Introduction to Reinforcement Learning

- What is reinforcement learning?
- Why is reinforcement learning exciting?
- Reward functions

What is Reinforcement Learning?

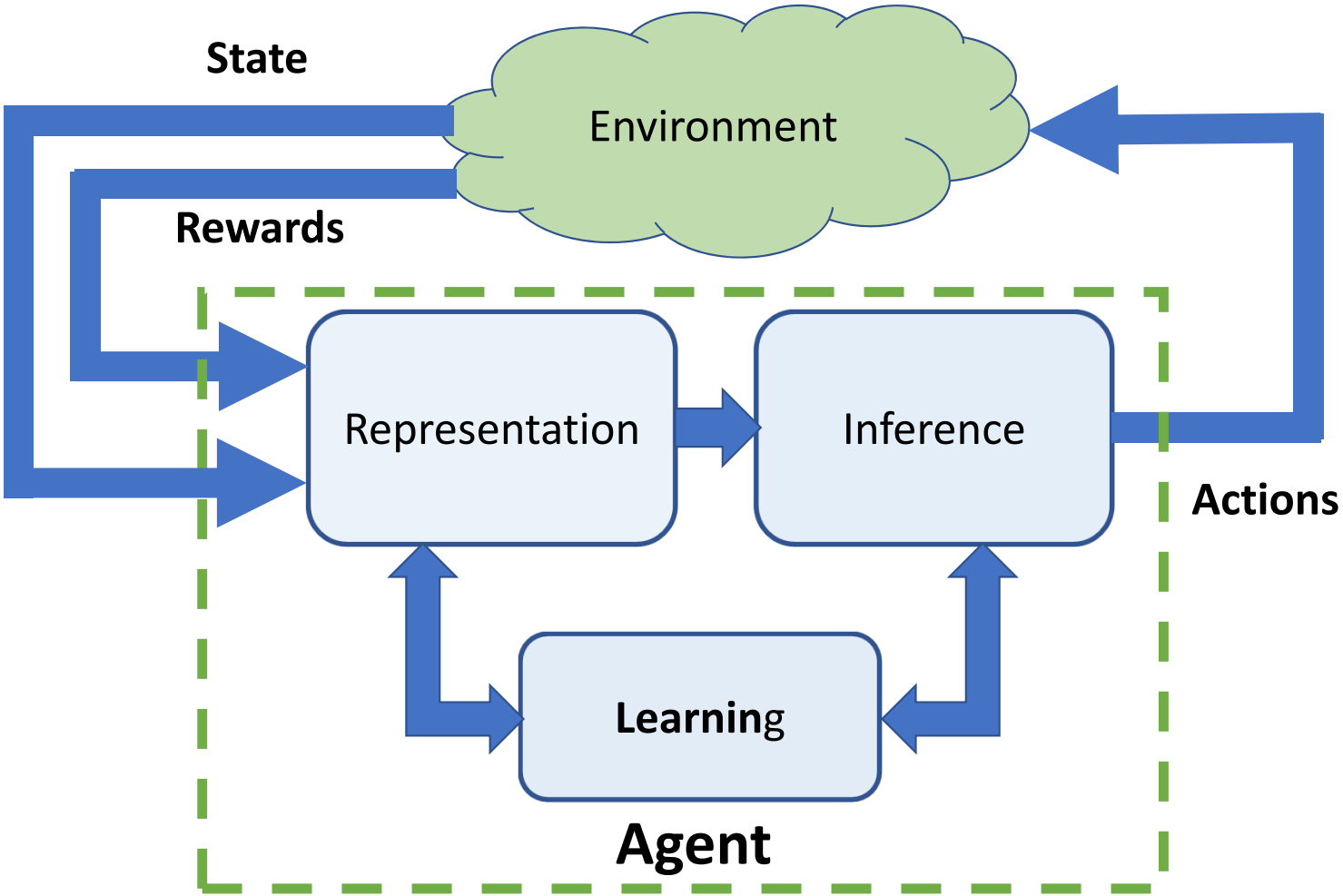
Model Type	Labeled Cases	Purpose	Metric
Supervised Machine Learning	Yes	Make Predictions	Error
Unsupervised Machine Learning	No	Find Structure	Error
Reinforcement Learning	No	Learn policy	Cumulative reward

What is Reinforcement Learning?

Key differences with other ML methods:

- Reinforcement learning agent creates a **policy** which **maximizes cumulative reward** or **utility**
- RL agent has no supervisor, only **reward signal**
- Cumulative reward feedback is **delayed**
- Agent **learns policy** for a given **task**
- Policy determines **actions**, given state
- Time matters; **sequential, non iid data**

The Reinforcement Learning Agent



What is Reinforcement Learning?

- Reinforcement learning **agent operates sequentially** over time steps:
  - From **state**, s_t
  - Executes **action**, a_t
  - Receives **observations**, o_t, and **updates state**, s_{t+1}
  - Receives scalar **reward**, r_t
- In response, the **environment**:
  - Receives and executes **action**, a_t
  - Emits **observations**, o_t
  - Emits reward, r_t

What is Reinforcement Learning?

- Agent **learns from experienceState** is the history of the actions, rewards, observations

  $S_t = (a_{t-n}, r_{t-n}, o_{t-n}, \ldots, a_{t-1}, r_{t-1}, o_{t-1}, a_t, r_t, o_t,)$

- Agent's **actions affect subsequent data**
- Time matters; **sequential process, non iid data**
- State is affected by actions

Why is Reinforcement Learning Exciting?

- Difficult robotics tasks
  - Walking robot
  - Drone flight control
  - Navigation
- Complex control problems
  - Control smart power grids
  - Allocate server resourses
  - Optimize elevator availability
- Play games at super-human level
  - Backgammon
  - Go
  - Atari
- Many more…….

Why is Reinforcement Learning Exciting?

- Rapid advances in algorithms
- Deep Q-Networks (DQN) only since 2013
- But there are **pitfalls**:
  - Learning can be slow
  - Gaining experience can be expensive
  - Unintended behaviors occur
- Many recent improvements improve learning rate, reduce required experience

Reward Functions

- A **good reward function** is key to success
- Reward function must be specific to a **task**
- Good reward function must reflect the goal
- Good reward function should be understandable and simple
- Poor reward function can lead to unexpected results

What is Reinforcement Learning?

Properties of reward functions, R_t:

- Reward is a **scalar feedback signal**
- Reward depends on agent's action
- Indicates how agent is doing at time t
- Time maters, **agent executes actions sequentially**
- **Non-instantaneous feedback**: non-zero values may be delayed

Reward Functions

- Reward function examples
- Agent plays a game:

  R(t) = +1 for win; -1 for loss

  Delayed reward; only at end of game

  No path penalty

Reward Functions

- Reward function examples
- Agent navigates robot to goal by shortest path:

  R(t) = -1 for step; +10 for goal

  Penalize for extra steps
- Poor reward function:

  R(t) = +10 for goal

  No penalty for long path

Reward Functions

- Reward function examples
- Agent directs walking robot:

  R(t) = +1 for step; -10 for falling

  Discourages falling
- Poor reward function:

  R(t) = +1 for step; +10 for getting up

  Falling increases cumulative reward!