# REINFORCE

# Announcements

- Reminder: Project writeups due Monday. (Turn in hard copy and email soft copy.)

- Guha out of town today and next week, back for final

- Final exam: **Tuesday, May 22 - 7:15-9:30 AM**
  - Last chance to ask questions 7:15 to 7:45AM. Exam will be then distributed. Leave as you finish.

- Today: Last presentations, R.L., and start of review. Monday: Finish review.

- How to prepare for final exam: (1) Closely pay attention to the review lectures and review their slides. (2) Skim back over old slides at high-level.

# Settings So Far

- Supervised
  - Transform X to Y
    - Classification
    - Regression

# Unsupervised

- k-Means
- GAN
- VAE

# Today

- Different Setting

# Terminology

- Agent
- Environment

# Agent

- Acts (picks one of several possible actions)
- Environment (gives reward)

# Over Time

- Maximize Expected Reward

# Examples

- Chess
    - Reward of 1 when you win the game, 0 in the middle (so v sparse)

# Examples

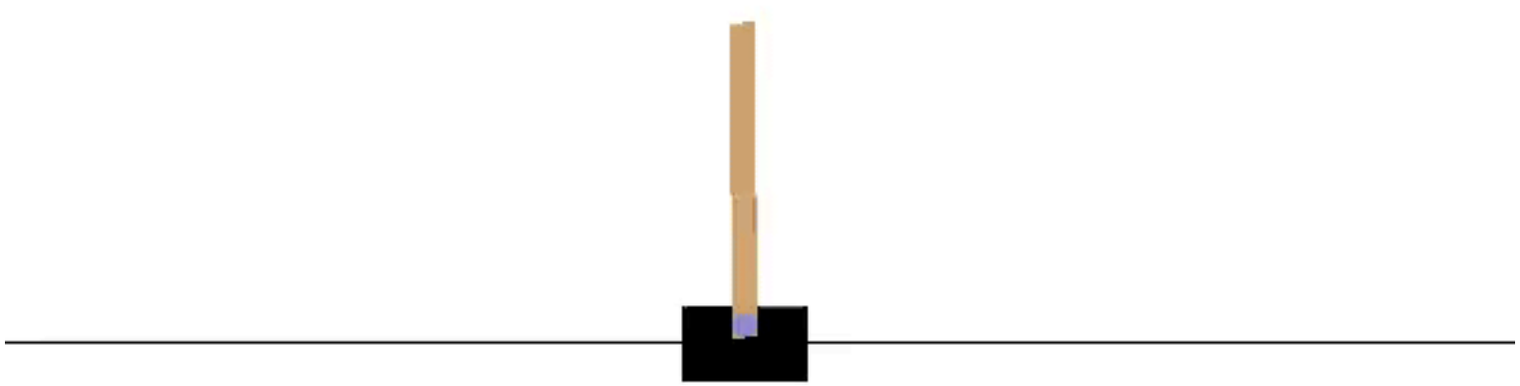
- Racing
- Reward 1 when you win
- Possibly negative if you crash

# Examples

- And many more

# Demo

- We’ll solve a simple problem
- Explore + Exploit



# Agent

- Allowed moves:
  - +1 force
  - -1 force

# Environment

- Reward of +1 for every time-step the pole is upright

- The episode ends when the pole is more than 15 degrees from vertical, or the cart moves more than 2.4 units from the center

# Environment

- Current State:
  - Θ - angle of the pole
  - x - position of the cart

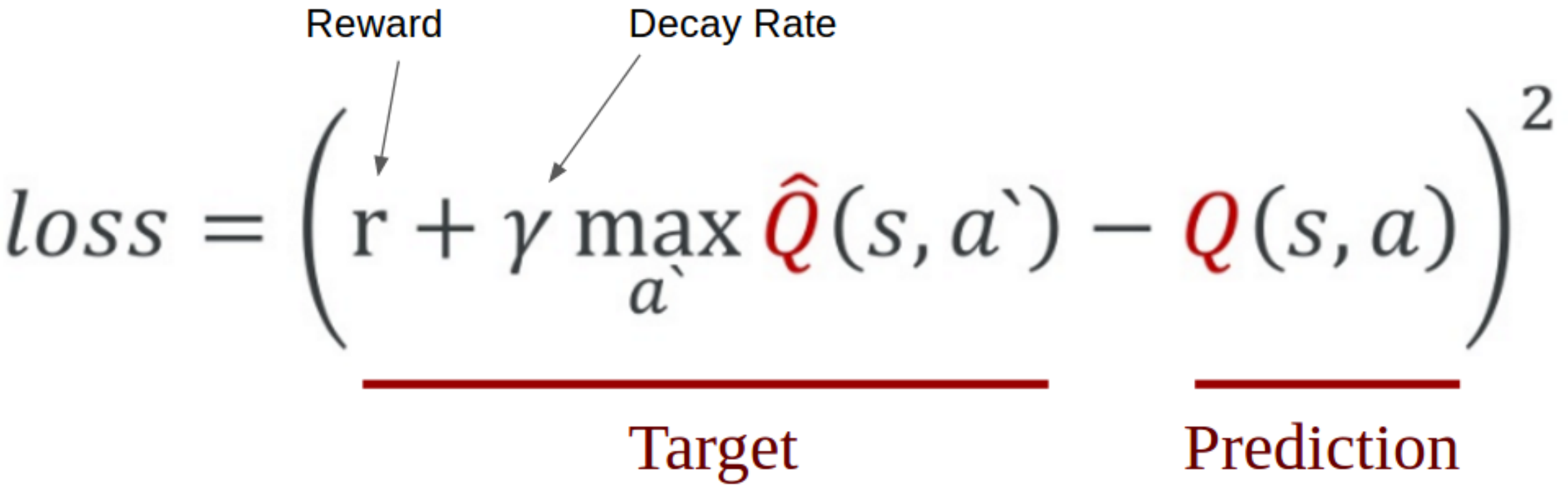
# Goal

- Learn to balance the pole

# Solution

- Play a bunch of episodes
- Learn a function:
  - Input: (state)
  - Output: (action to take in this state)

# Loss


Reward
Decay Rate
$loss = \left( r + \gamma \max_{a`} \hat{Q}(s, a`) - Q(s, a) \right)^2$
Target
Prediction

# This “function”

- Is a neural network

# Demo