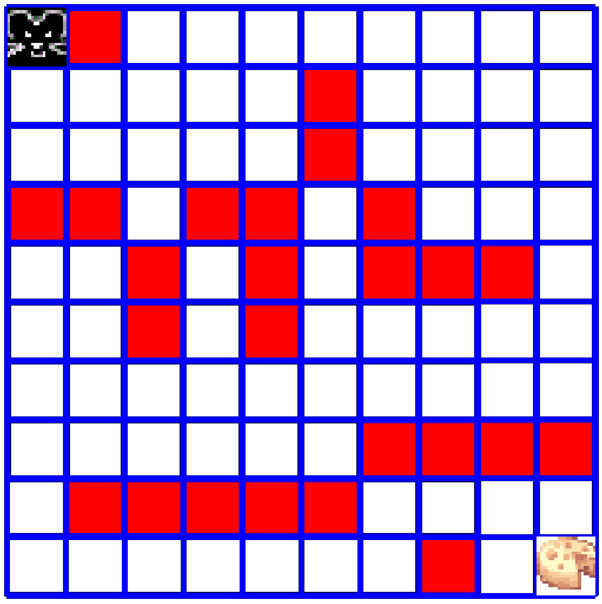
hugomcp@di.ubi.pt, 2025/26

Machine Learning

Practical Sheet 9: Reinforcement Learning

1. Create a Pyhton script that generates random "N x N" boards, with each cell having only two possible types: "**Safe**" (penalty=1) or "**Dangerous**" (penalty=100). In the example below, "Safe" cells appear in white, while "Dangerous" cells are represented in red.



   a. Generate two random cell positions (each one provided in a (row, column) format), corresponding to the current position of the agent and to the final position (e.g., "cheese" in the example above).

   b. Using Reinforcement Learning techniques (Q-learning), implement a solution for moving from the current state to the final position in as few movements as possible, avoiding the Dangerous cells as much as possible.

   c. Compare different strategies for defining the "state" in terms of the spatial computational cost of the algorithm (the amount of storage required) and the quality of the solutions generated (the cost of the overall path, i.e., the sum of the penalties for the cells composing the best path).

   d. Compare the variations in the results, with respect to the α, γ values used in the Q-Table update formula:

$$Q_{new}(s_t, at) \leftarrow (1-\alpha)Q(s_t, at) + \alpha(Rt + \gamma\, max_a\, Q(S_{t+1}\, a))$$

   e. Prepare a small report, describing your main findings.