Uso de Modelos Generativos en Aprendizaje por Refuerzo

Silvia Barroso Moreno **silviabm98@ugr.es**

Directores: **Juan Gómez Romero y Miguel Molina Solana**
Departamento Ciencia de la Computación e Inteligencia Artificial
Trabajo Fin de Máster: Ciencia de Datos e Ingeniería de Computadores
2022-2023

Índice

1. Introducción
   - Motivación y descripción del problema
2. Marco teórico
   - Aprendizaje por Refuerzo
   - Redes Generativas Adversarias (GANs)
3. Aprendizaje por Imitación Generativo Adversario (GAIL)
4. Hibridación Q-learning (HQL)
5. Experimentación
   - Entornos GYM OpenAI
   - Entornos Sinergym
6. Conclusiones y vías futuras

Índice

1 Introducción
Motivación y descripción del problema
2 Marco teórico
Aprendizaje por Refuerzo
Redes Generativas Adversarias (GANs)
3 Aprendizaje por Imitación Generativo Adversario (GAIL)
4 Hibridación Q-learning (HQL)
5 Experimentación
Entornos GYM OpenAI
Entornos Sinergym
6 Conclusiones y vías futuras

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo
- Conexión:

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo
- Conexión:
  1. Modelo generativo ← GANs

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo
- Conexión:
  1. Modelo generativo ← GANs
  2. Aprendizaje por refuerzo ← Aprendizaje por Imitación

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo
- Conexión:
  1. Modelo generativo ← GANs
  2. Aprendizaje por refuerzo ← Aprendizaje por Imitación
- **Aprendizaje por Imitación**: el agente observa e imita el comportamiento del EXPERTO. NO tiene acceso al entorno NI a la recompensa.

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo
- Conexión:
  1. Modelo generativo $\leftarrow$ GANs
  2. Aprendizaje por refuerzo $\leftarrow$ Aprendizaje por Imitación
- **Aprendizaje por Imitación**: el agente observa e imita el comportamiento del EXPERTO. NO tiene acceso al entorno NI a la recompensa.
  1. **Aprendizaje por Imitación Generativo Adversario (GAIL)**

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo
- Conexión:
  1. Modelo generativo $\leftarrow$ GANs
  2. Aprendizaje por refuerzo $\leftarrow$ Aprendizaje por Imitación
- **Aprendizaje por Imitación**: el agente observa e imita el comportamiento del EXPERTO. NO tiene acceso al entorno NI a la recompensa.
  1. **Aprendizaje por Imitación Generativo Adversario (GAIL)**
  2. **Hibridación Q-Learning (HQL)** $\rightarrow$ nueva propuesta

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo
- Conexión:
  1. Modelo generativo $\leftarrow$ GANs
  2. Aprendizaje por refuerzo $\leftarrow$ Aprendizaje por Imitación
- **Aprendizaje por Imitación**: el agente observa e imita el comportamiento del EXPERTO. NO tiene acceso al entorno NI a la recompensa.
  1. **Aprendizaje por Imitación Generativo Adversario (GAIL)**
  2. **Hibridación Q-Learning (HQL)** $\rightarrow$ nueva propuesta
- **Experimentación**

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo
- Conexión:
  1. Modelo generativo ← GANs
  2. Aprendizaje por refuerzo ← Aprendizaje por Imitación
- **Aprendizaje por Imitación**: el agente observa e imita el comportamiento del EXPERTO. NO tiene acceso al entorno NI a la recompensa.
  1. **Aprendizaje por Imitación Generativo Adversario (GAIL)**
  2. **Hibridación Q-Learning (HQL)** → nueva propuesta
- **Experimentación**
  1. GYM OpenAI → Taxi y CartPole

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo
- Conexión:
  1. Modelo generativo ← GANs
  2. Aprendizaje por refuerzo ← Aprendizaje por Imitación
- **Aprendizaje por Imitación**: el agente observa e imita el comportamiento del EXPERTO. NO tiene acceso al entorno NI a la recompensa.
  1. **Aprendizaje por Imitación Generativo Adversario (GAIL)**
  2. **Hibridación Q-Learning (HQL)** → nueva propuesta
- **Experimentación**
  1. GYM OpenAI → Taxi y CartPole
  2. Sinergym → 5Zone, Datacenter y Warehouse

Motivación y descripción del problema

- **Objetivo**: establecer conexión entre modelos generativos y aprendizaje por refuerzo
- Conexión:
  1. Modelo generativo ← GANs
  2. Aprendizaje por refuerzo ← Aprendizaje por Imitación
- **Aprendizaje por Imitación**: el agente observa e imita el comportamiento del EXPERTO. NO tiene acceso al entorno NI a la recompensa.
  1. **Aprendizaje por Imitación Generativo Adversario (GAIL)**
  2. **Hibridación Q-Learning (HQL)** → nueva propuesta
- **Experimentación**
  1. GYM OpenAI → Taxi y CartPole
  2. Sinergym → 5Zone, Datacenter y Warehouse
  3. Proyecto de investigación → IA4TES

Índice

Aprendizaje por Refuerzo: Elementos básicos

- Espacio de estados $\mathcal{S}$ y espacio de acciones $\mathcal{A}$, $\Pi = \{\pi : \mathcal{S} \rightarrow \mathcal{A}\}$

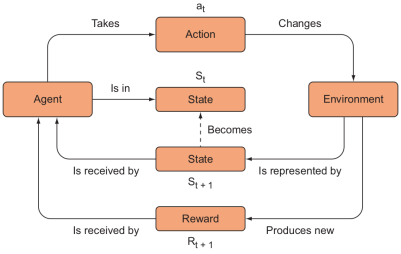


Figura: Funcionamiento de RL

Aprendizaje por Refuerzo: Elementos básicos

- Espacio de estados $\mathcal{S}$ y espacio de acciones $\mathcal{A}$, $\Pi = \{\pi : \mathcal{S} \rightarrow \mathcal{A}\}$
- La política se define como el conjunto de reglas que establece el mapeo de situaciones o estados del entorno a las acciones que el agente debe tomar con el fin de maximizar las recompensas a lo largo del tiempo.

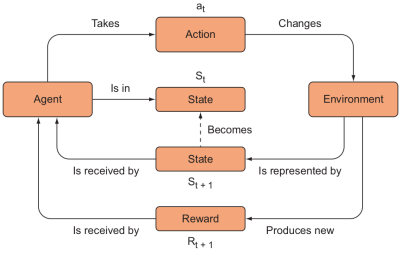


Figura: Funcionamiento de RL

Aprendizaje por Refuerzo: Elementos básicos

- Espacio de estados $\mathcal{S}$ y espacio de acciones $\mathcal{A}$, $\Pi = \{\pi : \mathcal{S} \rightarrow \mathcal{A}\}$
- La política se define como el conjunto de reglas que establece el mapeo de situaciones o estados del entorno a las acciones que el agente debe tomar con el fin de maximizar las recompensas a lo largo del tiempo.
- Señal de recompensa en el paso t: R_t

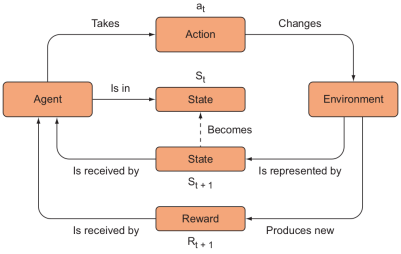


Figura: Funcionamiento de RL

Aprendizaje por Refuerzo: Elementos básicos

- Espacio de estados $\mathcal{S}$ y espacio de acciones $\mathcal{A}$, $\Pi = \{\pi : \mathcal{S} \rightarrow \mathcal{A}\}$
- La política se define como el conjunto de reglas que establece el mapeo de situaciones o estados del entorno a las acciones que el agente debe tomar con el fin de maximizar las recompensas a lo largo del tiempo.
- Señal de recompensa en el paso t: R_t
- Función valor: $V_\pi(S) = \mathbb{E}_\pi[G_t | S_t = S]$, $G_t = R_1 + R_2 + \ldots + R_t$

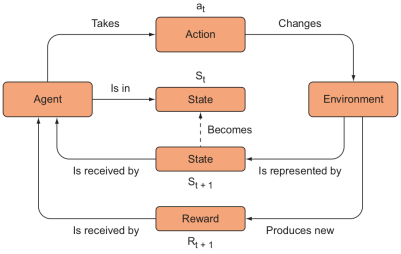


Figura: Funcionamiento de RL

Aprendizaje por Refuerzo: Elementos básicos

- Espacio de estados $\mathcal{S}$ y espacio de acciones $\mathcal{A}$, $\Pi = \{\pi : \mathcal{S} \rightarrow \mathcal{A}\}$
- La política se define como el conjunto de reglas que establece el mapeo de situaciones o estados del entorno a las acciones que el agente debe tomar con el fin de maximizar las recompensas a lo largo del tiempo.
- Señal de recompensa en el paso t: R_t
- Función valor: $V_\pi(S) = \mathbb{E}_\pi[G_t|S_t = S]$, $G_t = R_1 + R_2 + \ldots + R_t$
- Función acción-valor: $Q_\pi(S, A) = \mathbb{E}_\pi[G_t|S_t = S, A_t = A]$

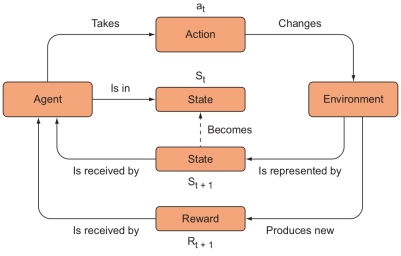


Figura: Funcionamiento de RL

Métodos para soluciones tabulares

Métodos para soluciones tabulares: **Algoritmo Q-Learning**

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma \,\text{máx}_a\, Q(S_{t+1}, a) - Q(S_t, A_t)]$$

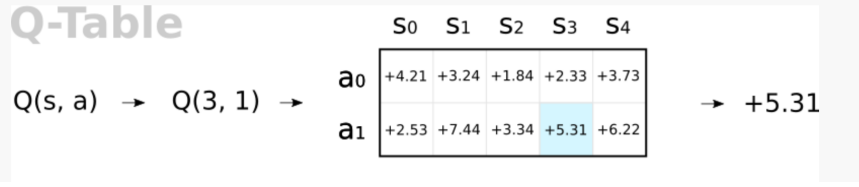


Figura: Acceso a la tabla Q(S,A)

Métodos para soluciones aproximadas

Métodos para soluciones aproximadas: **Proximal Policy Optimixation (PPO)**

- Incorpora una red neuronal para realizar la aproximación
- Corresponden a métodos que calculan gradientes de políticas

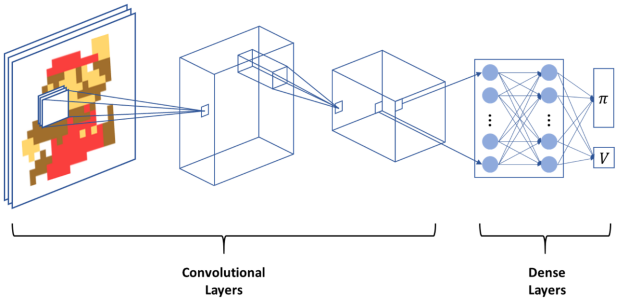


Figura: Ejemplo PPO: aprender a jugar a Mario Bros

Aprendizaje por Refuerzo Profundo (DRL)

La **estimación de la función ventaja** en el timestep t, $\bar{A}_t$, se define como

$$\bar{A}_t = \delta_t + (\gamma\lambda)\delta_{t+1} + \ldots + \ldots + (\gamma\lambda)^{T-t+1}\delta_{T-1} \text{ donde } \delta_t = r_t + \gamma V(s_{t+1}) - V(s_t)$$

- Recordemos, la función ventaja se define como $A_t := Q_t(s,a) - V_t(s)$

Nuestra **función de pérdida**:

$$L^{CPI}(\theta) = \mathbb{E}_t[min(r_t(\theta)\bar{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\bar{A}_t)]$$

Algoritmo PPO

for $i = 1$ **to** M **do**
  for $j = 1$ **to** N **do**
    - Ejecutar la política $\pi_{\theta_{Old}}$ en el entorno con T timesteps
    - Calcular las estimaciones de la función ventajas $\bar{A}_1, \bar{A}_2, \ldots, \bar{A}_T$
  end
  - Optimizar el objetivo clipped surrogated L^{CPI}, con K épocas y tamaño de minibatch $M \leq NT$
  - $\theta_{Old} \rightarrow \theta$
end

Redes Generativas Adversarias (GANs)

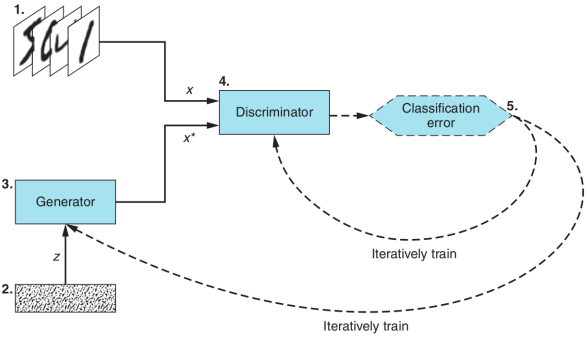


Figura: Funcionamiento de una GANs

TFG: Redes Generativas Adversarias para la creación de deepfakes:

`https://github.com/silviabm98/TFG`

Índice

Aprendizaje por Imitación Generativo Adversario (GAIL)

Descripción GAIL

- Aprendizaje de una política π (generador) y de un discriminador D
- El generador trata de imitar a la política experta, π_E, de la secuencia $[s, a]$, generando una secuencia falsa $[s, a]^*$ con la política π

Definimos

$$RL \circ IRL_\psi(\pi_E) = \arg\min_{\pi \in \Pi}(-H(\pi) + \psi^*(\rho_\pi - \rho_{\pi_E}))$$

1. siendo ρ_π la **medida de ocupación** de la política $\pi \in \Pi$, definida como $\rho_\pi(s, a) = \pi(a|s) \sum_{t=0}^{\infty} \gamma^t P(s_t = s|\pi)$
2. $H(\pi) \triangleq \mathbb{E}_\pi[-\log \pi(a|s)]$
3. ψ^* es la conjugada convexa de ψ

Aprendizaje por Imitación Generativo Adversario (GAIL)

Conexión entre aprendizaje por imitación y GANs

$$\psi_{GA} \triangleq \begin{cases} \mathbb{E}_{\pi_E}[g(c(s,a))] & si \quad c \leq 0 \\ +\infty & \text{en otro caso} \end{cases} \tag{1}$$

donde

$$g(x) = \begin{cases} -x - \log(1 - \exp x) & si \quad x \leq 0 \\ +\infty & \text{en otro caso} \end{cases} \tag{2}$$

$$\Downarrow$$

OBJETIVO: Encontrar un punto de silla (π, D) en la siguiente expresión

$$\mathbb{E}_{\pi}[\log(D(s,a))] + \mathbb{E}_{\pi_E}[\log(1 - D(s,a))] - \lambda H(\pi)$$

Algoritmo GAIL

Algoritmo GAIL:

Para cada trayectoria i**:**

for $i = 1$ **to** N **do**

1. Actualizamos los parámetros del Discriminador de ω_i a ω_{i+1} con el gradiente:

$$\mathbb{E}_{\pi_i}[\nabla_\omega \log(D_\omega(s,a))] + \mathbb{E}_{\pi_E}[\nabla_\omega \log(1 - D_\omega(s,a))]$$

2. Tomamos la política π_{θ_i} y actualizamos la política $\pi_{\theta_{i+1}}$ utilizando PPO con su función de coste. Realizamos la actualización del gradiente:

$$\mathbb{E}_{\pi_i}[\nabla_\theta \log \pi_\theta(a|s)\mathcal{Q}(s,a)] - \lambda \nabla_\theta H(\pi_\theta)$$

donde $\mathcal{Q}(\bar{s},\bar{a}) = \mathbb{E}_{\theta_i}[\log(D_{\omega_{i+1}}(s,a))|s_0 = \bar{s}, a_0 = \bar{a}]$

end

Índice

Hibridación Q-learning (HQL)

Descripción HQL

1. Aprendizaje de una política π (generador) y de un discriminador D
2. El generador trata de imitar a la política experta, π_E, de la tabla $Q[s, a]$, generando una tabla falsa $Q[s, a]^*$ con la política π

Base de datos experta:

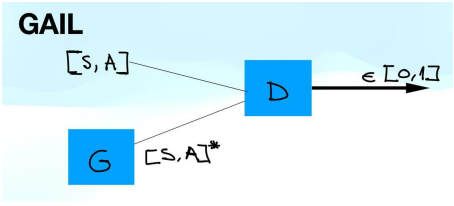
$$Q(S, A) = \{Q(S, A, 1), Q(S, A, 2), \dots Q(S, A, n)\}, \forall (S, A) \in \mathcal{S} \times \mathcal{A}$$

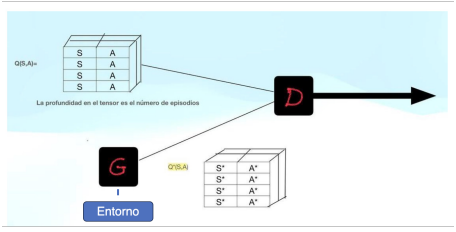
Vanilla GAN

$\min_G \max_D V(D, G) = \min_{Q^*} \max_D V(D, Q^*)$

$$= \min_{Q^*} \max_D (\mathbb{E}_{Q(S,A) \sim P_Q}[\log D(Q(S, A))] + \mathbb{E}_{Q^*(S,A) \sim P_{Q^*}}[\log(1 - D(Q^*(S, A)))]) \quad (3)$$

Hibridación Q-learning (HQL)





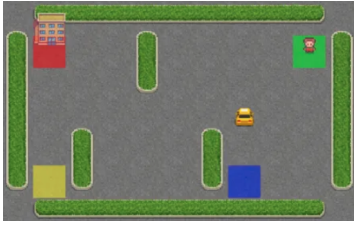
Índice

Experimentación

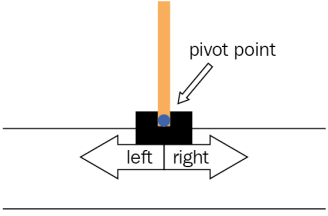
	Observacciones		Acciones		Algoritmo	
Entornos	**Discreto**	**Continuo**	**Discreto**	**Continuo**	**GAIL**	**HQL**
Taxi - Gym	×		×		×	×
CartPole - Gym		×	×		×	
5Zone - Sinergym		×	×		×	
Datacenter - Sinergym		×	×		×	
Warehouse - Sinergym		×	×		×	

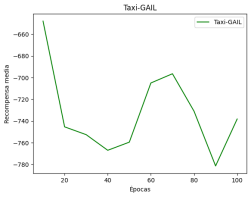
Entornos GYM OpenAI

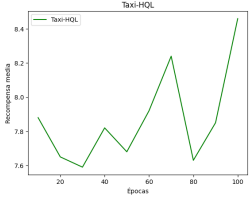
Taxi

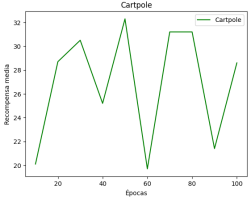


CartPole



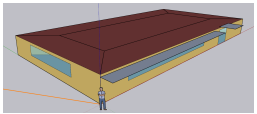


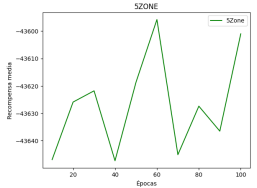




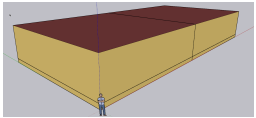
Entornos Sinergym

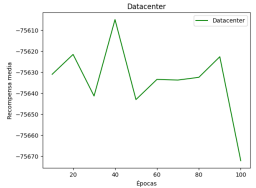
5Zone



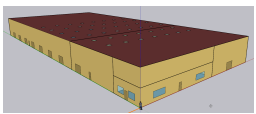


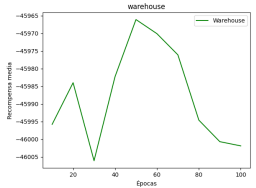
Datacenter





Warehouse





Comparativa

	GAIL	GAIL	GAIL	GAIL	GAIL	HQL
Épocas	5ZONE	DATACENTER	WAREHOUSE	CARTPOLE	TAXI	TAXI
10	−43646.9	−75631.07	−45995.8	20.1	−648.0	7.88
20	−43625.9	−75621.6	−45984.0	28.7	−745.4	7.65
30	−43621.8	−75641.4	−46006.1	30.5	−752.6	7.59
40	−43647.3	−75605	−45982.3	25.2	−767.0	7.82
50	−43618.7	−75643.09	−45966.1	32.3	−759.5	7.68
60	−43595.8	−75633.5	−45970.1	19.7	−705.0	7.92
70	−43645.1	−75633.8	−45976.10	31.2	−696.5	8.24
80	−43627.4	−75632.5	−45994.6	31.2	−731.0	7.63
90	−43636.5	−75622.7	−46000.7	21.4	−781.4	7.85
100	−43601.01	−75672.2	−46001.9	28.6	−738.2	8.46

Índice

Conclusiones y vías futuras

1. Conexión entre RL y los modelos generativos → GANs y Aprendizaje por Imitación
   - -GAIL
   - -HQL
   - -Distintos entornos: CartPole, Taxi, 5Zone, Datacenter, Warehouse
2. LINEA FUTURA: Establecer nueva conexión entre RL y modelos generativo
   - -Nuevo modelo generativo, por ejemplo Decision Difusser
   - -Mejorar los experimentos realizados con CartPole, Taxi, 5Zone, Datacenter, Warehouse
   - -Nuevos entornos distintos a CartPole, Taxi, 5Zone, Datacenter, Warehouse...
   - -Seguir investigando sobre la nueva propuesta Hibridación Q-Learning (HQL)

¡GRACIAS POR SU ATENCIÓN!