Homework 3

Bayesian Network

姓名：高茂航

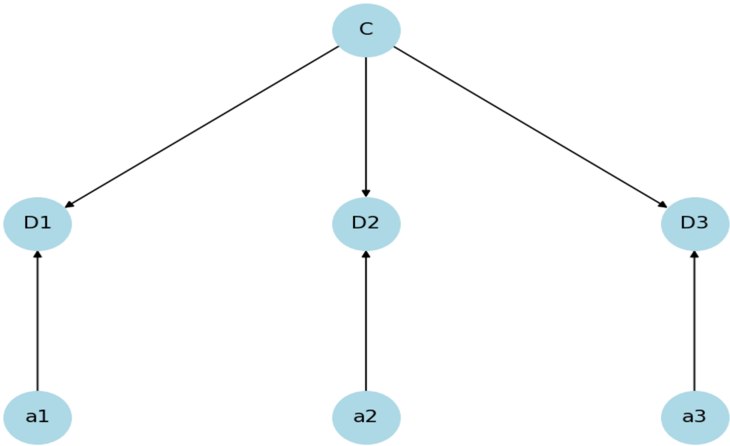
学号：PB22061161

日期：2024.6.1

Homework 3

1 概率推断 [25%]

1.1



1.2

$\because D_t$ 只依赖于 C 和 a_t，

$\therefore P(C = c, D_1 = d_1, D_2 = d_2, D_3 = d_3) = P(C = c)P(D_1 = d_1|C = c, a_1)P(D_2 = d_2|C = c, a_2)P(D_3 = d_3|C = c, a_3)$

1.3

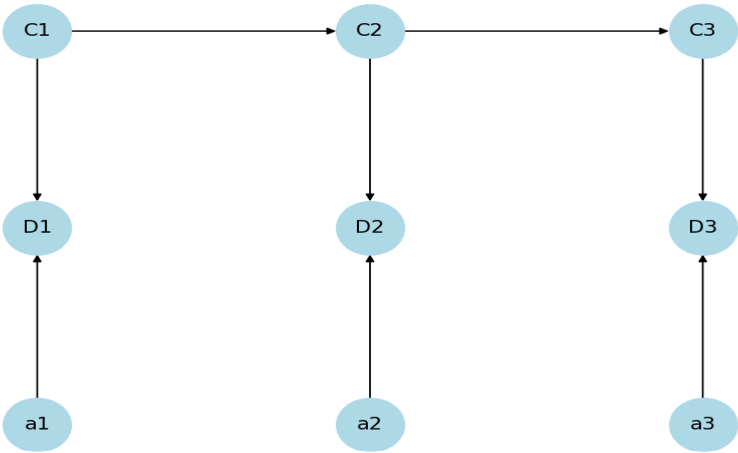
$$
\begin{aligned}
P(C = c|D_1 = d_1, ..., D_t = d_t) &= \frac{P(C = c, D_1 = d_1, ..., D_{t-1} = d_{t-1}, D_t = d_t)}{P(D_1 = d_1, ..., D_{t-1} = d_{t-1}, D_t = d_t)} \\
&= \frac{P(C = c)P(D_1 = d_1|C = c)P(D_2 = d_2|C = c)...P(D_t = d_t|C = c)}{P(D_1 = d1, ..., D_{t-1} = d_{t-1}, D_t = d_t)} \\
&= \frac{P(C = c, D_1 = d_1, ..., D_{t-1} = d_{t-1})P(D = d_t|C = c)}{P(D_1 = d_1, ..., D_{t-1} = d_{t-1}, D_t = d_t)} \\
&= \frac{P(C = c|D_1 = d_1, ..., D_{t-1} = d_{t-1})P(D = d_t|C = c)}{P(D_t = d_t|D_1 = d_1, ..., D_{t-1} = d_{t-1})} \\
&\propto P(C = c|D_1 = d_1, ..., D_{t-1} = d_{t-1})P(D = d_t|C = c)
\end{aligned}
$$

1.4

```
for i in range(self.belief.getNumRows()):
  for j in range(self.belief.getNumCols()):
    dist = math.sqrt((agentX - util.colToX(j))**2 + (agentY - util.rowToY(i))**2)
    prob = util.pdf(dist, Const.SONAR_STD, observedDist)
    self.belief.setProb(i, j, self.belief.getProb(i, j) * prob)
self.belief.normalize()
```

2 转移概率 [25%]

2.1



2.2

$\because D_t$ 依赖于 C_t 和 a_t，且 C_t 依赖于 C_{t-1}，

$\therefore P(C_1 = c_1, C_2 = c_2, C_3 = c_3, D_1 = d_1, D_2 = d_2, D_3 = d_3) = P(C_1 = c_1)P(D_1 = d_1|C_1 = c_1, a_1)P(C_2 = c_2|C_1 = c_1)P(D_2 = d_2|C_2 = c_2, a_2)P(C_3 = c_3|C_2 = c_2)P(D_3 = d_3|C_3 = c_3, a_3)$

2.3

$$
\begin{aligned}
P(C_{t+1} = c_{t+1}|D_1 = d_1, ..., D_t = d_t) &= \frac{P(C_{t+1} = c_{t+1}, D_1 = d_1, ..., D_{t-1} = d_{t-1}, D_t = d_t)}{P(D_1 = d_1, ..., D_{t-1} = d_{t-1}, D_t = d_t)} \\
&= \sum_{c_t} \frac{P(C_{t+1} = c_{t+1}, C_t = c_t, D_1 = d_1, ..., D_{t-1} = d_{t-1}, D_t = d_t)}{P(D_1 = d_1, ..., D_{t-1} = d_{t-1}, D_t = d_t)} \\
&= \sum_{c_t} \frac{P(C_t = c_t, D_1 = d_1, ..., D_t = d_t)P(C_{t+1} = c_{t+1}|C_t = c_t, ..., D_t = d_t)}{P(D_1 = d_1, ..., D_{t-1} = d_{t-1}, D_t = d_t)} \\
&= \sum_{c_t} \frac{P(C_t = c_t, D_1 = d_1, ..., D_t = d_t)P(C_{t+1} = c_{t+1}|C_t = c_t)}{P(D_1 = d_1, ..., D_{t-1} = d_{t-1}, D_t = d_t)} \\
&\propto \sum_{c_t} P(C_t = c_t|D_1 = d_1, ..., D_t = d_t)P(C_{t+1} = c_{t+1}|C_t = c_t)
\end{aligned}
$$

2.4

```
        newBelief = util.Belief(self.belief.getNumRows(), self.belief.getNumCols(), 0)
        for oldRow in range(self.belief.getNumRows()):
            for oldCol in range(self.belief.getNumCols()):
                grid = self.belief.grid[oldRow][oldCol]
                for newRow in range(self.belief.getNumRows()):
                    for newCol in range(self.belief.getNumCols()):
                        transProb = self.transProb.get(((oldRow, oldCol), (newRow, newCol)),
    0)
                        newBelief.addProb(newRow, newCol, grid * transProb)
```

```
        newBelief.normalize()
        self.belief = newBelief

```

3 是哪辆车?[30%]

3.1

$$
\begin{aligned}
p(c_{11}, c_{12}|e_1) &\propto p(C_{11} = c_{11}, C_{12} = c_{12}, E_1 = e_1) \\
&= p(c_{11}, c_{12}) p(e_1, e_2 | c_{11}, c_{12}) \\
&= \frac{1}{2} p(c_{11}) p(c_{12}) [p(D_{11} = e_{11}|c_{11}) p(D_{12} = e_{12}|c_{12}) + p(D_{11} = e_{12}|c_{11}) p(D_{12} = e_{11}|c_{12})] \\
&= \frac{1}{2} p(c_{11}) p(c_{12}) [p_N(e_{11}; \|a_1 - c_{11}\|^2, \sigma^2) p_N(e_{12}; \|a_1 - c_{12}\|^2, \sigma^2) \\
&\quad + p_N(e_{12}; \|a_1 - c_{11}\|^2, \sigma^2) p_N(e_{11}; \|a_1 - c_{12}\|^2, \sigma^2)]
\end{aligned}
$$

3.2

设循环群

$$
g = \begin{pmatrix} 1, 2, \ldots, K \\ 2, 3, \ldots, 1 \end{pmatrix}
$$

则 $z_t \in \{g^n | n = 0, 1, ..., K-1\}$，用于表示在每个时间步 t，传感器返回的车辆位置列表相对于真实位置列表的偏移量，即 $e_t = z_t(c_t)$。

$$
\begin{aligned}
p(c_t|c_{t-1}) &= \frac{p(c_t, c_{t-1})}{p(c_{t-1})} \\
&= \sum_{z_t} \frac{p(c_t, c_{t-1}, z_t)}{p(c_{t-1})} \\
&= \sum_{z_t} \frac{p(c_t, c_{t-1}, z_t) p(c_{t-1}, z_t)}{p(c_{t-1}, z_t) p(c_{t-1})} \\
&= \sum_{z_t} p(c_t | c_{t-1}, z_t) p(z_t | c_{t-1}) \\
&= \frac{1}{K} \sum_{z_t} p(c_t | c_{t-1}, z_t) \\
&= \frac{1}{K} \sum_{z_t} p(z_t^{-1}(e_t) | z_{t-1}^{-1}(e_{t-1}), z_t)
\end{aligned}
$$

3.3

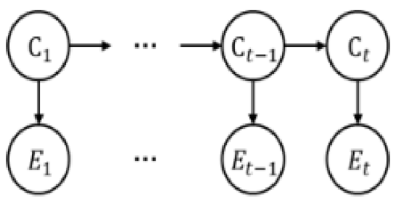


图 1: 贝叶斯网络

因为 E_t 由 C_t 和 Z_t 确定，且 Z_t 全为均匀分布（概率密度为 1/K），因此上图没有画 Z_t。

算法描述如下：

1. 前向

$$F_1(c_1) = P(C_1 = c_1)P(E_1 = e_1|C_1 = c_1) = \frac{1}{4}P(C_1 = c_1)$$

$$\begin{aligned} F_i(c_i) &= \sum_{C_{i-i}} F_{i-1}(C_{i-1})P(C_i = c_i|C_{i-1})P(E_i = e_i|C_i = c_i) \\ &= \frac{1}{4}\sum_{C_{i-i}} F_{i-1}(C_{i-1})P(C_i = c_i|C_{i-1}) \end{aligned}$$

2. 后向

$$B_T(c_T) = 1$$
$$B_i(c_i) = \frac{1}{4}\sum_{B_{i+i}} F_{i+1}(C_{i+1})P(C_i = c_i|C_{i+1})$$

3. 计算概率

$$P(c_t|e_1, ..., e_T) = \frac{F_i(c_i)B_i(c_i)}{\sum\limits_{C_t} F_i(C_i)B_i(C_i)}$$

3.4

精确推理试图直接计算出概率分布的完整表达式，因而耗时较久；相比之下粒子滤波方法更加高效，且在多车辆时 error 值更低、胜率更高。因为粒子滤波方法只关注了概率较大的区域，且由于采样时会减少对小概率区域的权重，因此小概率区域对大概率区域的影响不断减小，进而提高了准确度。

4 模型学习 [10%]

4.1

表 1: E step

X_1	X_2	Z_1	Z_2	$P(X_1, X_2, Z_1, Z_2)$	$q((Z_1, Z_2))$
true	false	true	true	0.2646	0.628
true	false	true	false	0.0648	0.154
true	false	false	true	0.028	0.066
true	false	false	false	0.064	0.152
true	true	true	true	0.1134	0.356
true	true	true	false	0.0972	0.305
true	true	false	true	0.012	0.038
true	true	false	false	0.096	0.301

表 2: M step

Z$_1$	count	$p_{Z_1}(Z_1)$
true	1.443	0.72
false	0.557	0.28

Z$_1$	Z$_2$	count	$p_{Z_2}(Z_2\|Z_1)$
true	true	0.984	0.68
true	false	0.459	0.32
false	true	0.104	0.19
false	false	0.453	0.81

5 反馈 [10%]

本次实验花了近一周的空闲时间，主要是在第三部分卡了很久，经过讨论，得知应该是题目本身就有问题。