#### **1. Reinforcement Learning (RL) – Tổng quan**

* RL là một nhánh của Machine Learning, nơi agent học cách tương tác với environment để tối đa hóa phần thưởng.
* Cốt lõi: Hành động → Phản hồi (Reward + State mới) → Học cách hành động tốt hơn.

#### **2. Markov Decision Processes (MDPs)**

* MDP là mô hình toán học cho bài toán RL.
* Thành phần: S (States), A (Actions), P(s'|s,a), R(s,a), γ (discount factor).
* Markov Property: Chuyển trạng thái chỉ phụ thuộc trạng thái hiện tại, không phụ thuộc quá khứ.

#### **3. Value Function & Q-Function**

* Value Function Vπ(s): Kỳ vọng tổng reward khi bắt đầu từ trạng thái s và theo policy π.
* Q-Function Qπ(s, a): Kỳ vọng tổng reward khi làm hành động a tại s rồi theo policy π.
* Mục tiêu: Tìm policy π\* sao cho Vπ\*(s) ≥ Vπ(s) với mọi π.

#### **4. Policy**

* Policy π: Chiến lược chọn hành động của agent.
* π(a|s): Xác suất chọn hành động a tại trạng thái s.
* Có thể là deterministic hoặc stochastic.

#### **5. Bellman Equations**

* Bellman Expectation Equation:  
   Vπ(s) = Σ\_a π(a|s) [ R(s,a) + γ Σ\_s' P(s'|s,a)Vπ(s') ]
* Bellman Optimality Equation:  
   V\*(s) = max\_a [ R(s,a) + γ Σ\_s' P(s'|s,a)V\*(s') ]

#### **6. Q-Learning**

* Thuật toán model-free, off-policy.
* Cập nhật Q-value:  
   Q(s,a) ← Q(s,a) + α [ r + γ \* max\_a' Q(s', a') - Q(s,a) ]
* Sau khi hội tụ: π(s) = argmax\_a Q(s,a)

#### **7. Exploration vs Exploitation**

* Exploration: Thử hành động mới để khám phá.
* Exploitation: Chọn hành động tốt nhất đã biết.
* Chiến lược ε-greedy:  
  + Xác suất ε: chọn hành động ngẫu nhiên (explore)
  + Xác suất 1-ε: chọn hành động tốt nhất (exploit)

#### **8. Một số thuật toán RL khác (nâng cao)**

* SARSA (on-policy), Deep Q-Learning (DQN),
* Policy Gradient, Actor-Critic,…