

# Ejercicios Lab Final

October 21, 2018

Se pide:

- 1) Dado el agente que utiliza una red neuronal para aproximar la función de valor, modificar la arquitectura de la misma en busca de una mejora en el desempeño del agente. La modificaciones a realizar puede ser:
  - Agregar mas capas ocultas a la red
  - Modificar las funciones de activación de las neuronas
  - Modificar el numero de neuronas en cada capa

**Recomendación #1:** Tener en cuenta que al incorporar mas capas y neuronas a una red neuronal, mas complejidad computacional se incorpora al modelo, lo que se traduce en una mayor duración (en tiempo) del entrenamiento del agente.

**Recomendación #2:** Una métrica eficiente para comparar del desempeño de diferentes agentes es la curva de “reward promedio acumulado”. Esta métrica permite visualizar el desempeño general del agente, como así también la rapidez (cantidad de episodios) que requiere un agente para alcanzar un determinado desempeño.

**Recomendación #3:** Al modificar distintos aspecto de la arquitectura del aproximador, realizar de a un cambio por evaluación.

- 2) Adaptar los agentes implementados en el notebook (incluido el implementado por ustedes) al entorno ‘MountainCar-v0’ (<http://gym.openai.com/envs/MountainCar-v0/>) y comparar el desempeño de los mismos.
- 3) Elaborar un informe que enumere lo realizado paso a paso, mostrando resultados parciales y finales.

**Recomendación General:** No se sugiere hacer este TP desde jupyter notebook/lab sino desde un IDE estilo Pycharm, debido a que los algoritmos de RL suelen requerir un debug paso a paso, tanto para corregir errores como para entender mejor cómo funcionan los mismos.

**Opcional:** Implementación de un agente DQN convolucional que aprende a jugar a Atari! Space Invaders! ([link](#))