

# HW4

r12922016 葉丞勛

## 1. Bayes Net Structure

：把五個 variable [PAC, GHOST0, GHOST1, OBS0, OBS1] 的 domain 決定好，其中 PAC、GHOST0、GHOST 的 domain 是全部的 position，OBS0、OBS1 則是 pacman 與 ghost 的所有可能的 Manhattan distances 加上 noise。

```
### Question q1: 2/2 ###
```

```
Finished at 17:26:54
```

```
Provisional grades
```

```
=====
```

```
Question q1: 2/2
```

```
-----
```

```
Total: 2/2
```

## 2. Join Factors

：首先將所有 conditionVariables 存在一個 set 裡面，所有 unConditioned Variables 存在另一個 set，接著所有在 conditionVariables set 裡面的 unConditioned Variables 移除掉，最後透過 Factor() 來產生新的 join Factor。而對於所有可能的 Assignment，我們去 given factors 裡面找出所有這個 Assignment 的機率，然後把他們與 1 相乘，最後就可以得到每個 Assignment 的機率。

```
### Question q2: 3/3 ###
```

```
Finished at 17:38:44
```

```
Provisional grades
```

```
=====
```

```
Question q2: 3/3
```

```
-----
```

```
Total: 3/3
```

### 3. Eliminate

：首先在 conditionVariables 跟 unConditionVariables 裡將 eliminated Variable 移掉，然後透過 Factor() 產生新的 eliminated factor。而對於所有可能的 Assignment，我們先去取得 eliminated Variable 的 domain，接著對 domain 裡每個可能的值，我們讓 Assignment 裡 eliminated Variable 的值等於這個值，接著去取得 given factor 裡，這個改過後的 Assignment 的機率，並累加起來，直到找完 domain 裡所有的值。最後我們就可以得到 eliminated factor 裡每個 Assignment 的機率了。

```
### Question q3: 2/2 ###
```

```
Finished at 18:20:32
```

```
Provisional grades
```

```
=====
```

```
Question q3: 2/2
```

```
-----
```

```
Total: 2/2
```

### 4. Variable Elimination

：首先透過 getAllCPTsWithEvidence() 得到已知 evidence 的 factor：currentFactorsList，接著對於 eliminationOrder 裡所有的 joinVariable，先透過 joinFactorsByVariable() 去找出新的 currentFactorsList 以及 factor，若是新的 factor 的 unConditioned variables 數量大於 1，就將此 factor 裡的 joinVariable 移除，並將處理完的 factor 新增至 currentFactorsList。處理完 eliminationOrder 裡所有的 variables 後，就將 currentFactorsList 丟進 joinFactors() 產生新的 joint，並對其做 normalize()，得到最後的結果。

```
### Question q4: 2/2 ###
```

```
Finished at 18:33:29
```

```
Provisional grades
```

```
=====
```

```
Question q4: 2/2
```

```
-----
```

```
Total: 2/2
```

## 5a. DiscreteDistribution Class

### (1) normalize

：先透過 `total()` 取得全部加總的值，若是  $total > 0$ ，則將所有 `item` 的值除以 `total`。

### (2) sample

：先透過 `total()` 取得全部加總的值，並透過 `random.uniform()` 取得一個  $0 \sim total$  的隨機 `threshold`，並設定目前累積的值=0。接著對於所有的 `item`，若是目前累積的值加上這個 `item` 的 `value` 大於 `threshold` 的話，就選擇這個 `item`；若沒有大於，就將累積的值加上這個 `item` 的 `value`，繼續做下去，直到找到一個符合條件的 `item`。

## 5b. Observation Probability

：首先判斷若是 `ghostPosition` 等於 `jailPosition`，當 `noisyDistance` 為 `None` 的時候就回傳 1，否則回傳 0。接著判斷若是 `noisyDistance` 為 `None` 的話，會直接回傳 0。然後先算出目前 `pacman` 與 `ghost` 的 `manhattanDistance`，然後透過 `getObservationProbability()` 去取得觀測到的機率，並回傳這個值。

```
### Question q5: 1/1 ###
```

```
Finished at 18:46:33
```

```
Provisional grades
```

```
=====
```

```
Question q5: 1/1
```

```
-----
```

```
Total: 1/1
```

## 6. Exact Inference Observation

：對於所有的位置，先透過 `getObservationProb()` 取得觀測到的機率，然後將這機率乘上這個位置本來的 `belief`，去做更新。

```
### Question q6: 2/2 ###
```

```
Finished at 21:04:33
```

```
Provisional grades
```

```
=====
```

```
Question q6: 2/2
```

```
-----
```

```
Total: 2/2
```

## 7. Exact Inference with Time Elapse

：首先透過 `DiscreteDistribution()` 產生一個空的 `newDistribution`。對於所有的位置，先透過 `getPositionDistribution()` 取得此位置的 `Distribution`，然後根據這個 `Distribution` 的包含的每個位置及對應的機率，將 `newDistribution` 這個位置的機率設定為這個位置本來的 `beliefs` 乘上剛剛拿到的對應的機率。全部做完之後對 `newDistribution` 做 `normalize()`，最後將這個 `newDistribution` 等於自己的 `beliefs`。

```
### Question q7: 2/2 ###
```

```
Finished at 19:50:10
```

```
Provisional grades
```

```
=====
```

```
Question q7: 2/2
```

```
-----
```

```
Total: 2/2
```

## 8. Exact Inference Full Test

：首先對 `livingGhostPositionDistributions` 做 `enumerate()`，只考慮還沒被抓到的鬼，然後利用 `argMax()` 將有著最高的機率的位置找出來，作為這隻鬼的位置，然後去計算這隻鬼跟 `pacman` 的距離，找出哪隻鬼與 `pacman` 最近。接著對於每個可能的 `action`，先算出 `pacman` 做完這個 `action` 後的位置，然後算出 `pacman` 這個位置與最近的鬼的距離，找出要做哪個 `action`，能讓 `pacman` 最靠近這隻鬼，而這個 `action` 就是要我們要找的。

```
### Question q8: 1/1 ###
```

```
Finished at 20:25:00
```

```
Provisional grades
```

```
=====
```

```
Question q8: 1/1
```

```
-----
```

```
Total: 1/1
```

## 9. Approximate Inference Initialization and Beliefs

### (1) initializeUniformly

：首先將全部的 particles 分成 `len(self.legalPositions)` 堆，然後紀錄分完剩下的 particles 數量，接著把沒分完的 particles 依序分到每堆裡面，直到分完為止。

### (2) getBeliefDistribution

：首先透過 `DiscreteDistribution()` 產生一個新的 `particleDistribution`，接著對於每個 particle，將 `particleDistribution[particles]` 加 1。做完後對其 `normalize()`，即可得到結果。

```
### Question q9: 1/1 ###
```

```
Finished at 20:51:40
```

```
Provisional grades
```

```
=====
```

```
Question q9: 1/1
```

```
-----
```

```
Total: 1/1
```

## 10. Approximate Inference Observation

：一樣先透過 `DiscreteDistribution()` 產生一個新的 `newDistribution`，接著對於每個 particle，先透過 `getObservationProb()` 取得觀測到的機率，然後將這個機率值累加到 `newDistribution[particle]`。做完之後，對 `newDistribution` 做 `normalize()`，若是 `newDistribution` 的總和為 0，則回傳 0，否則就對這個 `newDistribution` 取 `sample()`，然後將取到的 samples 更新到 `self.particles`。

```
### Question q10: 2/2 ###
```

```
Finished at 20:57:54
```

```
Provisional grades
```

```
=====
```

```
Question q10: 2/2
```

```
-----
```

```
Total: 2/2
```

## 11. Approximate Inference with Time Elapse

：先產生一個空的 afterParticles list，接著對於每個 particle，先透過 getPositionDistribution() 取得新的 newPosDistribution，然後對這過 distribution 做 sample，取得新的 particle 後，將新的 particle 新增到 afterParticles 裡。做完後，將 afterParticles 更新到 self.particles。

```
### Question q11: 2/2 ###
```

```
Finished at 21:03:59
```

```
Provisional grades
```

```
=====
```

```
Question q11: 2/2
```

```
-----
```

```
Total: 2/2
```