Assignment 4

**Is there a thing called strategy for rock scissor paper?**

**Description**

사람, 소녀, 조그만, 아이이(가) 표시된 사진

자동 생성된 설명

You and your friend eat lunch together almost every day. You and your friend always decide who will pay for lunch with rock scissor paper. Someday, you noticed that your friend tends to raise rock more than other two forms (scissor and paper). You thought that you may eat lunch with him at least 1,000 times during your college life. You think that some strategies will give higher probability of winning, so you can save your money. To find a strategy that work, you sit at your desk, and turn your laptop on, and start writing python program.

Your program plays the rock scissor paper game for *1000* times. The opponent, who samples the action (one of rock, scissor, or paper) from predefined probability distribution. If your program can learn the probability distribution, this becomes an easy problem. E.g., if you find out the opponent raises rock 80% all the time, you may always select paper. You should make your program discover the appropriate strategy.

Your goal is

1. Learn statistics from the first *1000* games.
2. Find a winning strategy from the statistics.
3. Examine your strategy for the next *1000* games.

**Note**

*Strategy (or, policy):*

It is a probability distribution over possible actions. For example, {“rock”: 0.5, “scissor”: 0.2, “paper”: 0.3} will sample an action for the rock scissor paper game. The sum of all action probabilities must be 1.0.

*Sampling from distribution*

If you a sample from a distribution {“rock”: 0.5, ”scissor”: 0.2, “paper”: 0.3}

You will likely get “rock” with 50% probability.

You will likely get “scissor” with 20% probability.

You will likely get “paper” with 30% probability.

*Opponent’s behavior*

In Easy mode, the probability distribution of the opponent is fixed and simple.

In hard mode, the probability distribution of the opponent is dependent on the form the opponent raised last time and my form of the last round.

Bit more formally, is fixed for all rounds.

For this assignment, a template program is given.

In the template program, you must implement two functions; one is “evaluate\_rsp()” and the other is “generate\_policy\_exploitation()”.

**Functions to be implemented**

|  |  |
| --- | --- |
| Name | Meaning |
| generate\_policy\_exploitation(): | Generate my policy given game statistics (the number of rocks, scissors, papers the opponent has raised, the last form I,and the opponent has raised).  *generate\_policy\_exploitation() is used in exploitation phase, where score is verdicted.* |
| evaluate\_rsp(…) | Given two forms, return result.  1 if A wins, 0 for draw, -1 otherwise. |
| NOTE  You somehow build your strategy upon from the opponents took during exploration phase, then exploit and make high scores in exploitation phase. For the first round, it is assumed that you and the opponent both have raised ‘rock’. You are allowed to use the parameters of the generate\_policy\_exploitation() only to make a policy. Zero grade will be given if you use other variables. | |

**Input & Output**

There is no Input. Do NOT change the ouput in the template program.

**Verdict**

Once you complete the program, you can test your strategy (policy) by running the template program. The program shows the average results of 10 trials of your winning strategy for 1000 games .

**Submission Form**

* A Python program.
* The file name must be “studentID.py” (ex. 2020333123.py)
* Submit on the i-campus assignment board.
* Submission due date: 6/18, 23:59
* -15 for a day late, -30 for two days late, -45 for three days. No late submission allowed after that.

**Scoring Criteria (Maximum 100 points)**

* Your program must run without any warning nor error. You will get zero grade because we cannot verdict that.
* evaluate\_rsp(*)* implemented correctly: 10pt
* In easy mode, your policy has greater score than 100, for more than 7 cases out of total 10 cases: 30pt
* In hard mode, your policy has greater score than 100, for more than 7 cases out of total 10 cases: 60pt

If you have any question about the assignment, use the google sheet.

**Caution**

Before asking a question, please read the template program code carefully.