# 第七章

## 7.1

D

## 7.2

BD

## 7.3

$P(G,S,R,T)=P(G|S,R)P(S|C)P(R|C)P(C)$

$P_{1111}=P(G=1|S=1,R=1)P(S=1|C=1)P(R=1|C=1)P(C=1)=0.99*0.1*0.8*0.5=0.0396$

$P_{1110}=P(G=1|S=1,R=1)P(S=1|C=0)P(R=1|C=0)P(C=0)=0.99*0.5*0.2*0.5=0.0495$

$P_{1101}=P(G=1|S=1,R=0)P(S=1|C=1)P(R=1|C=1)P(C=1)=0.9*0.1*0.2*0.5=0.009$

$P_{1100}=P(G=1|S=1,R=0)P(S=1|C=0)P(R=0|C=0)P(C=0)=0.9*0.5*0.8*0.5=0.018$

$P_{1011}=P(G=1|S=0,R=1)P(S=0|C=1)P(R=1|C=1)P(C=1)=0.9*0.8*0.8*0.5=0.0324$

$P_{1010}=P(G=1|S=0,R=1)P(S=0|C=0)P(R=1|C=0)P(C=0)=0.9*0.5*0.2*0.5=0.045$

$P_{1001}=P(G=1|S=0,R=0)P(S=0|C=1)P(R=0|C=1)P(C=1)=0.0*0.9*0.2*0.5=0.0$

$P_{1000}=P(G=1|S=0,R=0)P(S=0|C=0)P(R=0|C=0)P(C=0)=0.0*0.5*0.8*0.5=0.0$

$P(S=1|G=1)=\frac{P_{1111}+P_{1110}+P_{1101}+P_{1100}}{P_{1111}+P_{1110}+P_{1101}+P_{1100}+P_{1011}+P_{1010}+P_{1001}+P_{1000}}=\frac{2781}{6471}$

最后求得$P(S=1|R=1)=0.7079$

## 7.4

初始化参数$\theta^{(1)}$
E步：记$\theta^{(t)}$为第$t$次迭代参数的估计值，计算对数联合概率分布$ln(X,Z|\theta)$关于隐变量$Z$的后验概率分布的期望，即：
$Q(\theta|\theta^{(t)}=E_{Z|X,\theta^{(t)}}lnp(X,Z|\theta)=\int p(Z|X,\theta^{(t)})lnp(X,Z|\theta)dZ$
M步：求解使得$Q(\theta|\theta^{(t)}$最大化的$\theta$，得到第$t+1$次迭代的参数估计：
$\theta^{(t+1)}=argmax_\theta Q(\theta|\theta^{(t)}$
直到满足收敛条件

## 7.5

D

## 7.6

C

## 7.7

列出所有可能转移到观测序列（$O1O2O3$）的隐藏状态序列：

X1X2X3

X1X1X1

X1X1X2

X1X2X2

X2X2X2

X2X3X3

X2X2X3

X3X3X3

计算出所有隐藏状态观测序列为$O_1O_2O_3$的联合概率为：

$P(X1, X1, X1, O1, O2, O3) = 0.5 * 0.5 * 0.5 * 0.3 * 0.3 * 0.4$

$P(X1, X1, X2, O1, O2, O3) = 0.5 * 0.3 * 0.5 * 0.3 * 0.5 * 0.4$

$P(X1, X2, X2, O1, O2, O3) = 0.5 * 0.3 * 0.5 * 0.4 * 0.5 * 0.4$

$P(X1, X2, X3, O1, O2, O3) = 0.5 * 0.3 * 0.5 * 0.4 * 0.5 * 0.3$

$P(X2, X2, X2, O1, O2, O3) = 0.3 * 0.2 * 0.5 * 0.4 * 0.5 * 0.4$

$P(X2, X2, X3, O1, O2, O3) = 0.3 * 0.2 * 0.5 * 0.4 * 0.5 * 0.3$

$P(X2, X3, X3, O1, O2, O3) = 0.3 * 0.2 * 0.5 * 0.6 * 1 * 0.3$

$P(X3, X3, X3, O1, O2, O3) = 0.2 * 0.1 * 1 * 0.6 * 1 * 0.3$

P(X1X2X3|O1O2O3) = $\frac{4500}{4500+6000+4500+2400+1800+5400+3600} = \frac{45}{282}$

```python
import numpy as np

A = np.array(
    [
        [0.5, 0.5, 0],
        [0, 0.5, 0.5],
        [0, 0, 1]
    ]
)

B = np.array(
    [
        [0.3, 0.3, 0.4],
        [0.2, 0.4, 0.4],
        [0.1, 0.6, 0.3]
    ]
)

pi = np.array([0.5, 0.3, 0.2])

a = np.zeros((3, 3))
a[0] = pi * B.T[0]
for i in range(2):
    a[i + 1] = [np.dot(a[i], A.T[j]) for j in range(3)] * B.T[i + 1]

p_o = np.sum(a[2])

p = 0.5 * 0.3 * 0.5 * 0.4 * 0.5 * 0.3

print(p / p_o)
```

## 7.8

D

## 7.9

B

## 7.10

C

## 7.11

**（1）计算初值**
$\beta_4(i)=1\ \ i=1,2,3$
**（2）递推计算**
$\beta_3(1)=\sum_{j=1}^{3}a_{1j}b_j(O_4)\beta_4(j)=0.25+0.12+0.09=0.46$

$\beta_3(2)=\sum_{j=1}^{3}a_{2j}b_j(O_4)\beta_4(j)=0.15+0.3+0.09=0.51$

$\beta_3(3)=\sum_{j=1}^{3}a_{3j}b_j(O_4)\beta_4(j)=0.1+0.18+0.15=0.43$

$\beta_2(1)=\sum_{j=1}^{3}a_{1j}b_j(O_2)\beta_3(j)=0.25*0.46+0.08*0.51+0.21*0.43=0.2461$

$\beta_2(2)=\sum_{j=1}^{3}a_{2j}b_j(O_3)\beta_3(j)=0.15*0.46+0.2*0.51+0.14*0.43=0.2312$

$\beta_2(3)=\sum_{j=1}^{3}a_{3j}b_j(O_3)\beta_3(j)=0.1*0.46+0.12*0.51+0.35*0.43=0.2577$

$\beta_1(1)=\sum_{j=1}^{3}a_{1j}b_j(O_2)\beta_2(j)=0.25*0.2461+0.12*0.2312+0.09*0.2577=0.112462$

$\beta_1(2)=\sum_{j=1}^{3}a_{2j}b_j(O_2)\beta_2(j)=0.15*0.2461+0.3*0.2312+0.06*0.2577=0.121737$

$\beta_1(3)=\sum_{j=1}^{3}a_{3j}b_j(O_2)\beta_2(j)=0.1*0.2461+0.18*0.2312+0.15*0.2577=0.104881$

（3）终止
$P(O\mid\lambda)=0.2*0.5*0.112462+0.4*0.4*0.121737+0.4*0.7*0.104881=0.0600908$

```
A = np.array(
    [
        [0.5, 0.2, 0.3],
        [0.3, 0.5, 0.2],
        [0.2, 0.3, 0.5]
    ]
)

B = np.array([
    [0.5, 0.5],
    [0.4, 0.6],
    [0.7, 0.3]
])

pi = np.array([
    0.2, 0.4, 0.4
])

beta = np.ones((5, 5))
```

```python
for i in range(3, 0, -1):
    for j in range(0, 3):
        beta[i][j] = np.dot(beta[i + 1][0:3] * A[j], B.T[i % 2])

p = np.dot(pi, beta[1][0:3] * B.T[0])

print(p)
```