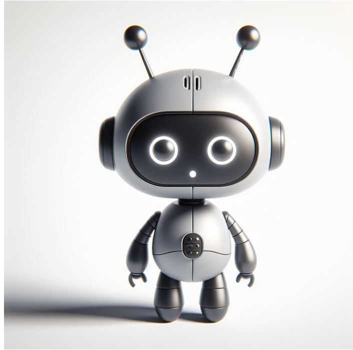
RL: Review

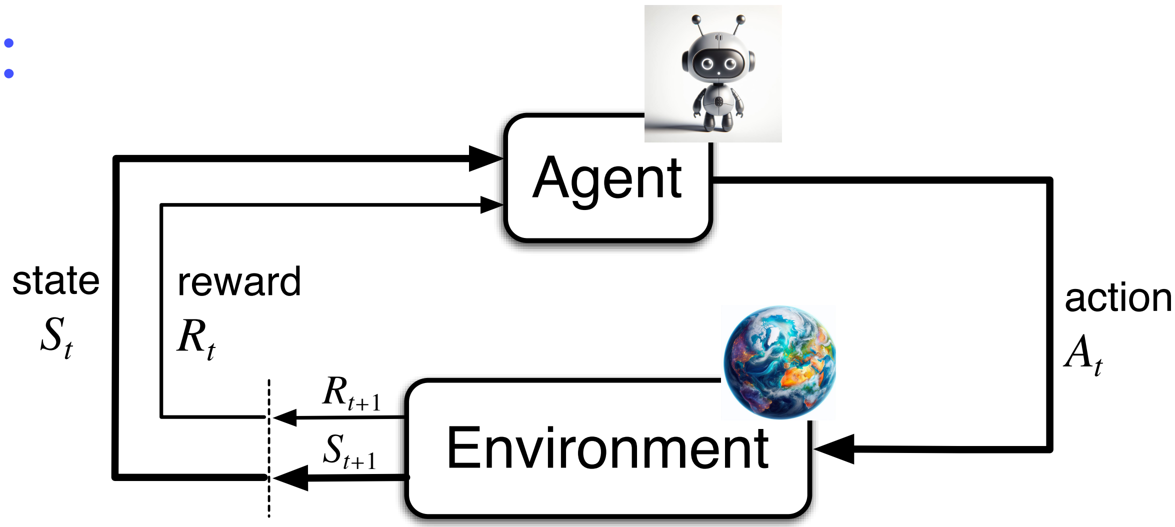
RL Setting:

Environment:

Agent:



RL Setting:



Agent and environment interact at discrete time steps: $t = 0, 1, 2, 3, \ldots$

Agent observes state at step t: $S_t \in \mathcal{S}$

produces action at step t : $A_t \in \mathcal{A}(S_t)$

gets resulting reward: $R_{t+1} \in \mathcal{R} \subset \mathbb{R}$

and resulting next state: $S_{t+1} \in \mathcal{S}^+$

Property of the Environment:

Property of the Environment:

Environment is Markov Decision Process (MDP)

$$p(S_{t+1}, R_{t+1} \,|\, A_t, S_t, A_{t-1}, S_{t-1}, \ldots, S_0)$$

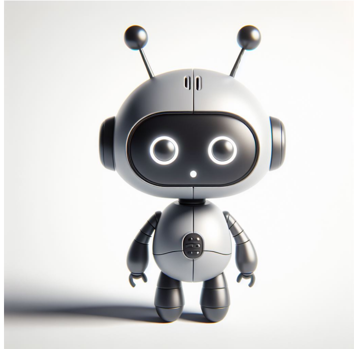
Property of the Environment:

Environment is Markov Decision Process (MDP)

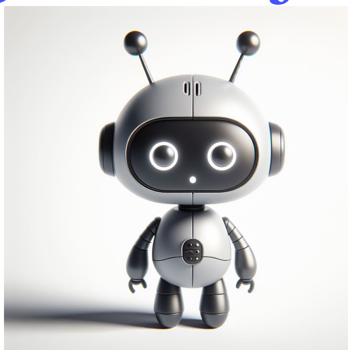
$$p(S_{t+1}, R_{t+1} \mid A_t, S_t, \cancel{A_{t-1}}, \cancel{S_{t-1}}, \cancel{\ldots}, \cancel{S_0})$$

$$= p(S_{t+1}, R_{t+1} \mid A_t, S_t)$$

Agent's objective:



Agent's objective:

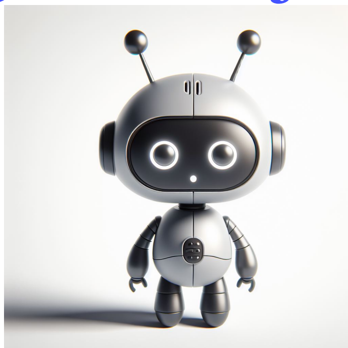


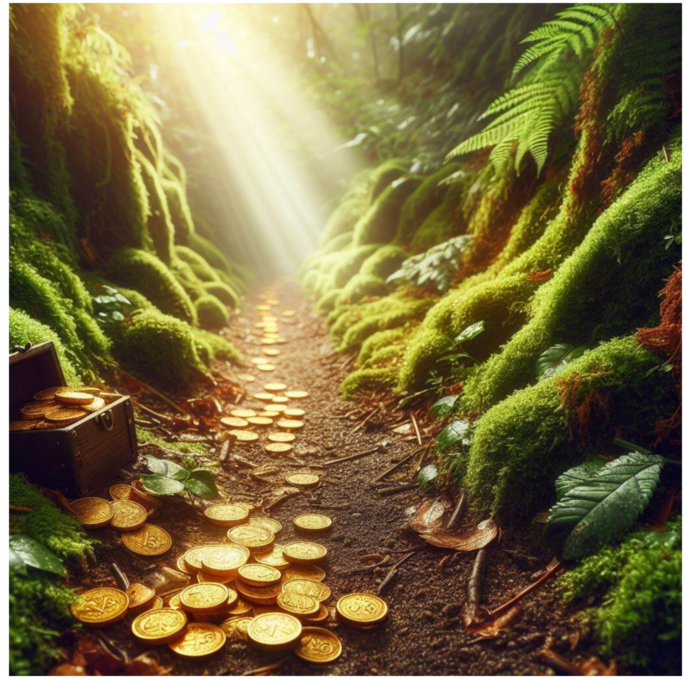
Maximize:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1},$$

where γ, $0 \le \gamma \le 1$, is the **discount rate**.

Agent's objective:



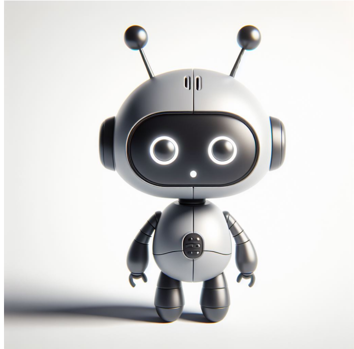


Maximize:

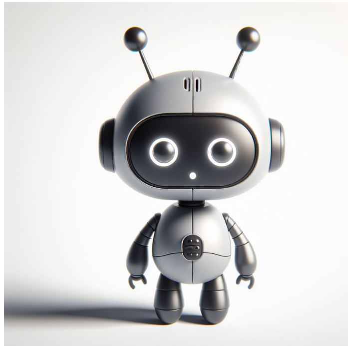
$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1},$$

where γ, $0 \le \gamma \le 1$, is the **discount rate**.

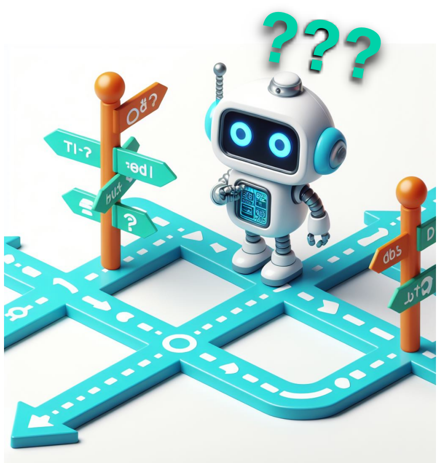
Agent does 2 things:



Agent does 2 things:

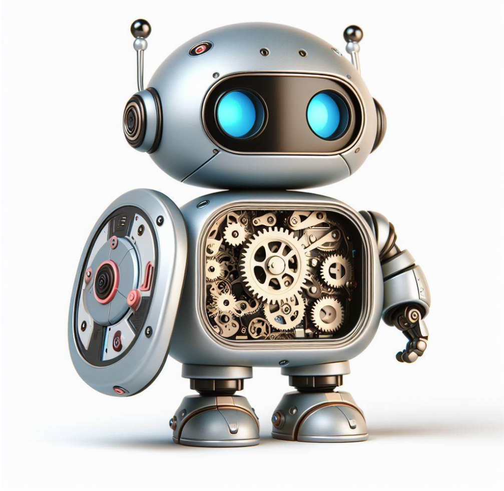


Choose actions:

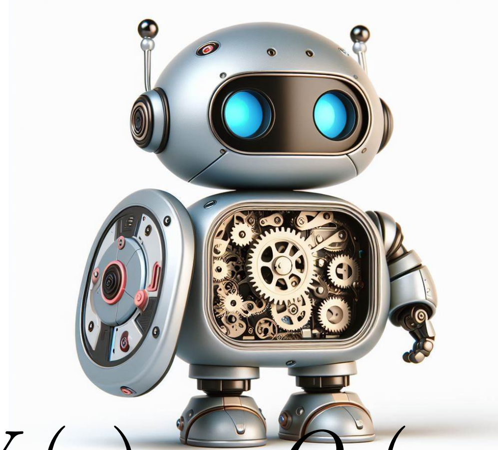


Learn how to choose better actions:

Different Parts of an Agent:



Different Parts of an Agent:



- Value Functions :

$$V_t(s) \qquad Q_t(s, a)$$

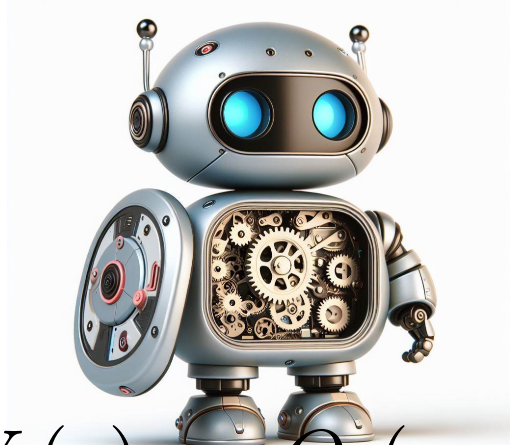
- World Model:

$$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$$

- Policy:

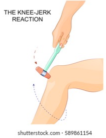
$$A_t = \pi(S_t, \theta)$$

Different Parts of an Agent:



- Value Functions :  $V_t(s)$ $Q_t(s, a)$

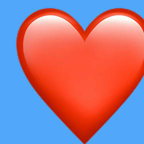
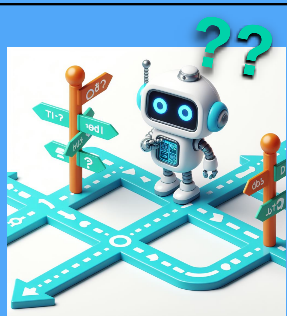
- World Model:  $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$

- Policy: 

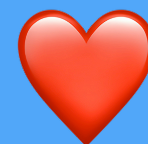
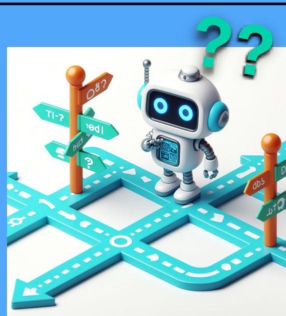
$A_t = \pi(S_t, \theta)$

- (Replay Buffer of past experience)

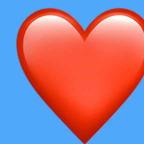
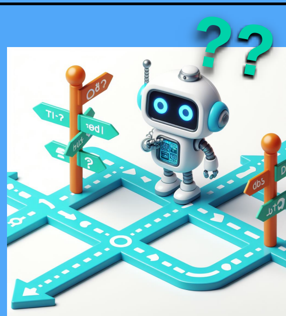
Grid of RL:

			
	Learn $\pi(S_t, \theta)$	Learn $V_t(s)$ $Q_t(s,a)$	Learn $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
	Use $\pi(S_t, \theta)$ to choose action.	Use $V_t(s)$ $Q_t(s,a)$ to choose action.	Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action.

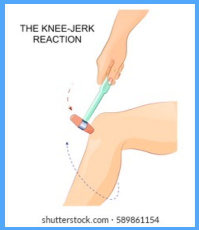
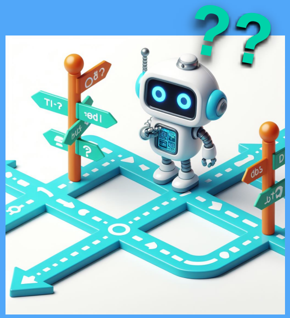
Grid of RL:

	 		
	Learn $\pi(S_t, \theta)$	Learn $V_t(s)$ $Q_t(s, a)$	Learn $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
	Use $\pi(S_t, \theta)$ to choose action.	Use $V_t(s)$ $Q_t(s, a)$ to choose action.	Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action.

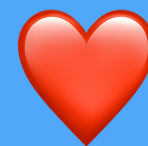
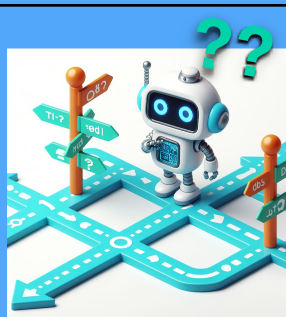
Grid of RL:

			
	Learn $\pi(S_t, \theta)$	Learn $V_t(s)$ $Q_t(s,a)$	Learn $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
	Use $\pi(S_t, \theta)$ to choose action.	Use $V_t(s)$ $Q_t(s,a)$ to choose action.	Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action.

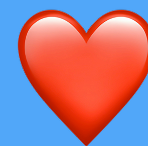
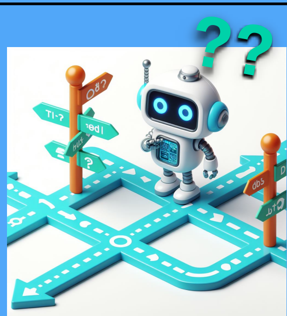
Grid of RL:

<table>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>Learn
$\pi(S_t, \theta)$</td><td>Learn
$V_t(s)$ $Q_t(s, a)$</td><td>Learn
$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$</td></tr>
<tr><td></td><td>Use $\pi(S_t, \theta)$ to choose action.</td><td>Use
$V_t(s)$ $Q_t(s, a)$
to choose action.</td><td>Use
$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
to choose action.</td></tr>
</table>

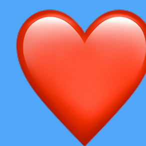
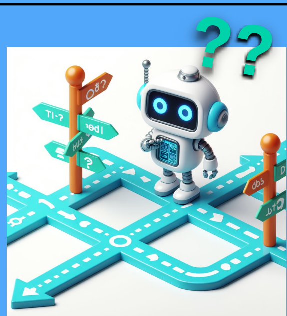
Grid of RL:

			
	Learn $\pi(S_t, \theta)$	Learn $V_t(s)$ $Q_t(s,a)$	Learn $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
	Use $\pi(S_t, \theta)$ to choose action.	Use $V_t(s)$ $Q_t(s,a)$ to choose action.	Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action.

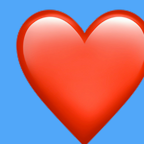
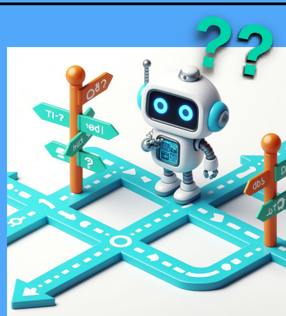
Grid of RL:

<table>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>Learn
$\pi(S_t, \theta)$</td><td>Learn
$V_t(s)$ $Q_t(s,a)$</td><td>Learn
$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$</td></tr>
<tr><td></td><td>Use $\pi(S_t, \theta)$
to choose
action.</td><td>Use
$V_t(s)$ $Q_t(s,a)$
to choose
action.</td><td>Use
$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
to choose
action.</td></tr>
</table>

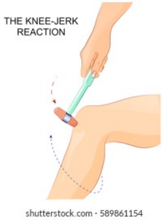
Grid of RL:

<table>
<tr>
<td></td>
<td>
</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Learn
$\pi(S_t, \theta)$</td>
<td>Learn
$V_t(s)$ $Q_t(s, a)$</td>
<td>Learn
$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$</td>
</tr>
<tr>
<td></td>
<td>Use $\pi(S_t, \theta)$
to choose
action.</td>
<td>Use
$V_t(s)$ $Q_t(s, a)$
to choose
action.</td>
<td>Use
$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
to choose
action.</td>
</tr>
</table>

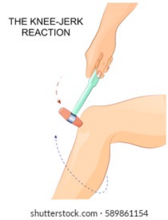
Grid of RL:

			
	Learn $\pi(S_t, \theta)$	Learn $V_t(s)$ $Q_t(s, a)$	Learn $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
	Use $\pi(S_t, \theta)$ to choose action.	Use $V_t(s)$ $Q_t(s, a)$ to choose action.	Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action.

TWO TYPES OF POLICY $\pi(S_t, \theta)$:



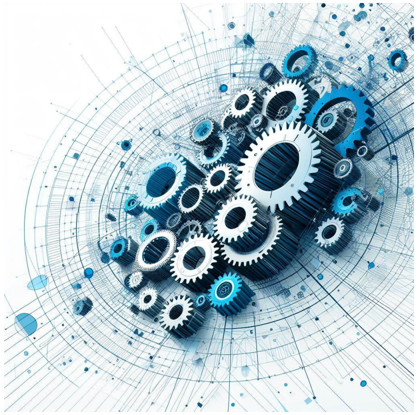
TWO TYPES OF POLICY $\pi(S_t, \theta)$:



Stochastic:

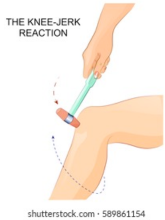
$\pi(A_t, S_t, \theta)$ is probability.

Deterministic:



$$A_t = \pi(S_t, \theta)$$

Stochastic: $\pi(A_t, S_t, \theta)$ is probability.



Act by sampling from the distribution:

Discrete Actions:

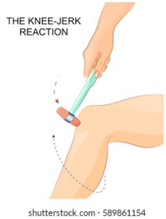
$$\pi(A_i \,|\, S) = \frac{\exp(\phi(A_i, S))}{\sum_j \exp(\phi(A_j, S))}$$

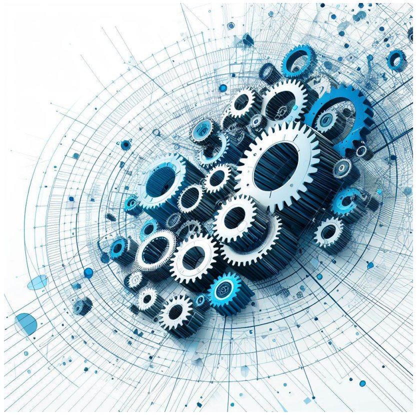
Continuous Actions:

$$\pi(A \,|\, S) = \mathcal{N}\left(\mu(S), \sigma(S)\right)$$

$$A = \mu(S) + \sigma(S)\epsilon, \qquad \epsilon \sim \mathcal{N}(0,1)$$

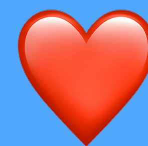
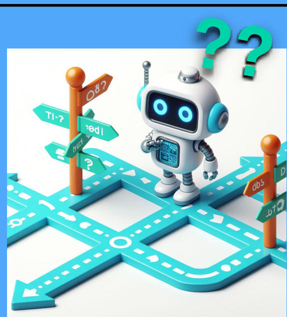
Deterministic: $A_t = \pi(S_t, \theta)$





Act by applying π to state: $A_t = \pi(S_t, \theta)$

Grid of RL:

			
	Learn $\pi(S_t, \theta)$	Learn $V_t(s)$ $Q_t(s, a)$	Learn $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
	Use $\pi(S_t, \theta)$ to choose action.	Use $V_t(s)$ $Q_t(s, a)$ to choose action.	Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action.

Value Functions

- The **value of a state** is the expected return starting from that state; depends on the agent's policy:

State - value function for policy π :

$$v_\pi(s) = E_\pi\left\{G_t \mid S_t = s\right\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s\right\}$$

- The **value of an action (in a state)** is the expected return starting after taking that action from that state; depends on the agent's policy:

Action - value function for policy π :

$$q_\pi(s,a) = E_\pi\left\{G_t \mid S_t = s, A_t = a\right\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s, A_t = a\right\}$$

Use $V_t(s)$ $Q_t(s,a)$ to choose action:

Action - value function for policy π :

$$q_\pi(s,a) = E_\pi\left\{G_t \mid S_t = s, A_t = a\right\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s, A_t = a\right\}$$

Use $V_t(s)$ $Q_t(s,a)$ to choose action:

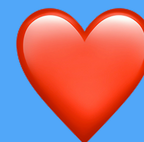
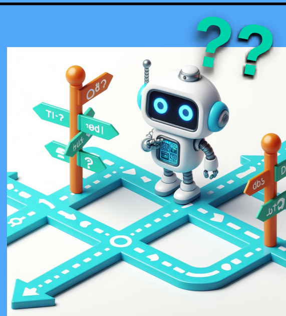
Action - value function for policy π :

$$q_\pi(s,a) = E_\pi\left\{G_t \mid S_t = s, A_t = a\right\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s, A_t = a\right\}$$

Act by taking the action which maximizes the expected return according to the estimate Q:

$$A_t = \text{argmax}_a Q(a, S_t)$$

Grid of RL:

			
	Learn $\pi(S_t, \theta)$	Learn $V_t(s)$ $Q_t(s, a)$	Learn $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
	Use $\pi(S_t, \theta)$ to choose action.	Use $V_t(s)$ $Q_t(s, a)$ to choose action.	Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action.

Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action:

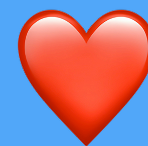
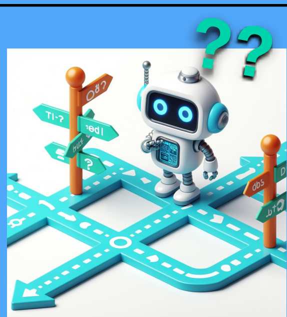
Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action:

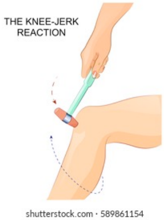
Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ in conjunction with planning algorithms (reactive planning):

Examples:

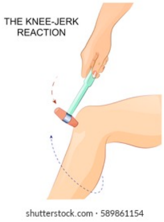
- Sampling future trajectories and taking the best one (as seen in PlaNet)
- Monte-Carlo Tree Search (as seen in MuZero)
- Cross-Entropy Method (with particles) (as seen in Dreamer Paper)

Grid of RL:

<table>
<tr>
<td></td>
<td>
</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Learn
$\pi(S_t, \theta)$</td>
<td>Learn
$V_t(s)$ $Q_t(s, a)$</td>
<td>Learn
$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$</td>
</tr>
<tr>
<td></td>
<td>Use $\pi(S_t, \theta)$
to choose action.</td>
<td>Use
$V_t(s)$ $Q_t(s, a)$
to choose action.</td>
<td>Use
$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
to choose action.</td>
</tr>
</table>



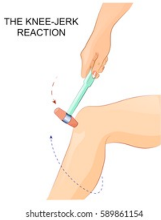
Learning $\pi(S_t, \theta)$:



Learning $\pi(S_t, \theta)$:

Action - value function for policy π :

$$q_\pi(s,a) = E_\pi\left\{G_t \mid S_t = s, A_t = a\right\} = E_\pi\left\{\sum_{k=0}^{\infty}\gamma^k R_{t+k+1} \mid S_t = s, A_t = a\right\}$$

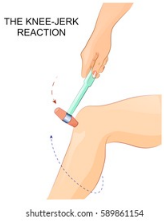


Learning $\pi(S_t, \theta)$:

Action - value function for policy π :

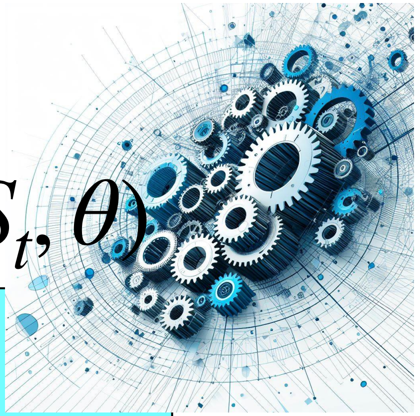
$$q_\pi(s,a) = E_\pi\left\{G_t \mid S_t = s, A_t = a\right\} = E_\pi\left\{\sum_{k=0}^{\infty}\gamma^k R_{t+k+1} \mid S_t = s, A_t = a\right\}$$

$$\theta_{t+1} = \theta_t + \alpha \underbrace{\nabla_\theta q_\pi}_{\nabla_\theta J}$$



Learning $\pi(S_t, \theta)$:

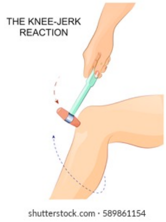
Deterministic and Continuous: $A_t = \pi(S_t, \theta)$



Action - value function for policy π :

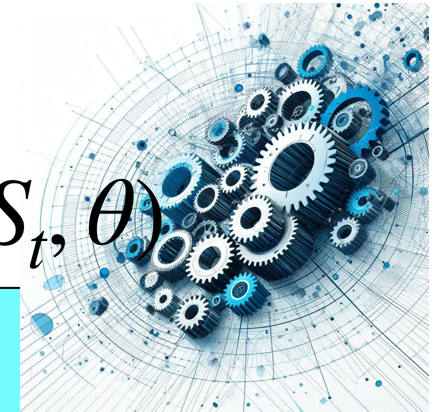
$$q_\pi(s,a) = E_\pi\left\{G_t \mid S_t = s, A_t = a\right\} = E_\pi\left\{\sum_{k=0}^{\infty}\gamma^k R_{t+k+1} \mid S_t = s, A_t = a\right\}$$

$$J_\theta(\pi \mid S_0 = S) = q_\pi(\pi(S), S) \approx Q_\pi(\pi(S,\theta), S)$$



Learning $\pi(S_t, \theta)$:

Deterministic and Continuous: $A_t = \pi(S_t, \theta)$



Action - value function for policy π :

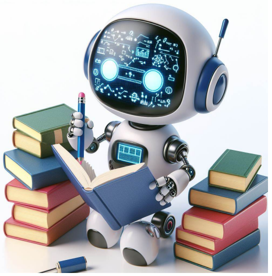
$$q_\pi(s,a) = E_\pi\left\{G_t \mid S_t = s, A_t = a\right\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s, A_t = a\right\}$$

$$J_\theta(\pi \mid S_0 = S) = q_\pi(\pi(S), S) \approx Q_\pi(\pi(S, \theta), S)$$

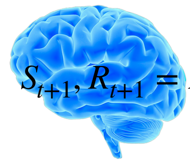
$$\nabla_\theta J_\theta(\pi \mid S_0 = S) \approx \nabla_\theta Q_\pi(\pi(S, \theta), S)$$

$$= \sum_i^m \frac{\partial Q_\pi(A = \pi(S, \theta), S)}{\partial a_i} \nabla_\theta \pi_i(S, \theta)$$

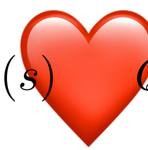
$$= \nabla_A Q_\pi\left(A = \pi(S, \theta), S\right) \nabla_\theta \pi(S, \theta)$$



RL Learning Map:



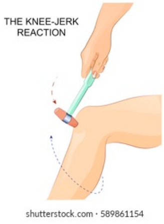
$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$



$V_t(s)$ $Q_t(s, a)$

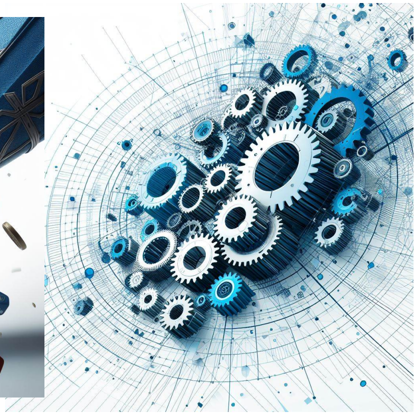
Deterministic Policy Gradient, DDPG

$\pi(S_t, \theta)$

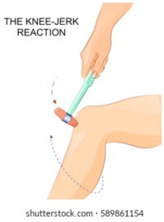


Learning $\pi(S_t, \theta)$:

Deterministic or Stochastic:

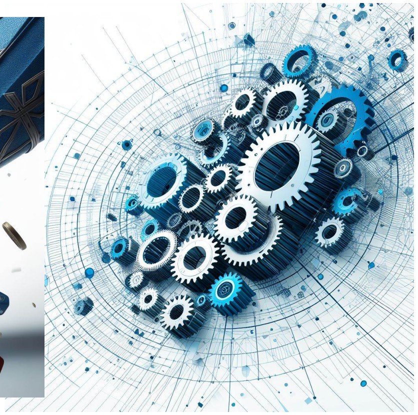


If we can plan with a World-Model M,
and planning gives us a next action A:

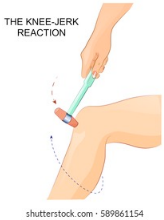


Learning $\pi(S_t, \theta)$:

Deterministic or Stochastic:

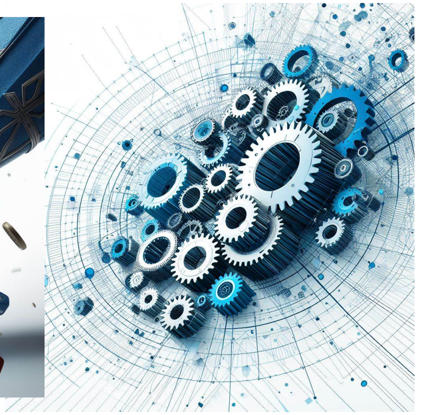


If we can plan with a World-Model M, and planning gives us a next action A: We can can use A as a target for supervised learning of π.



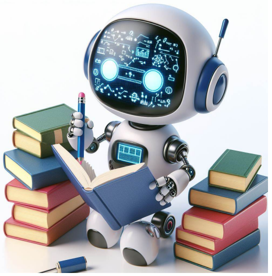
Learning $\pi(S_t, \theta)$:

Deterministic or Stochastic:

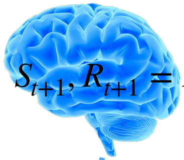


If we can plan with a World-Model M,
and planning gives us a next action A:
We can can use A as a target for
supervised learning of π.

Continuous A: regression problem: MSE loss
Discrete A, stochastic π: classification problem: Cross-Entropy loss



RL Learning Map:

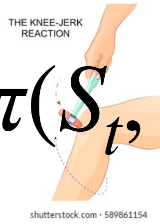


$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$

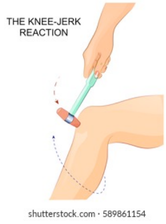
Supervised-Learning
of Plan

$V_t(s)$ $Q_t(s, a)$

Deterministic Policy Gradient,
DDPG



$\pi(S_t, \theta)$

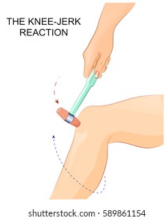


Learning $\pi(S_t, \theta)$:

Stochastic: $\pi(A_t, S_t, \theta)$ is probability.

Policy Gradient Theorem:

$$\nabla_{\boldsymbol{\theta}} J_{\boldsymbol{\theta}}(\pi) = \mathbb{E}_\pi[\sum_{t=0}^{T} \left(\gamma^t G_t\right) \nabla_{\boldsymbol{\theta}} \log \pi(A_t | S_t)]$$

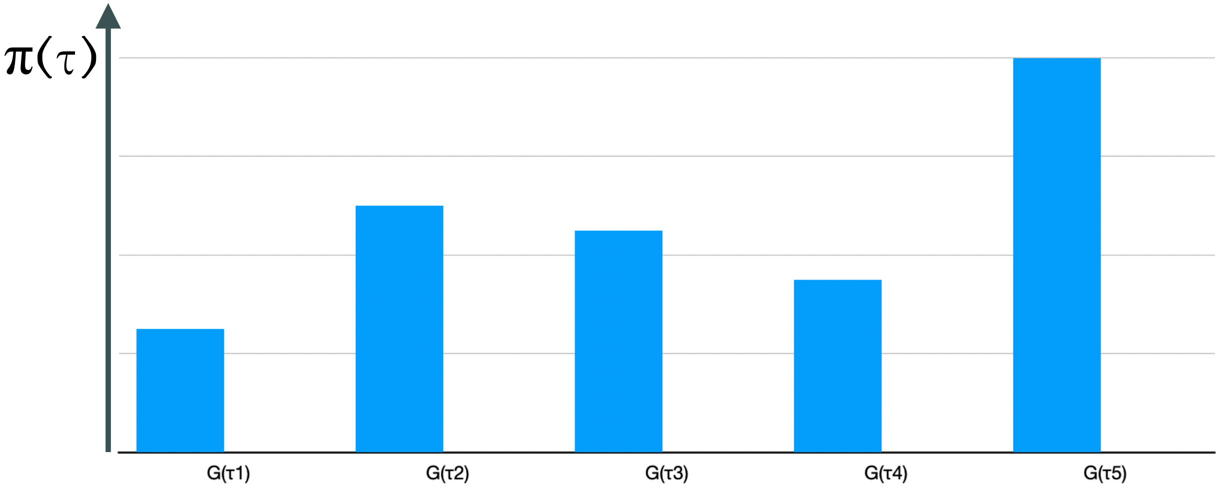


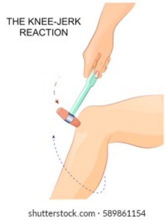
Learning $\pi(S_t, \theta)$:

Stochastic: $\pi(A_t, S_t, \theta)$ is probability.

Policy Gradient Theorem:

$$\nabla_{\boldsymbol{\theta}} J_{\boldsymbol{\theta}}(\pi) = \mathbb{E}_{\pi}[\sum_{t=0}^{T} (\gamma^t G_t)\, \nabla_{\boldsymbol{\theta}} \log \pi(A_t | S_t)]$$





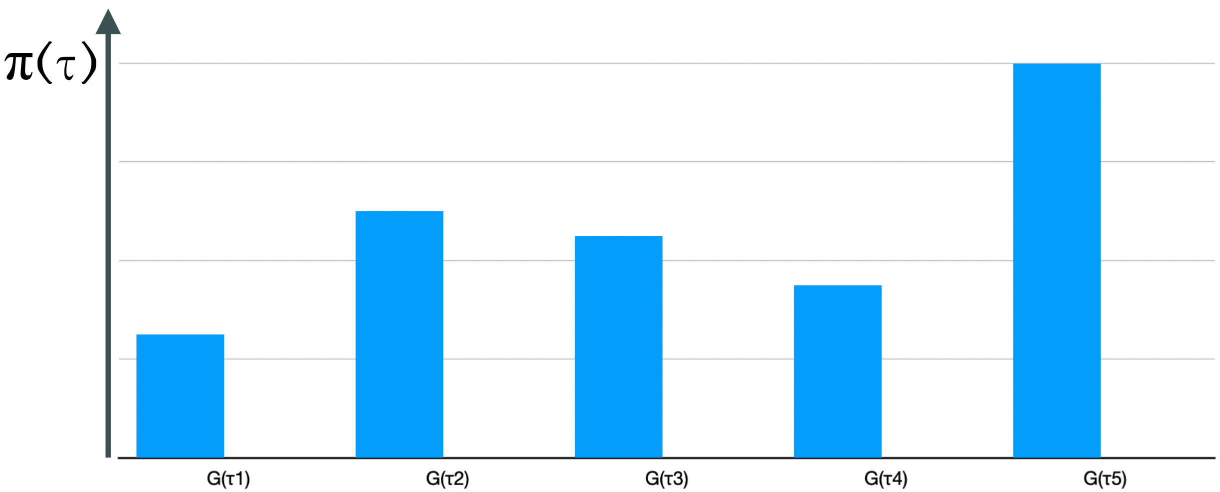
Learning $\pi(S_t, \theta)$:

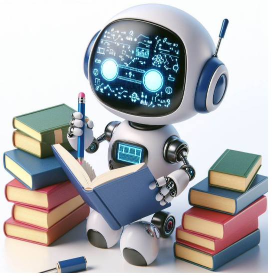
Stochastic: $\pi(A_t, S_t, \theta)$ is probability.

Policy Gradient Theorem:

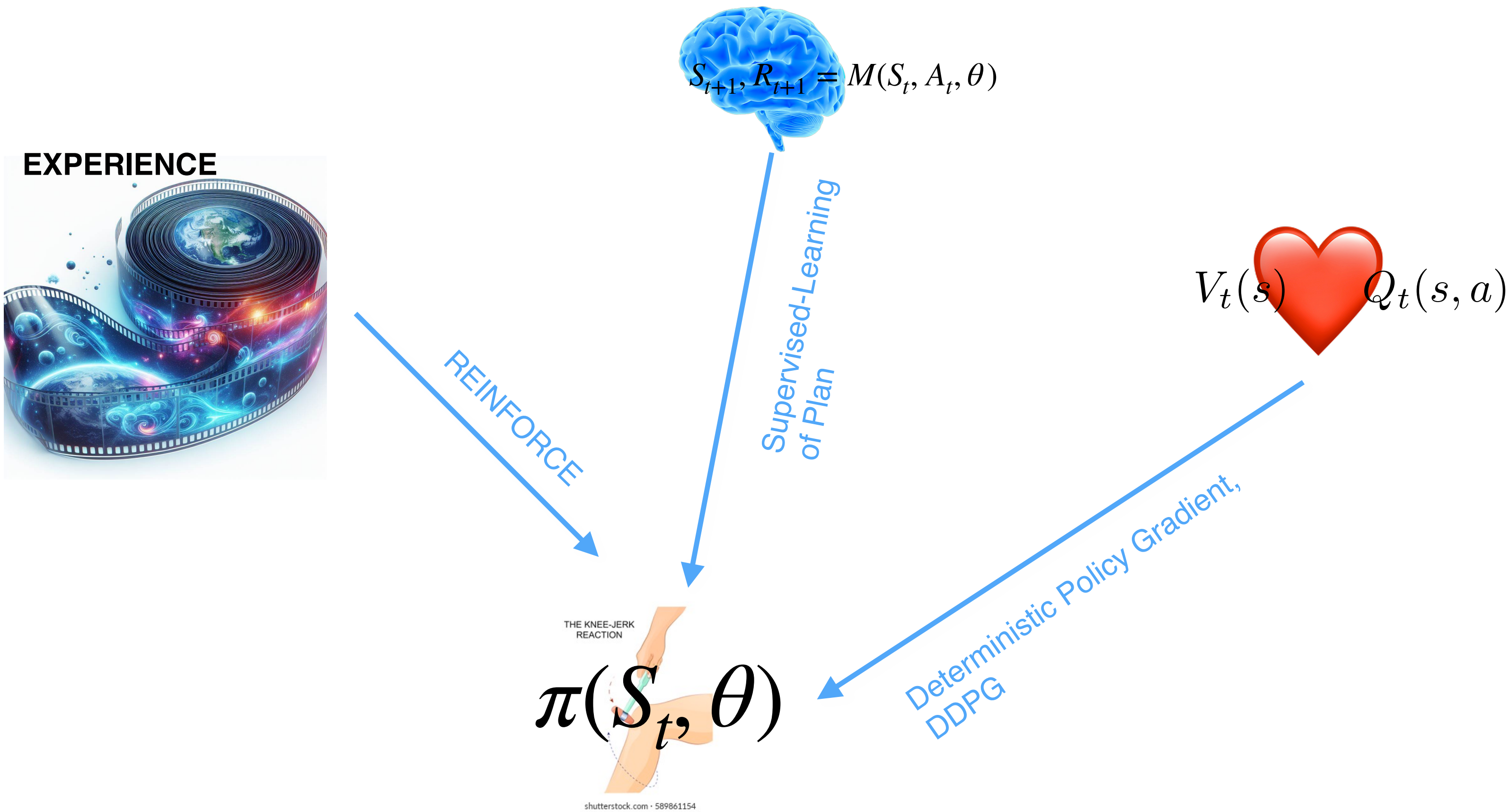
REINFORCE Estimates G with Monte-Carlo

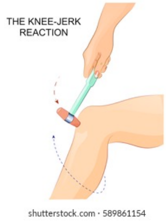
$$\nabla_{\theta} J_{\theta}(\pi) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{T} (\gamma^t G_t)\, \nabla_{\theta} \log \pi(A_t | S_t)\right]$$





RL Learning Map:



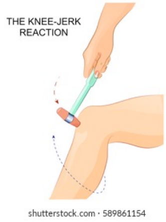


Learning $\pi(S_t, \theta)$:

Stochastic: $\pi(A_t, S_t, \theta)$ is probability.

Policy Gradient Theorem:

$$\nabla_{\boldsymbol{\theta}} J_{\boldsymbol{\theta}}(\pi) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{T} \left(\gamma^{t} G_{t}\right) \nabla_{\boldsymbol{\theta}} \log \pi(A_t | S_t)\right]$$



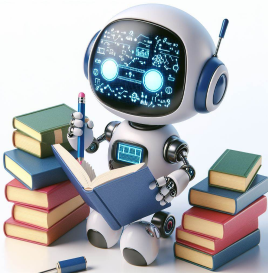
Learning $\pi(S_t, \theta)$:

Stochastic: $\pi(A_t, S_t, \theta)$ is probability.

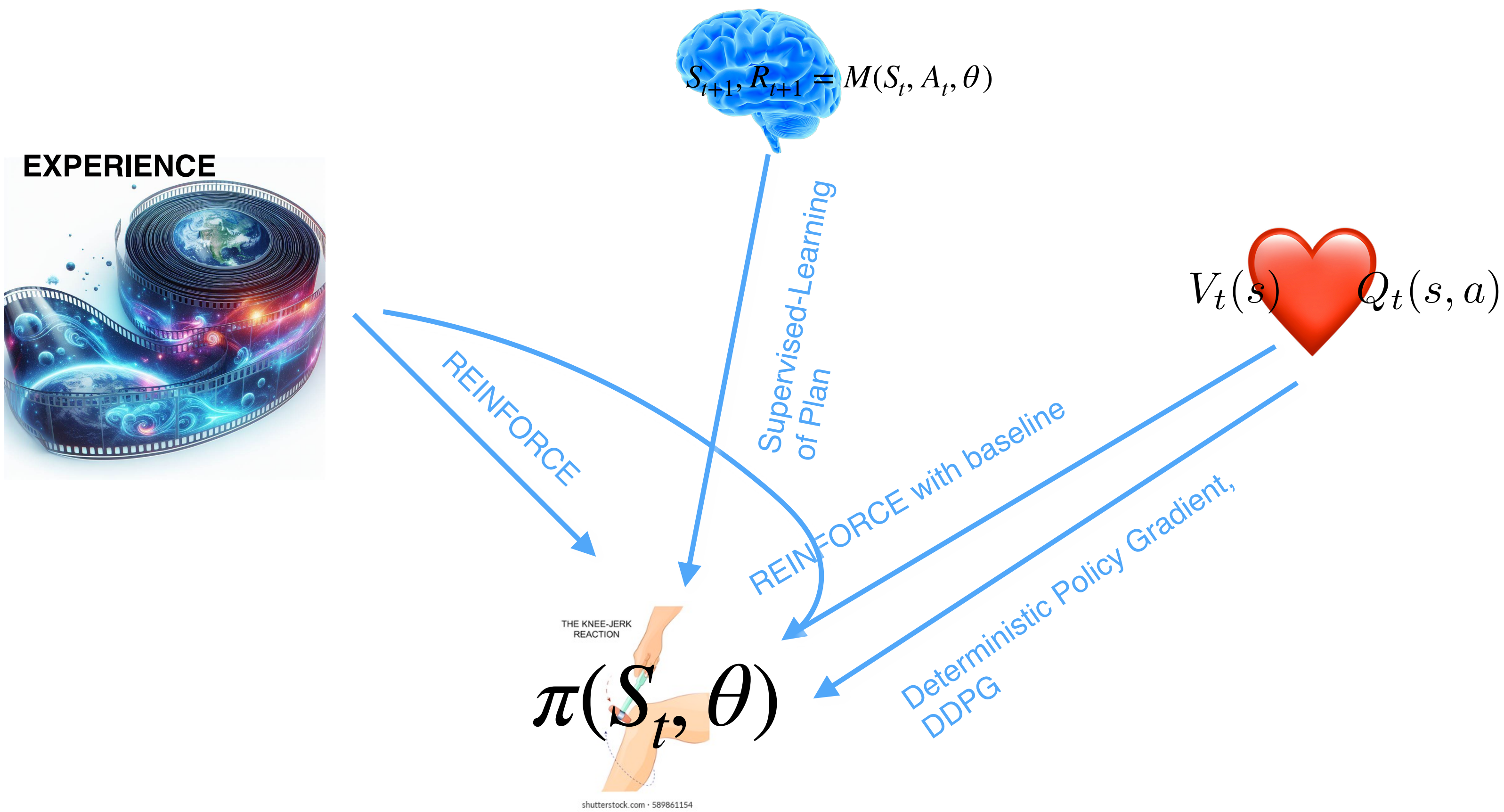
Policy Gradient Theorem:

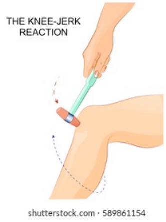
REINFORCE Estimates G with Monte-Carlo

$$\nabla_\theta J_\theta(\pi) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t \underbrace{\left(q_\pi(S_t, A_t) - v_\pi(S_t)\right)}_{\text{Advantage}} \nabla_\theta \log(\pi)\right]$$



RL Learning Map:





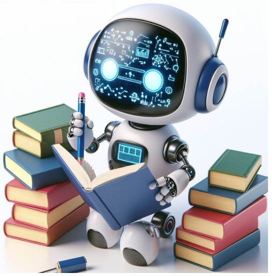
Learning $\pi(S_t, \theta)$:

Stochastic: $\pi(A_t, S_t, \theta)$ is probability.

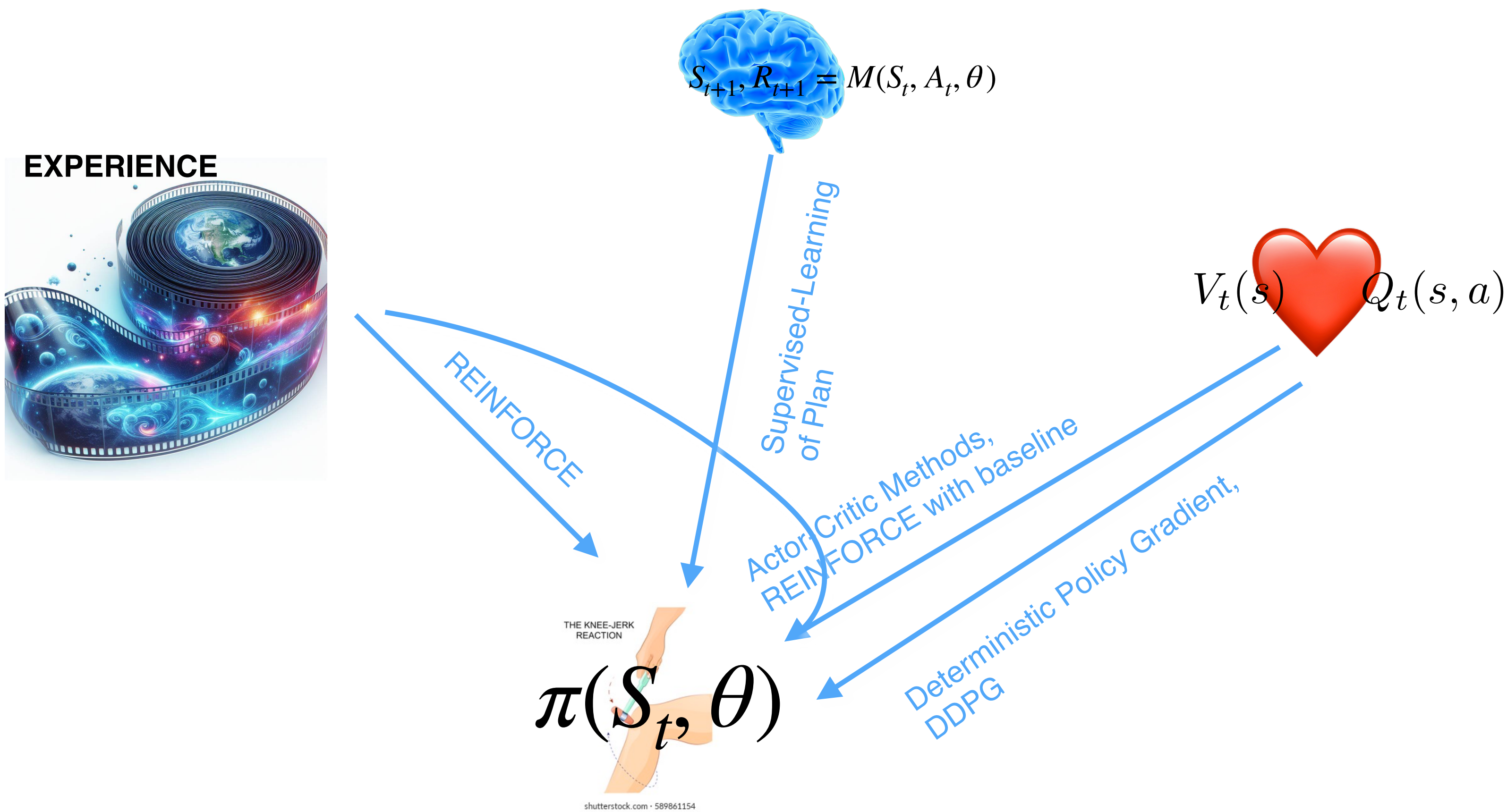
Policy Gradient Theorem:

$$\nabla_\theta J_\theta(\pi) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t \left(\underbrace{q_\pi(S_t, A_t) - v_\pi(S_t)}_{\textbf{Advantage}}\right) \nabla_\theta \log(\pi)\right]$$

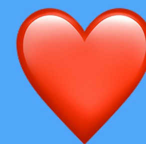
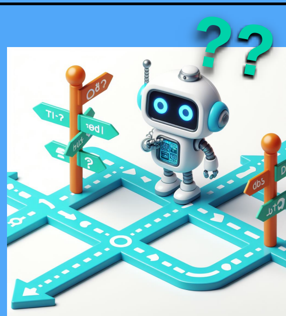
Actor-Critic: use V and/or Q to estimate G or Advantage, e.g. TD(λ)



RL Learning Map:



Grid of RL:

			
	Learn $\pi(S_t, \theta)$	Learn $V_t(s)$ $Q_t(s, a)$	Learn $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
	Use $\pi(S_t, \theta)$ to choose action.	Use $V_t(s)$ $Q_t(s, a)$ to choose action.	Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action.

Learn $V_t(s)$ $Q_t(s,a)$:

Action - value function for policy π :

$$q_\pi(s,a) = E_\pi\left\{G_t \mid S_t = s, A_t = a\right\} = E_\pi\left\{\sum_{k=0}^{\infty}\gamma^k R_{t+k+1} \mid S_t = s, A_t = a\right\}$$

State - value function for policy π :

$$v_\pi(s) = E_\pi\left\{G_t \mid S_t = s\right\} = E_\pi\left\{\sum_{k=0}^{\infty}\gamma^k R_{t+k+1} \mid S_t = s\right\}$$

❤️ Learn $V_t(s)$ $Q_t(s,a)$:

4 value functions

	state values	action values
prediction	v_π	q_π
control	v_*	q_*

- All theoretical objects, expected values
- Distinct from their estimates: $V_t(s)$ $Q_t(s,a)$

Learn $V_t(s)$ $Q_t(s,a)$**:**

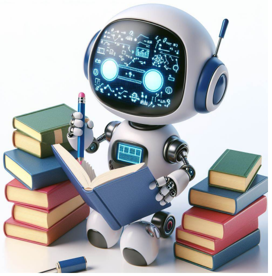
$$q_\pi(s,a) = \mathbb{E}\{G_t \mid S_t = s, A_t = a, A_{t+1:\infty} \sim \pi\} \qquad q_\pi : \mathcal{S} \times \mathcal{A} \to \Re$$

Monte-Carlo Estimate :

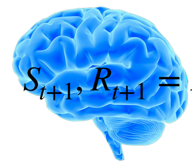
$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

where γ, $0 \le \gamma \le 1$, is the **discount rate**.

- *Every-Visit MC:* average returns for *every* time *s* is visited in an episode
- *First-visit MC:* average returns only for *first* time *s* is visited in an episode
- Both converge asymptotically



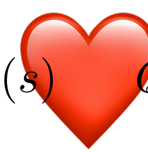
RL Learning Map:

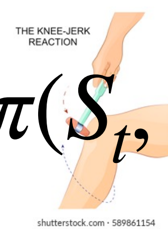


$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$

EXPERIENCE

Monte-Carlo Policy Evaluation

$V_t(s)$  $Q_t(s, a)$



$\pi(S_t, \theta)$

Learn $V_t(s)$ $Q_t(s,a)$:

$$q_\pi(s,a) = \mathbb{E}\{G_t \mid S_t = s, A_t = a, A_{t+1:\infty} \sim \pi\} \qquad q_\pi : \mathcal{S} \times \mathcal{A} \to \Re$$

Bootstrapping :

- TD: $G_t^{(1)} \doteq R_{t+1} + \gamma V_t(S_{t+1})$
  - Use V_t to estimate remaining return

- n-step TD:
  - 2 step return: $G_t^{(2)} \doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 V_t(S_{t+2})$
  - n-step return: $G_t^{(n)} \doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 + \cdots + \gamma^{n-1} R_{t+n} + \gamma^n V_t(S_{t+n})$

    with $G_t^{(n)} \doteq G_t$ if $t + n \geq T$

Learn $V_t(s)$ $Q_t(s,a)$:

$$q_\pi(s,a) = \mathbb{E}\{G_t \mid S_t = s, A_t = a, A_{t+1:\infty} \sim \pi\} \qquad q_\pi : \mathcal{S} \times \mathcal{A} \to \Re$$

(Expected) SARSA (Bellman Eqn) :

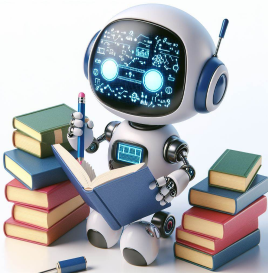
$$\begin{aligned} Q(S_t, A_t) &\leftarrow Q(S_t, A_t) + \alpha \Big[R_{t+1} + \gamma \mathbb{E}[Q(S_{t+1}, A_{t+1}) \mid S_{t+1}] - Q(S_t, A_t) \Big] \\ &\leftarrow Q(S_t, A_t) + \alpha \Big[R_{t+1} + \gamma \sum \pi(a|S_{t+1}) Q(S_{t+1}, a) - Q(S_t, A_t) \Big] \end{aligned}$$

Learn $V_t(s)$ $Q_t(s,a)$:

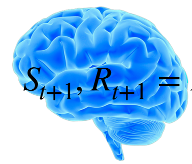
$$q_*(s,a) = \max_\pi q_\pi(s,a) \qquad q_* : \mathcal{S} \times \mathcal{A} \rightarrow \Re$$

Q-Learning (Bellman Optimality Eqn):

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



RL Learning Map:

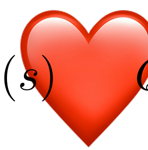


$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$

Bootstrap methods:
Q-Learning, SARSA, TD(λ), n-step TD

EXPERIENCE

Monte-Carlo Policy Evaluation

$V_t(s)$  $Q_t(s, a)$

$\pi(S_t, \theta)$

❤️ Learn $V_t(s)$ $Q_t(s,a)$ through pro-active planning:

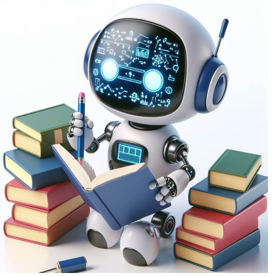
USE IMAGINED EXPERIENCE USING MODEL M:

(Expected) SARSA (Bellman Eqn) :

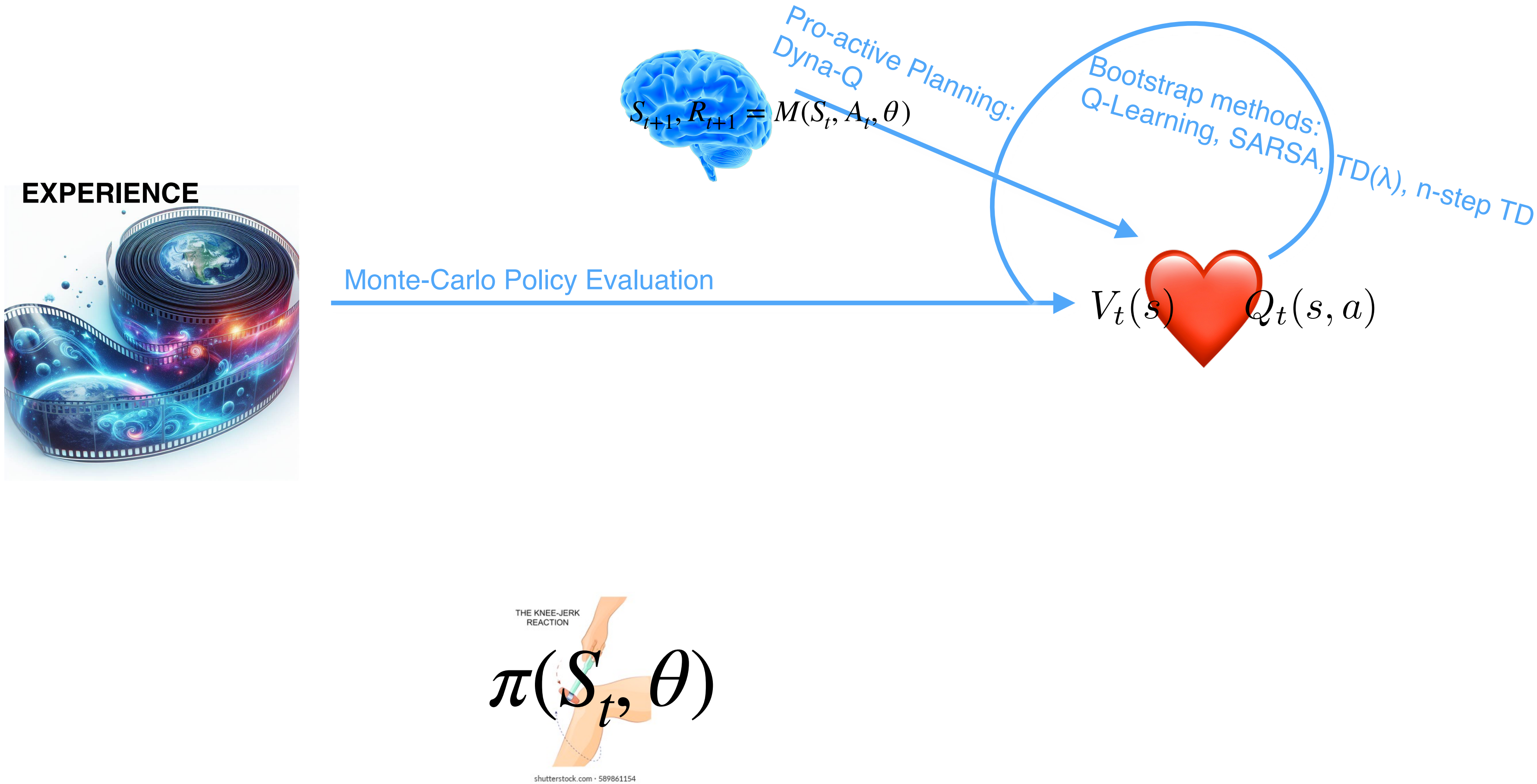
$$\begin{aligned} Q(S_t, A_t) &\leftarrow Q(S_t, A_t) + \alpha \Big[R_{t+1} + \gamma \mathbb{E}[Q(S_{t+1}, A_{t+1}) \mid S_{t+1}] - Q(S_t, A_t) \Big] \\ &\leftarrow Q(S_t, A_t) + \alpha \Big[R_{t+1} + \gamma \sum \pi(a|S_{t+1}) Q(S_{t+1}, a) - Q(S_t, A_t) \Big] \end{aligned}$$

Q-Learning (Bellman Optimality Eqn):

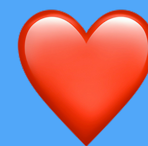
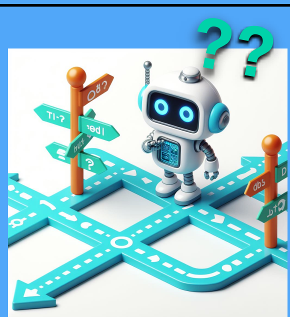
$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \Big[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t) \Big]$$



RL Learning Map:



Grid of RL:

			
	Learn $\pi(S_t, \theta)$	Learn $V_t(s)$ $Q_t(s,a)$	Learn $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$
	Use $\pi(S_t, \theta)$ to choose action.	Use $V_t(s)$ $Q_t(s,a)$ to choose action.	Use $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$ to choose action.

Learn $S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$:

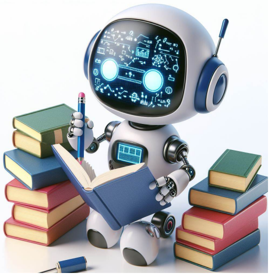
Use transition $S_t, A_t, R_{t+1}, S_{t+1}$:

Supervised learning

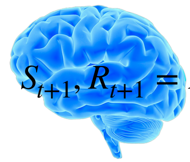
- Target for $\hat{S}_{t+1}, \hat{R}_{t+1} = M(S_t, A_t, \theta)$ is S_{t+1}, R_{t+1}
- Target for inverse model

$$\hat{S}_t, \hat{R}_{t+1} = M_{inv}(S_{t+1}, A_t, \psi)$$

is S_t, R_{t+1}



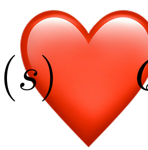
RL Learning Map:



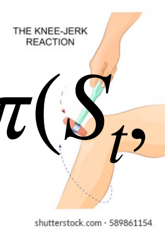
$S_{t+1}, R_{t+1} = M(S_t, A_t, \theta)$

Supervised Model Learning

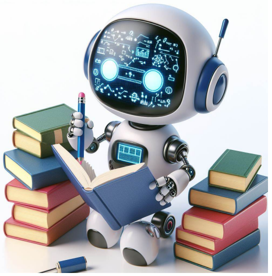
EXPERIENCE



$V_t(s)$ $Q_t(s, a)$



$\pi(S_t, \theta)$



RL Learning Map: