| Universitat de València |
| --- |
| Reinforcement Learning |
| Car Racing – GYM Library |
| Javier Ferrero Micó  7-7-2022 |

# Planteamiento

Para este proyecto se ha afrontado el reto Car Racing, planteado a través del motor gráfico para juegos Box2D. CarRacing consiste en mantener un coche dentro de la pista a lo largo de un circuito.

En la biblioteca de GYM, la última versión disponible es la V1; sin embargo, en el entorno de desarrollo utilizado no se ha podido lanzar esta última versión y, por tanto, se ha trabajado con la V0.

La diferencia entre ambas versiones es:

* v1: Cambio de la lógica de finalización de la pista y adición de la aleatorización del dominio (0.24.0)
* v0: Versión original

Las referencias utilizadas para el desarrollo del entorno de entrenamiento y de pruebas han sido:

* https://github.com/andywu0913/OpenAI-GYM-CarRacing-DQN;
* https://github.com/wpiszlogin/driver\_critic/blob/main/base\_solution.py;
* https://www.gymlibrary.ml/environments/box2d/car\_racing/

Se ha seguido una estrategia basada en el algoritmo Deep Deterministic Policy Gradient o DDPG. Se han configurado cuatro redes neuronales: un actor, que es el que juega; un crítico, que evalua al actor; y los dos targets, que generan los objetivos del juego para evaluar el aprendizaje. Este planteamiento ha sido usado en clase para solventar el problema de Mountain Car.

Aunque el entorno del Car Racing permite habilitar la opción de discretizar el espacio de acciones, se requiere un espacio de acciones contínuo para utilizar DDPG. Discretizar el espacio de acciones, además, cambiaría su definición, pasando de una dimensionalidad 3 (continuo) a 5.

# Entrenamiento y evaluación

Se ha configurado un virtual env en una máquina local con python3.8, y se ha lanzado el proceso de entrenamiento, utilizando una GPU NVIDIA 2070 RTX. El entrenamiento ha sido abortado a los 877 episodios puesto que llevaba más de 400 sin mejorar.

En la siguiente gráfica podemos ver el progreso de las métricas utilizadas para la evaluación del entrenamiento.

Chart, line chart

Description automatically generated

En rojo se dibuja la mayor recompensa obtenia en cada episodio y, en azúl, la media de la recompensa a lo largo de todas las iteraciones del episodio. El modelo guardado está señalado con una cruz sobre el episodio 320. Para optimizar el entrenamiento, se ha establecido que si no existe una mejora en la recompensa tras 200 iteraciones, el episodio sea abortado.

Para evaluar los resultados de forma conjunta, se han guardado los distintos modelos en formato .h5, cada vez que se conseguía una mejora en la recompensa obtenida.

Text

Description automatically generated

# Resultados

Se adjunta un vídeo en el archivo zip. En él, se ha grabado la pantalla de ejecución del entorno de simulación utilizando el modelo entrenado con mayor recompensa. En líneas generales, el coche zigzaguea demasiado para lo cual habría que mejorar el sistema de recompensas de manera que fuera más centrado en el camino y evitase más el interactuar con verde (fuera de pista: hierba).

La siguiente imagen muestra una captura del entorno de simulación y el log que se genera con las recompensas. Como vemos, al zigzaguear, las recompensas positivas no son muy elevadas y las recompensas negativas són demasiado frecuentes.

Graphical user interface, application

Description automatically generated

# Mejoras

A continuación, se proponen algunas de las actuaciones que podrían considerarse para obtener un modelo más estable:

* Introducir otro tipo de políticas de decisión de acción a realizar: políticas basadas en la acción que represente la mayor recompensa, políticas de acción aleatoria…
* Modificar la función de generación del ruido para explorar mejor el environment y poder proporcionar mejor información al actor.
* El crítico debería ser más restrictivo para: conseguir que el coche haga los giros de forma más fina, vaya más centrado en la pista y sea penalizado más cada vez que se sitúa sobre la hierba.
* Controlar el rendimiento del entrenamiento de la red (agente) empleando un entorno de pruebas previo para cada modelo obtenido.