



OpenAI Gym: BlackjackEnv

In this lesson, you will write code to teach an agent to play Blackjack.



Source: <https://www.blackjackinfo.com/img/2-card-21.png>

Please read about the game of Blackjack in Example 5.1 of the [textbook](#).

When you have finished, please review the corresponding [GitHub file](#), by reading the commented block in the `BlackjackEnv` class. (*While you do **not** need to understand how all of the code works, please read the commented block that explains the dynamics of the environment.*) For clarity, we have also pasted the description of the environment below:



Blackjack is a card game where the goal is to obtain cards that sum to as

near as possible to 21 without going over. They're playing against a fixed dealer.

Face cards (Jack, Queen, King) have point value 10.

Aces can either count as 11 or 1, and it's called 'usable' at 11.

This game is played with an infinite deck (or with replacement).

The game starts with each (player and dealer) having one face up and one face down card.

The player can request additional cards (`hit=1`) until they decide to stop (`stick=0`) or exceed 21 (bust).

After the player sticks, the dealer reveals their facedown card, and draws until their sum is 17 or greater. If the dealer goes bust the player wins.

If neither player nor dealer busts, the outcome (win, lose, draw) is decided by whose sum is closer to 21. The reward for winning is +1, drawing is 0, and losing is -1.

The observation of a 3-tuple of: the player's current sum, the dealer's one showing card (1-10 where 1 is ace), and whether or not the player holds a usable ace (0 or 1).

This environment corresponds to the version of the blackjack problem

described in Example 5.1 in Reinforcement Learning: An Intr



`http://incompleteideas.net/sutton/book/the-book.html`

"" ""

NEXT