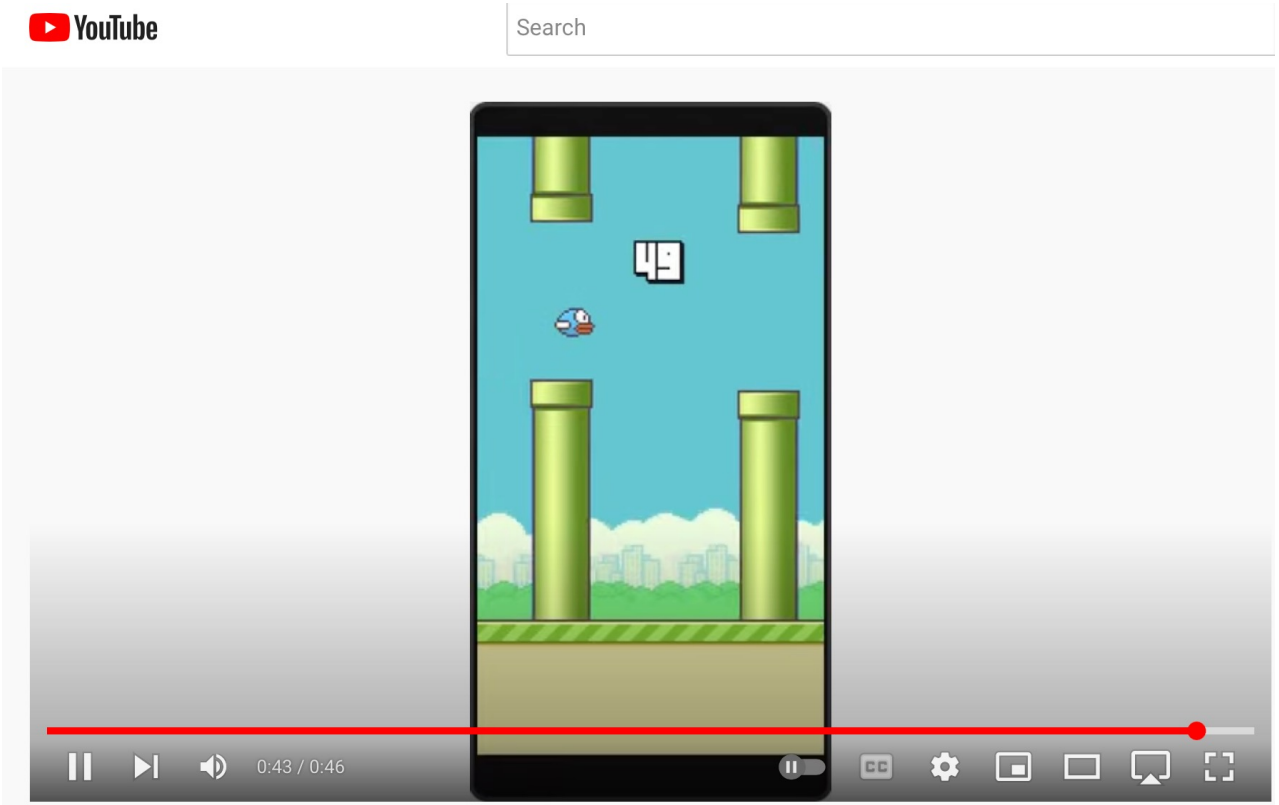
Overview of Reinforcement Learning

Dr. Lotfi ben Othmane
University of North Texas

The Flappy Bird Game

https://www.youtube.com/watch?v=BV7a4rufMOg&t=5s

Can you do it with supervised machine learning?

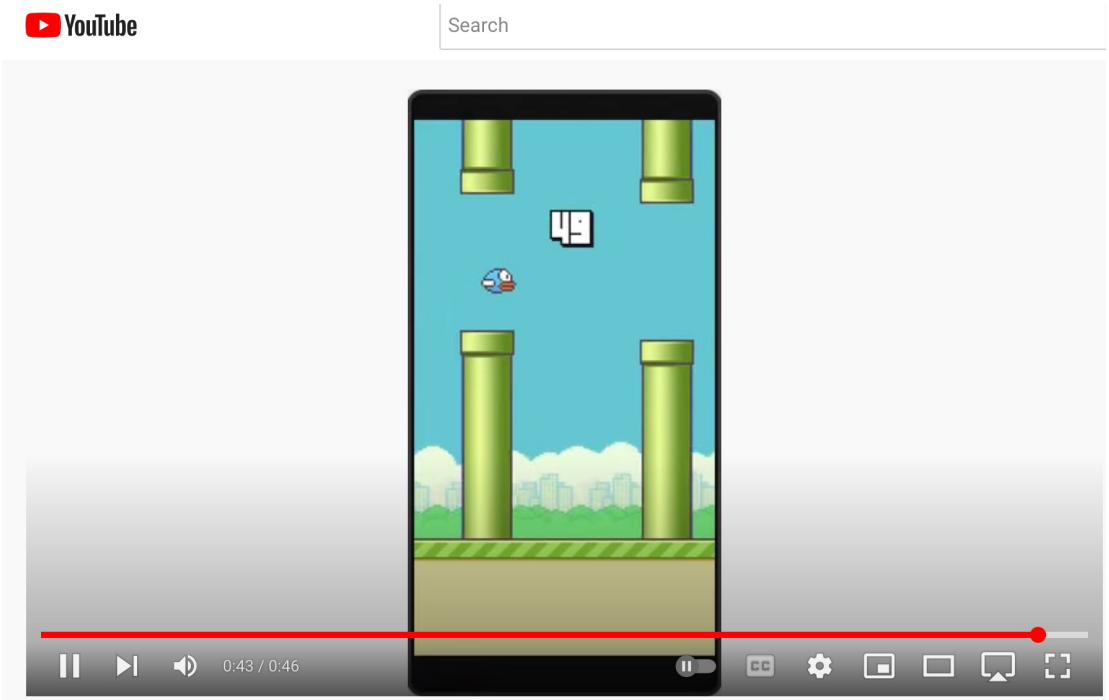


The Flappy Bird Game

Reward

+1 if Flappy Bird is still alive

-1000 if Flappy Bird is dead



The Flappy Bird Game Program

http://sarvagyavaish.github.io/FlappyBirdRL/

Inputs
- S is a set of states
- A is a set of actions
- γ the discount
- α is the step size

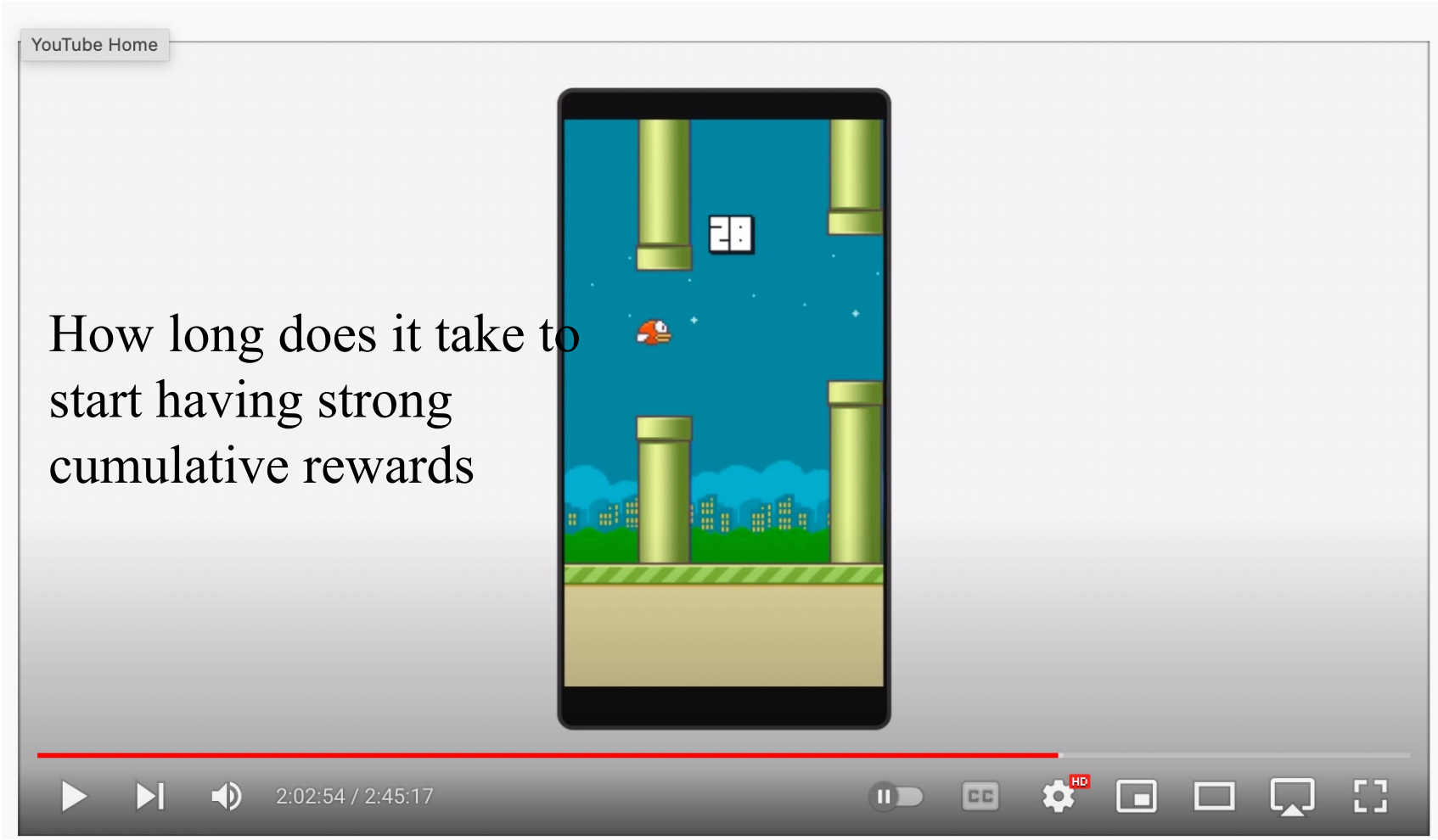
Local
- real array $Q[S,A]$
- previous state s
- previous action a

initialize $Q[S,A]$ arbitrarily
observe current state s
repeat
- select and carry out an action a
- observe reward r and state s'
- $Q[s,a] \leftarrow Q[s,a] + \alpha(r + \gamma \max_{a'} Q[s',a'] - Q[s,a])$
- $s \leftarrow s'$ **until** termination

S: State as the step
A: Actions are: (1) jump or (2) do nothing
α: Learning rate– value 0.7
r: Reward +1 or -1000
γ: Discount factor (set to 1)

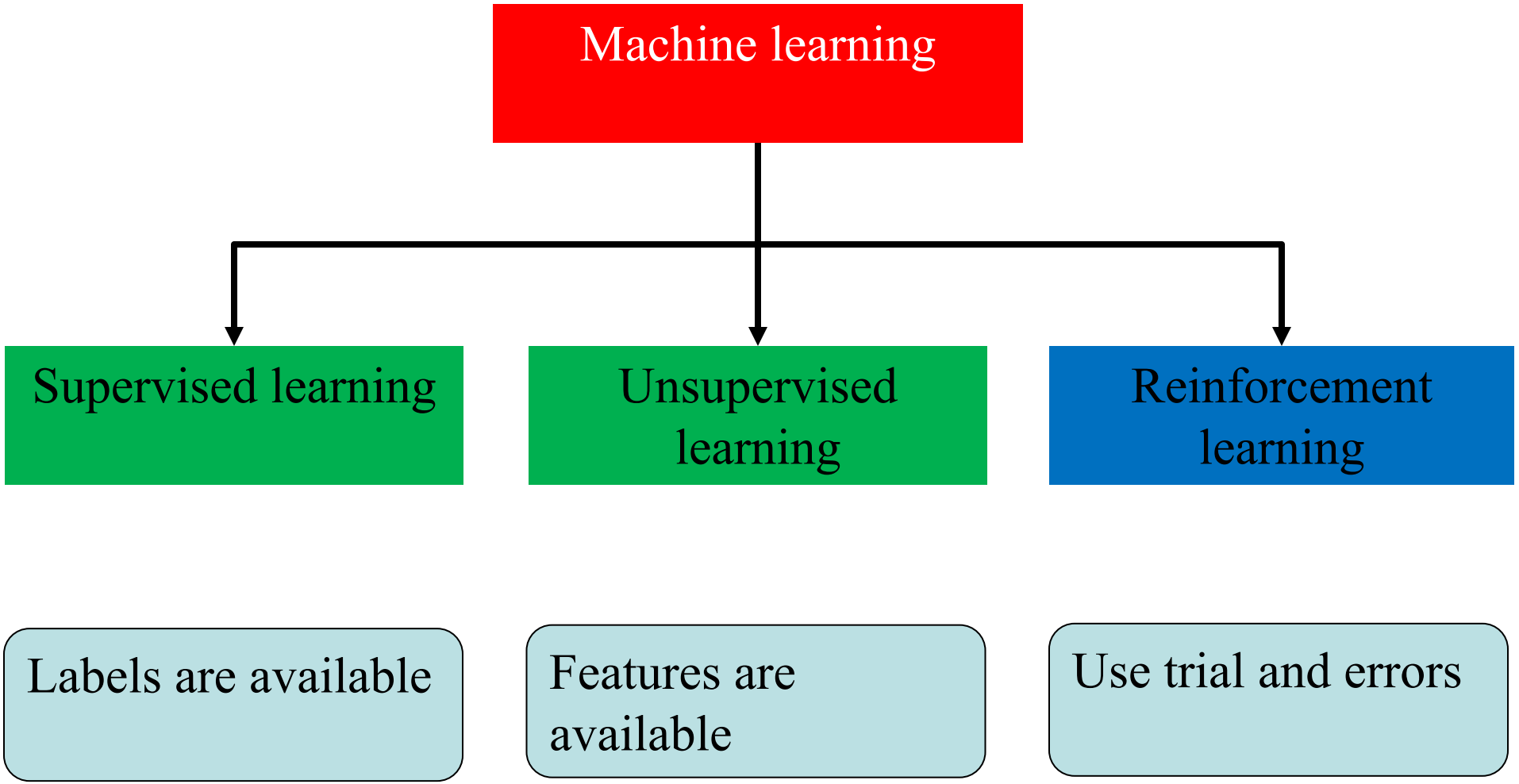
What about the function Q learning?

The Flappy Bird Game – Q Learning Over Time



https://www.youtube.com/watch?v=OJw4HTWvGdY&t=21s

Reinforcement Learning

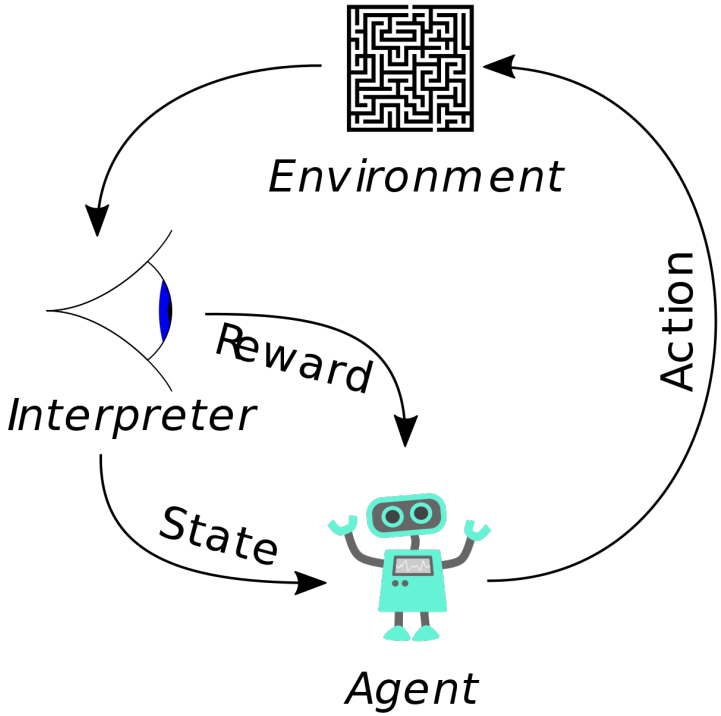


Reinforcement Learning

- In a lot of cases, like in games, the data is not good enough to have an intelligent machine. But, we have knowledge on how to make decisions.

Reinforcement Learning

Reinforcement learning is the problem faced by an agent that must learn behavior through trial-and-error interactions with dynamic environment.



https://en.wikipedia.org/wiki/Reinforcement

Reinforcement Learning

The agent interacts with its environment in discrete time steps. At each time t, the agent receives the current state s_t and reward r_t. It chooses an action a_t from the set of available actions, which is subsequently sent to the environment. The environment moves to a new state s_{t+1} and the reward r_{t+1} associated with the *transition* (s_t,a_t,s_{t+1}) is determined.

In a policy RL, the goal of a reinforcement learning agent is to learn a *policy*

$\pi: A \times S \rightarrow [0,1]$, $\pi(a,s)=\Pr(a_t=a|s_t=s)$ which maximizes the expected cumulative reward.

Reinforcement Learning

Reinforcement learning could be modeled as a Markov Decision Process.

S: Set of environment and agent states

A: Set of actions of the agent

$P_a(s, s') = P(s_{t+1} = s' | s_t = s, a_t = a)$ is the probability of transition (at time **t**) from state **s** to state **s'** under action **a.**

$R_a(s,s')$ is the immediate reward after transition from **s** to **s'** with action **a**.

Goal: Maximize the cumulative reward $R_a(s, s')$

Simple Approach

Brute force approach

1. For each possible policy, sample returns while following it
2. Choose the policy with the largest expected return

One problem with this is that the number of policies can be large, or even infinite.

RL Exploration vs. Exploitation

- Good decisions are through exploitation of knowledge
  => Pick the action that has the highest utility
- Exploration: Pick a randomly reasonably good action
- The utility weights the sub-optimal exploitation decision with $0 < \epsilon < 1$

Applications of RL

- There many applications of RL including:
  1. Games
  2. Robotics and control

Exercise 1

- Formulate cleaning robot using reinforcement learning
  - What is the rewards?
  - What are the states?
  - What are the transitions?

Exercise 2

- Formulate the robot using reinforcement learning
  - What is the rewards?
  - What are the states?
  - What are the transitions?

RL for RoboCup Soccer

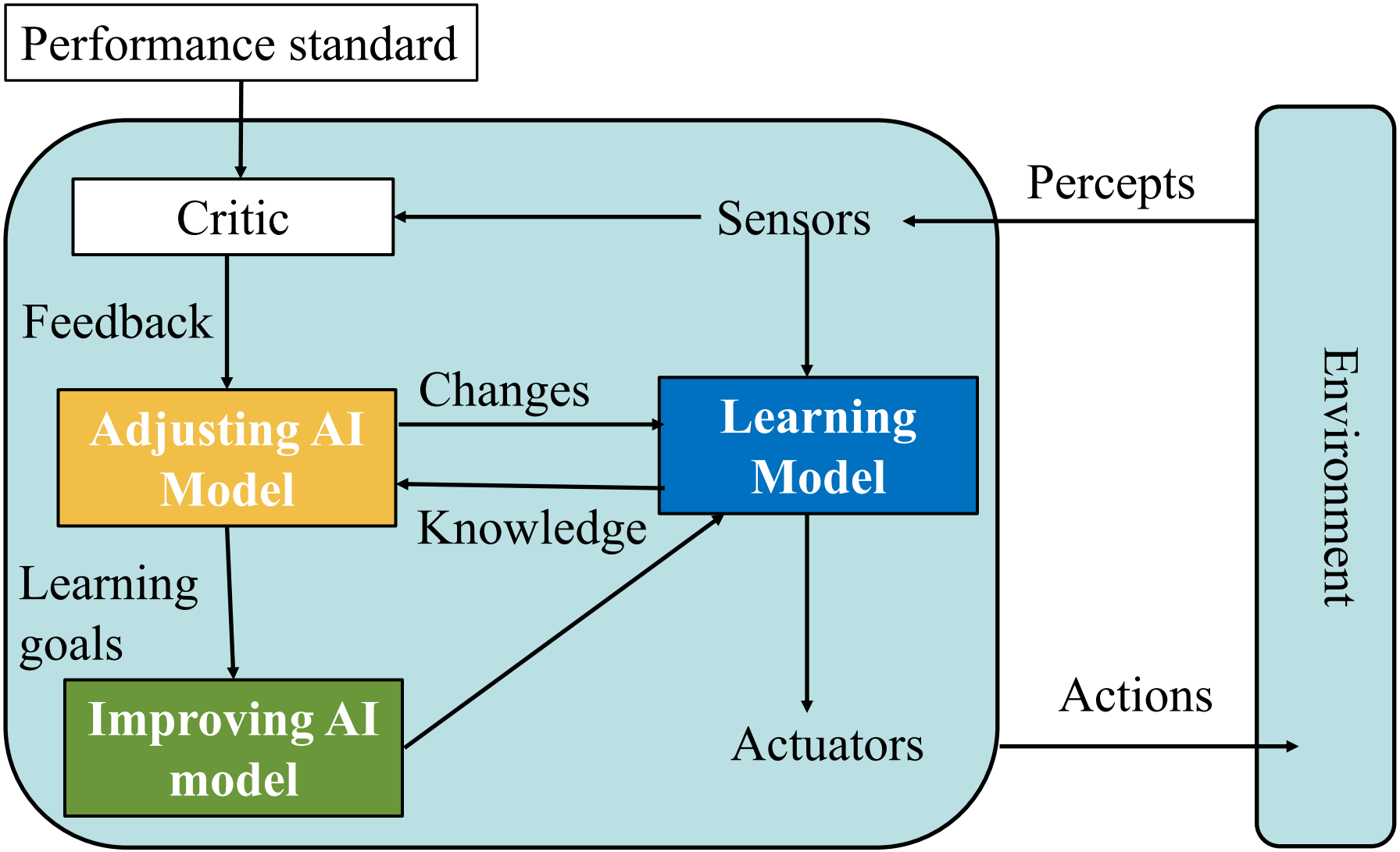
https://www.youtube.com/watch?v=51L_gq_fu5U

RL with Unknown Reward

- Receive a reward when the agent puts the ball in the net.
- The agent needs to catch the ball, give the ball to team members, advance with the ball, and shoot the ball.
- How RL can help in such scenarios

https://rcsoccersim.github.io
https://github.com/rcsoccersim/

Learning Agents



Summary

- Reinforcement learning is the problem faced by an agent that must learn behavior through trial-and-error interactions with dynamic environment.
- Reinforcement learning methods looks to optimize the reward from the decisions.
- It has many applications mostly in games and robotics/control theory

Thank you

Any Question?