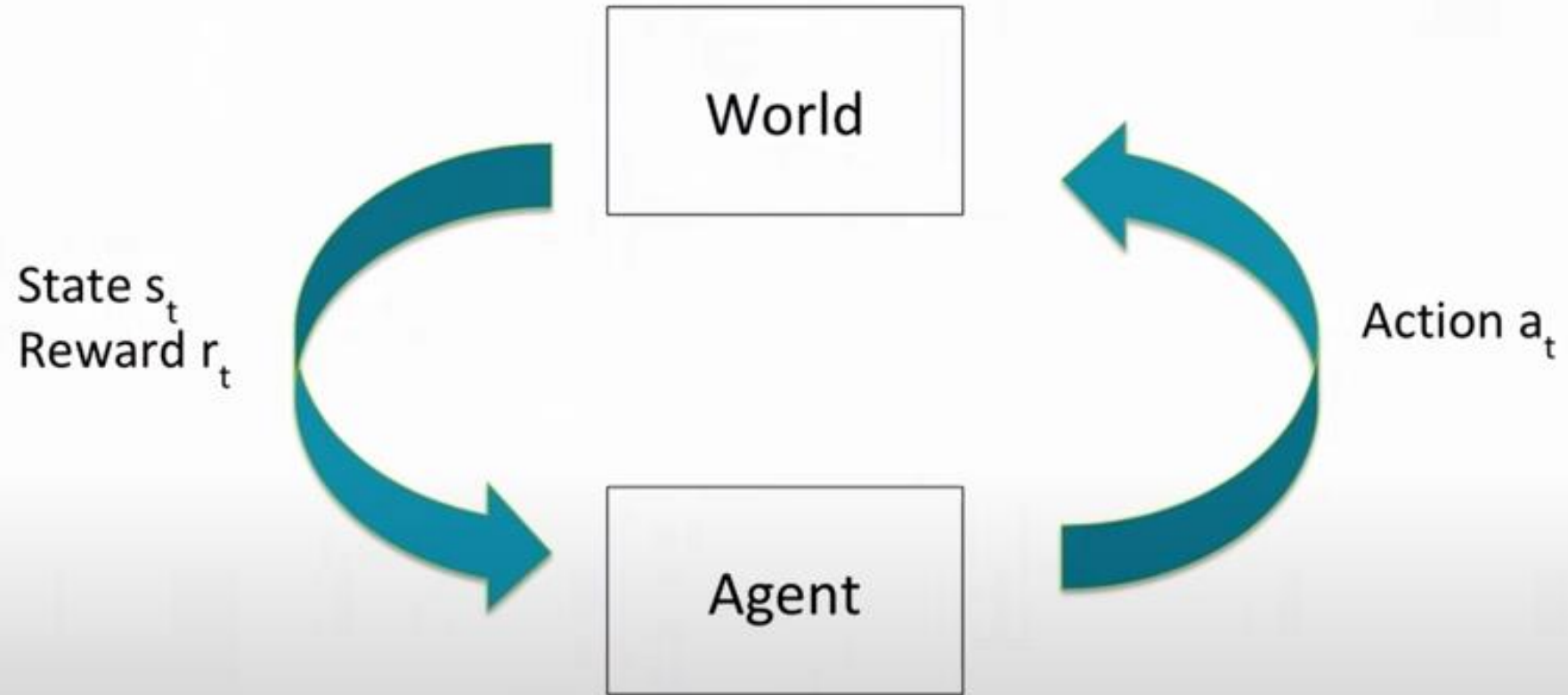


Clase 6.1

Análisis Markoviano

Jorge Vasquez

Proceso de Decisión de Márkov (MDP)



Supuesto de Márkov

- Información del Estado: puedes predecir la probabilidad de lo que va a pasar
- Estado **s_t** es **Markov** si y solo si:
- El futuro solo depende del valor presente

$$p(s_{t+1}|s_t, a_t) = p(s_{t+1}|h_t, a_t)$$

Supuesto de Markov

- El futuro solo depende de su valor presente
 - **Control de hipertensión:** el estado es la presión de sangre, y la acción es tomar o no tomar el medicamento. ¿Es este un sistema markoviano?
 - **Compra online:** estado es el producto que estoy mirando, y la acción es que otro producto te recomiendo como software. ¿Es este un sistema markoviano?

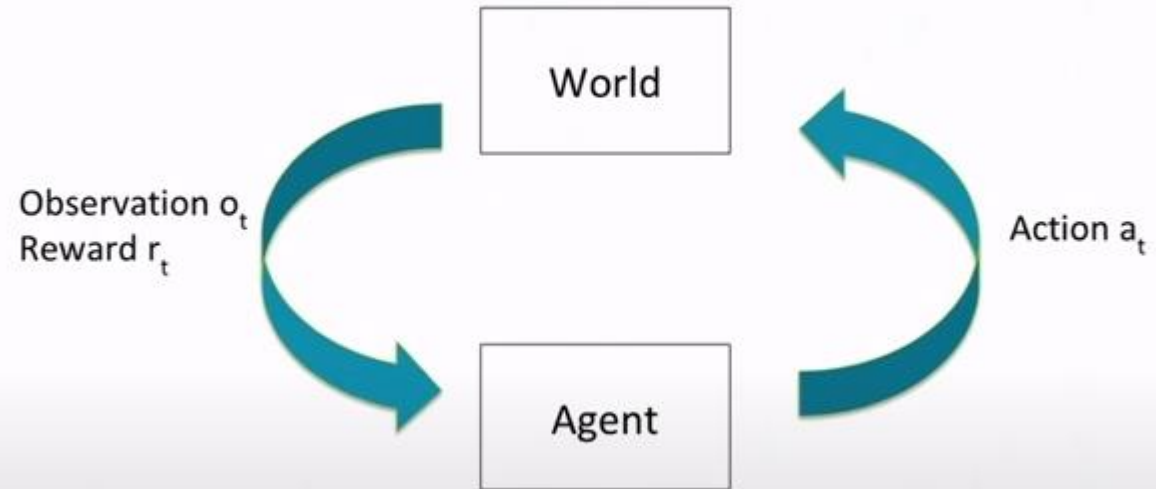
Supuesto de Markov

- Pero, todos puedes transformarlos en modelos markovianos
 - Seteando el estado por una historia
- En la práctica, se usa muchas veces las ultimas observaciones como estadística suficiente para crear una historia
- La Representación de Estados tiene muchas implicancias:
 - Complejidad Computacional
 - Data requerida
 - Rendimiento esperado

¿Qué es la Historia?

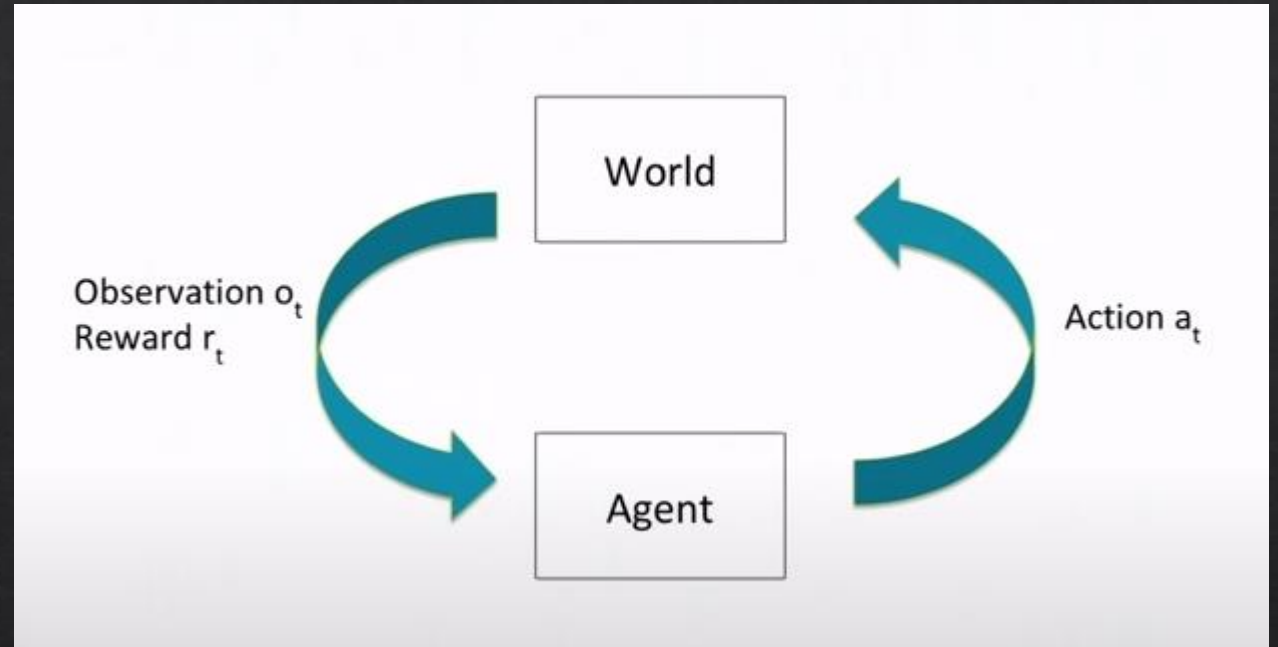
- Historia $ht = (a_1, s_1, r_1, \dots, a_t, s_t, r_t)$
- Agente escoge su acción basada en su historia
- Estado es information asumida para determinar que pasa despues
 - $S_t = (ht)$

Observación Full MDP



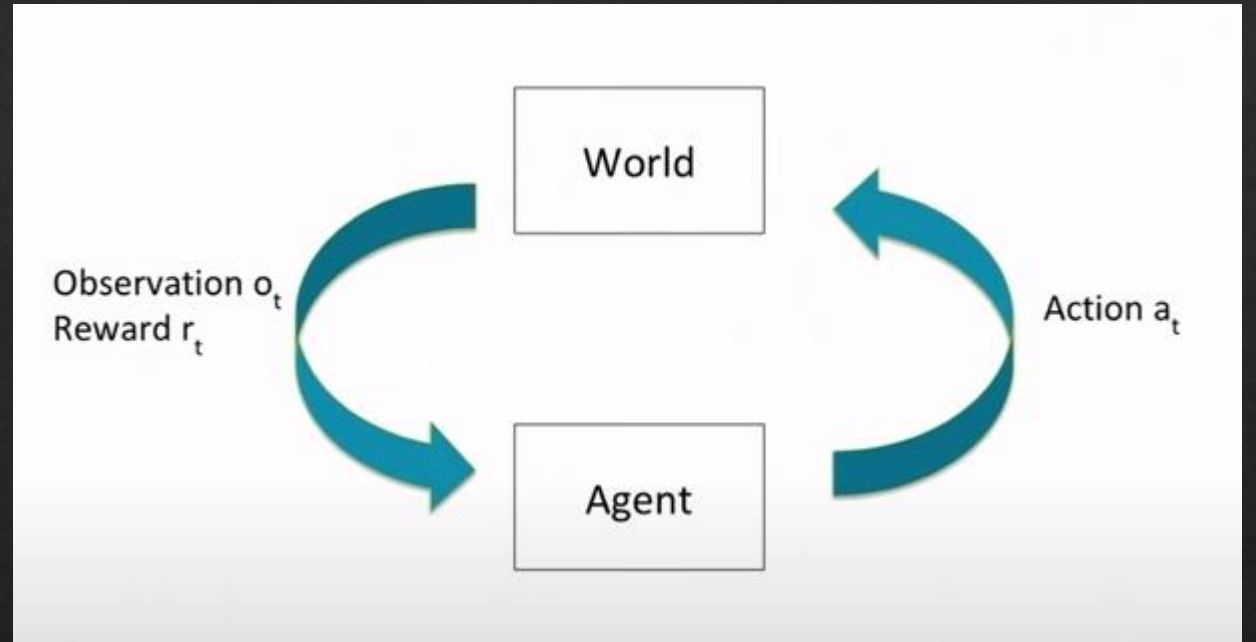
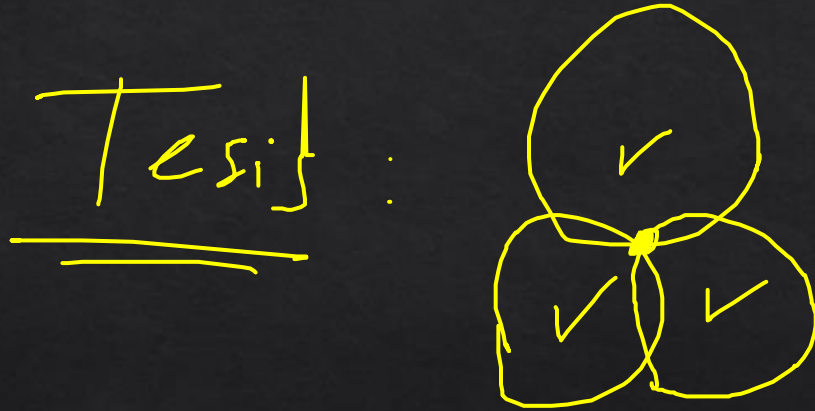
MDP Parcialmente Observable o POMDP

- El estado del agente **no es el mismo** que el estado del entorno
- El agente construye su propio estado
 - $S_t = h_t$
 - Sensores parciales
 - RNN



Proceso de Decisión Secuenciales: Bandits

- **Acciones** no tienen influencia en estados siguientes
- No hay **recompensas** atrasadas



Tipos de Entorno

Determinísticos:

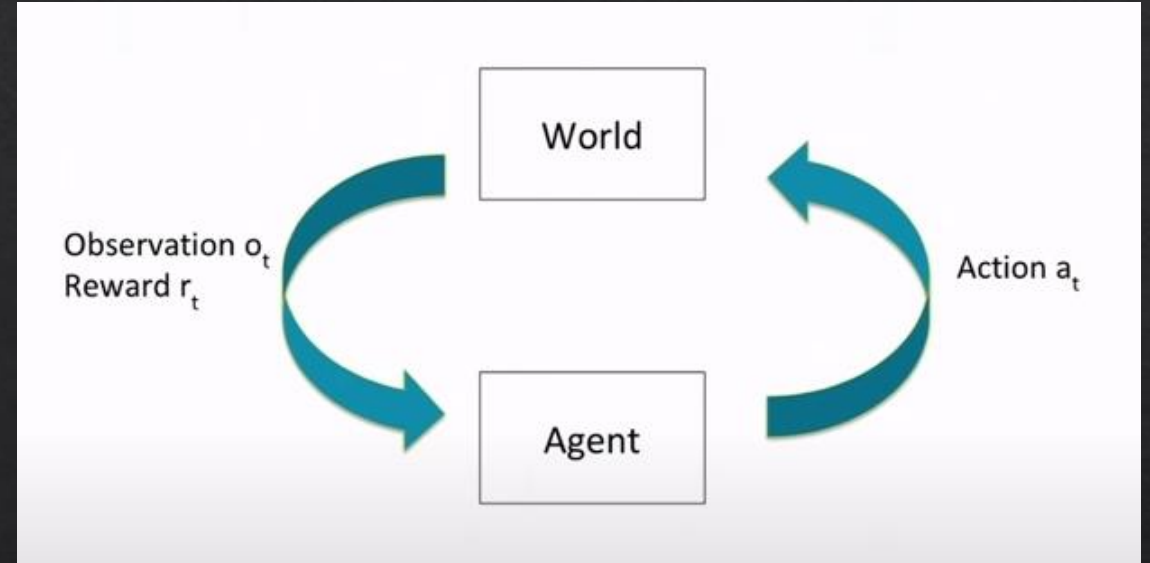
- Dado secuencia de estados (historia) y acciones, observación única y recompensas.
- Común supuesto en robótica y control automático

Estocástico:

- Dado secuencia de estados y acción, multiple potenciales observaciones y recompensas
- Común en supuestos para clientes, pacientes.

Paradigma

→ D vs E




Tipos de Entorno - Ejemplo Rover

S

- Estados: Ubicación del Rover (s_1, \dots, s_7)

- Acciones: TryLeft o TryRight A

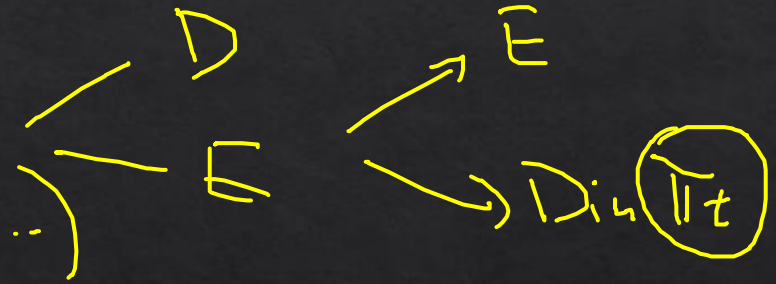
- Recompensas:
 - +1 en estado s_1
 - +10 en estado s_7
 - 0 en todo el resto de los estados

✓	✓	✓	✓	✓	✓	✓
s_1	s_2	s_3	s_4	s_5	s_6	s_7
+1	0	0	 0	0	0	+10



Componentes de un Algoritmo de RL

- Modelo $\xrightarrow{\text{Modelo}} \text{Dinámica} \mid \text{Transición}$ $\underline{P(s' \mid s, a)}$
 - Representación de como el mundo cambia en respuesta a la acción de un agente.
- Política $\pi : S \rightarrow A$
 - Función de mapeo del agente para pasar de estados a acciones
- Función de Valor $E(V_0 + \gamma^1 V_1 + \gamma^2 V_2 + \dots)$
 - Recompensas futuras por estar en un estado y/o acción siguiendo una política particular.



Componentes de un Algoritmo de RL

- Modelo:

- Representación de como el mundo cambia en respuesta a la acción de un agente.

➡ Dinámica o Transiciones del modelo predice el estado del agente en el siguiente estado

➡ Modelo de Recompensas predice recompensas inmediatas


$$p(s_{t+1} = s' | s_t = s, a_t = a)$$

$$r(s_t = s, a_t = a) = \mathbb{E}[r_t | s_t = s, a_t = a]$$

$$\begin{aligned} &- V(s) \\ &- V(s, a) \\ &- V(s, a, s') \end{aligned}$$

Componentes de un Algoritmo de RL

- Modelo:

- Modelo de Recompensas:

s_1	s_2	s_3	s_4	s_5	s_6	s_7
$\hat{r} = 0$	$\hat{r} = 0$	$\hat{r} = 0$	$\hat{r} = 0$	$\hat{r} = 0$	$\hat{r} = 0$	$\hat{r} = 0$

- Modelo de Transiciones:

$$\begin{aligned} 0.5 &= P(s_1|s_1, \text{TryRight}) = P(s_2|s_1, \text{TryRight}) \\ 0.5 &= P(s_2|s_2, \text{TryRight}) = P(s_3|s_2, \text{TryRight}) \dots \end{aligned}$$

- El modelo puede estar equivocado



Componentes de un Algoritmo de RL

- Política

- Función de mapeo del agente para pasar de estados a acciones
- Π determina como el agente escoge acciones

- $\pi: S \rightarrow A$

- Política Determinística:

$$\pi(s) = a$$

- Política Estocástica:

$$\pi(a|s) = Pr(a_t = a | s_t = s)$$

π (Controlador)

Componentes de un Algoritmo de RL

- Función de Valor $\gamma = [0, 1]$
 - $\gamma = 0$
 - $\pi(s_1) = \pi(s_2) = \dots = \pi(s_7) =$
TryRight
 - Números muestran el valor de $V^\pi(s)$ para esta política π y este factor de descuento γ

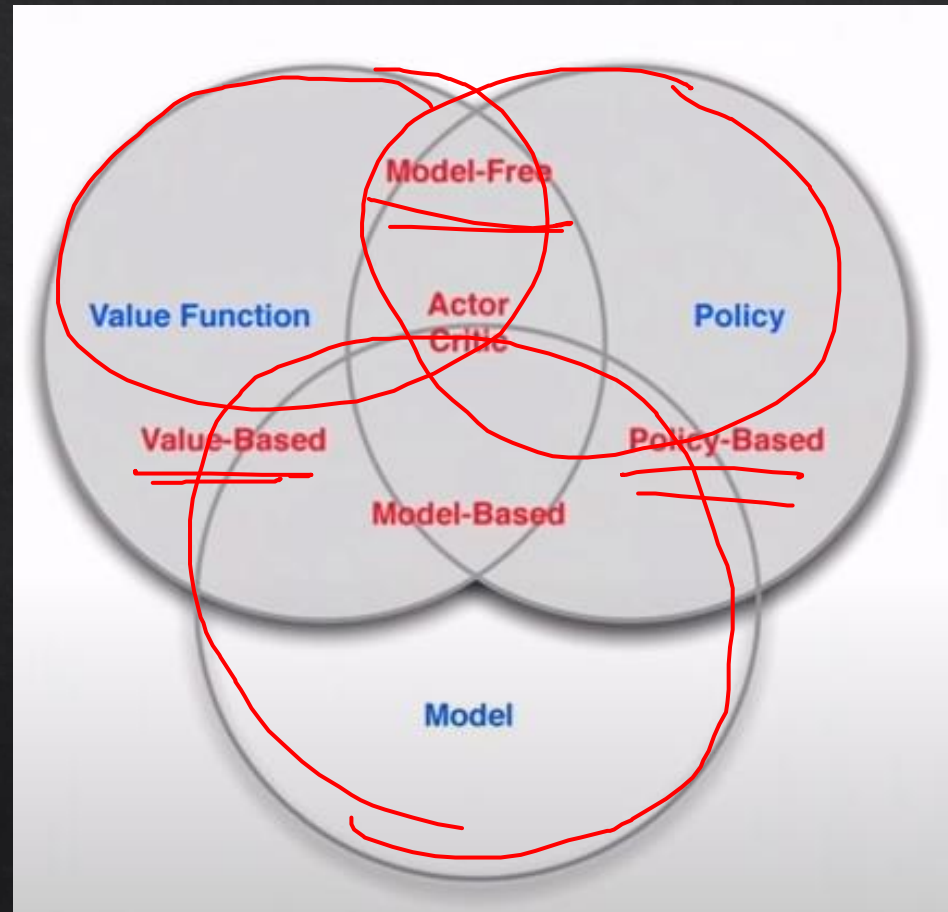
s_1	s_2	s_3	s_4	s_5	s_6	s_7
$V^\pi(s_1) = +1$	$V^\pi(s_2) = 0$	$V^\pi(s_3) = 0$	$V^\pi(s_4) = 0$	$V^\pi(s_5) = 0$	$V^\pi(s_6) = 0$	$V^\pi(s_7) = +10$

Tipos de Agentes RL

- **Basados en Modelo**
 - Modelo solo ✓
 - Puede o no puede tener una política o una function de valor ✓
- **Libre de Modelo**
 - Función de valor y/o Función de Política
 - No hay modelo

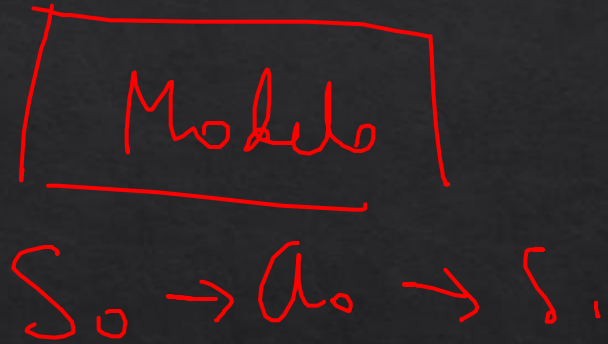
Model-free
B. Policy Direct!

Tipos de Agentes RL



Desafíos en aprender a hacer una buena secuencia de decisiones

- **Planificación**
 - Modelo dado de como el entorno funciona
 - Dinámica y Modelo de recompensas
 - Desenrollar
 - Algoritmo computa para maximizar recompensas esperadas
 - Esto sin interacción con el mundo real
- **Aprendizaje por Refuerzo**
 - Agente no sabe como funciona el mundo
 - Interactúa con el mundo para implícita o explícitamente aprender
 - El agente mejora su política (puede tener planificación)



Planning

- Ejemplo de Planificación (Planning)
 - Solitario
 - Saber todas las reglas del juego , modelo perfecto
 - Si tomas acción a desde el estado s
 - Puede computar una distribución de prob. Sobre el siguiente estado
 - Puede computar puntaje
 - Puede planear hacia delante para decidir la acción optimal
 - Programación dinámica, tree search

Ejemplos de Exploración vs Explotación

- Películas
 - Explotar una película ✓
 - Explorar una nueva
- Advertising
 - Mostrar la mas efectiva hasta le momento
 - Mostrar una diferente
- Manejo
 - Explotar la ruta mas rápida hasta el momento
 - Intentar una ruta diferente

greedy


experiencia ↑↑

Evaluación vs Control

- Evaluación
 - Estimar o predecir recompensas esperados siguiendo una política dada π
- Control
 - Es una optimización, encontrar la mejor política



Control de Política del Rover

s_1	s_2	s_3	s_4	s_5	s_6	s_7
						

- Factor de Descuento , $\gamma = 0$
- Cual es la política que optimiza la suma de recompensas esperadas con descuento

