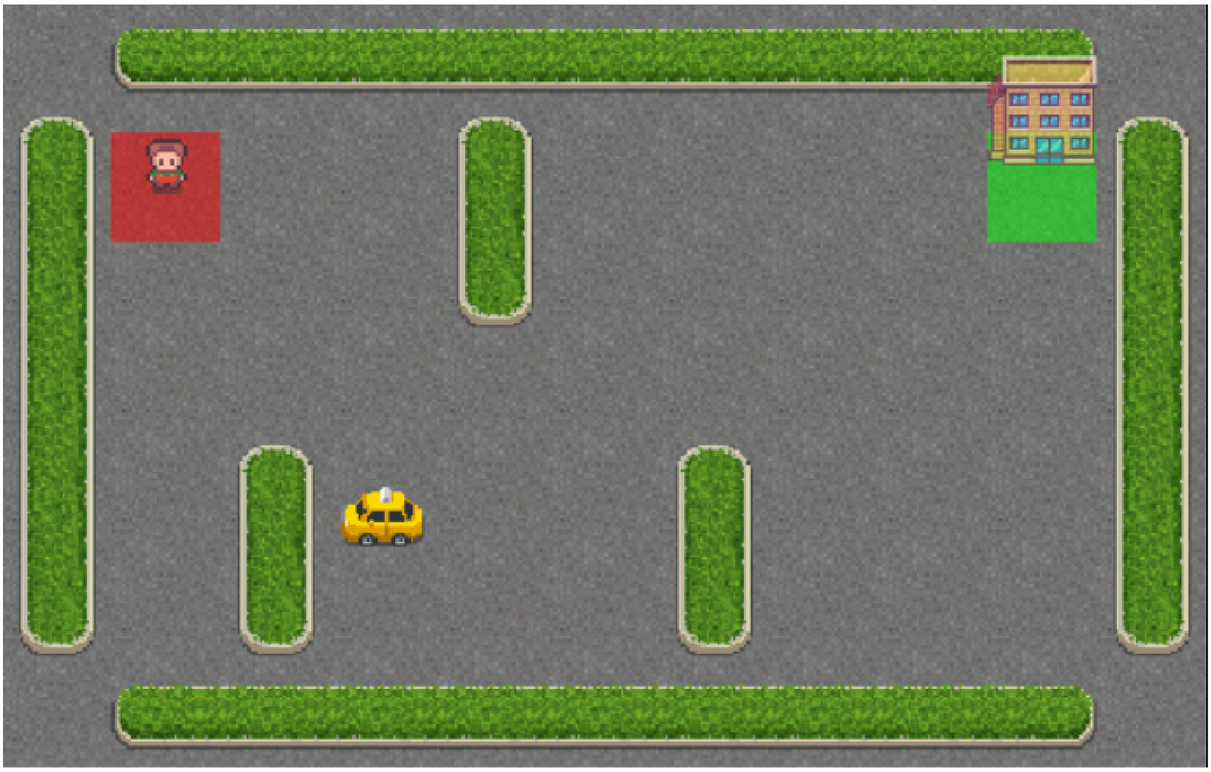
Tutorial 1: Data Generation

Universidad Carlos III de Madrid
Grado en Ingeniería Robótica
Asignatura: Aprendizaje Automático

Made by Lucas Beltrán Rúa (NIA 100523114)
& Ignacio González Meléndez (NIA 100522976)



1. Which are the parameters of the make function? What are they used for? Do the same for the reset and step functions.

- The function *make* initializes environments. It returns an *env* to interact with.

- The render mode determines whether you see a visual window (“human”), get image arrays (“rgb_array”), or run without visuals (None - fastest for training).

- The function step executes the selected action

2. Is the state space discrete or continuous? How about the action space?

- For the taxi environment we have 25 possible positions, as the grid is 5x5. The passenger can be in 5 different positions, 4 are fixed and the other one inside the taxi. And there are 4 possible destinations. Which means that we have 500 discrete space states.

- About the action space we have 6 possible actions (0-5), this means that the action space is also discrete.

3. How is the observation space defined? What are the possible actions in the game?

- The observation space is what your agent can see, (images, numbers, structured data, etc.).

- Possible actions in the game are: 0 - south, 1 - north, 2 - east, 3 - west, 4 - pick-up, 5 - drop-off.

4. How is the reward defined? When does an episode end?

- The reward is the feedback that the agent receives.

- If the environment has terminated due to the task being completed or failed, this is returned by step() as the parameter terminated is True. If we want the environment to end after a fixed number of timesteps (like a time limit), the environment uses Truncated equal to True signal.

5. In your opinion, which data could be useful to decide what the taxi should do on each tick?

- Useful data would be:

- Taxi position, in order to know where the cab is. With this information the reward can be calculated. For example if the cab tries to pick-up the passenger in an erroneous position the reward would be -10.

- Taxi action, this way the cab will perform the wanted movement.

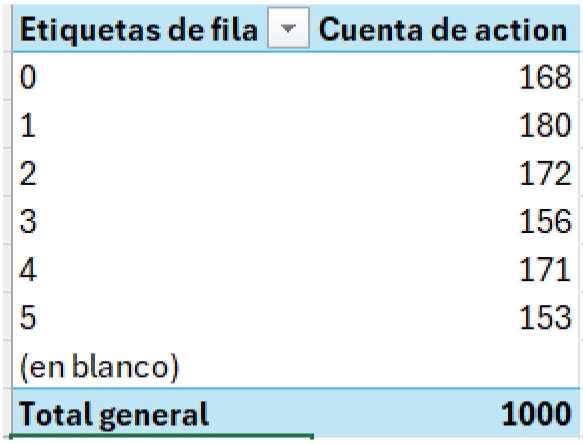
- Passenger position, with this the reward will be correctly given if the cab drops him in the desired position.

6. What happens if you define the environment with the flag is raining set to True?

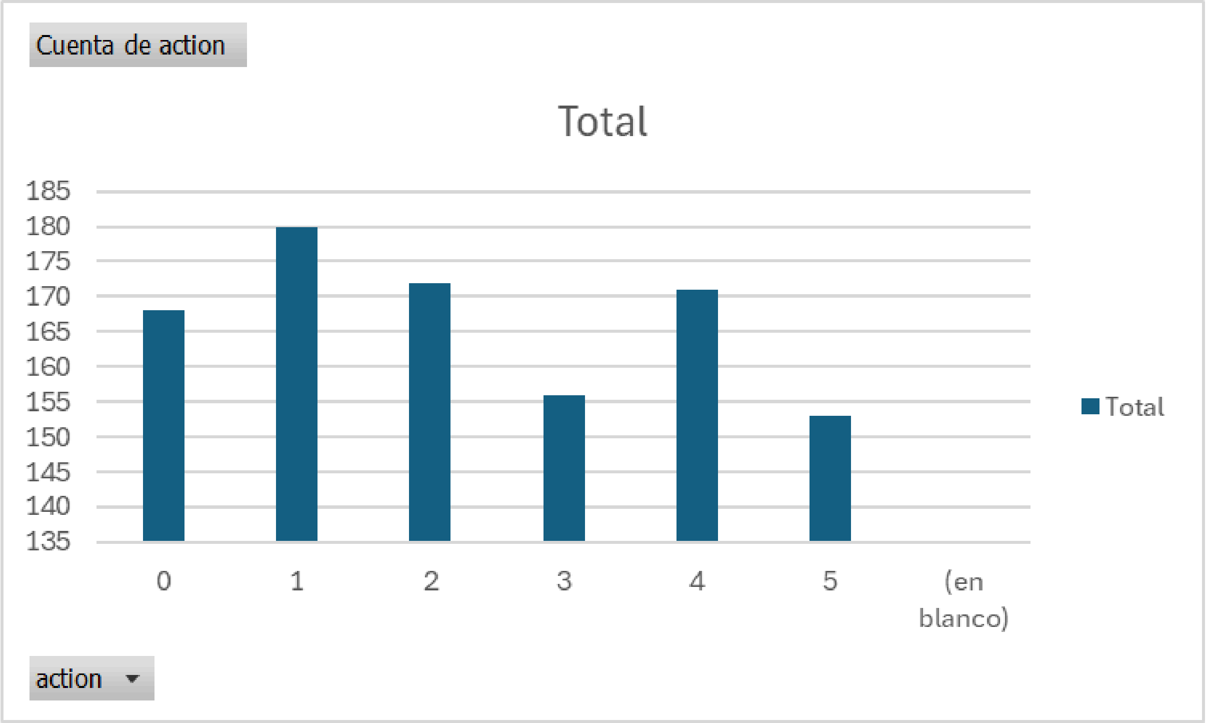
- If True the cab will move in the intended direction with probability of 80% else it will move in either left or right of target direction with equal probability of 10% in both directions. This means that the cab has a chance of moving in a random direction, either left or right.

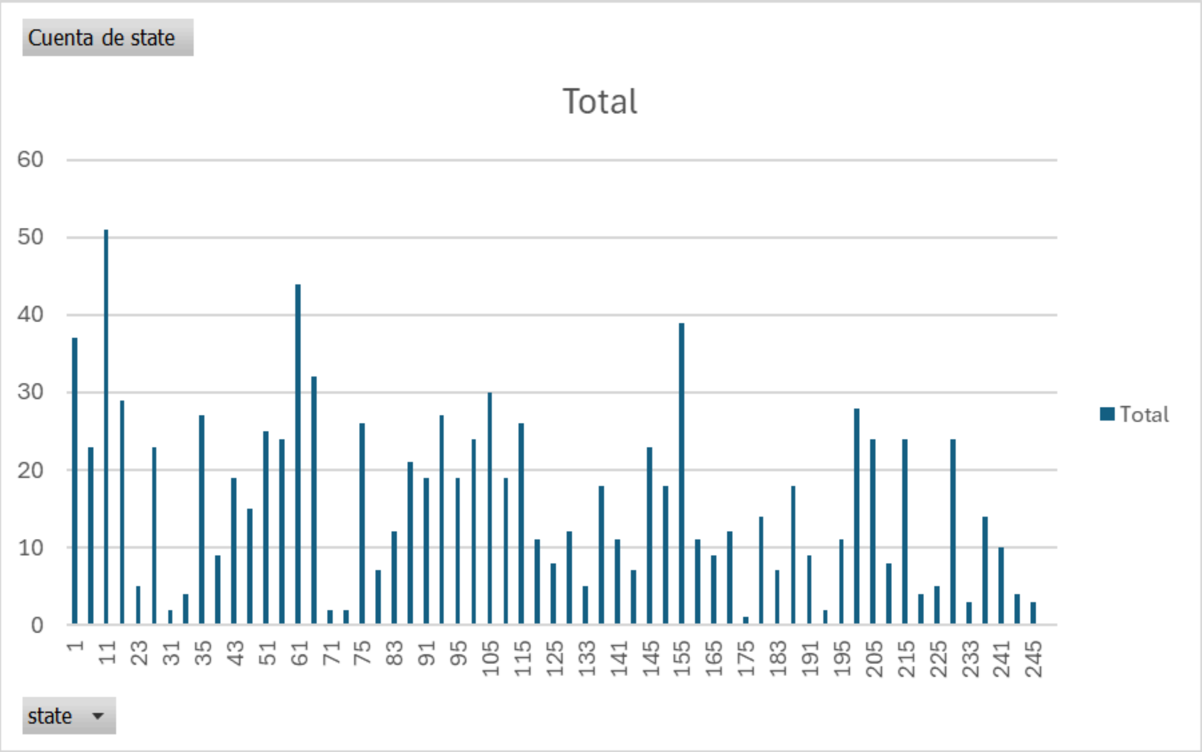
8. Analyze the data gathered.

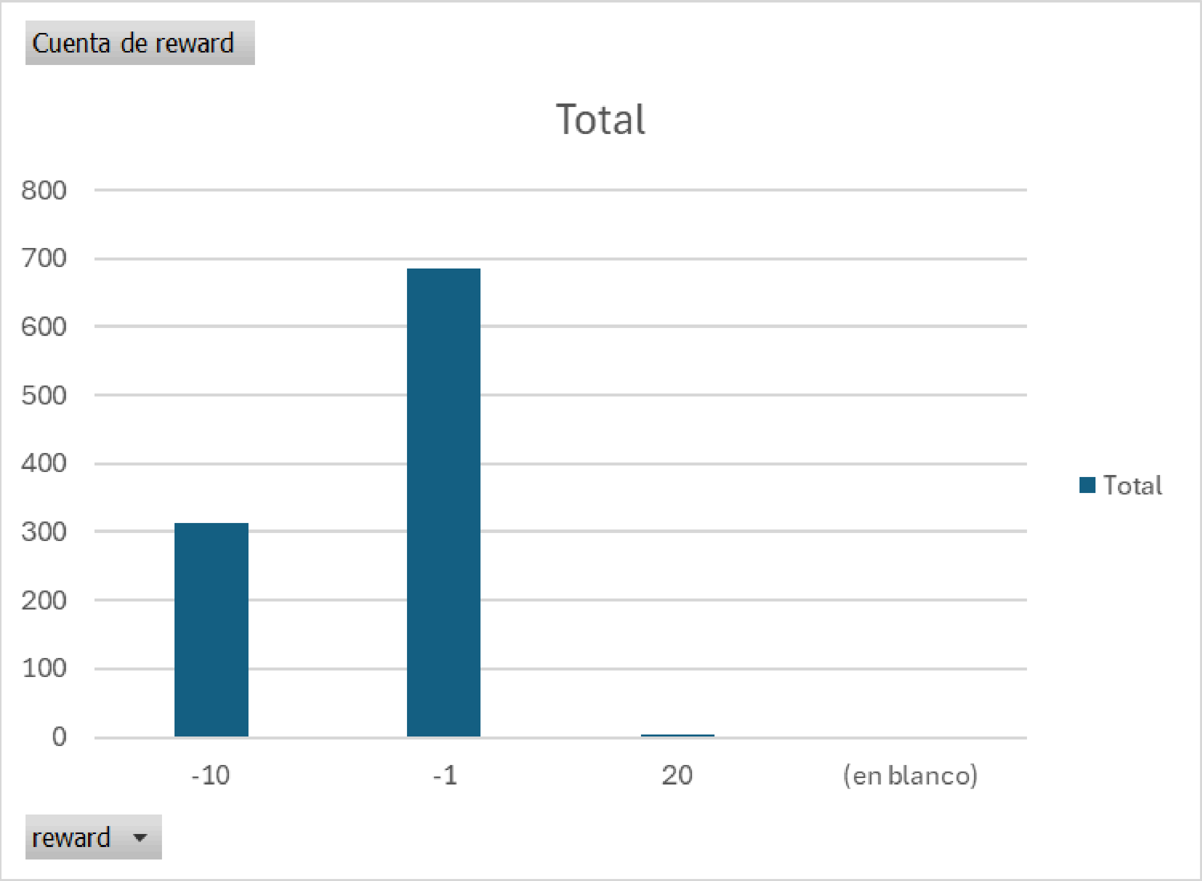
- By analyzing the data we can clearly see that not all actions are equally distributed, as the move and pick/drop actions are the most performed. Also in this simulation there was no terminated = True state, which means that the passenger has not been dropped correctly.



Etiquetas de fila	Cuenta de action
0	168
1	180
2	172
3	156
4	171
5	153
(en blanco)	
Total general	**1000**




Cuenta de state
Total
60
50
40
30
20
10
0
1
11
23
31
35
43
51
61
71
75
83
91
95
105
115
125
133
141
145
155
165
175
183
191
195
205
215
225
233
241
245
Total
state


Cuenta de reward
Total
800
700
600
500
400
300
200
100
0
-10
-1
20
(en blanco)
Total
reward

Code Description:

- The implemented method for the keyboard controller is the usage of a listener, which will perform an on_press function continuously. This function defines the current_action value to perform it with env.step(action).

- For saving the state of each tick we implemented the library “pandas” in order to save the important information in a csv. For saving this information we gathered it as a dictionary and then appended it to a created list called "experiences". To sum up, this list is transformed into a csv called “experiences_auto” with the function “.to_csv”.