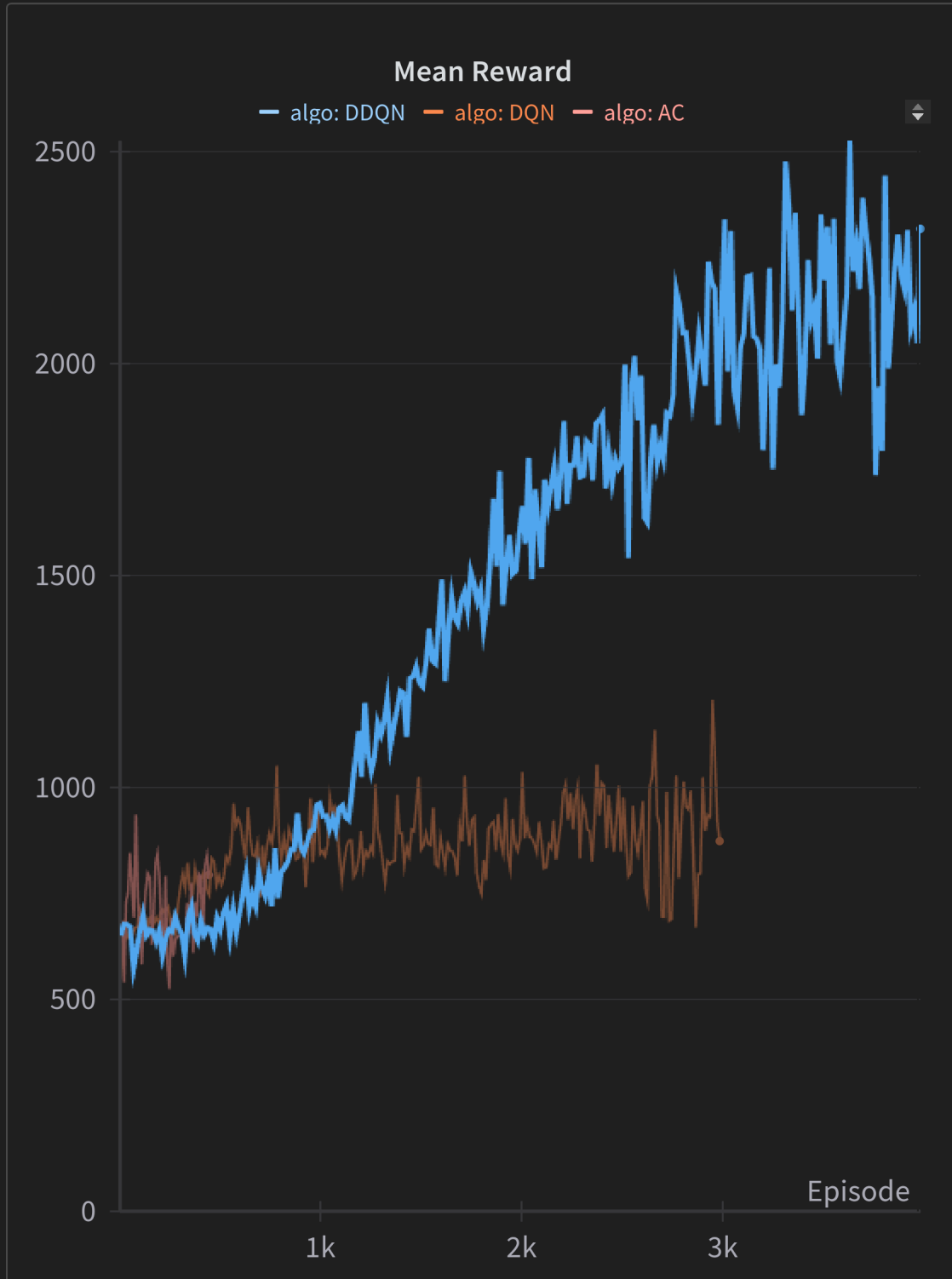


[Share](#)[Comment](#)[Star](#)[...](#)

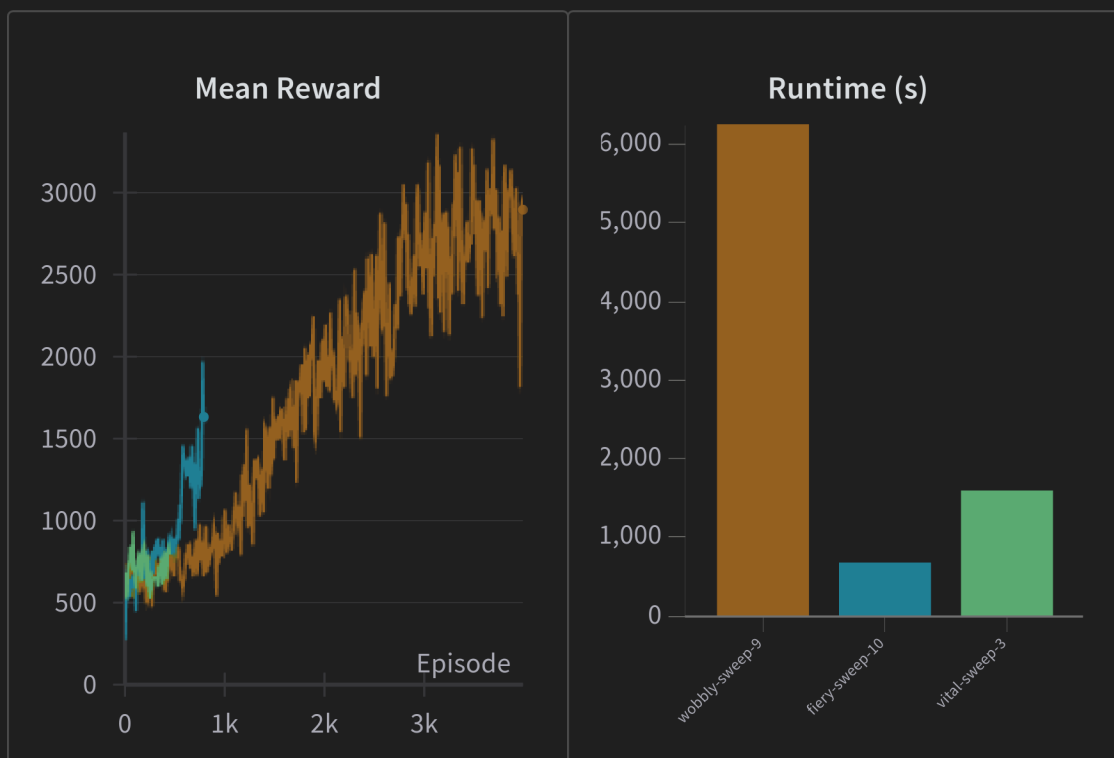
Reinforcement Learning

Matias Molinolo, Juan Pedro da Silva

<https://github.com/JuanCommits/reinforcement-learning>



Agrupamos las corridas de los distintos algoritmos probados: DQN, DoubleDQN y Actor-Critic.



▼ Actor Critic

Como podemos ver en la gráfica anterior, las corridas hechas con el método de gradiente de política Actor Critic (AC) tienen muchos menos episodios. Esto se debe a que estas corridas siempre terminan en una explosión de gradientes.

Lo poco que se puede apreciar de la gráfica es que el entrenamiento con esta técnica es muy inestable y ya han salido otras técnicas que resuelven estos problemas, como por ejemplo TRPO o PPO.

Algunos tips que encontramos para combatir este problema sin ser el cambio de técnica son:

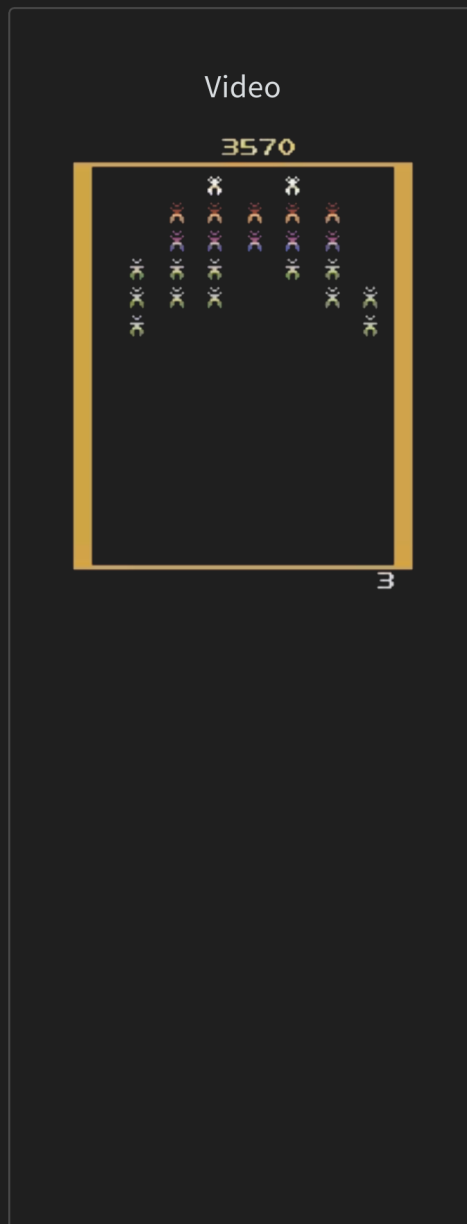
- Utilizar gradient clipping.
- Reducir el learning rate.
- Que el learning rate del actor sea mucho menor que el del crítico.

Si bien probamos todo esto no pudimos evitar el problema mencionado anteriormente. Como futuro trabajo queda explorar diferentes variaciones de esta técnica y las otras técnicas mencionadas anteriormente.

▼ Deep-Q Network

Si bien corrimos varios sweeps, por cuestiones de tiempo, no pudimos probar ejecuciones de DQN con mayor cantidad de episodios, que nos hubiera gustado hacer para observar el comportamiento del algoritmo y como aprende.

Podemos ver la mejor corrida de DQN en el siguiente video:



[Link](#) a la corrida.

▼ Double Deep-Q Network

Adjuntamos el video de la mejor corrida de Double DQN:



[Link](#) de la corrida.

▼ Apreciaciones finales

Podemos observar como DQN parece tener un crecimiento más abrupto en cuanto a la recompensa media pero tambien más inestable. Si bien en nuestra implementación de congelar un segundo modelo, identico al principal, y ponerlo al día luego de


varias actualizaciones del modelo principal obtenemos una mayor estabilidad, esta sigue siendo menor a la de DDQN.

Como mencionamos antes, DDQN tiene un crecimiento más lento pero llega a resultados muy buenos y con mayor estabilidad.

Otra apreciación que podemos hacer es que los dos algoritmos tienden a tener mayor variación cuando llegan a mejores políticas. Esto se debe a que un cambio menor en la política puede hacer que un movimiento temprano en el juego sea malo y perder antes.

También tenemos que tener en cuenta que si dejamos de explorar los estados iniciales podemos caer en un comportamiento similar al fenómeno de "catastrophic forgetting" que vemos en transfer learning y olvidar como jugar en ellos.

Como fue mencionado anteriormente, Actor-Critic es muy inestable en su aprendizaje y cae en el problema de exploding gradients, con lo cual no pudimos finalizar una corrida con este algoritmo.

Created with  on Weights & Biases.

https://wandb.ai/jpds_mm/Reinforcement%20Learning/reports/Reinforcement-Learning--VmIldzo4NDU0NDQ4