Simulación Estocástica: Teoría y Laboratorio
Equipo Docente: Joaquín Fontbona; Camilo Carvajal, Arie Wortsman, Pablo Zúñiga

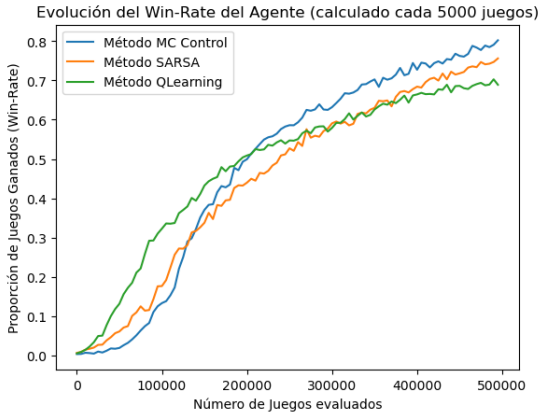
Integrantes: Javier Maass
Juan Pablo Sepúlveda

Resumen Proyecto Final
Reinforcement Learning en el juego de *Cacho/Dudo*

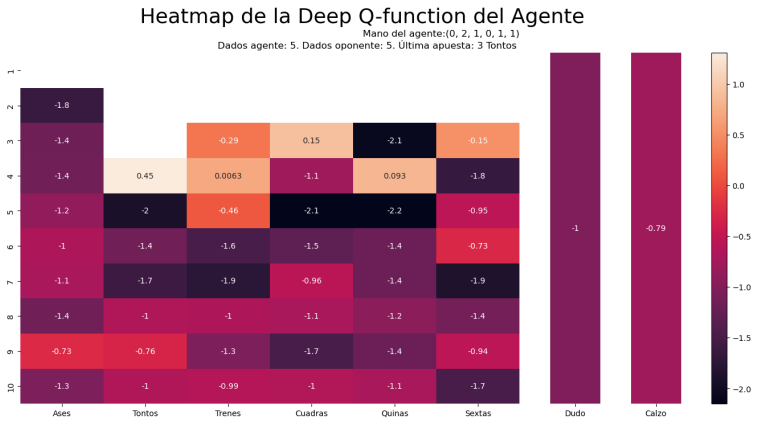
El Reinforcement Learning (RL) es una de las técnicas que ha revolucionado el área de *ML* en el último tiempo, siendo usado por las grandes compañías de tecnología, en proyectos como *Deep Mind*. Más en general, ha revolucionado la forma en que enseñamos a las máquinas a tomar *decisiones en tiempo real*. Por otro lado, el *Cacho/Dudo* (ver este link) es un juego popular, con una importante componente aleatoria, que se basa en la *toma de decisiones condicionadas a un estado* y que no ha sido mayormente estudiado. Buscaremos implementar las ideas del RL para crear *AlphaCacho*.

El setting básico de RL es el de un **Markov Decision Process** (MDP), en que un agente toma (en cada instante de tiempo t discreto) **acciones** (A_t) que, según el **medio ambiente**, llevan al agente a un nuevo **estado** (S_t) en el cual se recibe cierta **recompensa** (R_t). El agente elige su siguiente acción (conociendo el estado en que se encuentra), siguiendo una *política* (distribución de probabilidad condicional $\pi(a|s)$). El objetivo del RL es **encontrar la política óptima** π^* que maximice el retorno esperado del agente ($\pi^* \in \arg\max_\pi \mathbb{E}_\pi[\sum_{t=1}^{\infty} \gamma^{t-1} R_t]$). Hay resultados teóricos que garantizan que esto puede hacerse si conocemos la *matriz de transición* del ambiente (con *Programación Dinámica*); y en caso contrario, se puede aproximar π^* usando métodos de Monte Carlo (en el llamado *MC-Control*[1]) con exploración *aleatoria* $\varepsilon-$**greedy** (Eligiendo la acción de *máximo* valor, con $\mathbb{P}(\cdot) = 1 - \varepsilon$)[2].

Se implementó el juego de *Cacho* en código, y se implementaron las técnicas de RL mencionadas anteriormente, para entrenar a un agente capaz de jugar contra un **único adversario**, de **política determinista** (para simplificar la implementación); o así también contra un humano. Se estudió la *efectividad del agente* (winrate), según el número de *samples* usadas para el aprendizaje (con los distintos métodos), así como también el costo computacional de cada método, y la influencia del parámetro exploratorio ε en los resultados.



Evolución del WinRate del Agente a lo largo del entrenamiento.



Visualización de la QTable del Agente

[1] O sus variantes: *SARSA*, *Q-Learning*, *Deep Q-Learning*, entre otras
[2] Similar a la temperatura en *Simulated Annealing*

Referencias Bibliográficas:

- Sutton, R. S., & Barto, A. G. (1998). *Reinforcement Learning:An Introduction*. Bradford Books.
- Rich, D. (2022). Reinforcement Learning Fundamentals. Youtube: Link. GitHub: Link.