

# AGENTES INTELIGENTES PARA EXPLoDING KITTENS MEDIANTE DEEP Q- LEARNING

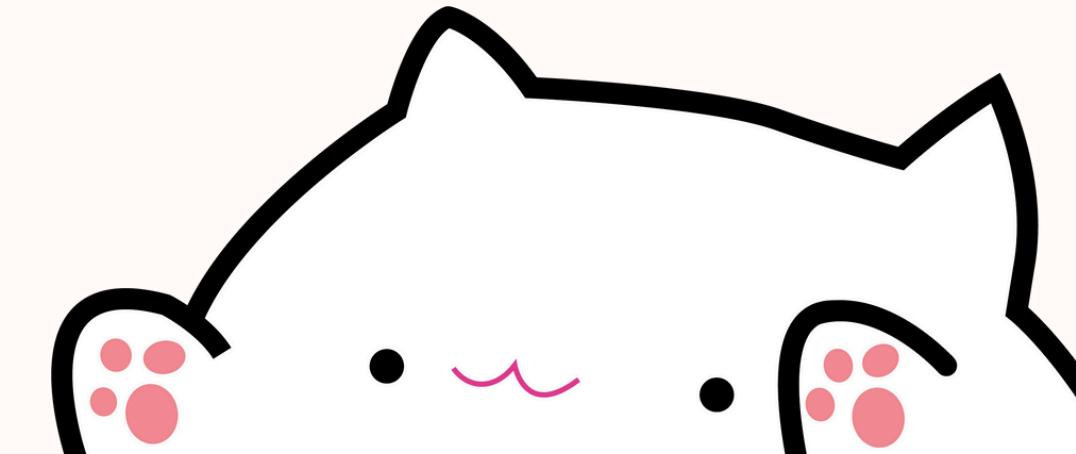
Grupo #1

# PROBLEMA





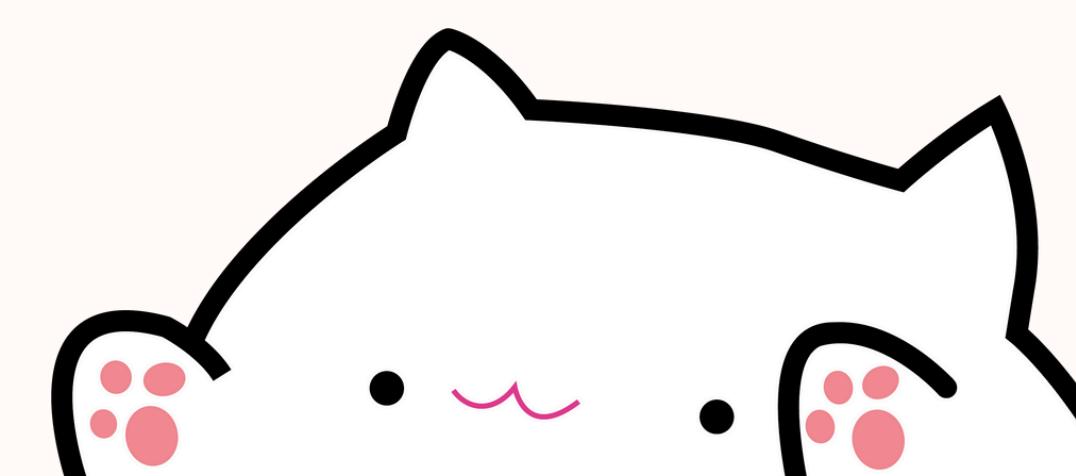
# PROBLEMA



- Espacio de estados complejo (probabilidad de bomba, cartas en mano, mano del oponente, oportunidad de defuse, etc)
- Espacio de acciones discreto, cartas del juego que el agente tiene.
- Recompensas escasas, victoria o derrota al final del juego.



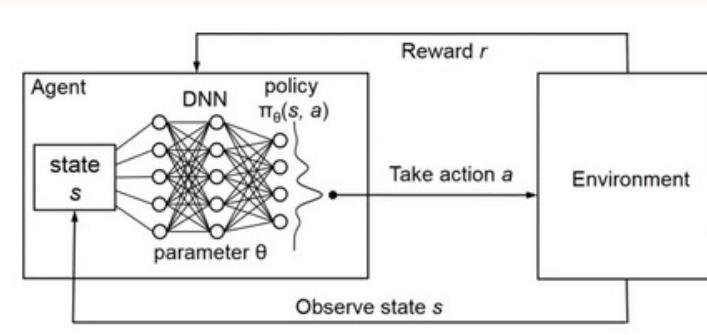
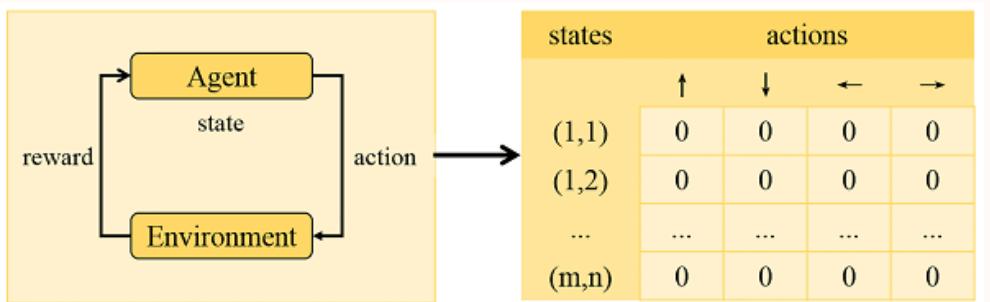
# ANÁLISIS



- El agente debe hacer una decisión secuencial bajo incertidumbre, cada turno.
- Tiene información parcial, solamente variables como cuantas cartas quedan, cuantas bombas, cartas propias, # cartas que tiene el oponente, y su ultima carta jugada.
- Recompensas sparse
- Espacio de estado continuo, grande.

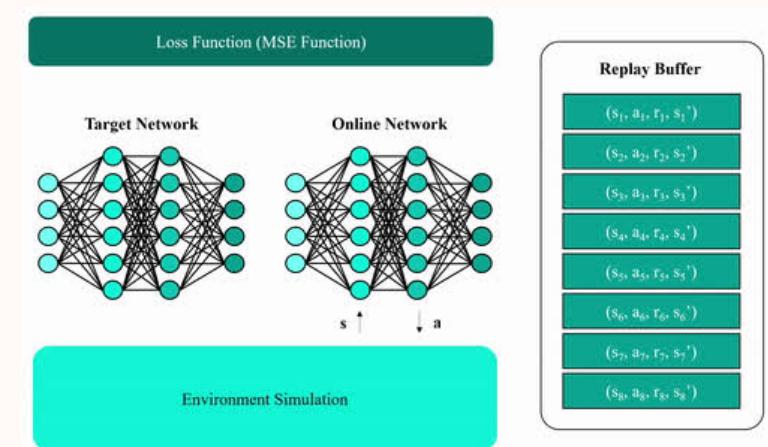


# MÉTODOS EVALUADOS

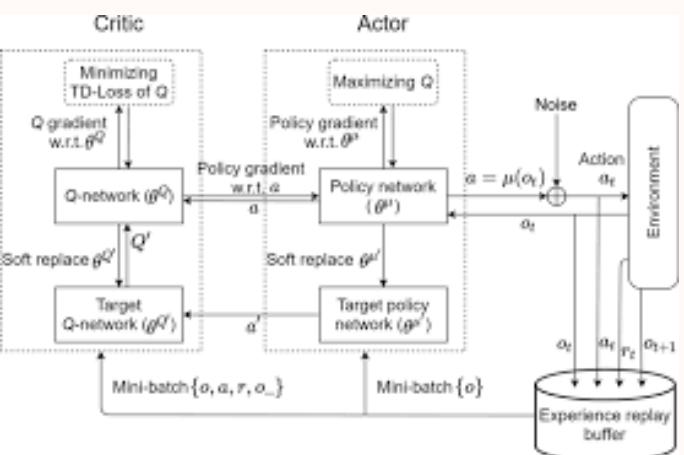


Q-Learning Tabular

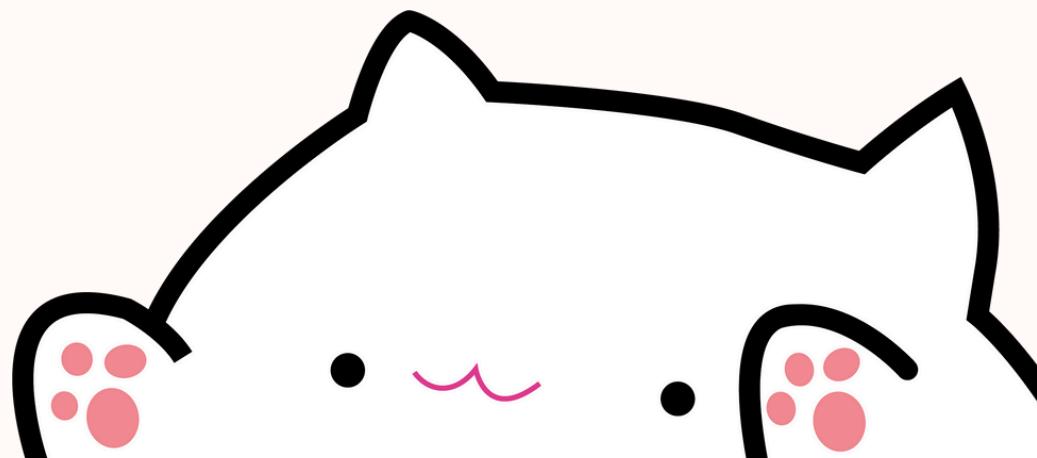
DQN



Double DQN

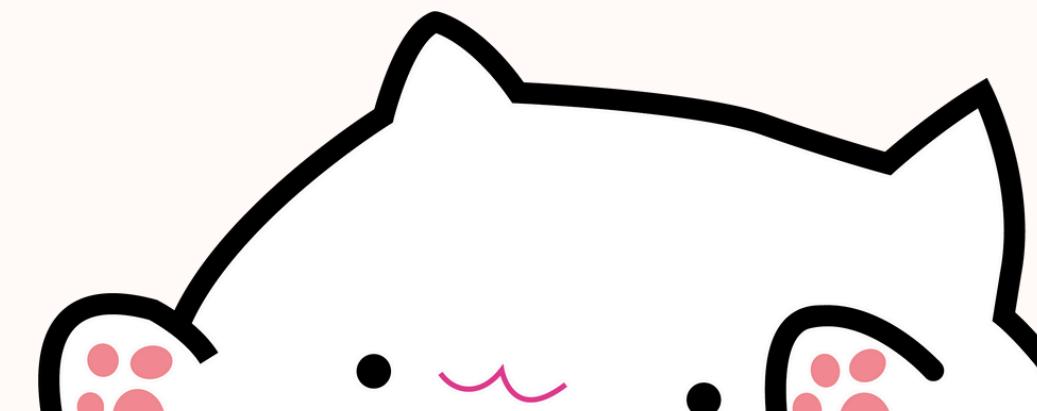


Policy Gradient / Actor Critic

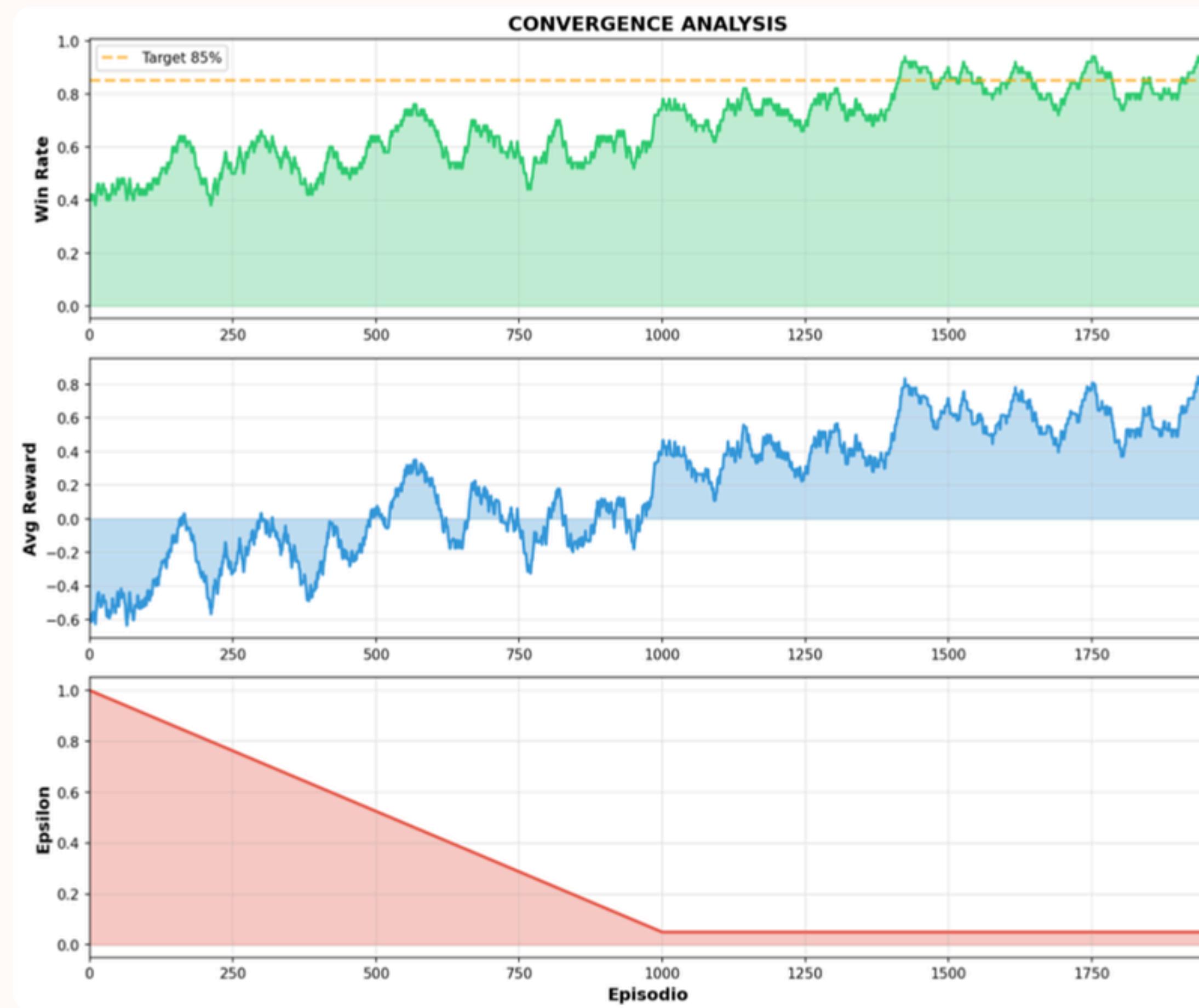




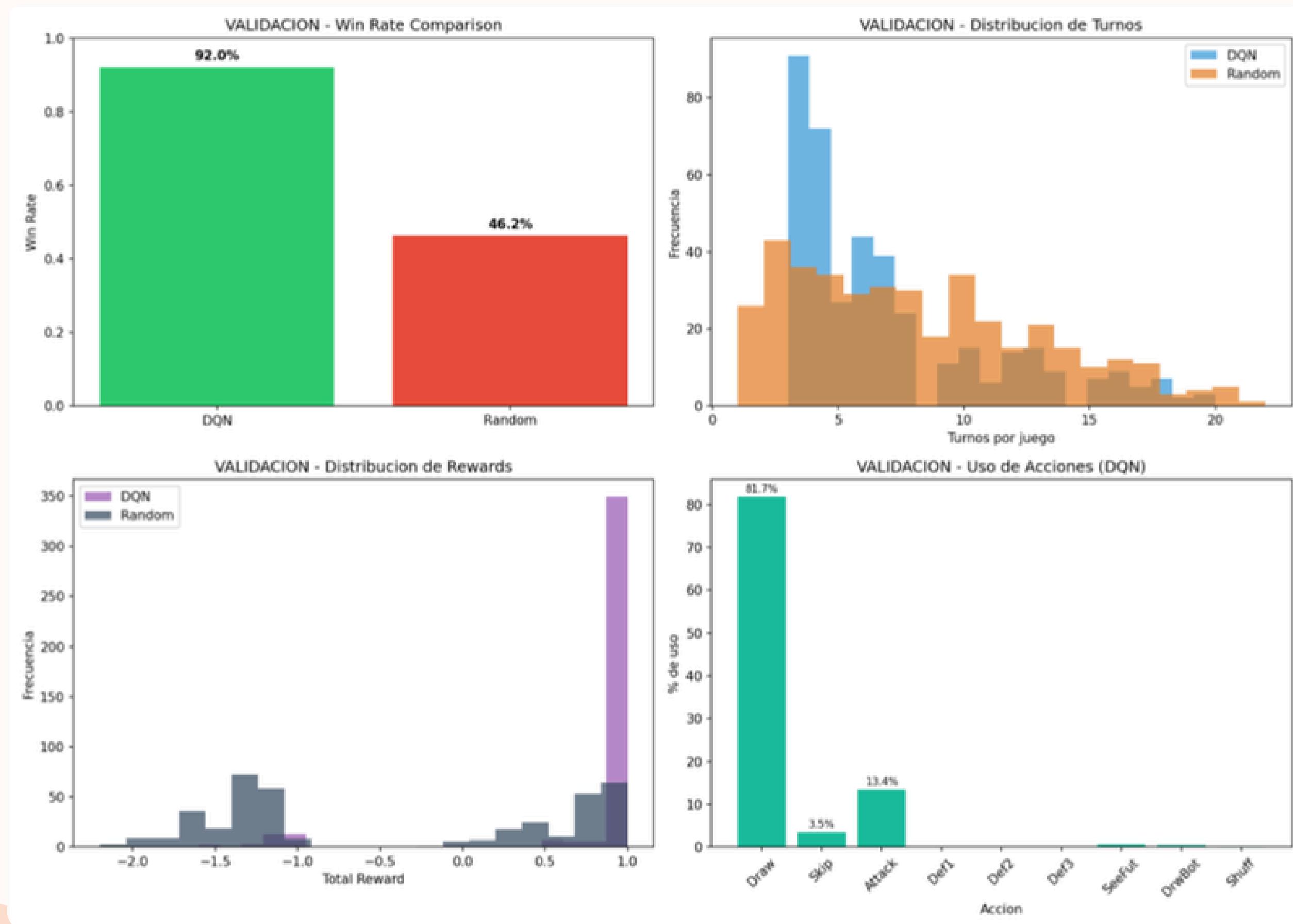
# PROPUUESTA DE SOLUCIÓN

- Entorno de juego tipo Gym que modela partida 1v1, mazo, y reglas del juego.
  - Agente Double DQN que aprende sobre un espacio de estados continuo con 11 dimensiones y tiene 9 acciones discretas posibles.
  - Entrenamiento basado en experience replay, target network actualizado periódicamente, reward shaping ligero y estrategia de exploración e-greedy con decay lineal.
- 

# RESULTADOS

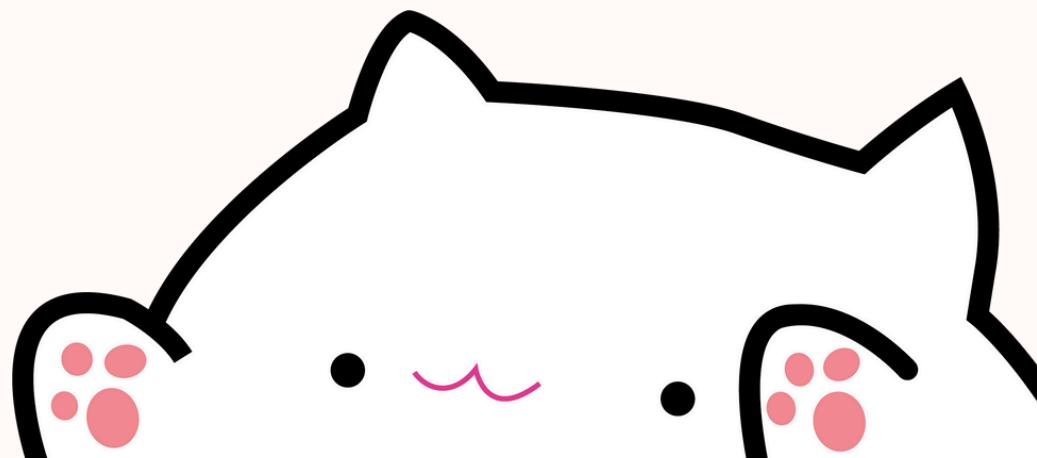


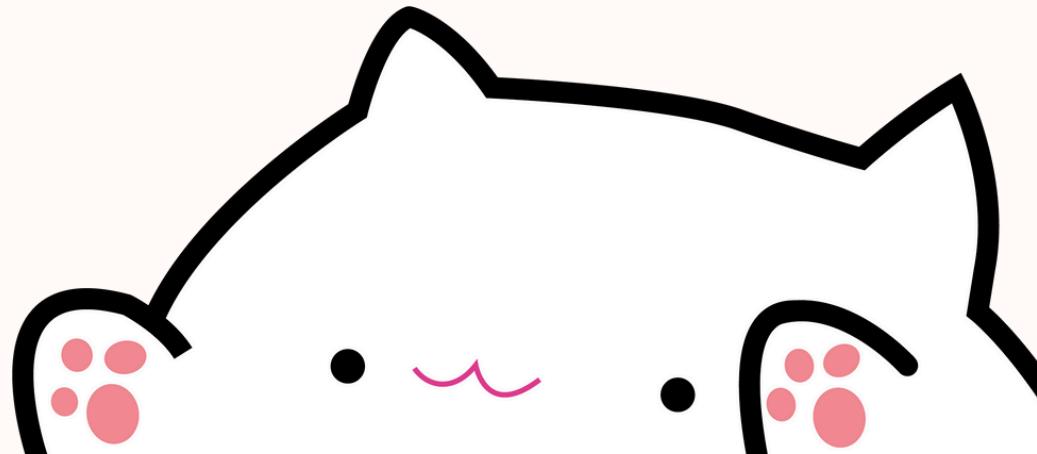
# RESULTADOS



# CONCLUSIONES

- Se desarrolló un agente capaz de aprender estrategias óptimas para el juego, mediante Double DQN.
- Mantener un espacio de estado optimizado con 11 variables fue clave para mejorar la eficiencia y rendimiento del modelo.
- Explorar con e-greedy y decaimiento lineal controlado permitió una transición progresiva y equilibrio, favoreciendo la convergencia sin perder diversidad de acciones.





JUGUEMOS  
CONTRA NUESTRO  
AGENTE



GRACIAS

