Project 3: Easy21 Game

Introduction

Please apply reinforcement learning to solve a simple game called Easy21. Rules are defined as below.

- Act as **the player** to play with **the dealer**. Initial scores are both 0. There are infinite deck of cards (i.e. cards are sampled with replacement).
- Each draw from the deck results in a value $\mathbf{v} \in \{\mathbf{1}, \ldots, \mathbf{10}\}$ (uniformly distributed) with a colour of red (probability $\frac{\mathbf{1}}{\mathbf{3}}$) or black (probability $\frac{\mathbf{2}}{\mathbf{3}}$).
- Add (black cards) the value of a card to or subtract (red cards) that from the corresponding score.
- At the start of the game both the player and the dealer draw one black card.
- Player first. Each turn the player may either **stick** or **hit** (two actions in total). If the player hits then he draws another card from the deck. If the player sticks he receives no further cards.
- If the player's score exceeds **21**, or becomes less than **1**, then he "goes bust" and loses the game (**reward** -1).
- If the player sticks then the dealer starts taking turns. The dealer always sticks on any sum of **16** (not 17) or greater, and hits otherwise. If the dealer goes bust, then the player wins (reward $+1$). Otherwise, compare their scores. The player wins (reward $+1$), loses (reward -1), or draws (reward 0) accordingly.

Requirements

- Implement Q-learning to play with the dealer.

Optional codes for reference are provided. Read the file *README.md* if you want to use them. You can also try more algorithms (e.g. SARSA which is similar to Q-learning).

Submission

1. Please submit a file in ZIP or RAR format containing your **report** (Chinese or English) and **codes** on Canvas before **2022-01-08**, **23 : 59**. Name it as $StudentID\_Name$ (eg. 5180xxxxxxxx_张三).

2. Describe and analyze your implementation and understanding of Q-learning.

3. Do not cheat.