UNIVERSIDAD EAFIT

Escuela de Ciencias Aplicadas e Ingeniería

Artificial Intelligence

Lecture10 – Reinforcement Learning

Contenido

UNIVERSIDAD EAFIT
Escuela de Ciencias Aplicadas e Ingeniería

1. Introducción
2. Complicaciones del RL
3. Fundamentos del RL
4. Q Learning

INTRODUCCIÓN

Introducción

UNIVERSIDAD EAFIT
Escuela de Ciencias Aplicadas e Ingeniería

Que es el reinforcement learning?

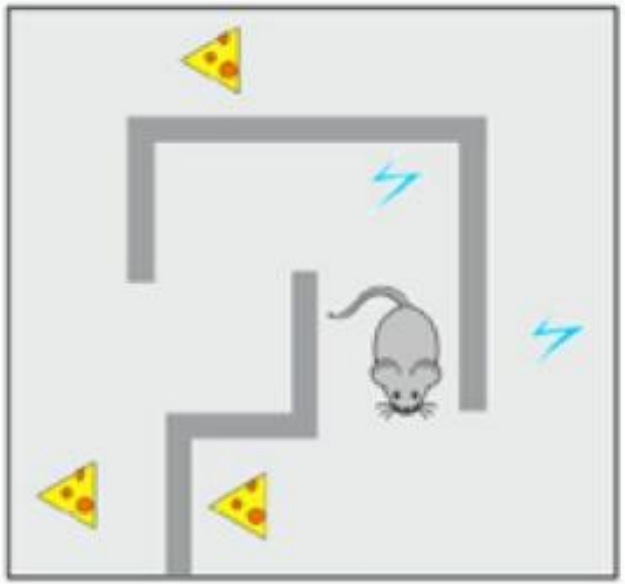
Se refiere al proceso de aprender qué hacer y mapear situaciones a ciertas acciones para maximizar la recompensa.

En la mayoría de los paradigmas de aprendizaje automático, a un agente de aprendizaje se le dice qué acciones tomar para lograr ciertos resultados. En el caso del aprendizaje por refuerzo, no se le dice al agente de aprendizaje qué acciones tomar.

Introducción

Que es el reinforcement learning?

Imagina que tienes un agente que necesita realizar acciones en un entorno determinado.



El ratón robot intenta encontrar tanta comida como sea posible, evitando recibir descargas eléctricas siempre que sea posible. Estas señales de comida y electricidad son la recompensa que el entorno otorga al agente (ratón robot) como retroalimentación adicional sobre las acciones del agente

Introducción

Que son las recompensas en RL?

Una recompensa es un numero (generalmente un escalar) que el entorno envía al agente.

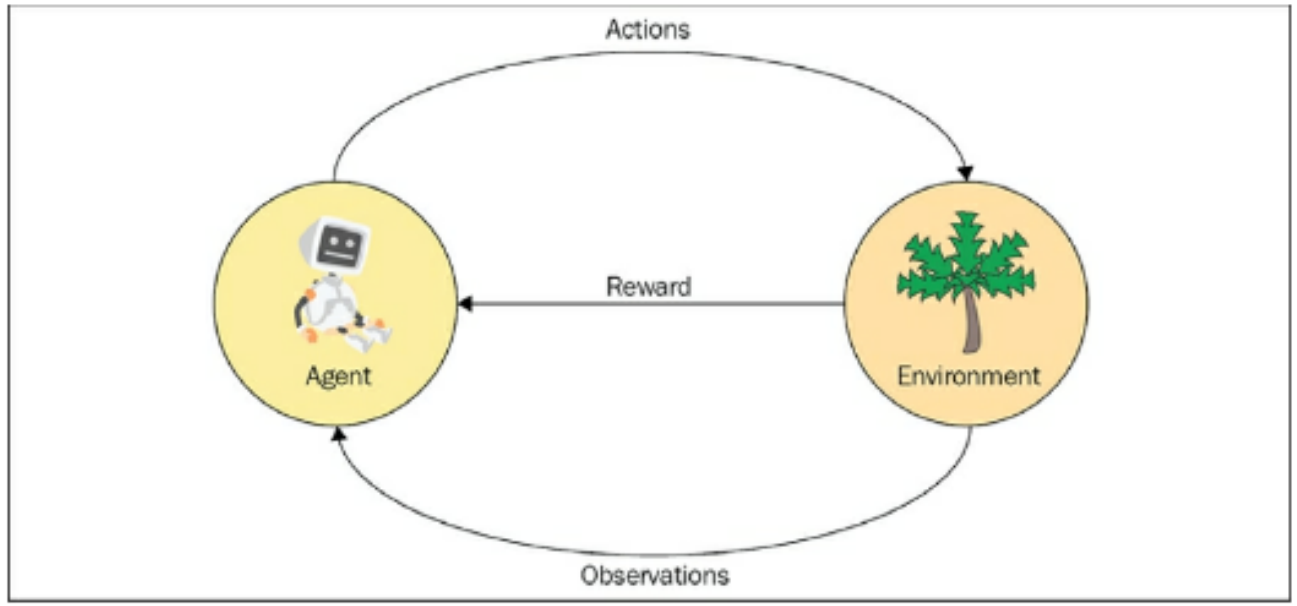


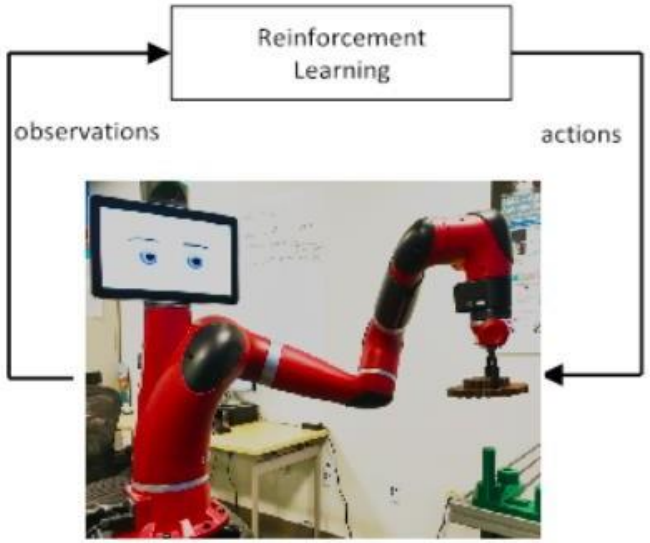
Figure 1.2: RL entities and their communication channels

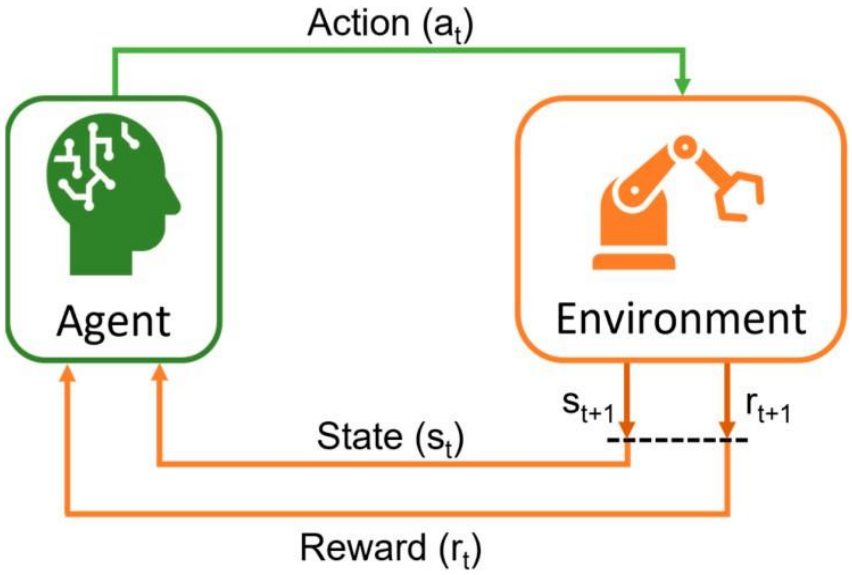
El objetivo final del agente es obtener la mayor recompensa total posible. En nuestro ejemplo concreto, el ratón robot podría sufrir una ligera descarga eléctrica para llegar a un lugar con mucha comida, lo que sería un resultado mejor para el ratón robot que quedarse quieto y no ganar nada.

Introducción

Aplicaciones del RL

- Juegos de mesa
- Robótica
- Controladores Industriales
- Bebés





Introducción

UNIVERSIDAD EAFIT
Escuela de Ciencias Aplicadas e Ingeniería

Componentes del RL

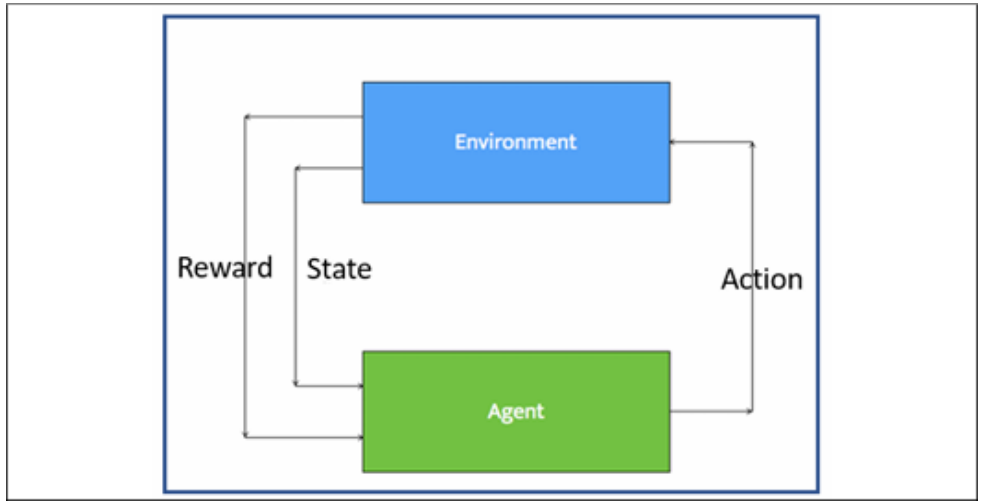
Conjunto de estados relacionados con el agente y el entorno

Políticas que determinan qué acción debe tomarse

El agente toma la acción

El entorno reacciona en respuesta a esa acción

El agente calcula y registra la información sobre esta recompensa



COMPLICACIONES DEL RL

Complicaciones del RL

Lo primero que hay que tener en cuenta es que la observación en RL depende del comportamiento de un agente y, en cierta medida, es el resultado de este comportamiento. Si tu agente decide hacer cosas ineficaces, las observaciones no te dirán nada sobre lo que ha hecho mal y lo que se debería hacer para mejorar el resultado (el agente solo recibirá comentarios negativos todo el tiempo).

La segunda cosa que complica la vida de nuestro agente es que no solo necesita **explotar** el conocimiento que ha aprendido, sino también **explorar** activamente el entorno, porque tal vez hacer las cosas de manera diferente mejore significativamente el resultado.

Complicaciones del RL

El tercer factor de complicación radica en el hecho de que **la recompensa puede retrasarse considerablemente después de las acciones.** En el ajedrez, por ejemplo, una sola jugada fuerte en medio de la partida puede cambiar el equilibrio.

Durante el aprendizaje, necesitamos descubrir esos eventos, que pueden ser difíciles de discernir durante el transcurso del tiempo y nuestras acciones.

FUNDAMENTOS DEL RL

Fundamentos del RL

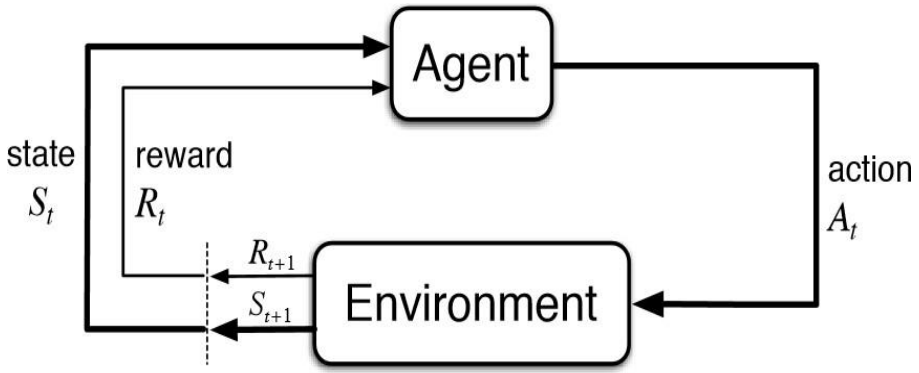
UNIVERSIDAD EAFIT | Escuela de Ciencias Aplicadas e Ingeniería

REWARD

En RL, es solo un valor escalar que obtenemos periódicamente del entorno.

La recompensa puede ser positiva o negativa, grande o pequeña, pero es solo un número.

El propósito de la recompensa es indicar a nuestro agente lo bien que se ha comportado.



La recompensa es local, lo que significa que refleja el éxito de la actividad reciente del agente y no todos los éxitos logrados por el agente hasta el momento.

Fundamentos del RL

UNIVERSIDAD EAFIT
Escuela de Ciencias Aplicadas e Ingeniería

Operaciones financieras: una cantidad de ganancias es una recompensa para un operador que compra y vende acciones.

Sistema de dopamina en el cerebro: Hay una parte del cerebro (sistema límbico) que produce dopamina cada vez que necesita enviar una señal positiva al resto del cerebro. Las concentraciones más altas de dopamina provocan una sensación de placer, lo que refuerza las actividades que este sistema considera buenas.

Juegos de PC: suelen proporcionar una retroalimentación evidente al jugador, que puede ser el número de enemigos eliminados o la puntuación obtenida. En estos casos la recompensa ya está acumulada, por lo que la recompensa RL para los juegos arcade debería ser la derivada de la puntuación, es decir, +1 cada vez que se elimina a un nuevo enemigo y 0 en todos los demás intervalos de tiempo.

Fundamentos del RL

AGENTS
Un agente es alguien o algo que **interactúa con el entorno** ejecutando determinadas acciones, realizando observaciones y recibiendo recompensas por ello. En la mayoría de los escenarios prácticos de RL, el agente es nuestro software, que se supone que debe resolver algún problema de una manera más o menos eficiente.

Operaciones financieras: un sistema de operaciones o un operador que toma decisiones sobre la ejecución de órdenes.

Ajedrez: un jugador o un programa informático.

Sistema de dopamina: el propio cerebro, que, según los datos sensoriales, decide si ha sido una buena experiencia.

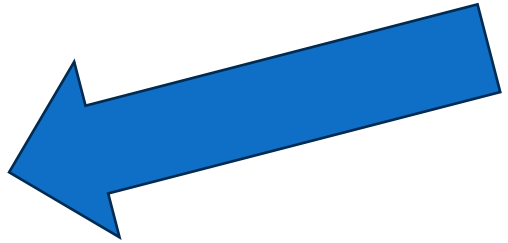
Juegos de ordenador: el jugador que disfruta del juego o el programa informático.

Fundamentos del RL

ENVIRONMENT

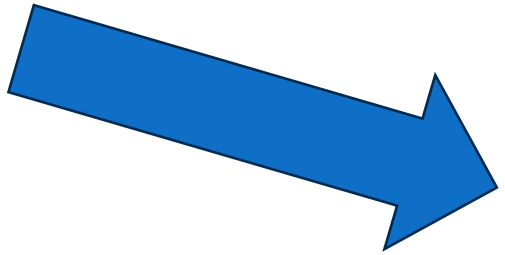
El entorno es todo lo que hay fuera de un agente. En el sentido más general, es el resto del universo

La comunicación del agente con el entorno se limita a la recompensa (obtenida del entorno), las acciones (ejecutadas por el agente y dadas al entorno) y las observaciones (alguna información además de la recompensa que el agente recibe del entorno).



ACTIONS

son cosas que un agente puede hacer en el entorno. Las acciones pueden ser, por ejemplo, movimientos permitidos por las reglas del juego (si se trata de un juego)

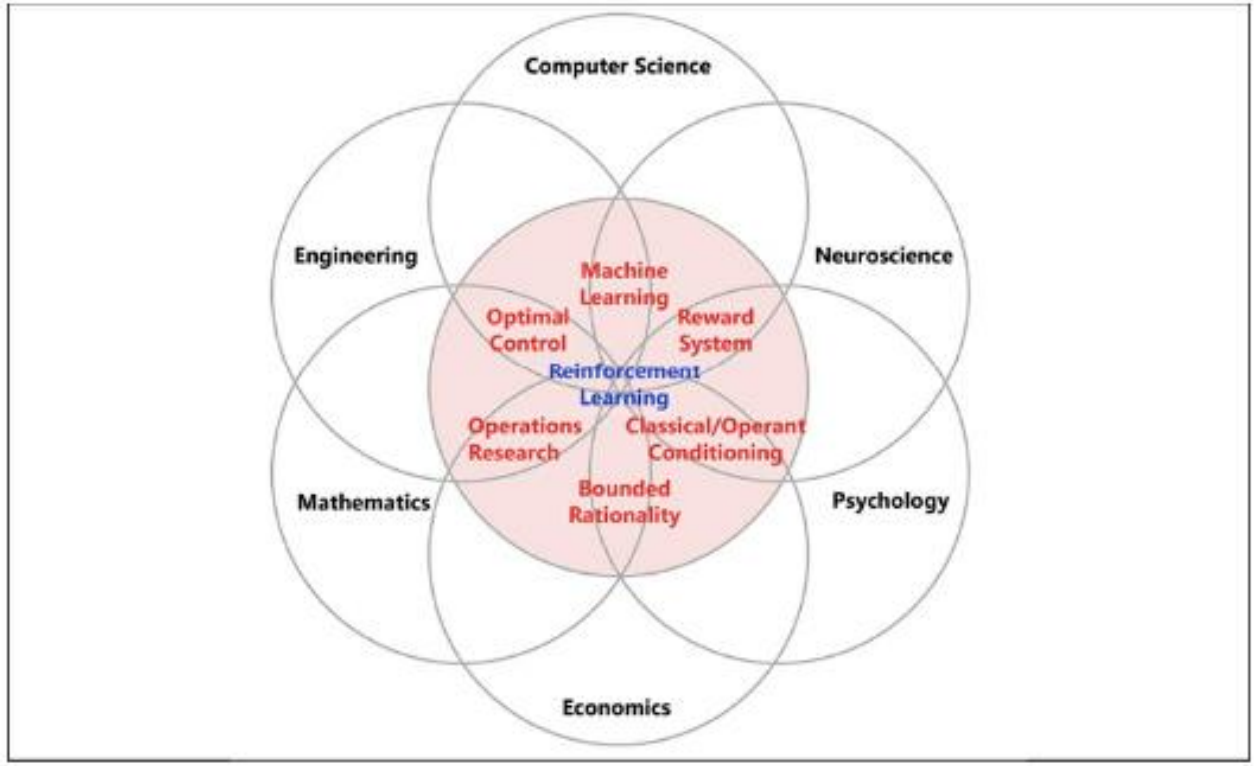


OBSERVATIONS

Las observaciones son datos que el entorno proporciona al agente y que le indican lo que está sucediendo a su alrededor.

Fundamentos del RL

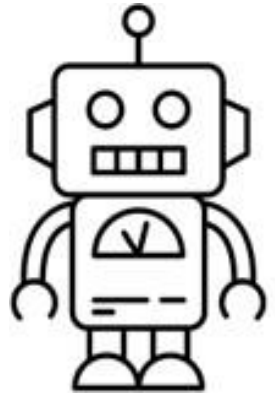
Hay muchas otras áreas que contribuyen o se relacionan con el RL. Las más significativas se muestran en el siguiente diagrama, que incluye seis grandes dominios que se superponen en gran medida entre sí en cuanto a los métodos y temas específicos relacionados con la toma de decisiones



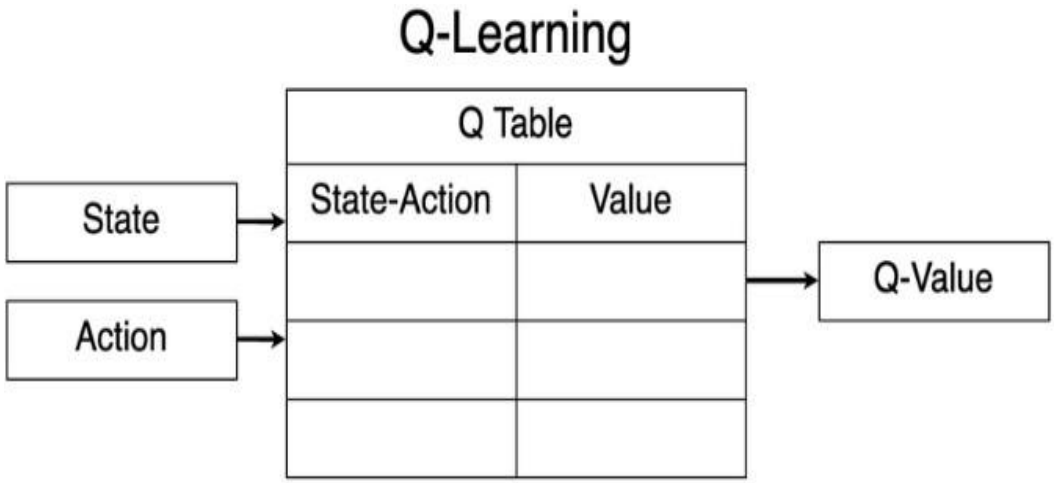
Q-LEARNING

Q learning

Es un algoritmo de aprendizaje por refuerzo que permite a un agente aprender a tomar decisiones en un entorno de manera óptima a través de prueba y error. Es particularmente útil en problemas en los que el agente necesita tomar una secuencia de acciones para maximizar una recompensa acumulada en el tiempo.



Q learning



$$Q = \begin{pmatrix} Q(s_1, a_1) & Q(s_1, a_2) \\ Q(s_2, a_1) & Q(s_2, a_2) \end{pmatrix}$$

Es una técnica de RL basada en la idea de aprender una función de valores, llamada *función Q*, que estima el valor de tomar una acción específica en un estado específico.

Q-learning es más adecuado para problemas de RL discretos y con un espacio de estado/acción pequeño, ya que requiere almacenar y actualizar una tabla Q.

Q learning

UNIVERSIDAD EAFIT

Escuela de Ciencias Aplicadas e Ingeniería

Método basado en valores

Se centra en aprender una *función de valor de acción* Q(s,a)

Actualizacion interactiva

La función Q se actualiza de forma iterativa con cada acción que el agente toma, utilizando una fórmula de actualización basada en la recompensa inmediata y el valor Q del siguiente estado

Balance entre exploración y explotación

Utiliza un mecanismo (como la política epsilon-greedy) para equilibrar entre *exploración* (probar nuevas acciones para descubrir recompensas potenciales) y *explotación* (elegir la mejor acción conocida según la función Q actual).

Q learning

1. Inicialización de la tabla Q

Se inicializa la tabla Q con valores arbitrarios (usualmente ceros) para cada posible par estado-acción (s,a)

2. Exploración del entorno

El agente comienza en un estado inicial y, en cada paso, selecciona una acción a basada en la política actual

3. Ejecuta la acción y observa el resultado

Tras ejecutar la acción a en el estado s, el agente observa: La recompensa r obtenida y el nuevo estado.

4.Actualización de la función Q:

La tabla Q se actualiza utilizando la siguiente fórmula de *actualización Q*:

$$Q(s,a) = Q(s,a) + \alpha \left[r + \gamma \max_{a'} Q(s',a') - Q(s,a) \right]$$

Q learning

$$Q(s,a) = Q(s,a) + \alpha \left[r + \gamma \max_{a'} Q(s',a') - Q(s,a) \right]$$

- **Donde α** es la *tasa de aprendizaje*, que controla cuánto se actualiza el valor de Q.
- **Γ** es el *factor de descuento*, que pondera la importancia de las recompensas futuras frente a las recompensas inmediatas.
- r es la recompensa inmediata obtenida por tomar la acción a en el estado s.
- **max Q(s',a)** es el valor Q máximo en el siguiente estado *s'*, lo que representa el valor estimado de la mejor acción en el próximo estado.

Q learning

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

New Q-value estimation

Former Q-value estimation

Learning Rate

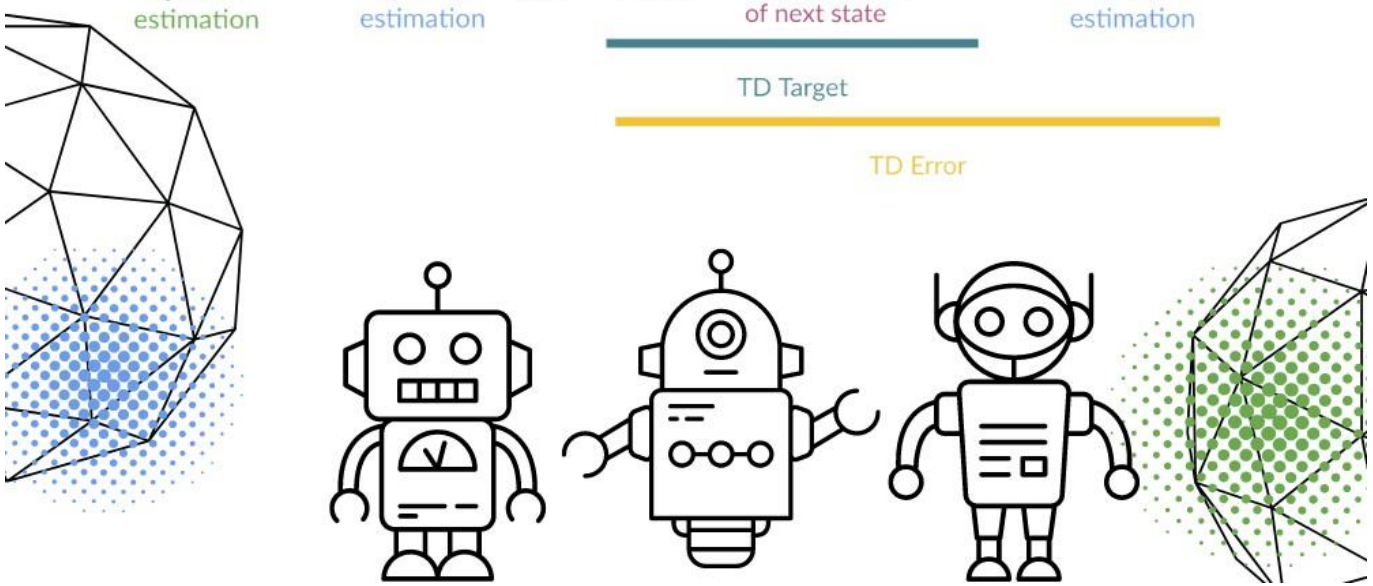
Immediate Reward

Discounted Estimate optimal Q-value of next state

Former Q-value estimation

TD Target

TD Error



5.Repetición hasta la convergencia

Este proceso de explorar, seleccionar acciones y actualizar la función Q se repite a través de múltiples episodios hasta que la tabla Q converge a valores estables.

Cuando esto ocurre, la función Q representa la política óptima, permitiendo al agente seleccionar la acción que maximiza la recompensa acumulada en cada estado.

Q learning

EJEMPLO

Supongamos un entorno con dos estados, s1y s2, y dos acciones posibles en cada estado, a1 y a2. La tabla Q podría verse así al inicio:

$$Q = \begin{pmatrix} Q(s_1, a_1) & Q(s_1, a_2) \\ Q(s_2, a_1) & Q(s_2, a_2) \end{pmatrix}$$

Al comienzo, todos los valores son cero. El agente empieza en un estado (digamos s1), elige una acción (por ejemplo, a1), observa la recompensa inmediata y el próximo estado, y actualiza el valor Q(s1,a1)) usando la fórmula de actualización.

Q learning

UNIVERSIDAD EAFIT
Escuela de Ciencias Aplicadas e Ingeniería

LIMITACIONES

- No es eficiente en problemas con grandes espacios de estado (problemas de escalabilidad), ya que requiere una tabla Q de gran tamaño.
- Dificultades con entornos con espacios de acción continuos.
- Puede ser lento en converger si el entorno es complejo o si la tasa de aprendizaje y el factor de descuento no están bien ajustados.

EJERCICIO DE CLASE!!