

# Fictitious Play

小川慶将

2014.6.28

- Fictitious Play とは？
  - 日本名では仮想ゲームなどと呼ばれる。
  - 各プレイヤーがそれぞれの予想のもとで最適反応戦略を選択する動学ゲーム。

## Fictitious play の説明

- 予想の立て方各  $t$  時点において、プレイヤー 0 は「プレイヤー 1 は、確率  $1 - x_0(t)$  で行動 0 をとり、確率  $x_0(t)$  で行動 1 をとる」と考え、自分の期待利得が最大になるような行動をとる。 $x_0(t)$  を「 $t$  時点における、プレイヤー 0 の、プレイヤー 1 の行動に関する信念 (belief)」と呼ぶことにする。(ゼミ課題文より)
- 予想形成

$x_0(t)$  は

$$x_0(t+1) = x_0(t) + \frac{1}{t+2}(a_1(t) - x_0(t))$$

と再帰的に書くことができる。

- switch **を使用**

その場で入力を要求させる case0

MachinPennies を case1

Coordination Game を case2

Prisoner's Dilemma を case3

## コードの説明とか

- コードの表示の例

```
if NUM == 0:
    print '2 × 2 の利得表をひとつずつ記入してください'

    pr_0_00 = float(raw_input('プレイヤー0左上:'))
    pr_0_01 = float(raw_input('プレイヤー0右上:'))
    pr_0_10 = float(raw_input('プレイヤー0左下:'))
    pr_0_11 = float(raw_input('プレイヤー0右下:'))
    pr_1_00 = float(raw_input('プレイヤー1左上:'))
    pr_1_01 = float(raw_input('プレイヤー1右上:'))
    pr_1_10 = float(raw_input('プレイヤー1左下:'))
    pr_1_11 = float(raw_input('プレイヤー1右下:'))
    titlename = raw_input('タイトル名:')
    #player
    p0 = np.array([[pr_0_00,pr_0_01],[pr_0_10,pr_0_11]])
    p1 = np.array([[pr_1_00,pr_1_01],[pr_1_10,pr_1_11]])
```

- コードの表示の例

```
#Matching Pennies
if NUM == 1:
    titlename = 'Matching Pennies'
    p0 = np.array([[1,-1],[-1,1]])
    p1 = np.array([[1,-1],[-1,1]])
#Coordination Game
if NUM == 2:
    titlename = 'Coordination Game'
    p0 = np.array([[4,0],[3,2]])
    p1 = np.array([[4,0],[3,2]])
#Prisoner's Dilemma
if NUM == 3:
    titlename = 'Prisoners Dilemma'
    p0 = np.array([[5,0],[-10,-3]])
    p1 = np.array([[5,0],[-10,-3]])
```

# Fictitious Play

小川慶  
将

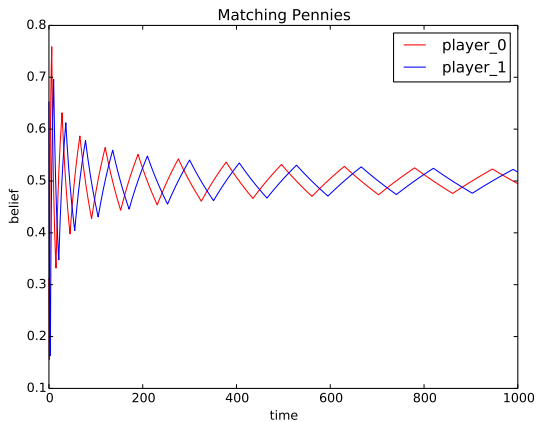


Figure : 図の表示

- まとめ
  - $2 \times 2$  の FictitiousPlay がプログラムを動かすことでナッシュ均衡がすぐに見つかる。
  - 囚人のジレンマは信念とか全く関係ないですけど動かしてみたらちゃんとみんな自白しました。
- よくわかっていない点とか
  - スライドの作り方へたくそすぎましたすいません。
  - 次はもっときれいな作ります
- 今後の課題とか
  - $3 \times 3$  ゲームでのプログラムも作りたいです。