Introduction to Reinforcement Learning

DEPARTMENT OF COMPUTER SCIENCE
ST. FRANCIS XAVIER UNIVERSITY

CSCI-531 - Reinforcement Learning

Fall 2024

What is Reinforcement Learning (RL)?

- Reinforcement learning is **what to do** to **maximize a reward**.

What is Reinforcement Learning (RL)?

- Reinforcement learning is **what to do** to **maximize a reward**.
- We can give a more "formal" definition.

What is Reinforcement Learning (RL)?

- Reinforcement learning is **what to do** to **maximize a reward**.
- We can give a more "formal" definition.

Definition: Reinforcement Learning

Reinforcement Learning is calculating a function that maps **situations** to **actions**.

What is Reinforcement Learning (RL)?

- Reinforcement learning is **what to do** to **maximize a reward**.
- We can give a more "formal" definition.

Definition: Reinforcement Learning

Reinforcement Learning is calculating a function that maps **situations** to **actions**.

We said that we want to maximize a reward, but **what is a reward?**

What is a Reward?

Activity

Try to explain what a reward is.

Key Characteristics

- To maximize a reward the learner can do **different actions**.

Key Characteristics

- To maximize a reward the learner can do **different actions**.
- If the learner was **passive**, it could not maximize anything.

Key Characteristics

- To maximize a reward the learner can do **different actions**.
- If the learner was **passive**, it could not maximize anything.
- Usually, the learner starts with **no prior knowledge** about what action it should do.

Key Characteristics

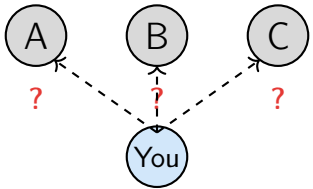
- To maximize a reward the learner can do **different actions**.
- If the learner was **passive**, it could not maximize anything.
- Usually, the learner starts with **no prior knowledge** about what action it should do.

What would you do to maximize a reward if you had no idea which action you should do?

Strategy with No Knowledge

Activity

What would you do to maximize a reward if you had **no idea** which action you should do?

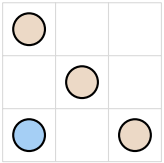


A Simple Example: Learning to Navigate

Scenario

Imagine you're in a new city trying to find the best coffee shop.

- **Your goal:** Find great coffee (maximize reward)

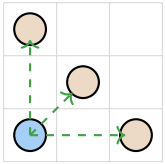


A Simple Example: Learning to Navigate

Scenario

Imagine you're in a new city trying to find the best coffee shop.

- **Your goal:** Find great coffee (maximize reward)
- **Your actions:** Choose which direction to walk, which shops to try

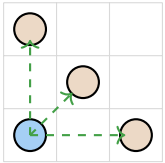


A Simple Example: Learning to Navigate

Scenario

Imagine you're in a new city trying to find the best coffee shop.

- **Your goal:** Find great coffee (maximize reward)
- **Your actions:** Choose which direction to walk, which shops to try
- **Your feedback:** Coffee quality (immediate), but also learning about the neighborhood (delayed)

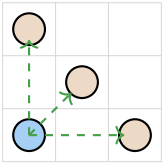


A Simple Example: Learning to Navigate

Scenario

Imagine you're in a new city trying to find the best coffee shop.

- **Your goal:** Find great coffee (maximize reward)
- **Your actions:** Choose which direction to walk, which shops to try
- **Your feedback:** Coffee quality (immediate), but also learning about the neighborhood (delayed)
- **The challenge:** Balance trying new places vs. returning to known good ones

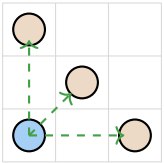


A Simple Example: Learning to Navigate

Scenario

Imagine you're in a new city trying to find the best coffee shop.

- **Your goal:** Find great coffee (maximize reward)
- **Your actions:** Choose which direction to walk, which shops to try
- **Your feedback:** Coffee quality (immediate), but also learning about the neighborhood (delayed)
- **The challenge:** Balance trying new places vs. returning to known good ones



This captures the essence of reinforcement learning!

Coffee Shop Essence of RL

This captures the essence of reinforcement learning:

1. You have a **goal** (good coffee)

Coffee Shop Essence of RL

This captures the essence of reinforcement learning:

1. You have a **goal** (good coffee)
2. You take **actions** (choose directions/shops)

Coffee Shop Essence of RL

This captures the essence of reinforcement learning:

1. You have a **goal** (good coffee)
2. You take **actions** (choose directions/shops)
3. You get **feedback** (coffee quality)

Coffee Shop Essence of RL

This captures the essence of reinforcement learning:

1. You have a **goal** (good coffee)
2. You take **actions** (choose directions/shops)
3. You get **feedback** (coffee quality)
4. You **learn and improve** your strategy over time

Coffee Shop Essence of RL

This captures the essence of reinforcement learning:

1. You have a **goal** (good coffee)
2. You take **actions** (choose directions/shops)
3. You get **feedback** (coffee quality)
4. You **learn and improve** your strategy over time
5. You must balance **exploration** (new places) vs **exploitation** (known good places)

Coffee Shop Essence of RL

This captures the essence of reinforcement learning:

1. You have a **goal** (good coffee)
2. You take **actions** (choose directions/shops)
3. You get **feedback** (coffee quality)
4. You **learn and improve** your strategy over time
5. You must balance **exploration** (new places) vs **exploitation** (known good places)

Coffee Shop Strategies

Activity

In the coffee shop example:

1. What would happen if you only went to the **first decent shop** you found?

Coffee Shop Strategies

Activity

In the coffee shop example:

1. What would happen if you only went to the **first decent shop** you found?
2. What would happen if you tried a **completely new shop every single day**?

Coffee Shop Strategies

Activity

In the coffee shop example:

1. What would happen if you only went to the **first decent shop** you found?
2. What would happen if you tried a **completely new shop every single day**?
3. What's a **good strategy for the long term**?

Coffee Shop Strategies

Activity

In the coffee shop example:

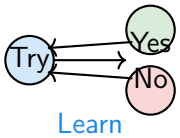
1. What would happen if you only went to the **first decent shop** you found?
2. What would happen if you tried a **completely new shop every single day**?
3. What's a **good strategy for the long term**?

Exploit	Explore	Balance
Safe but limited	Risky but learning	Optimal strategy

Key Concepts - The Basics

Reinforcement learning has two main characteristics:

- **Trial-and-error search:** Learning by trying things and seeing what works



Why these matter:

- **Trial-and-error** means no teacher provides "correct" answers - you learn by experience

Key Concepts - The Basics

Reinforcement learning has two main characteristics:

- **Trial-and-error search:** Learning by trying things and seeing what works
- **Delayed rewards:** Actions now may have consequences much later

Why these matter:

- **Trial-and-error** means no teacher provides "correct" answers - you learn by experience
- **Delayed rewards** means you must connect actions to outcomes that happen later

Important Distinction

Warning

Reinforcement learning is a name that regroups different concepts:

- It's a **type of problem**.

Important Distinction

Warning

Reinforcement learning is a name that regroups different concepts:

- It's a **type of problem**.
- It's also a **class of solution methods**.

Important Distinction

Warning

Reinforcement learning is a name that regroups different concepts:

- It's a **type of problem**.
- It's also a **class of solution methods**.
- And it's the **field** that studies the two previous points as well.

Important Distinction

Warning

Reinforcement learning is a name that regroups different concepts:

- It's a **type of problem**.
- It's also a **class of solution methods**.
- And it's the **field** that studies the two previous points as well.

You need to understand the distinction.

What is the Reinforcement Learning Problem? (Simplified)

- The reinforcement learning problem is an idea coming from **dynamical system theory**.

What is the Reinforcement Learning Problem? (Simplified)

- The reinforcement learning problem is an idea coming from **dynamical system theory**.
- And more specifically from the **Markov Decision Processes**.

What is the Reinforcement Learning Problem? (Simplified)

- The reinforcement learning problem is an idea coming from **dynamical system theory**.
- And more specifically from the **Markov Decision Processes**.
- The basic ideas are:

What is the Reinforcement Learning Problem? (Simplified)

- The reinforcement learning problem is an idea coming from **dynamical system theory**.
- And more specifically from the **Markov Decision Processes**.
- The basic ideas are:
  - A learning agent must **sense** the state of the environment.

What is the Reinforcement Learning Problem? (Simplified)

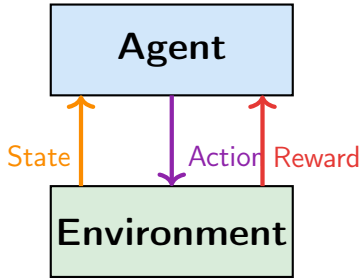
- The reinforcement learning problem is an idea coming from **dynamical system theory**.
- And more specifically from the **Markov Decision Processes**.
- The basic ideas are:
  - A learning agent must **sense** the state of the environment.
  - The agent must be able to take **actions** that affect the state.

What is the Reinforcement Learning Problem? (Simplified)

- The reinforcement learning problem is an idea coming from **dynamical system theory**.
- And more specifically from the **Markov Decision Processes**.
- The basic ideas are:
  - A learning agent must **sense** the state of the environment.
  - The agent must be able to take **actions** that affect the state.
  - It must have a **goal** or goals relating to the state of the environment.

What is the Reinforcement Learning Problem? (Simplified)

- The reinforcement learning problem is an idea coming from **dynamical system theory**.
- And more specifically from the **Markov Decision Processes**.
- The basic ideas are:
  - A learning agent must **sense** the state of the environment.
  - The agent must be able to take **actions** that affect the state.
  - It must have a **goal** or goals relating to the state of the environment.

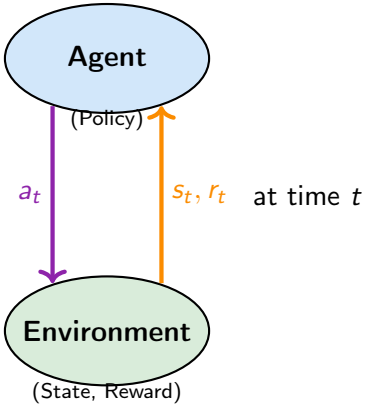


Agent-Environment Loop

This simple agent-environment loop might look basic, but it powers some of the most impressive AI breakthroughs.

Agent-Environment Loop

This simple agent-environment loop might look basic, but it powers some of the most impressive AI breakthroughs.



Why Reinforcement Learning Matters

Real-World Success Stories

- **Game Playing:** AlphaGo defeated world champions, mastering strategy through self-play

Why Reinforcement Learning Matters

Real-World Success Stories

- **Game Playing:** AlphaGo defeated world champions, mastering strategy through self-play
- **Autonomous Systems:** Self-driving cars make split-second decisions in complex environments

Why Reinforcement Learning Matters

Real-World Success Stories

- **Game Playing:** AlphaGo defeated world champions, mastering strategy through self-play
- **Autonomous Systems:** Self-driving cars make split-second decisions in complex environments
- **Finance & Trading:** Systems optimize investment decisions over time under uncertainty

What Makes These Problems Special?

Why Traditional AI Approaches Fail

These problems require:

1. **Learning without a teacher** - no dataset of "perfect" decisions exists

What Makes These Problems Special?

Why Traditional AI Approaches Fail

These problems require:

1. **Learning without a teacher** - no dataset of "perfect" decisions exists
2. **Handling delayed consequences** - actions now affect outcomes much later

What Makes These Problems Special?

Why Traditional AI Approaches Fail

These problems require:

1. **Learning without a teacher** - no dataset of "perfect" decisions exists
2. **Handling delayed consequences** - actions now affect outcomes much later
3. **Adapting to change** - environment responds to your actions

What Makes These Problems Special?

Why Traditional AI Approaches Fail

These problems require:

1. **Learning without a teacher** - no dataset of "perfect" decisions exists
2. **Handling delayed consequences** - actions now affect outcomes much later
3. **Adapting to change** - environment responds to your actions
4. **Balancing exploration vs exploitation** - try new things vs use current knowledge

What Makes These Problems Special?

Why Traditional AI Approaches Fail

These problems require:

1. **Learning without a teacher** - no dataset of "perfect" decisions exists
2. **Handling delayed consequences** - actions now affect outcomes much later
3. **Adapting to change** - environment responds to your actions
4. **Balancing exploration vs exploitation** - try new things vs use current knowledge

You might wonder: why couldn't traditional machine learning solve these problems? Understanding what RL is requires understanding what it's not.

What Reinforcement Learning is NOT?

Understanding RL is easier when we compare it to other learning paradigms:

Learning Type	Data Required	Feedback Type	Goal
Supervised	Labeled examples	Immediate correct answers	Predict/classify
Unsupervised	Unlabeled data	No feedback	Find patterns
Reinforcement	Environment interaction	Delayed rewards	Maximize reward

What Reinforcement Learning is NOT?

Understanding RL is easier when we compare it to other learning paradigms:

Learning Type	Data Required	Feedback Type	Goal
Supervised	Labeled examples	Immediate correct answers	Predict/classify
Unsupervised	Unlabeled data	No feedback	Find patterns
Reinforcement	Environment interaction	Delayed rewards	Maximize reward

Example: Email spam detection

- Has labeled data

Example: Game playing

What Reinforcement Learning is NOT?

Understanding RL is easier when we compare it to other learning paradigms:

Learning Type	Data Required	Feedback Type	Goal
Supervised	Labeled examples	Immediate correct answers	Predict/classify
Unsupervised	Unlabeled data	No feedback	Find patterns
Reinforcement	Environment interaction	Delayed rewards	Maximize reward

Example: Email spam detection

- Has labeled data
- Immediate "correct" classification

Example: Game playing

What Reinforcement Learning is NOT?

Understanding RL is easier when we compare it to other learning paradigms:

Learning Type	Data Required	Feedback Type	Goal
Supervised	Labeled examples	Immediate correct answers	Predict/classify
Unsupervised	Unlabeled data	No feedback	Find patterns
Reinforcement	Environment interaction	Delayed rewards	Maximize reward

Example: Email spam detection

- Has labeled data
- Immediate "correct" classification
- $\rightarrow$ Supervised Learning

Example: Game playing

What Reinforcement Learning is NOT?

Understanding RL is easier when we compare it to other learning paradigms:

Learning Type	Data Required	Feedback Type	Goal
Supervised	Labeled examples	Immediate correct answers	Predict/classify
Unsupervised	Unlabeled data	No feedback	Find patterns
Reinforcement	Environment interaction	Delayed rewards	Maximize reward

Example: Email spam detection

- Has labeled data
- Immediate "correct" classification
- $\rightarrow$ Supervised Learning

Example: Game playing

- No "correct" move dataset

What Reinforcement Learning is NOT?

Understanding RL is easier when we compare it to other learning paradigms:

Learning Type	Data Required	Feedback Type	Goal
Supervised	Labeled examples	Immediate correct answers	Predict/classify
Unsupervised	Unlabeled data	No feedback	Find patterns
Reinforcement	Environment interaction	Delayed rewards	Maximize reward

Example: Email spam detection

- Has labeled data
- Immediate "correct" classification
- $\rightarrow$ Supervised Learning

Example: Game playing

- No "correct" move dataset
- Delayed consequences

What Reinforcement Learning is NOT?

Understanding RL is easier when we compare it to other learning paradigms:

Learning Type	Data Required	Feedback Type	Goal
Supervised	Labeled examples	Immediate correct answers	Predict/classify
Unsupervised	Unlabeled data	No feedback	Find patterns
Reinforcement	Environment interaction	Delayed rewards	Maximize reward

Example: Email spam detection

- Has labeled data
- Immediate "correct" classification
- $\rightarrow$ Supervised Learning

Example: Game playing

- No "correct" move dataset
- Delayed consequences
- $\rightarrow$ Reinforcement Learning

Key Differences Explained

Why supervised learning fails for RL problems:

- **No perfect dataset:** There's no collection of "correct" actions for every situation

Why unsupervised learning isn't enough:

Key Differences Explained

Why supervised learning fails for RL problems:

- **No perfect dataset:** There's no collection of "correct" actions for every situation
- **Context dependency:** The best action depends on long-term consequences, not just current state

Why unsupervised learning isn't enough:

Key Differences Explained

Why supervised learning fails for RL problems:

- **No perfect dataset:** There's no collection of "correct" actions for every situation
- **Context dependency:** The best action depends on long-term consequences, not just current state
- **Interactive nature:** The environment changes based on your actions

Why unsupervised learning isn't enough:

Key Differences Explained

Why supervised learning fails for RL problems:

- **No perfect dataset:** There's no collection of "correct" actions for every situation
- **Context dependency:** The best action depends on long-term consequences, not just current state
- **Interactive nature:** The environment changes based on your actions

Why unsupervised learning isn't enough:

- **No objective:** Finding patterns doesn't tell you which actions are good

Key Differences Explained

Why supervised learning fails for RL problems:

- **No perfect dataset:** There's no collection of "correct" actions for every situation
- **Context dependency:** The best action depends on long-term consequences, not just current state
- **Interactive nature:** The environment changes based on your actions

Why unsupervised learning isn't enough:

- **No objective:** Finding patterns doesn't tell you which actions are good
- **No feedback:** You can't improve without knowing if you're doing well

Learning Paradigms

1. Why can't you use supervised learning to learn chess strategy?

Learning Paradigms

1. Why can't you use supervised learning to learn chess strategy?
2. What would unsupervised learning find in a chess game, and why isn't that sufficient?

Learning Paradigms

1. Why can't you use supervised learning to learn chess strategy?
2. What would unsupervised learning find in a chess game, and why isn't that sufficient?
3. Give an example of a problem where you'd need each type of learning.

The Challenges of Reinforcement Learning

Reinforcement learning faces unique challenges that make it different from other machine learning approaches:

The Exploration-Exploitation Trade-off

The Challenges of Reinforcement Learning

Reinforcement learning faces unique challenges that make it different from other machine learning approaches:

The Exploration-Exploitation Trade-off

This is the fundamental challenge in RL

The Exploration-Exploitation Trade-off

Strategy	Description	Pros	Cons
Exploitation	Use current knowledge	Immediate gains	Miss better options
Exploration	Try new actions	Discover better options	Short-term costs
Balance	Mix both strategies	Long-term optimal	Complex to implement

The Exploration-Exploitation Trade-off

Strategy	Description	Pros	Cons
Exploitation	Use current knowledge	Immediate gains	Miss better options
Exploration	Try new actions	Discover better options	Short-term costs
Balance	Mix both strategies	Long-term optimal	Complex to implement

Examples:

- **Exploitation:** Always go to your favorite restaurant

The Exploration-Exploitation Trade-off

Strategy	Description	Pros	Cons
Exploitation	Use current knowledge	Immediate gains	Miss better options
Exploration	Try new actions	Discover better options	Short-term costs
Balance	Mix both strategies	Long-term optimal	Complex to implement

Examples:

- **Exploitation:** Always go to your favorite restaurant
- **Exploration:** Try a new restaurant (might be bad)

The Exploration-Exploitation Trade-off

Strategy	Description	Pros	Cons
Exploitation	Use current knowledge	Immediate gains	Miss better options
Exploration	Try new actions	Discover better options	Short-term costs
Balance	Mix both strategies	Long-term optimal	Complex to implement

Examples:

- **Exploitation:** Always go to your favorite restaurant
- **Exploration:** Try a new restaurant (might be bad)
- **Balance:** Sometimes try new places, sometimes stick to favorites

The Whole Problem Challenge

The Whole Problem Challenge

- **Complete system:** RL considers the entire problem from start to finish

The Whole Problem Challenge

The Whole Problem Challenge

- **Complete system:** RL considers the entire problem from start to finish
- **Goal-seeking agent:** Must actively pursue objectives, not just respond to inputs

The Whole Problem Challenge

The Whole Problem Challenge

- **Complete system:** RL considers the entire problem from start to finish
- **Goal-seeking agent:** Must actively pursue objectives, not just respond to inputs
- **Uncertainty handling:** Must operate effectively despite incomplete information about the environment

The Whole Problem Challenge

The Whole Problem Challenge

- **Complete system:** RL considers the entire problem from start to finish
- **Goal-seeking agent:** Must actively pursue objectives, not just respond to inputs
- **Uncertainty handling:** Must operate effectively despite incomplete information about the environment

Understanding these challenges prepares us to examine what makes RL systems work. Every RL agent relies on the same core building blocks.

The Four Core Elements

Element	Purpose	Think of it as...	Required?
Policy	Decision maker	The brain that chooses actions	Essential
Reward Function	Goal definition	The scoring system	Essential
Value Function	Long-term predictor	The strategic advisor	Essential
Model	Environment simulator	The crystal ball	Optional

The Four Core Elements

Element	Purpose	Think of it as...	Required?
Policy	Decision maker	The brain that chooses actions	Essential
Reward Function	Goal definition	The scoring system	Essential
Value Function	Long-term predictor	The strategic advisor	Essential
Model	Environment simulator	The crystal ball	Optional

These four elements work together like a team - each has a specific role, but their real power comes from how they interact in the learning process.

1. Policy - The Decision Maker

Definition: Policy

A policy is a function that maps each state to an action.

What it does:

- Defines the behavior of an agent at any given time

1. Policy - The Decision Maker

Definition: Policy

A policy is a function that maps each state to an action.

What it does:

- Defines the behavior of an agent at any given time
- Core of the RL agent - determines all actions

1. Policy - The Decision Maker

Definition: Policy

A policy is a function that maps each state to an action.

What it does:

- Defines the behavior of an agent at any given time
- Core of the RL agent - determines all actions
- Can be deterministic (same action every time) or stochastic (probabilistic)

1. Policy - The Decision Maker

Definition: Policy

A policy is a function that maps each state to an action.

What it does:

- Defines the behavior of an agent at any given time
- Core of the RL agent - determines all actions
- Can be deterministic (same action every time) or stochastic (probabilistic)

Examples:

- **Chess:** "If opponent threatens my queen, move it to safety"
- **Trading:** "If price drops 5%, sell 20% of holdings"

$s_1 \rightarrow a_2$
$s_2 \rightarrow a_1$
$s_3 \rightarrow a_3$

2. Reward Function - The Goal Definition

Definition: Reward

A reward is a value returned by the environment at a time step t.

What it does:

- Defines the goal of the RL problem

2. Reward Function - The Goal Definition

Definition: Reward

A reward is a value returned by the environment at a time step t.

What it does:

- Defines the goal of the RL problem
- Provides immediate feedback for actions

2. Reward Function - The Goal Definition

Definition: Reward

A reward is a value returned by the environment at a time step t.

What it does:

- Defines the goal of the RL problem
- Provides immediate feedback for actions
- Agent's objective: maximize total reward over time

2. Reward Function - The Goal Definition

Definition: Reward

A reward is a value returned by the environment at a time step t.

What it does:

- Defines the goal of the RL problem
- Provides immediate feedback for actions
- Agent's objective: maximize total reward over time

Examples:

- **Game:** +10 for winning, -1 for losing, 0 for draw
- **Robot:** +1 for forward movement, -10 for collision
- **Trading:** +profit for good trades, -loss for bad ones

3. Value Function - The Strategic Advisor

Definition: Value Function

A value function is a function returning for each state the total expected reward starting from this state.

What it does:

- Estimates long-term expected reward from each state

3. Value Function - The Strategic Advisor

Definition: Value Function

A value function is a function returning for each state the total expected reward starting from this state.

What it does:

- Estimates long-term expected reward from each state
- Helps agent think strategically, not just about immediate rewards

3. Value Function - The Strategic Advisor

Definition: Value Function

A value function is a function returning for each state the total expected reward starting from this state.

What it does:

- Estimates long-term expected reward from each state
- Helps agent think strategically, not just about immediate rewards
- Much harder to determine than immediate rewards

3. Value Function - The Strategic Advisor

Definition: Value Function

A value function is a function returning for each state the total expected reward starting from this state.

What it does:

- Estimates long-term expected reward from each state
- Helps agent think strategically, not just about immediate rewards
- Much harder to determine than immediate rewards

Key insight: We seek actions that lead to states with higher **value**, not higher immediate **reward**.

Activity: Reward vs Value

Interactive Activity

Reward vs Value - Which would you choose?

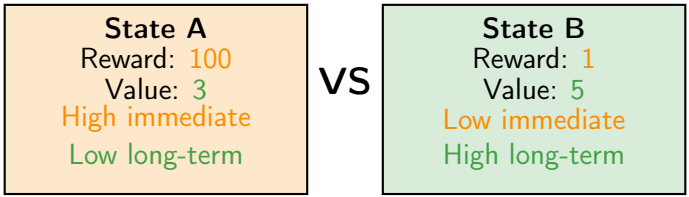
- State A: immediate reward = 100, long-term value = 3
- State B: immediate reward = 1, long-term value = 5

Activity: Reward vs Value

Interactive Activity

Reward vs Value - Which would you choose?

- State A: immediate reward = 100, long-term value = 3
- State B: immediate reward = 1, long-term value = 5

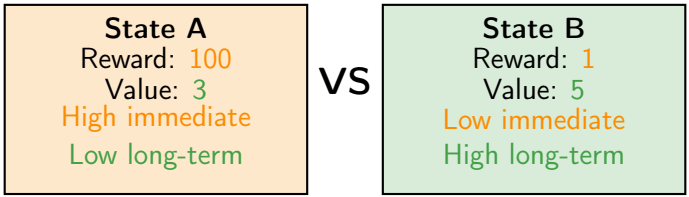


Activity: Reward vs Value

Interactive Activity

Reward vs Value - Which would you choose?

- State A: immediate reward = 100, long-term value = 3
- State B: immediate reward = 1, long-term value = 5



Question: Why might State B be better despite lower immediate reward?

Value vs Reward Examples

Examples of choosing value over immediate reward:

- **Chess:** Sacrificing a piece (negative reward) to gain better position (higher value)

Value vs Reward Examples

Examples of choosing value over immediate reward:

- **Chess:** Sacrificing a piece (negative reward) to gain better position (higher value)

Value vs Reward Examples

Examples of choosing value over immediate reward:

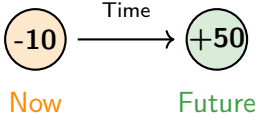
- **Chess:** Sacrificing a piece (negative reward) to gain better position (higher value)

Value vs Reward Examples

Examples of choosing value over immediate reward:

- **Chess:** Sacrificing a piece (negative reward) to gain better position (higher value)

Strategic Thinking



4. Model - The Crystal Ball (Optional)

Definition: Model

The model of the environment is the representation of the dynamic of the problem.

What it does:

- Predicts what happens next: given current state and action, what's the next state and reward?

4. Model - The Crystal Ball (Optional)

Definition: Model

The model of the environment is the representation of the dynamic of the problem.

What it does:

- Predicts what happens next: given current state and action, what's the next state and reward?
- Allows agent to plan ahead and simulate different strategies

4. Model - The Crystal Ball (Optional)

Definition: Model

The model of the environment is the representation of the dynamic of the problem.

What it does:

- Predicts what happens next: given current state and action, what's the next state and reward?
- Allows agent to plan ahead and simulate different strategies
- Not always available or practical to learn

4. Model - The Crystal Ball (Optional)

Definition: Model

The model of the environment is the representation of the dynamic of the problem.

What it does:

- Predicts what happens next: given current state and action, what's the next state and reward?
- Allows agent to plan ahead and simulate different strategies
- Not always available or practical to learn

Type	Model?	Characteristics	Examples
Model-based	Yes	Can plan ahead, simulate scenarios	Chess engines
Model-free	No	Learn directly from experience	Most game AI

Summary

You now understand the core building blocks of reinforcement learning!

From our coffee shop example to these fundamental elements, you've seen how RL systems learn through experience.

What we've covered:

- **Core RL Concepts:** Trial-and-error learning with delayed rewards

Summary

You now understand the core building blocks of reinforcement learning!

From our coffee shop example to these fundamental elements, you've seen how RL systems learn through experience.

What we've covered:

- **Core RL Concepts:** Trial-and-error learning with delayed rewards
- **Key Elements:** Policy, rewards, values, and models

Summary

You now understand the core building blocks of reinforcement learning!

From our coffee shop example to these fundamental elements, you've seen how RL systems learn through experience.

What we've covered:

- **Core RL Concepts:** Trial-and-error learning with delayed rewards
- **Key Elements:** Policy, rewards, values, and models
- **Main Challenge:** Balancing exploration vs exploitation

Summary

You now understand the core building blocks of reinforcement learning!

From our coffee shop example to these fundamental elements, you've seen how RL systems learn through experience.

What we've covered:

- **Core RL Concepts:** Trial-and-error learning with delayed rewards
- **Key Elements:** Policy, rewards, values, and models
- **Main Challenge:** Balancing exploration vs exploitation
- **Why RL Matters:** Real-world applications where traditional AI fails

Summary

You now understand the core building blocks of reinforcement learning!

From our coffee shop example to these fundamental elements, you've seen how RL systems learn through experience.

What we've covered:

- **Core RL Concepts:** Trial-and-error learning with delayed rewards
- **Key Elements:** Policy, rewards, values, and models
- **Main Challenge:** Balancing exploration vs exploitation
- **Why RL Matters:** Real-world applications where traditional AI fails

Up Next: Multi-Armed Bandits

Current Limitations & Assumptions

RL is powerful but has important limitations to keep in mind:

Challenge	Description	Impact
State Design	How to represent the current situation	Learning speed
State Space Size	Too many possible states	Learning speed
Partial Observability	Can't see everything relevant	Incomplete info

Key Points:

- **State is everything:** The quality of state representation determines learning success

Current Limitations & Assumptions

RL is powerful but has important limitations to keep in mind:

Challenge	Description	Impact
State Design	How to represent the current situation	Learning speed
State Space Size	Too many possible states	Learning speed
Partial Observability	Can't see everything relevant	Incomplete info

Key Points:

- **State is everything:** The quality of state representation determines learning success
- **Not magic:** RL assumes you can define reasonable states, actions, and rewards

Current Limitations & Assumptions

RL is powerful but has important limitations to keep in mind:

Challenge	Description	Impact
State Design	How to represent the current situation	Learning speed
State Space Size	Too many possible states	Learning speed
Partial Observability	Can't see everything relevant	Incomplete info

Key Points:

- **State is everything:** The quality of state representation determines learning success
- **Not magic:** RL assumes you can define reasonable states, actions, and rewards
- **Design matters:** How you frame the problem affects what the agent can learn

Activity: Driving Example

Interactive Activity

Think about teaching someone to drive:

1. What information (state) do they need to make good decisions?

Activity: Driving Example

Interactive Activity

Think about teaching someone to drive:

1. What information (state) do they need to make good decisions?
2. What if they could only see through a small window - how would this affect learning?

Activity: Driving Example

Interactive Activity

Think about teaching someone to drive:

1. What information (state) do they need to make good decisions?
2. What if they could only see through a small window - how would this affect learning?
3. How would you define "good driving" as rewards?