型的概率计算 02:11:38

例 10.2 考虑盒子和球模型 $\lambda = (A, B, \pi)$, 状态集合 $Q = \{1, 2, 3\}$, 观测集合 $V = \{\text{红}, \text{白}\}$,

$$A = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \\ 0.7 & 0.3 \end{bmatrix}, \quad \pi = \begin{bmatrix} 0.2 \\ 0.4 \\ 0.4 \end{bmatrix}$$

设 $T = 3$, $O = (\text{红}, \text{白}, \text{红})$, 试用前向算法计算 $P(O|\lambda)$ 。 0.13022

3. $\beta_3(\text{白}) = \beta_3(\text{红}) = \beta_3(\text{白}) = 1$

2. $\beta_2(\text{白}) = P(O_3 = \text{红} | i_2 = \text{白}) = 0.54$
 $\beta_2(\text{红}) = 0.49, \beta_2(\text{白}) = 0.57$

1. $\beta_1(\text{白}) = P(O_2 = \text{白}, O_3 = \text{红} | i_1 = \text{白}) = 0.2451$
 $\beta_1(\text{红}) = 0.2622, \beta_1(\text{白}) = 0.2277$

0. $0.13022 = \pi_1 b_1(1) \beta_1(\text{白}) + \pi_2 b_2(1) \beta_1(\text{红}) + \pi_3 b_3(1) \beta_1(\text{白})$

- **模型参数说明**:
 - 状态集合 $Q = \{1, 2, 3\}$, 观测集合 $V = \{\text{红}, \text{白}\}$
 - 转移矩阵 A 、观测矩阵 B 和初始概率分布 π 的具体数值
 - 给定 $T = 3$, 观测序列 $O = (\text{红}, \text{白}, \text{红})$, 要求计算 $P(O|\lambda)$
- 观测序列概率的后向算法 02:15:40
 - **终止条件初始化**:
 - 在终点时刻 T , 后向概率 $\beta_T(i) = 1$ 对所有状态 $i = 1, 2, \dots, N$ 成立
 - 这是因为终点时刻不需要从后续状态汇聚概率, 直接设为1保证递推起点
 - **递推公式**:
 - $\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)$
 - 表示 t 时刻状态 i 转移到 $t+1$ 时刻所有可能状态 j 的概率加权和
 - 包含状态转移概率 a_{ij} 、观测概率 $b_j(o_{t+1})$ 和后向概率 $\beta_{t+1}(j)$
- 后向算法的递推与结果计算 02:16:18
 - **递推过程**:
 - 从 $T-1$ 时刻开始反向递推到 $t=1$
 - 每个时刻 t 需要计算 N 个状态的后向概率
 - 递推时需要考虑 $t+1$ 时刻所有可能的 N 个状态转移
 - **最终概率计算**:
 - $P(O|\lambda) = \sum_{i=1}^N \pi_i b_i(o_1) \beta_1(i)$
 - 结合初始概率 π_i 、第一个观测概率 $b_i(o_1)$ 和后向概率 $\beta_1(i)$
 - 解决了 $\beta_1(i)$ 不包含第一个观测的问题

- 后向算法的时间复杂度 02:17:33

- 计算复杂度分析:

- 每个时刻 t 需要计算 N 个状态的后向概率
 - 每个状态计算需要 N 次求和 (对应 N 种可能转移)
 - 总时间复杂度为 $O(N^2T)$, 与前向算法相同

- 效率优势:

- 相比直接计算法 (复杂度 $O(TN^T)$) 显著提高效率
 - 适用于长观测序列的概率计算

- 前向算法与后向算法的结合 02:18:23

- 联合计算原理:

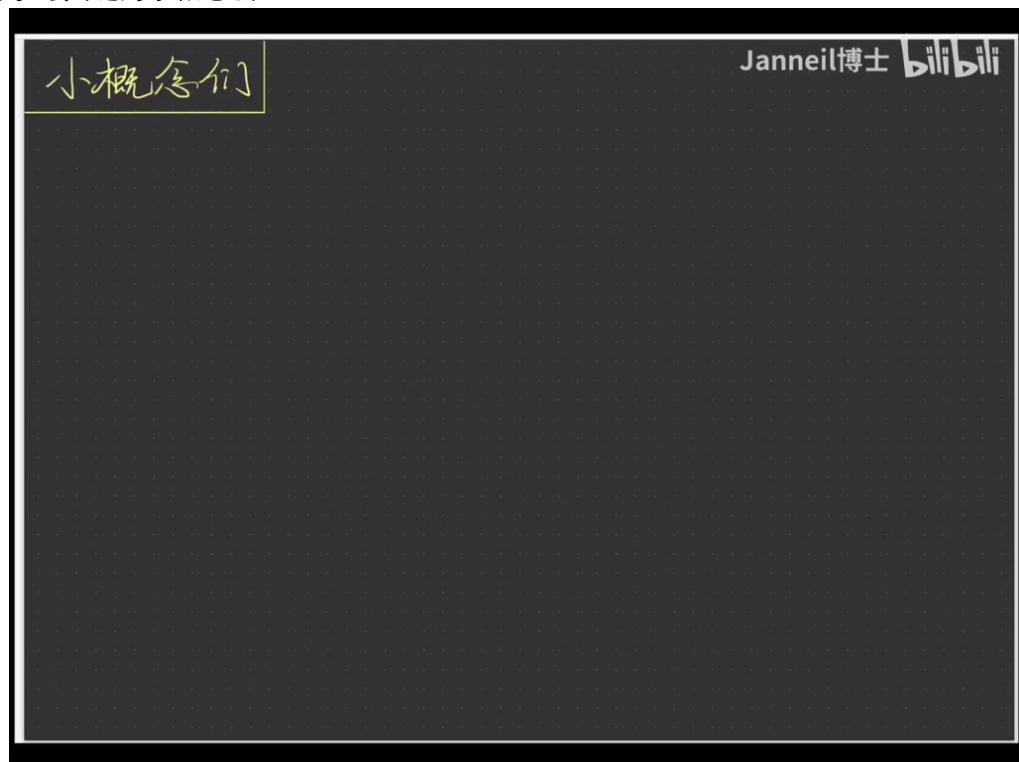
- 在任意时刻 t , 可以结合前向概率 $\alpha_t(i)$ 和后向概率 $\beta_{t+1}(j)$
 - 计算公式: $\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)$
 - 表示通过状态 i 到 j 的转移连接前后两部分概率

- 应用场景:

- 可用于计算特定时刻处于某状态的概率
 - 在参数估计等问题中有重要应用
 - 类似逐步回归的思想, 前后信息结合提高计算效率

- 9) 小概念们 02:21:05

- 隐马尔可夫链的小概念引入 02:21:18



- 基本组成: 包含 N 种状态和 M 种观测结果, 重点关注 t 时刻的第 i 个状态概率计算
 - 算法结合: 前向算法 α 和后向算法 β 在特定时刻交汇, 形成"八爪鱼"式计算结构
 - 概率计算: 通过 $\alpha_t(i)$ 和 $\beta_t(i)$ 的乘积表示 t 时刻处于 q_i 状态的概率

- 隐马尔可夫模型的三要素 02:22:20

- 初始分布 π : 决定模型初始状态的概率分布
 - 转移矩阵 A : 表示状态间的转移概率, a_{ij} 为从 i 到 j 的转移概率
 - 观测矩阵 B : 表示在特定状态下观测到不同结果的概率
 - 观测序列 O : 包含从 o_1 到 o_T 的观测结果序列

- 观测序列概率与条件概率 02:22:47

- 条件概率展开: $P(i_t = q_i | O, \lambda) = \frac{P(i_t = q_i, O | \lambda)}{P(O | \lambda)}$
- 分子构成: $\alpha_t(i)\beta_t(i)$ 表示所有观测结果及 t 时刻 q_i 状态的联合概率
- 分母处理: 需要对所有可能状态求和 $\sum_{j=1}^N \alpha_t(j)\beta_t(j)$

- 前向算法与后向算法的结合 02:24:40

算法 10.3 (观测序列概率的后向算法)
 输入: 隐马尔可夫模型 λ , 观测序列 O ;
 输出: 观测序列概率 $P(O | \lambda)$ 。

(1)

$$\beta_T(i) = 1, \quad i = 1, 2, \dots, N \quad (10.19)$$

(2) 对 $t = T-1, T-2, \dots, 1$

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(o_{t+1}) \beta_{t+1}(j), \quad i = 1, 2, \dots, N \quad (10.20)$$

(3)

$$P(O | \lambda) = \sum_{i=1}^N \pi_i b_i(o_1) \beta_1(i) \quad (10.21)$$

Janneil博士 bilibili

$O(N^2T)$

前向算法

$P(O | \lambda)$

$= \sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} \cdot b_j(o_{t+1}) \cdot \beta_{t+1}(j)$



- 前向部分: $\alpha_t(i)$ 表示 o_1 到 o_t 观测及 $i_t = q_i$ 的概率
- 后向部分: $\beta_t(i)$ 表示从 $t+1$ 到 T 观测的条件概率

- 联合表示: $\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^N \alpha_t(j)\beta_t(j)}$

- 观测序列概率的后向算法 02:31:04

- 初始化: $\beta_T(i) = 1, \quad i = 1, 2, \dots, N$
- 递推计算: 对 $t = T-1, T-2, \dots, 1, \quad \beta_t(i) = \sum_{j=1}^N a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)$
- 概率计算: $P(O | \lambda) = \sum_{i=1}^N \pi_i b_i(o_1) \beta_1(i)$

- 隐马尔可夫模型中的期望值 02:32:51

小概念们

Janneil博士 bilibili

1. 已知 $\lambda = (A, B, \pi)^T$, $O = (o_1, o_2, \dots, o_T)$

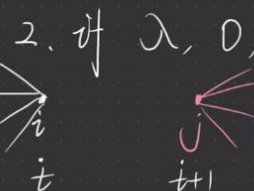


$$\gamma_t(i) = P(i_t = q_i | \lambda, O) \quad \text{条件概率}$$

$$= \frac{P(i_t = q_i, O | \lambda)}{P(O | \lambda)} = \frac{\alpha_t(i) \beta_t(i)}{\sum_{j=1}^N \alpha_t(j) \beta_t(j)}$$

$$\alpha_t(i) = P(o_1, o_2, \dots, o_t, i_t = q_i | \lambda)$$

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \dots, o_T | i_t = q_i, \lambda)$$



$$\xi_t(i, j) = P(i_t = q_i, i_{t+1} = q_j | O, \lambda)$$

$$= \frac{P(i_t = q_i, i_{t+1} = q_j, O | \lambda)}{P(O | \lambda)} = \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{i,j=1}^N \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}$$

- 状态期望: $\sum_{t=1}^T \gamma_t(i)$ 表示观测序列中处于 q_i 状态的期望次数
- 转移期望: $\sum_{t=1}^{T-1} \xi_t(i, j)$ 表示从 q_i 转移到 q_j 的期望次数
- 类比关系: 与混合高斯模型中计算观测值属于特定分布的期望类似
- 隐马尔可夫模型与EM算法 02:36:30
 - 算法关系: Baum-Welch算法是EM算法在HMM中的特例
 - 历史背景: Baum-Welch提出早于EM算法, 但仅适用于HMM
 - 参数估计: 通过期望值计算更新模型参数 $\lambda = (A, B, \pi)$
- EM算法在隐马尔可夫模型中的应用 02:37:35

小概念们

Janneil博士 bilibili

1. 已知 $\lambda = (A, B, \pi)^T$, $O = (o_1, o_2, \dots, o_T)$

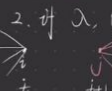


$$\gamma_t(i) = P(i_t = q_i | \lambda, O) \quad \text{条件概率}$$

$$= \frac{P(i_t = q_i, O | \lambda)}{P(O | \lambda)} = \frac{\alpha_t(i) \beta_t(i)}{\sum_{j=1}^N \alpha_t(j) \beta_t(j)}$$

$$\alpha_t(i) = P(o_1, o_2, \dots, o_t, i_t = q_i | \lambda)$$

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \dots, o_T | i_t = q_i, \lambda)$$



$$\xi_t(i, j) = P(i_t = q_i, i_{t+1} = q_j | O, \lambda)$$

$$= \frac{P(i_t = q_i, i_{t+1} = q_j, O | \lambda)}{P(O | \lambda)} = \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{i,j=1}^N \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}$$

- **E步骤**: 计算 $\gamma_t(i)$ 和 $\xi_t(i,j)$ 期望值
- **M步骤**: 基于期望值重新估计模型参数
- **迭代优化**: 重复E-M步骤直至参数收敛

5. 学习方法 02:38:03

1) 混合高斯模型 02:38:14

- **完全信息下的参数估计**: 当已知每个观测值所属的高斯分布时, 可以直接用观测值估计各分布的参数。例如对于 k 个高斯分布 $N(\mu_1, \sigma_1^2)$ 到 $N(\mu_k, \sigma_k^2)$, 若已知 y_1, y_3, y_n 属于第一个分布, 则:
 - μ_1 的估计: 取这些观测值的平均值
 - σ_1^2 的估计: 计算这些观测值的方差
- **极大似然估计示例**: 以盒子中有10个小球 (2黑8白) 为例, 摸到黑球的概率估计就是频率 $\frac{2}{10}$, 体现了"以频率估计概率"的思想

2) 初始状态分布的学习 02:41:58

学习算法 π, A, B EM

混合高斯

$N(\mu_1, \sigma_1^2)$

$N(\mu_2, \sigma_2^2)$

$\vdots$

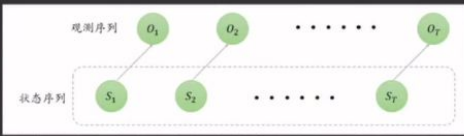
$N(\mu_k, \sigma_k^2)$

全部信息 MLE

○○○○○
 ●●○○○

$\hat{\theta} = \frac{2}{10}$

HMM Janneil博士 Bilibili



- **参数定义**: 初始状态分布 π 包含 N 个元素 $\pi_1, \pi_2, \dots, \pi_N$, 表示初始时刻处于各状态的概率
- **估计方法**:
 - 分子: 初始时刻处于第 i 个状态的样本个数
 - 分母: 初始时刻所有样本总个数
 - 公式: $\hat{\pi}_i = \frac{\text{初始处于 } i \text{ 状态的样本数}}{\text{总样本数}}$

- **实现条件**: 需要从总体生成多条马尔可夫链才能进行有效计数, 单条链无法完成估计

3) 状态转移概率矩阵的学习 02:46:03

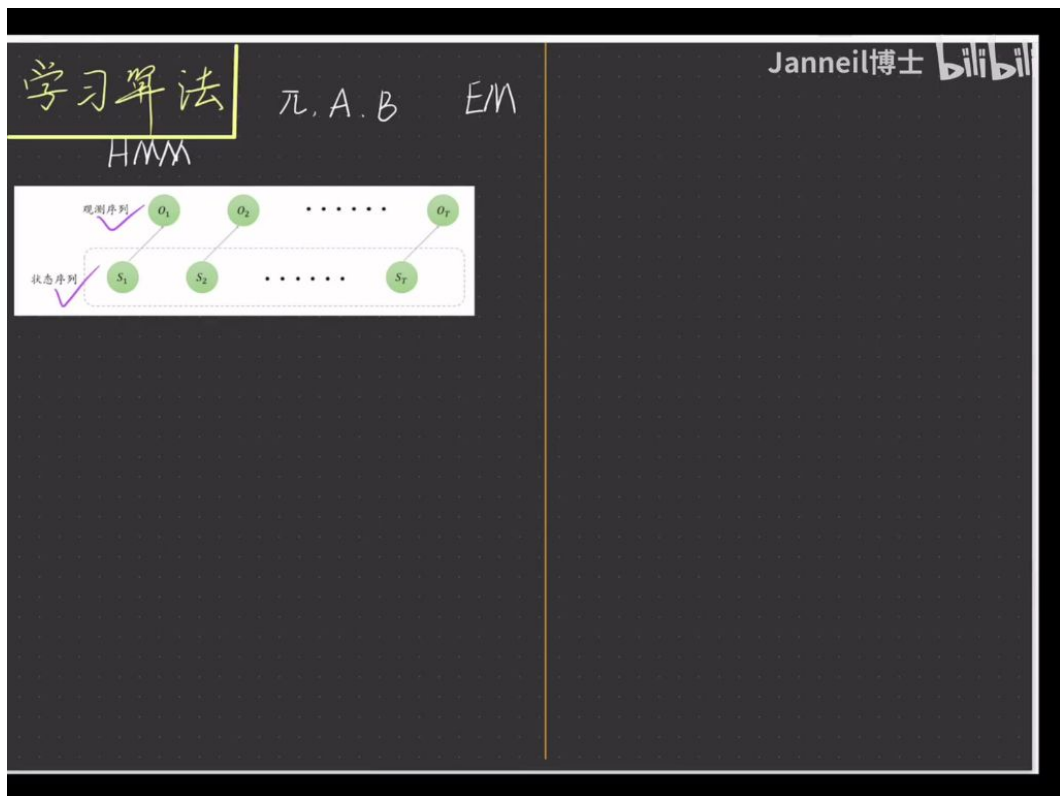
- **矩阵元素含义**: a_{ij} 表示 t 时刻从状态 i 转移到状态 j 的概率
- **估计方法**:
 - 分子: t 时刻从 i 转移到 j 的样本个数

- 分母：t时刻所有处于i状态的样本总数

- 公式： $\hat{a}_{ij} = \frac{\text{从} i \text{ 转移到 } j \text{ 的样本数}}{\text{处于 } i \text{ 状态的样本总数}}$

- 数学表达：设从i到j的转移次数为 A_{ij} ，则 $\hat{a}_{ij} = \frac{A_{ij}}{\sum_{k=1}^N A_{ik}}$

4) 观测矩阵的学习 02:50:14



- 矩阵元素含义： b_{jk} 表示处于状态j时观测到第k种结果的概率

- 估计方法：

- 分子：处于j状态且观测到k的样本个数

- 分母：所有处于j状态的样本总数

- 公式： $b_{jk} = \frac{\text{处于 } j \text{ 且观测到 } k \text{ 的样本数}}{\text{处于 } j \text{ 状态的样本总数}}$

- 数学表达：设处于j状态观测到k的次数为 B_{jk} ，则 $b_{jk} = \frac{B_{jk}}{\sum_{m=1}^M B_{jm}}$

- 注意事项：虽然 a_{ij} 和 b_{jk} 的分母可能对应相同状态，但它们的估计视角不同（状态转移vs观测生成）

5) 鲍姆-威尔奇算法 02:57:26

- 若尔当不等式 03:00:06

Baum-Welch 算法

Jensen 不等式 Janneil 博士 bilibili

$\bar{\lambda}$ 上一组迭代参数

$P(I | \bar{\lambda}, 0)$

西蒙斯：征服市场的人

MLE

联合概率 $L(\lambda)$

$P(O | \lambda) = \sum_I P(O, I | \lambda)$

任意已知 $i_1, \dots, i_T$

似然函数

$\log L(\lambda) = \log \sum_{I \rightarrow \text{所有组合}} P(O, I | \lambda)$

$\log \sum_I P(I | \bar{\lambda}, 0) \frac{P(O, I | \lambda)}{P(I | \bar{\lambda}, 0)}$

$\geq \sum_I P(I | \bar{\lambda}, 0) \log \frac{P(O, I | \lambda)}{P(I | \bar{\lambda}, 0)}$

$= \sum_I P(I | \bar{\lambda}, 0) \log P(O, I | \lambda)$

$- \sum_I P(I | \bar{\lambda}, 0) \log P(I | \bar{\lambda}, 0)$

- 应用条件：需要使用真三不等式时，必须找到一个概率分布。在迭代方法中，采用上一组迭代参数 $\lambda^{(t-1)}$ 作为已知条件。
- 推导过程：
 - 在已知观测序列 O 和上一组参数 $\lambda^{(t-1)}$ 的情况下，通过条件概率分布 $P(I | O, \lambda^{(t-1)})$ 将隐变量转化为完全数据
 - 利用 Jensen 不等式将求和移到对数外部，形成期望形式： $E[\log P(O, I | \lambda)]$
 - 目标函数简化为 $Q(\lambda, \lambda^{(t-1)}) = \sum_I P(I | O, \lambda^{(t-1)}) \log P(O, I | \lambda)$
- M 步 03:09:18
 - 参数分解

M 步

$Q(\lambda, \bar{\lambda}) = \sum_I P(O, I | \bar{\lambda}) \log P(O, I | \lambda)$

$P(O, I | \lambda) = \pi_{i_1} b_{i_1}(o_1) \dots a_{i_{t-1}, i_t} b_{i_t}(o_t) \pi_{i_t}$

$\pi_{i_1} \dots \pi_{i_t}$ π

$b_{i_1}(o_1) \dots b_{i_t}(o_t)$ B

a_{i_{t-1}, i_t} A

Q'

Janneil 博士 bilibili

■ 三要素分解:

- 初始概率 π : 对应 $P(I_1 = i)$
- 转移概率 a : 对应 $P(I_{t+1} = j | I_t = i)$ 的乘积
- 发射概率 b : 对应 $P(O_t | I_t = j)$ 的乘积

■ 对数转换: 将联合概率 $P(O, I | \lambda)$ 的对数分解为三部分之和, 便于分别优化

○ 初始概率估计

第三步

Janneil博士 bilibili

$$Q(\lambda, \bar{\lambda}) = \sum_{\mathcal{I}} P(O, \mathcal{I} | \bar{\lambda}) \log P(O, \mathcal{I} | \lambda)$$

$$P(O, \mathcal{I} | \lambda) = \pi_{i_1} b_{i_1}(o_1) a_{i_1 i_2} b_{i_2}(o_2) \cdots a_{i_{T-1} i_T} b_{i_T}(o_T)$$

π_{i_1}
 $b_{i_1}(o_1) \cdots b_{i_T}(o_T)$
 $a_{i_1 i_2} \cdots a_{i_{T-1} i_T}$

π
 B
 A

$$Q(\lambda, \bar{\lambda}) = \sum_{\mathcal{I}} P(O, \mathcal{I} | \bar{\lambda}) \log \{ \pi_{i_1} (a_{i_1 i_2} \cdots a_{i_{T-1} i_T}) (b_{i_1}(o_1) \cdots b_{i_T}(o_T)) \}$$

$$= \sum_{\mathcal{I}} P(O, \mathcal{I} | \bar{\lambda}) \left\{ \log \pi_{i_1} + \sum_{t=1}^{T-1} \log a_{i_t i_{t+1}} + \sum_{t=1}^T \log b_{i_t}(o_t) \right\}$$

$$= \underbrace{\sum_{\mathcal{I}} P(O, \mathcal{I} | \bar{\lambda}) \log \pi_{i_1}}_{\pi} + \underbrace{\sum_{\mathcal{I}} \sum_{t=1}^{T-1} P(O, \mathcal{I} | \bar{\lambda}) \log a_{i_t i_{t+1}}}_{A} + \underbrace{\sum_{\mathcal{I}} \sum_{t=1}^T P(O, \mathcal{I} | \bar{\lambda}) \log b_{i_t}(o_t)}_{B}$$

■ 约束条件: $\sum_{i=1}^N \pi_i = 1$

■ 拉格朗日求解:

- 构建目标函数 Q_1 包含 $\log \pi_i$ 项和约束条件
- 通过求偏导得到估计式: $\hat{\pi}_i = \frac{P(O, I_1 = i | \lambda(t-1))}{P(O | \lambda(t-1))}$
- 物理意义: 初始处于状态 i 的样本比例

○ 转移概率估计

M步

$$Q(\lambda, \lambda) = \sum_i P(0, 1 | \lambda) \log P(0, 1 | \lambda)$$

$$P(0, 1 | \lambda) = \pi_{i_1} b_{i_1}(0, 1) a_{i_1, i_2} b_{i_2}(0, 1) \dots a_{i_{T-1}, i_T} b_{i_T}(0, 1)$$

$$\begin{matrix} \pi_{i_1} & \pi \\ b_{i_1}(0, 1) & B \\ a_{i_1, i_2} & A \end{matrix}$$

拉格朗日乘子法

$$Q_1(\pi, \lambda) = \sum_i P(0, 1 | \lambda) \log \pi_{i_1} + \gamma \left(\sum_{i=1}^N \pi_{i_1} - 1 \right)$$

费马原理

$$\frac{\partial Q_1(\pi, \lambda)}{\partial \pi_{i_1}} = \frac{P(0, i_1 = i | \lambda)}{\pi_{i_1}} + \gamma = 0$$

$$\begin{aligned} \log \{ \pi_{i_1} (a_{i_1, i_2} \dots a_{i_{T-1}, i_T}) (b_{i_1}(0, 1) \dots b_{i_T}(0, 1)) \} \\ = \sum_i P(0, 1 | \lambda) \left\{ \log \pi_{i_1} + \sum_{t=1}^{T-1} \log a_{i_t, i_{t+1}} + \sum_{t=1}^T \log b_{i_t}(0, 1) \right\} \\ = \underbrace{\sum_i P(0, 1 | \lambda) \log \pi_{i_1}}_{\pi} + \underbrace{\sum_{t=1}^{T-1} \sum_i P(0, 1 | \lambda) \log a_{i_t, i_{t+1}}}_{A} + \underbrace{\sum_{t=1}^T \sum_i P(0, 1 | \lambda) \log b_{i_t}(0, 1)}_{B} \end{aligned}$$

约束条件: $\sum_{j=1}^N a_{ij} = 1$ (行和为1)

估计公式:

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} P(O, I_t = i, I_{t+1} = j | \lambda(t-1))}{\sum_{t=1}^{T-1} P(O, I_t = i | \lambda(t-1))}$$

- 分子: 从i转移到j的期望次数
- 分母: 处于状态i的总期望次数

发射概率估计

A

$$\sum_{j=1}^N a_{ij} = 1 \text{ 条件 } \arg \max_{\lambda} Q(\lambda, \lambda)$$

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} P(0, i_t = i, i_{t+1} = j | \lambda)}{\sum_{t=1}^{T-1} P(0, i_t = i | \lambda)}$$

拉格朗日乘子法

$$Q_2(\pi, \lambda) = \sum_{i=1}^N \sum_{j=1}^N P(0, i_t = i, i_{t+1} = j | \lambda) \log a_{ij} + \gamma \left(\sum_{j=1}^N a_{ij} - 1 \right)$$

费马原理

$$\frac{\partial Q_2(\pi, \lambda)}{\partial a_{ij}} = \frac{\sum_{t=1}^{T-1} P(0, i_t = i, i_{t+1} = j | \lambda)}{a_{ij}} + \gamma = 0$$

$$\Rightarrow \sum_{j=1}^N a_{ij} \gamma = - \sum_{j=1}^N \sum_{t=1}^{T-1} P(0, i_t = i, i_{t+1} = j | \lambda) = - \sum_{t=1}^{T-1} P(0, i_t = i | \lambda)$$

约束条件: $\sum_{k=1}^M b_j(k) = 1$ (行和为1)

估计公式:

$$b_j(k) = \frac{\sum_{t=1}^T P(O, I_t = j | \lambda^{(t-1)}) I(O_t = v_k)}{\sum_{t=1}^T P(O, I_t = j | \lambda^{(t-1)})}$$

- 分子：在状态j观察到 v_k 的期望次数
- 分母：处于状态j的总期望次数

■ 算法总结

Janneil博士 bilibili

$$\sum_{i=1}^N P(O, I_1 = i | \lambda) \log \pi_{i_1} + \sum_{i=1}^N \sum_{j=1}^N P(O, I_1 = i | \lambda) \log a_{ij} + \sum_{i=1}^N \sum_{j=1}^N P(O, I_1 = i | \lambda) \log b_{ij}(O_2)$$

A $\sum_{j=1}^N a_{ij} = 1$ 条件 $\arg \max_{\lambda} Q(\lambda, \lambda)$ $\hat{a}_{ij} = \frac{\sum_{t=1}^T P(O, i_t = i, i_{t+1} = j | \lambda)}{\sum_{t=1}^T P(O, i_t = i | \lambda)}$

拉格朗日乘子法

$$Q_2(\pi, \lambda) = \sum_{i=1}^N \sum_{j=1}^N P(O, i_t = i, i_{t+1} = j | \lambda) \log a_{ij} + \sum_{i=1}^N \gamma_i (\sum_{j=1}^N a_{ij} - 1)$$

$$= \sum_{i=1}^N \sum_{j=1}^N \sum_{t=1}^T P(O, i_t = i, i_{t+1} = j | \lambda) \log a_{ij} + \sum_{i=1}^N \gamma_i (\sum_{j=1}^N a_{ij} - 1)$$

费马原理 $\frac{\partial Q_2(\pi, \lambda)}{\partial a_{ij}} = \frac{\sum_{t=1}^T P(O, i_t = i, i_{t+1} = j | \lambda)}{a_{ij}} + \gamma_i = 0$

$$\Rightarrow \sum_{j=1}^N a_{ij} \gamma_i = - \sum_{j=1}^N \sum_{t=1}^T P(O, i_t = i, i_{t+1} = j | \lambda) = - \sum_{t=1}^T P(O, i_t = i | \lambda)$$

■ 迭代过程：

- 初始化模型参数 $\lambda^{(0)} = (A^{(0)}, B^{(0)}, \pi^{(0)})$
- E步：计算 $\gamma_t(i)$ 和 $\xi_t(i, j)$
- M步：用估计公式更新参数
- 重复直到收敛

- 与MLE关系：相当于在"完整数据"（包含隐变量）上的极大似然估计，用期望次数代替实际观察次数

二、隐马尔可夫模型 03:39:45

1. 预测算法

1) 近似算法 03:41:16

- 近似算法：微观视角与状态预测 03:41:26
 - 微观视角：近似算法从每个时刻点单独出发，独立预测每个时刻的最佳状态，不考虑状态间的时序关系。
 - 实现方式：在每个时刻t，选择使 $P(i_t = q_i | \lambda, O)$ 最大的状态 q_i 作为预测结果。
 - 计算优势：相比维特比算法更简单直观，因为只需处理单个时刻的状态选择问题。
- 维特比算法：整体视角与动态规划 03:42:03
 - 宏观视角：维特比算法关注整个状态序列的全局最优路径，通过动态规划方法在所有可能的 n^T 条路径中找到最佳路径。
 - 复杂度：需要逐步更新和比较路径概率，计算复杂度显著高于近似算法。
 - 动态规划：采用"走一步看一步"的迭代方式，每一步都保留当前最优路径。

- 最佳状态的选择：概率转移与观测概率矩阵 03:43:36
 - **选择依据**：两种算法都基于隐马尔可夫模型的参数（转移概率矩阵和观测概率矩阵）进行状态预测。
 - **概率最大化**：通常选择使状态概率最大的结果，近似算法在每个时刻独立进行这种选择。
- 近似算法的概率计算与状态选择 03:44:12
 - **概率表示**：用 $\gamma_t(i)$ 表示t时刻处于状态 q_i 的概率，定义为 $P(i_t = q_i | \lambda, O)$ 。
 - **简化计算**：由于分母 $P(O | \lambda)$ 固定，最大化 $\gamma_t(i)$ 等价于最大化分子 $P(i_t = q_i, O | \lambda)$ 。
 - **状态选择**：每个时刻选择使 $\gamma_t(i)$ 最大的状态 i_t^* ，最终得到完整的状态序列。
- 状态概率的计算方法 03:45:17
 - **前向后向结合**： $\gamma_t(i)$ 可以表示为 $\alpha_t(i)\beta_t(i)$ ，其中：
 - $\alpha_t(i)$ ：前向算法得到的t时刻处于 q_i 且观测到 $o_1, \dots, o_t$ 的概率
 - $\beta_t(i)$ ：后向算法得到的从 $t+1$ 到T时刻观测到 $o_{t+1}, \dots, o_T$ 的概率
 - **计算转化**：最大化 $\gamma_t(i)$ 转化为最大化 $\alpha_t(i)\beta_t(i)$ 的乘积。
- 例题1:近似算法的应用 03:48:59

Janneil博士 bilibili



$\alpha_1(1)=0.10, \alpha_1(2)=0.16, \alpha_1(3)=0.28$
 $\alpha_2(1)=0.077, \alpha_2(2)=0.1104, \alpha_2(3)=0.0606$
 $\alpha_3(1)=0.04187, \alpha_3(2)=0.03551, \alpha_3(3)=0.05284$

$\beta_1(1)=0.2451, \beta_1(2)=0.2622, \beta_1(3)=0.2277$
 $\beta_2(1)=0.54, \beta_2(2)=0.49, \beta_2(3)=0.57$
 $\beta_3(1)=1, \beta_3(2)=1, \beta_3(3)=1$

Janneil博士

红





时刻 1

白





时刻 2

红





时刻 3

$\alpha_1(1)=0.10, \alpha_1(2)=0.16, \alpha_1(3)=0.28$
 $\alpha_2(1)=0.077, \alpha_2(2)=0.1104, \alpha_2(3)=0.0606$
 $\alpha_3(1)=0.04187, \alpha_3(2)=0.03551, \alpha_3(3)=0.05284$

$\beta_1(1)=0.2451, \beta_1(2)=0.2622, \beta_1(3)=0.2277$
 $\beta_2(1)=0.54, \beta_2(2)=0.49, \beta_2(3)=0.57$
 $\beta_3(1)=1, \beta_3(2)=1, \beta_3(3)=1$

时刻 1: $\alpha_i(i) \beta_i(i)$

□ $0.10 \times 0.2451 = 0.02451$

□ $0.16 \times 0.2622 = 0.041952$

□ $0.28 \times 0.2277 = 0.063756$

时刻 2: $\alpha_i(i) \beta_i(i)$

□ $0.077 \times 0.54 = 0.04158$

□ $0.1104 \times 0.49 = 0.054096$

□ $0.0606 \times 0.57 = 0.034542$

○

题目解析

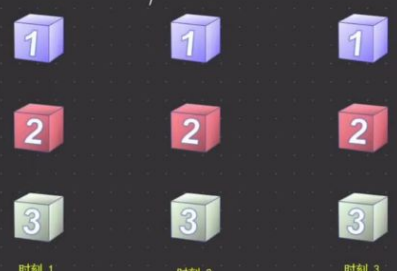
- 模型设定：使用小球与盒子模型，观测序列为红-白-红
- 时刻1计算：
 - 盒子1: $0.10 \times 0.2451 = 0.02451$
 - 盒子2: $0.16 \times 0.2622 = 0.041952$
 - 盒子3: $0.28 \times 0.2277 = 0.063756$ (最大)
- 时刻2计算：
 - 盒子1: $0.077 \times 0.54 = 0.04158$
 - 盒子2: $0.1104 \times 0.49 = 0.054096$ (最大)
 - 盒子3: $0.0606 \times 0.57 = 0.034542$
- 时刻3计算：直接比较 $\alpha_3(i)$ ，盒子3的值最大
- 最终路径：3→2→3
- 算法特点：这是局部最优解，可能不是全局最优路径

2) 维特比算法 03:54:22

- 例题:盒子与小球观测状态判定 03:55:23
 - 维特比算法原理

Janneil博士 bilibili

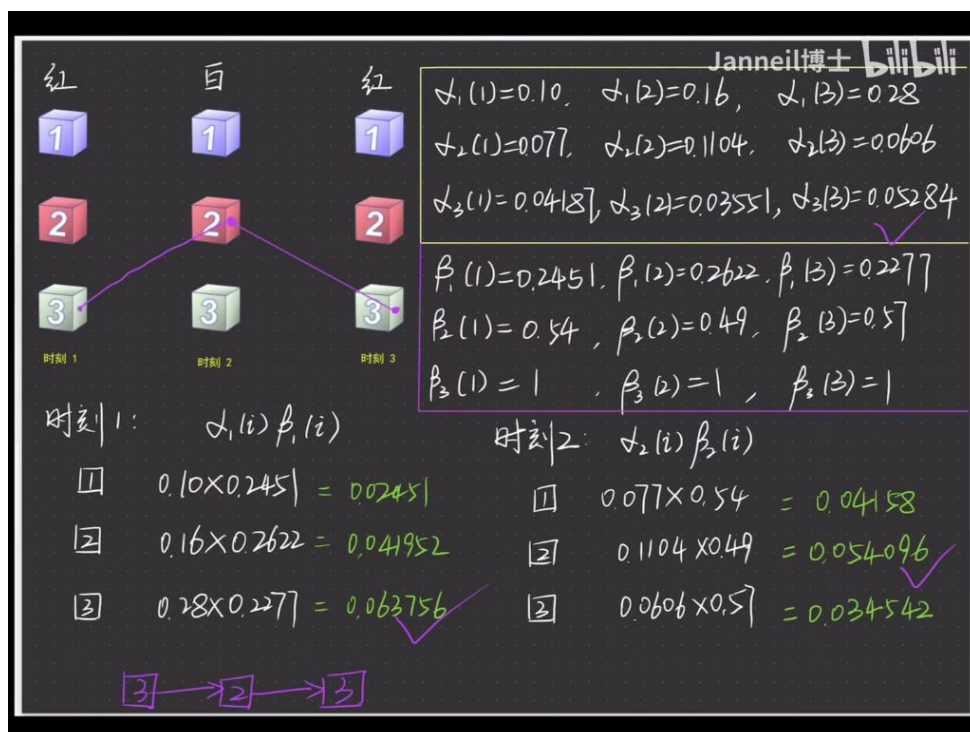
② 维特比算法 宏观 全局
动态规划



时刻 1 时刻 2 时刻 3

$$A = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} \quad B = \begin{bmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \\ 0.7 & 0.3 \end{bmatrix} \quad \pi = \begin{bmatrix} 0.2 \\ 0.4 \\ 0.4 \end{bmatrix}$$

- 宏观全局思想：采用动态规划方法，通过时刻1到时刻3的状态转移寻找最优路径
- 初始时刻计算： $\delta_1(i) = \pi_i \times b_i(\text{红})$
 - 盒子1： $0.2 \times 0.5 = 0.10$
 - 盒子2： $0.4 \times 0.4 = 0.16$
 - 盒子3： $0.4 \times 0.7 = 0.28$
- 路径记录： $\varphi_1(i) = 0$ （初始时刻无前驱状态）
- 时刻2的状态转移
 - 递推公式： $\delta_2(i) = \max[\delta_1(j) \times a_{ji}] \times b_i(\text{白})$
 - 盒子1路径选择：
 - 路径1→1： $0.10 \times 0.5 = 0.05$
 - 路径2→1： $0.16 \times 0.3 = 0.048$
 - 路径3→1： $0.28 \times 0.2 = 0.056$ （最优）
 - 最终值： $0.056 \times 0.5 = 0.028$
 - 盒子2路径选择：
 - 最优路径3→2： $0.28 \times 0.3 = 0.084$
 - 最终值： $0.084 \times 0.6 = 0.0504$
 - 盒子3路径选择：
 - 最优路径3→3： $0.28 \times 0.5 = 0.14$
 - 最终值： $0.14 \times 0.3 = 0.042$
- 时刻3的最优路径确定



- 递推计算:
 - 盒子1: $\max[0.028 \times 0.5, 0.0504 \times 0.3, 0.042 \times 0.2] \times 0.5 = 0.00756$ (路径2→1)
 - 盒子2: $\max[0.028 \times 0.2, 0.0504 \times 0.5, 0.042 \times 0.3] \times 0.4 = 0.01008$ (路径2→2)
 - 盒子3: $\max[0.028 \times 0.3, 0.0504 \times 0.2, 0.042 \times 0.5] \times 0.7 = 0.0147$ (路径3→3)
- 最优路径选择: 比较三个终态概率0.00756, 0.01008, 0.0147, 选择最大概率
 路径3→3→3
- 算法特点总结
 - 全局最优性: 与近似算法结果 (3→2→3) 不同, 维特比算法保证找到全局最优路径
 - 动态规划优势: 通过递推计算避免穷举所有可能路径, 提高计算效率
 - 路径回溯: 利用 φ 记录每个状态的最优前驱节点, 最终反向追踪得到完整路径

3) 维特比算法总结 04:18:13

- 维特比算法的输入输出
 - 输入: 模型参数 $\lambda = (A, B, \pi)$ 和观测序列 $O = (o_1, o_2, \dots, o_T)$
 - 输出: 最优状态路径 $I^* = (i_1^*, i_2^*, \dots, i_T^*)$
- 维特比算法的迭代公式 04:18:22

算法 10.5 (维特比算法)

输入: 模型 $\lambda = (A, B, \pi)$ 和观测 $O = (o_1, o_2, \dots, o_T)$;

输出: 最优路径 $I^* = (i_1^*, i_2^*, \dots, i_T^*)$ 。

(1) 初始化

$$\delta_1(i) = \pi_i b_i(o_1), \quad i = 1, 2, \dots, N$$

$$\psi_1(i) = 0, \quad i = 1, 2, \dots, N$$

(2) 递推。对 $t = 2, 3, \dots, T$

$$\delta_t(i) = \max_{1 \leq j \leq N} [\delta_{t-1}(j) a_{ji}] b_i(o_t), \quad i = 1, 2, \dots, N$$

$$\psi_t(i) = \arg \max_{1 \leq j \leq N} [\delta_{t-1}(j) a_{ji}], \quad i = 1, 2, \dots, N$$

(3) 终止

$$P^* = \max_{1 \leq i \leq N} \delta_T(i)$$

$$i_T^* = \arg \max_{1 \leq i \leq N} [\delta_T(i)]$$

(4) 最优路径回溯。对 $t = T-1, T-2, \dots, 1$

$$i_t^* = \psi_{t+1}(i_{t+1}^*)$$

求得最优路径 $I^* = (i_1^*, i_2^*, \dots, i_T^*)$ 。

-
- **核心变量:**
 - $\delta_t(i)$: 表示t时刻位于状态i的所有路径中的最大概率
 - $\psi_t(i)$: 记录t时刻状态i的最优前驱状态
- **迭代关系:**
 - $\delta_t(i)$ 通过前一个时刻所有可能状态的转移概率最大值计算
 - 记录使 $\delta_{t-1}(j)a_{ji}$ 最大的j值
- **维特比算法的初始化 04:20:03**
 - **初始条件:**
 - $\delta_1(i) = \pi_i b_i(o_1), \quad i = 1, 2, \dots, N$
 - $\psi_1(i) = 0, \quad i = 1, 2, \dots, N$
 - **物理意义:** 第一个时刻没有前驱状态, 概率仅由初始状态和观测决定
- **维特比算法的递推 04:20:33**

图维特比算法 宏观全局

动态规划

时刻1: $\delta_1(i) = \pi_i b_i(o_1)$

$\delta_1(1) = 0.2 \times 0.5 = 0.10$
 $\delta_1(2) = 0.4 \times 0.4 = 0.16$
 $\delta_1(3) = 0.4 \times 0.7 = 0.28$

$\Psi_1(i) = 0$

时刻2: $\delta_2(i) = \max_j \{\delta_1(j) a_{ji} b_i(o_2)\}$

$\delta_2(1) = 0.056 \times 0.5 = 0.028$
 $\delta_2(2) = 0.0504$
 $\delta_2(3) = 0.042$

时刻3: $\delta_3(i) = \max_j \{\delta_2(j) a_{ji} b_i(o_3)\}$

$\delta_3(1) = 0.00756$
 $\delta_3(2) = 0.01008$
 $\delta_3(3) = 0.0147$

$\Psi_3(1) = 2$
 $\Psi_3(2) = 2$
 $\Psi_3(3) = 3$

最优路径: $3 \rightarrow 3 \rightarrow 3$

维特比算法的终止: $i_1^* = i_2^* = i_3^* = 3$

最终概率: $P^* = \delta_3(3) = 0.0147$

○

○ 递推步骤:

■ 对每个时刻 $t=2, 3, \dots, T$:

- 计算 $\delta_t(i) = \max_{1 \leq j \leq N} [\delta_{t-1}(j) a_{ji} b_i(o_t)]$
- 记录

○ 实际案例: 在示例中, $t=2$ 时刻三个状态的最优前驱都是状态3

● 维特比算法的终止 04:21:23

算法 10.5 (维特比算法)

输入: 模型 $\lambda = (A, B, \pi)$ 和观测 $O = (o_1, o_2, \dots, o_T)$;
 输出: 最优路径 $I^* = (i_1^*, i_2^*, \dots, i_T^*)$.

(1) 初始化

$\delta_1(i) = \pi_i b_i(o_1), \quad i = 1, 2, \dots, N$
 $\Psi_1(i) = 0, \quad i = 1, 2, \dots, N$

(2) 递推. 对 $t = 2, 3, \dots, T$

$\delta_t(i) = \max_{1 \leq j \leq N} [\delta_{t-1}(j) a_{ji} b_i(o_t)], \quad i = 1, 2, \dots, N$
 $\Psi_t(i) = \arg \max_{1 \leq j \leq N} [\delta_{t-1}(j) a_{ji}], \quad i = 1, 2, \dots, N$

(3) 终止

$P^* = \max_{1 \leq i \leq N} \delta_T(i)$
 $i_T^* = \arg \max_{1 \leq i \leq N} [\delta_T(i)]$

(4) 最优路径回溯. 对 $t = T-1, T-2, \dots, 1$

$i_t^* = \Psi_{t+1}(i_{t+1}^*)$

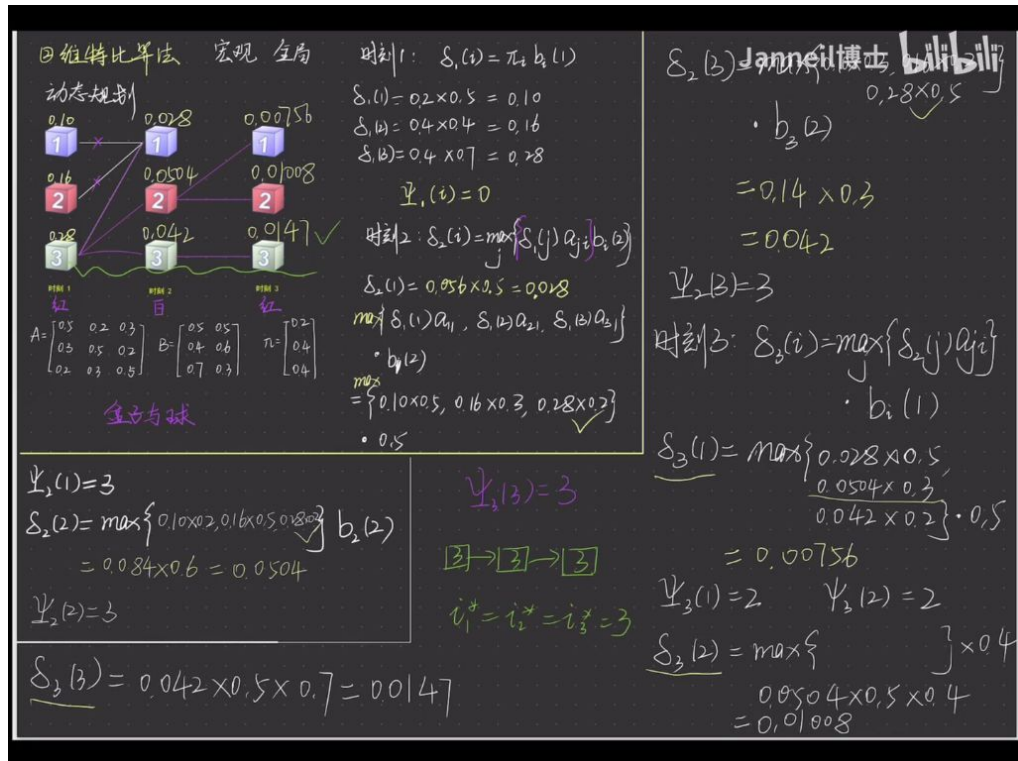
求得最优路径 $I^* = (i_1^*, i_2^*, \dots, i_T^*)$.

○

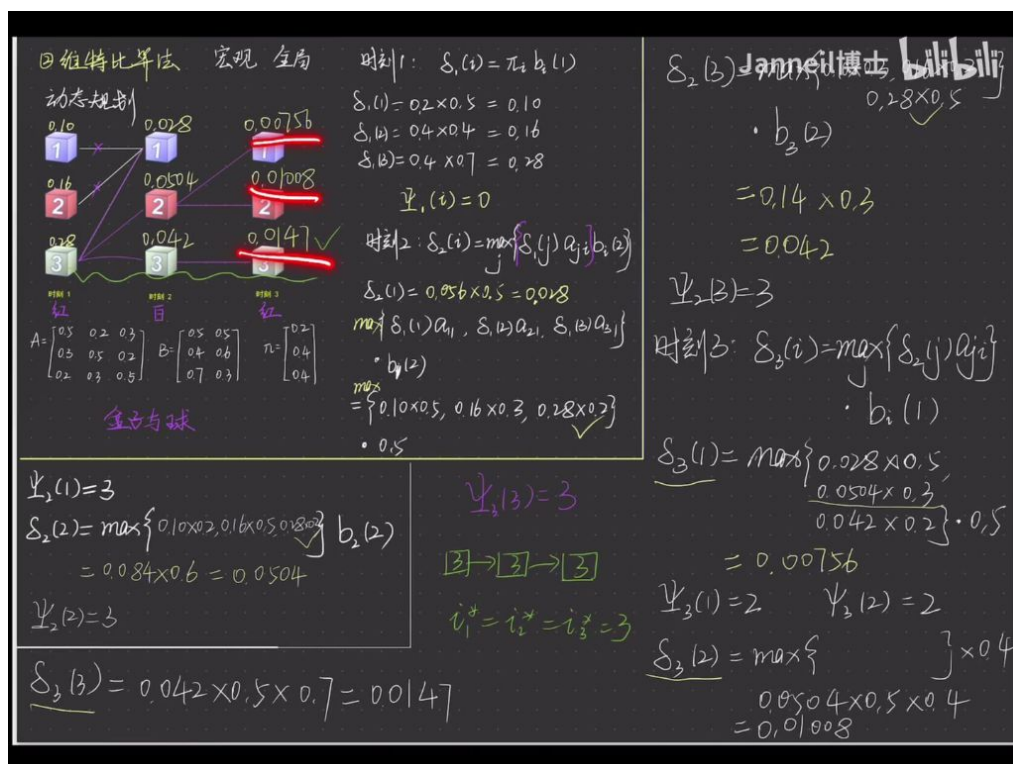
○ 终止条件:

■ $P^* = \max_{1 \leq i \leq N} \delta_T(i)$

- 物理意义：找到最终时刻概率最大的状态作为路径终点
- 维特比算法的最优路径回溯 04:21:54



- 回溯方法:
 - 对 $t = T-1, T-2, \dots, 1$:
- 选择依据：比较多条候选路径的概率，选择概率最大的路径
- 维特比算法与费马原理的相似性 04:22:16
 - 类比关系：
 - 费马原理：光沿能量消耗最小的路径传播
 - 维特比算法：选择概率最大的状态转移路径
 - 本质联系：都是优化问题，一个最小化能量消耗，一个最大化概率
- 隐马尔可夫模型回顾 04:23:55



三、知识小结

知识点	核心内容	关键公式/算法	应用示例
马尔可夫模型	两个核心要素: 初始状态分布 π 和状态转移矩阵 A 未来只与现在有关的马尔可夫性	$P(s_{t+1} s_t)$ 状态转移方程	天气预报状态转移
隐马尔可夫模型	新增观测概率矩阵 B 构成三要素 状态序列不可见, 只能观测输出值	$\lambda = (\pi, A, B)$ 观测生成过程	三硬币问题 盒子小球实验
前向算法	按时间步正向计算概率 时间复杂度 $O(N^2T)$	$\alpha_t(i) = P(o_1, \dots, o_t, i_t = q_i \lambda)$ $\alpha_{t+1}(j) = [\sum_{i=1}^N \alpha_t(i) a_{ij}] b_j(o_{t+1})$	计算 $P(\text{红, 白, 红})$ 观测序列概率
后向算法	逆向递归计算概率 与前向算法复杂度相同	$\beta_t(i) = P(o_{t+1}, \dots, o_T i_t = q_i, \lambda)$ $\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)$	与前向结合计算状态概率

Baum-Welch算法	EM算法在HMM中的实现 通过迭代优化参数	E步: 计算 ξ 和 γ M步: $\pi_i = \gamma_1(i)$ $a_{ij} = \frac{\sum \xi_t(i,j)}{\sum \gamma_t(i)}$ $b_j(k) = \frac{\sum_{o_t=v_k} \gamma_t(j)}{\sum \gamma_t(j)}$	仅凭观测序列 学习模型参数
维特比算法	动态规划求最优路径 全局最优解	$\delta_t(i) = \max_{i_{1:t-1}} P(i_{1:t-1}, i_t$ $= q_i o_{1:t} \lambda)$	预测最可能的 状态序列 (3→3→3)
近似算法	贪心算法求局部最优 计算效率高但可能非全局最优	$\gamma_t(i) = P(i_t = q_i O, \lambda)$	预测状态序列 (3→2→3)