Fictitious Play
Assignment # 1

By
312581020
許瀚丰

Game Theory and Its Applications
Fall 2023
Date Submitted: October 23, 2023

- Setup of the Experiment
  - 在本次作業中，所有的任務皆是使用同一份程式碼完成，每個 question 皆以不同組 belief table 測試 5000 次，且每次皆跑 1000 rounds，並計算所有最終結果扣除初始 belief table 後的平均值與標準差。
  - 每次模擬中，初始的 belief table 皆是從 0~100 隨機 sample p 與 q 兩個數值，並設定 p2's belief = [p, 100 – p]，p1's belief = [q, 100 - q]。

```
# sample p, q from uniform distribution(0, 100)
p, q = np.random.random(2) * 100
# store the init p and q
_p.append(p)
_q.append(q)

# p2's belief of p1's action
cnt1 = np.array([p, 100 - p])
# p1's belief of p2's action
cnt2 = np.array([q, 100 - q])
```

  - 在調整 belief 時，若 payoff 計算結果相同，則透過隨機選擇一種來調整 belief。

```
# calculate payoff
payoff1, payoff2 = np.zeros((2, n))
for action_1 in range(n):
    for action_2 in range(m):
        payoff1[action_1] += cnt2[action_2] * game_matrix[action_1][action_2][0]
        payoff2[action_2] += cnt1[action_1] * game_matrix[action_1][action_2][1]
# choose the idx of max expected payoff
# if there are multiple max, choose one randomly
# update the belief
idx_1 = np.argwhere(payoff1 == np.max(payoff1)).flatten()
idx_1 = np.random.choice(idx_1, 1)
idx_2 = np.argwhere(payoff2 == np.max(payoff2)).flatten()
idx_2 = np.random.choice(idx_2, 1)
cnt1[idx_1] += 1
cnt2[idx_2] += 1
```

- Results of the Experiment

Q1. One pure-strategy Nash Equilibrium

- 在此題中，只有一個 pure-strategy 且一定會收斂於(r_2, c_2)。
- 雖初始的 belief table 有所不同，但由於(r_1, c_1)為負，且初始時並沒有 P(r_1)與 P(c_1)為負的情況，因此選擇 r_1 或 c_1 的 payoff 一定小於 0，而選擇 r_2 或 c_2 的 payoff 一定大於等於 0，因此不管是 p1 或 p2 來說，皆一定會選擇 r_2 與 c_2。
-

```
p1's belief mean: [   0. 1000.], p2 belief mean's: [   0. 1000.]
p1's belief std: [0. 0.], p2 belief std's: [0. 0.]
```

Q2. Two or more pure-strategy NE

- 在此題中，皆有可能收斂於 pure-strategy (r_1, c_1)與(r_2, c_2)。
- 在我的初始 belief table 中，若 p 與 q 兩者皆大於 50，則一定會收斂於(r_1, c_1)，而對於 p 與 q 皆小於 50 的情況，一定會收斂於(r_2, c_2)，若非上述情況，在我的觀察中由於(r_2, c_2)的 utility 較大，因此機率是 uniform 的情況下，會有較多的機率收斂於(r_2, c_2)的位置。
-

```
p1's belief mean: [499.767 500.233], p2 belief mean's: [499.772 500.228]
p1's belief std: [490.669 490.669], p2 belief std's: [490.728 490.728]
```

Q3. Two or more pure-strategy NE (Conti.)

- 在此題中，只會收斂至(r_1, c_1)。
- 與 Q1 類似，由於選擇 r_2 或 c_2 的 payoff 一定為 0，而在不考慮初始 belief table 選擇 r_1 或 c_1 的機率為 0 的情況下，選擇 r_1 或 c_1 的 payoff 必定大於 0，因此此題只會收斂於(r_1, c_1)。
-

```
p1's belief mean: [1000.    0.], p2 belief mean's: [1000.    0.]
p1's belief std: [0. 0.], p2 belief std's: [0. 0.]
```

Q4. Mixed-Strategy Nash Equilibrium mixed-strategy Nash equilibrium

- 在此題中，使用 fictitious play 會以 mixed-strategy Nash equilibrium 的方式收斂。
- 在我的實驗中，不管設定何種的初始 belief table，最後的 p1 與 p2 的 belief 結果靠近於[0.5, 0.5]與[0.8, 0.2]，且標準差並不大，因此此題會以會以 mixed-strategy Nash equilibrium 的形式收斂。
-
```
p1's belief mean: [452.276 547.724], p2 belief mean's: [831.215 168.785]
p1's belief std: [71.462 71.462], p2 belief std's: [52.127 52.127]
```

Q5. Best-reply path

- 在此題中，使用 fictitious play 會以 mixed-strategy Nash equilibrium 的方式收斂。
- 在我實驗中，不管設定何種的初始 belief table，最後的 p1 與 p2 的 belief 皆趨近於[0.5, 0.5]，標準差並不大，因此此題會以會以 mixed-strategy Nash equilibrium 的形式收斂。
-
```
p1's belief mean: [500.876 499.124], p2 belief mean's: [500.744 499.256]
p1's belief std: [46.659 46.659], p2 belief std's: [47.468 47.468]
```

Q6. Pure-Coordination Game

- 在此題中，多數情況皆會收斂於(r_1, c_1)與(r_2, c_2)，但仍會有 mixed-strategy Nash equilibrium 的情況。
- 在實驗中，與 Q2 類似，若初始的 p 與 q 皆大於 50，則會收斂於(r_1, c_1)，若皆小於 50，則收斂於(r_2, c_2)，其他情況再經過模擬後也會依照比例收斂至(r_1, c_1)與(r_2, c_2)。然而，若剛好初始的 p+q≈100，則會以 mixed-strategy Nash equilibrium 的方式，p1 與 p2 的 belief 皆會收斂於[0.5, 0.5]。
-
```
p1's belief mean: [496.719 503.281], p2 belief mean's: [496.408 503.592]
p1's belief std: [491.11 491.11], p2 belief std's: [490.719 490.719]
```

Q7. Anti-Coordination game

- 在此題中，多數情況皆會收斂於(r_1, c_2)與(r_2, c_1)，但仍會有 mixed-strategy Nash equilibrium 的情況。
- 此題與 Q6 類似，只是收斂條件有所調整，若 p 大於 50 且 q 小於 50，則一定會收斂於(r_1, c_2)，若 p 小於 50 且 q 大於 50，則一定會收斂於(r_2, c_1)，其他情況則經過模擬後有會依照比例收斂至(r_1, c_2)與(r_2, c_1)。而對於 mixed-strategy Nash equilibrium 的情況，則是出現於 p≈q 時。
-
```
p1's belief mean: [504.397 495.603], p2 belief mean's: [495.487 504.513]
p1's belief std: [490.883 490.883], p2 belief std's: [491.043 491.043]
```

Q8. Battle of the Sexes

- 在此題中，多數情況皆會收斂於(r_1, c_1)與(r_2, c_2)，但仍會有 mixed-strategy Nash equilibrium 的情況。
- 此題與 Q6 相同，皆會在初始 p, q 大於 50 時收斂於(r_1, c_1)，小於 50 時在收斂於(r_2, c_2)，其他情況則依照比例收斂至(r_1, c_1)與(r_2, c_2)。而 mixed-strategy Nash equilibrium 的情況，則是出現於 p+q≈100 時。

```
p1's belief mean: [494.615 505.385], p2 belief mean's: [510.293 489.707]
p1's belief std: [488.718 488.718], p2 belief std's: [488.976 488.976]
```

Q9. Stag Hunt Game

- 在此題中，多數情況皆會收斂於(r_1, c_1)與(r_2, c_2)，但仍會有 mixed-strategy Nash equilibrium 的情況。
- 對於此題來說，雖與 Q6 以 p 與 q 初始值計算 payoff 時有些不同，但最終的結論仍有初始 p, q 大於 50 時收斂於(r_1, c_1)，皆小於時則會收斂於(r_2, c_2)，而其他情況則會依比例收斂於(r_1, c_1)與(r_2, c_2)。而對於 mixed-strategy Nash equilibrium，也是出現於 p+q≈100 時
-
```
p1's belief mean: [507.006 492.994], p2 belief mean's: [506.709 493.291]
p1's belief std: [490.62 490.62], p2 belief std's: [491.022 491.022]
```

Q10. Observation and Conclusion

- 否，例如 Q3 的情況，若有出現如$(r_2, c_2) = (0, 0)$的情況，雖說其是 Nash equilibrium，但卻被 weakly dominated，在此情況下，使用 fictitious play 就可能無法找到所有的 Nash equilibrium。
- 另外，Q1 到 Q9 中，皆假設 game matrix 是固定的，僅使用對手選擇不同行為的次數來調整 belief table。然而，若遊戲中對手的選擇是動態的，也就是說對手可能會在過程中可能會調整其選擇行為的策略與 utility，就可能導致 fictitious play 無法正常收斂。
- 最後，fictitious play 也只適用於 perfect complete information game。若缺少前者，則無法得知對手過去的行為，導致無法以此調整自己的 belief table。若缺少後者，則無法知道 game matrix 的樣子，進而無法計算 payoff 來挑整 belief table。