第三次作业（贝叶斯网络）

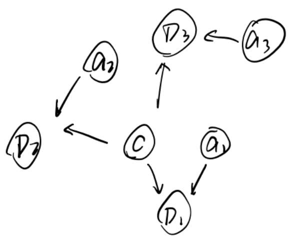
崔士强　PB22151743

2024 年 6 月 9 日

问题 1：概率推断

a.

1. (a)、



b.

(b)、$P(C=c,\ D_1=d_1,\ D_2=d_2,\ D_3=d_3)$

$$= p(c)\,p(d_1|c,a_1)\,p(d_2|c,a_2)\,p(d_3|c,a_3)$$

c.

(c)、$LHS = \dfrac{P(C=c, D_1=d_1, \cdots, D_t=d_t)}{P(D_1=d_1, \cdots, D_t=d_t)}$

$$= \frac{p(c)p(d_1|c)\cdots p(d_t|c)}{p(d_t|d_1,\cdots,d_{t-1})\cdot p(d_1\cdots d_{t-1})}$$

$RHS = \dfrac{p(c)p(d_1|c)\cdots p(d_{t-1}|c)}{P(D=d_1,\cdots,D_{t-1}=d_{t-1})} \cdot p(d_t|c)$

$= \dfrac{LHS}{p(d_t|d_1,\cdots,d_{t-1})}$　d_1 至 d_{t-1} 已知　则 LHS = RHS · Const.

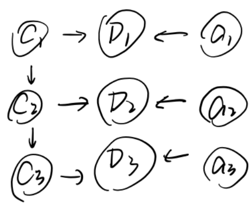
d.

```python
def observe(self, agentX: int, agentY: int, observedDist: float) -> None:
    # BEGIN_YOUR_CODE (our solution is 7 lines of code, but don't worry if you
        deviate from this)
    for row in range(self.belief.numRows):
        for col in range(self.belief.numCols):
            prob = self.belief.getProb(row, col)    # Probability at current index
            targetX, targetY = util.colToX(col), util.rowToY(row)   # Get location
                of current index
            distance = math.sqrt(math.pow(agentX-targetX, 2)+math.pow(agentY-targetY
                , 2))
            multiplier = util.pdf(distance, Const.SONAR_STD, observedDist)
            self.belief.setProb(row, col, prob*multiplier)
    self.belief.normalize()
    # END_YOUR_CODE
```

问题 2：转移概率

a.

2. (a).



b.

(b). $P(C_1=c_1, C_2=c_2, C_3=c_3, D_1=d_1, D_2=d_2, D_3=d_3)$

$= p(c_1)p(c_2|c_1)p(c_3|c_2)p(d_1|c_1,a_1)p(d_2|c_2,a_2)p(d_3|c_3,a_3)$

c.

(c)、 $LHS = \sum_{C_t} P(C_{t+1}, C_t \mid d_1, \cdots, d_t)$

$$P(C_{t+1}, C_t \mid d_1, \cdots, d_t) = \frac{P(C_{t+1}, C_t, d_1, \cdots, d_t)}{P(d_1, \cdots, d_t)}$$

$$= \frac{P(C_{t+1} \mid C_t, d_1, \cdots, d_t)\, P(C_t, d_1, \cdots, d_t)}{P(d_1, \cdots, d_t)}$$

$$\parallel$$

$$P(C_{t+1} \mid C_t)$$

$$= P(C_t \mid d_1 \cdots d_t)\, P(C_{t+1} \mid C_t)$$

$$LHS = \sum_{C_t} P(C_t \mid d_1, \cdots, d_t)\, P(C_{t+1} \mid C_t)$$

d.

```
def elapseTime(self) -> None:
    if self.skipElapse: ### ONLY FOR THE GRADER TO USE IN Problem 1
        return
    # BEGIN_YOUR_CODE (our solution is 7 lines of code, but don't worry if you
        deviate from this)
    temp_belief = self.belief
    for row in range(self.belief.numRows):
        for col in range(self.belief.numCols):
            prob = 0.0
            for key in self.transProb:
                if key[1] == (row, col):
                    prob += self.belief.getProb(key[0][0], key[0][1])*self.transProb
                        [key]
            temp_belief.setProb(row, col, prob)
    temp_belief.normalize()
    self.belief = temp_belief
    # END_YOUR_CODE
```

问题 3：是哪辆车?

a.

3、(a)、

$$p(c_{11}, c_{12} \mid e_1) = \frac{p(c_{11}, c_{12}, e_1)}{p(e_1)} = \frac{p(e_1 \mid c_{11}, c_{12})\, p(c_{11}, c_{12})}{p(e_1)}$$

$= p(c_{11})p(c_{12})$

$$p(e_1 \mid c_{11}, c_{12}) = 0.5 \cdot p_N(e_1; \|a_1 - c_{11}\|_2, \sigma^2) + 0.5 \cdot p_N(e_1; \|a_1 - c_{12}\|_2, \sigma^2)$$

即 $p(c_{11}, c_{12} \mid e_1) \propto$

$$[p_N(e_1; \|a_1 - c_{11}\|_2, \sigma^2) + p_N(e_1; \|a_1 - c_{12}\|_2, \sigma^2)] \cdot p(c_{11})\, p(c_{12})$$

b.

(b). 设 z_t 为 t 时间步索引向右移动的数目，取值为 $0, 1, \cdots K-1$

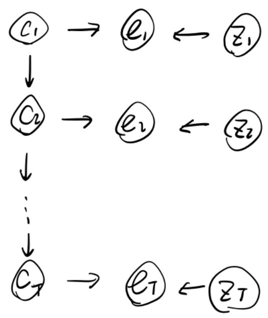
对于每一个 z_t，取 $[0, K-1]$ 上各整数值概率均等，i.e. $\frac{1}{K}$

$$p(c_t \mid c_{t-1}) = \sum_{z_t} p(c_t \mid c_{t-1}, z_t)\, p(z_t)$$

$$= \frac{1}{K} \sum_{z_t} p(c_t \mid c_{t-1}, z_t)$$

c.

(c)、



$$p(C_t, e_1, \cdots, e_T) = p(C_t|C_{t-1})\, p(e_1|C_1, z_1) \cdots p(e_T|C_T, z_T)$$
$$= p(C_t|C_{t-1})$$
$$= p(C_t|e_1, \cdots e_T)\, p(e_1, \cdots, e_T)$$

$$p(e_1, \cdots, e_T) = p(e_1|C_1, z_1) \cdots p(e_T|C_T, z_T)$$

$$p(C_t|e_1, \cdots, e_T) = \frac{1}{k} \sum_{z_t} p(C_t|C_{t-1}, z_t)$$

d.

粒子滤波相比精确推理，速度上有明显的提升，另外成功率也有显著改善，在车辆数增加的情况下也是如此。

原因是精确推理需要存储整个空间上的概率分布，导致计算的复杂度很高。而粒子滤波使用一组有限数量的粒子，计算复杂度主要取决于粒子的数量而非状态空间。

问题 4：模型学习

a.

4.(a).

$$\begin{tikzcd} Z_1 \arrow[r] \arrow[d] & Z_2 \arrow[d] \\ X_1 & X_2 \end{tikzcd}$$

X_1	X_2
T	F
T	T

E步： 第一个分量：

$$P(z_1|x_1) = \frac{P(x_1|z_1)P(z_1)}{P(x_1)}$$

$$P(z_1=true|x_1=true) = \frac{0.9\cdot 0.6}{1} = 0.54$$

$$P(z_1=false|x_1=true) = 0.46$$

$$P(z_2|x_2) = \frac{P(x_2|z_2)P(z_2|z_1)}{P(x_2)}$$

$$P(z_2=true|x_2=false)$$

$$= 0.7\cdot(0.54\times 0.7 + 0.46\times 0.2) = 0.329$$

$$P(z_2=false|x_2=false) = 0.671$$

第二个分量 (即 x_2 改为 true)

$$P(z_2=true \mid x_2=true)$$

$$= 0.3\cdot(0.57\times 0.7 + 0.46\times 0.2) = 0.141$$

$$P(z_2=false \mid x_2=true) = 0.859$$

反馈

- 代码部分没有难度，书写部分 3，4 题感觉有些难度，无论是在理解题意上还是在解题上。
- 整体用时在 20h 左右，大部分是在做计算以及证明题。