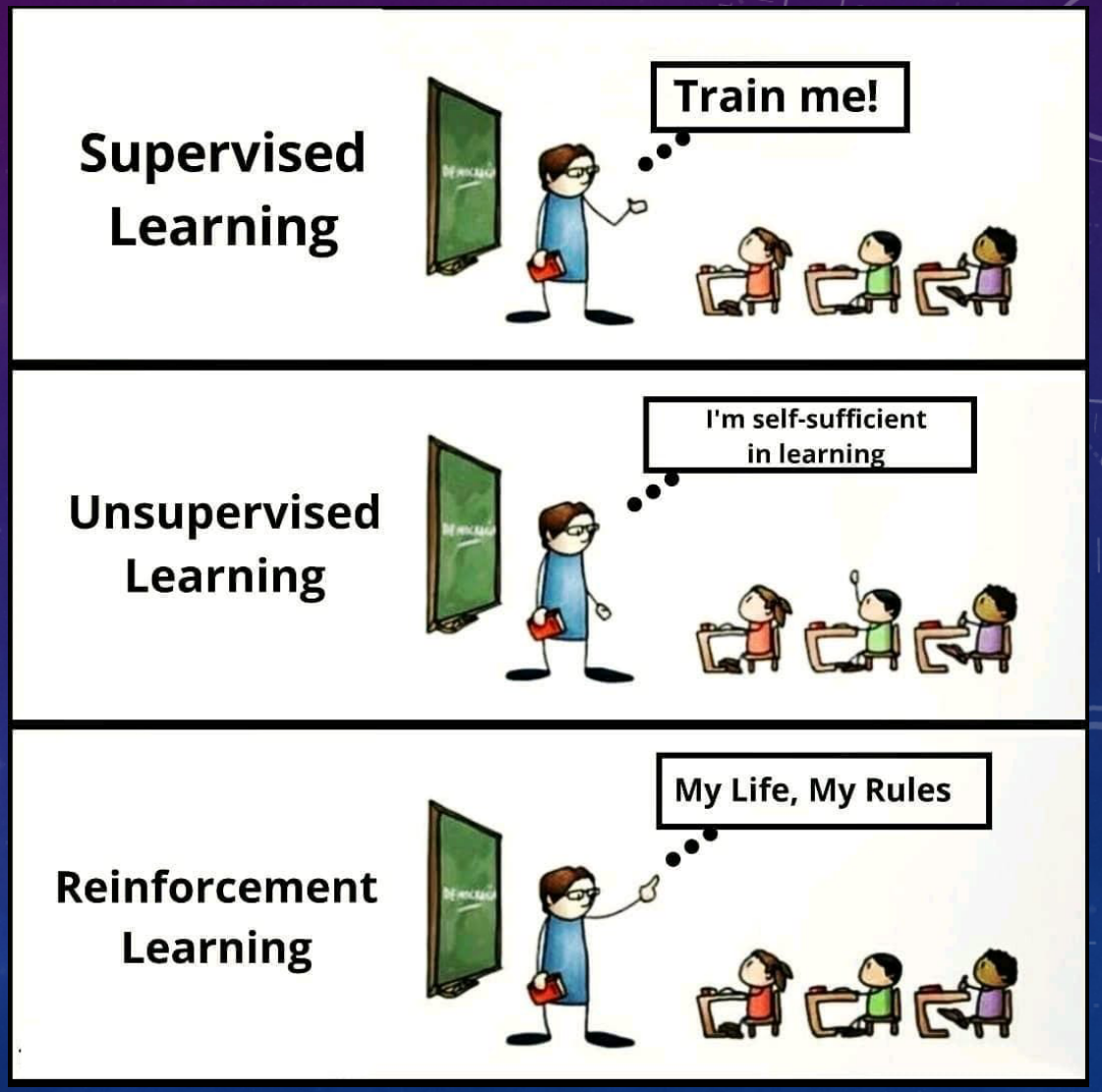
Reinforcement Learning

UCSB Robotics, Winter 2021 | Alex Mei

DAILY RUNDOWN

- What is Reinforcement Learning?
- Reinforcement Learning Use Cases
- Project Sprint 1 Retrospective
- Project Sprint 2 Work Time

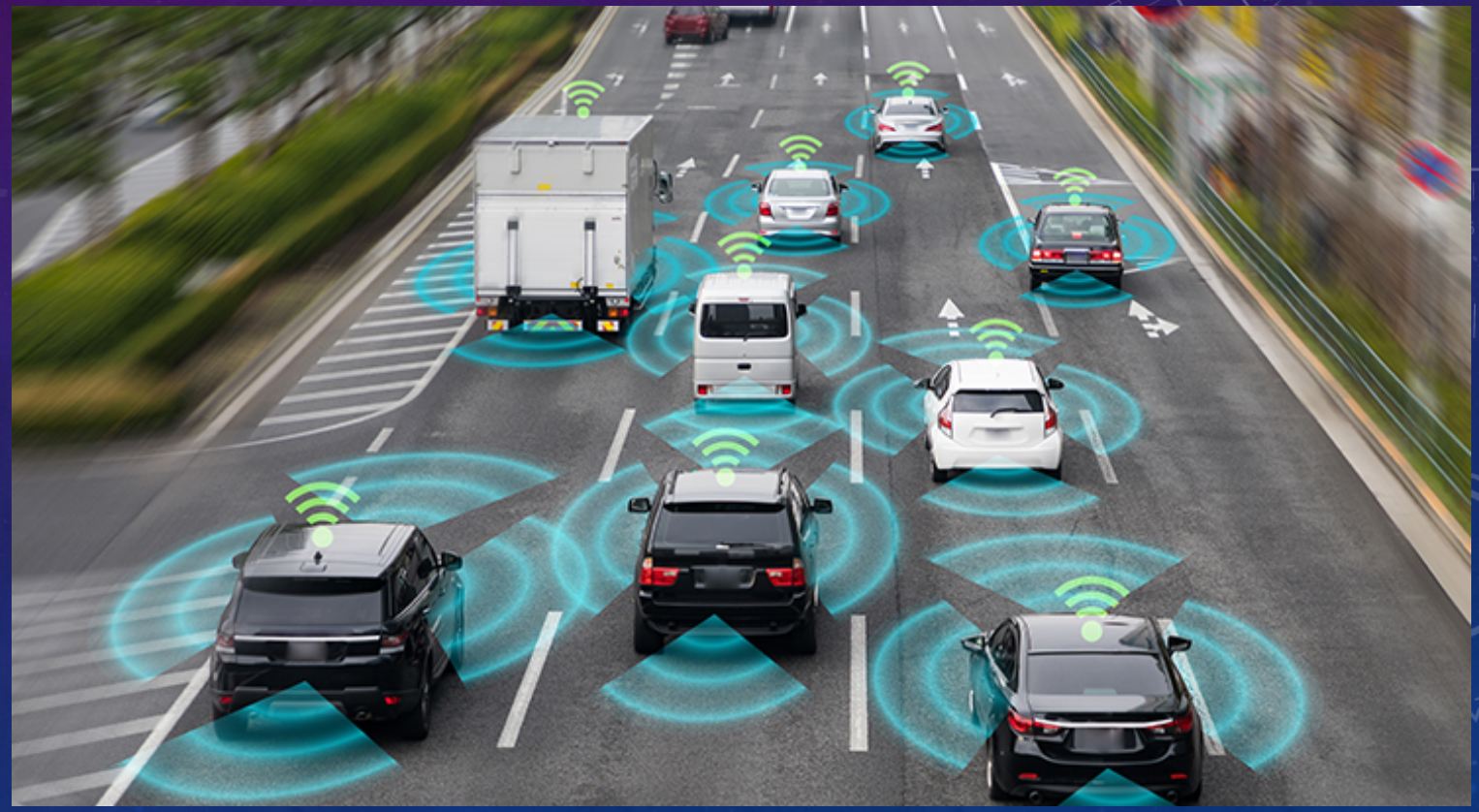


THE IDEA

- Learn the dominant strategy through trial and error
- Reward for success; penalty for failure
- Goal: maximize reward and minimize penalty

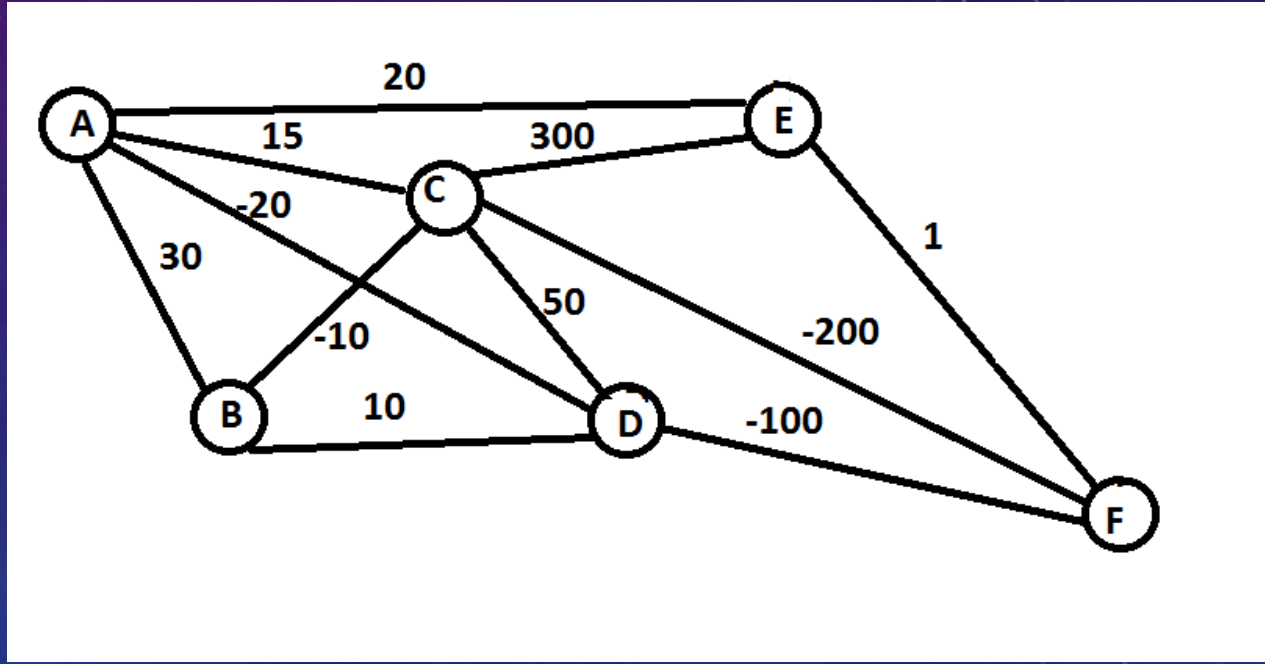
PRACTICAL EXAMPLE: AUTONOMOUS VEHICLE

- Safety and Regulations
- Ride Time
- Environmental Impacts

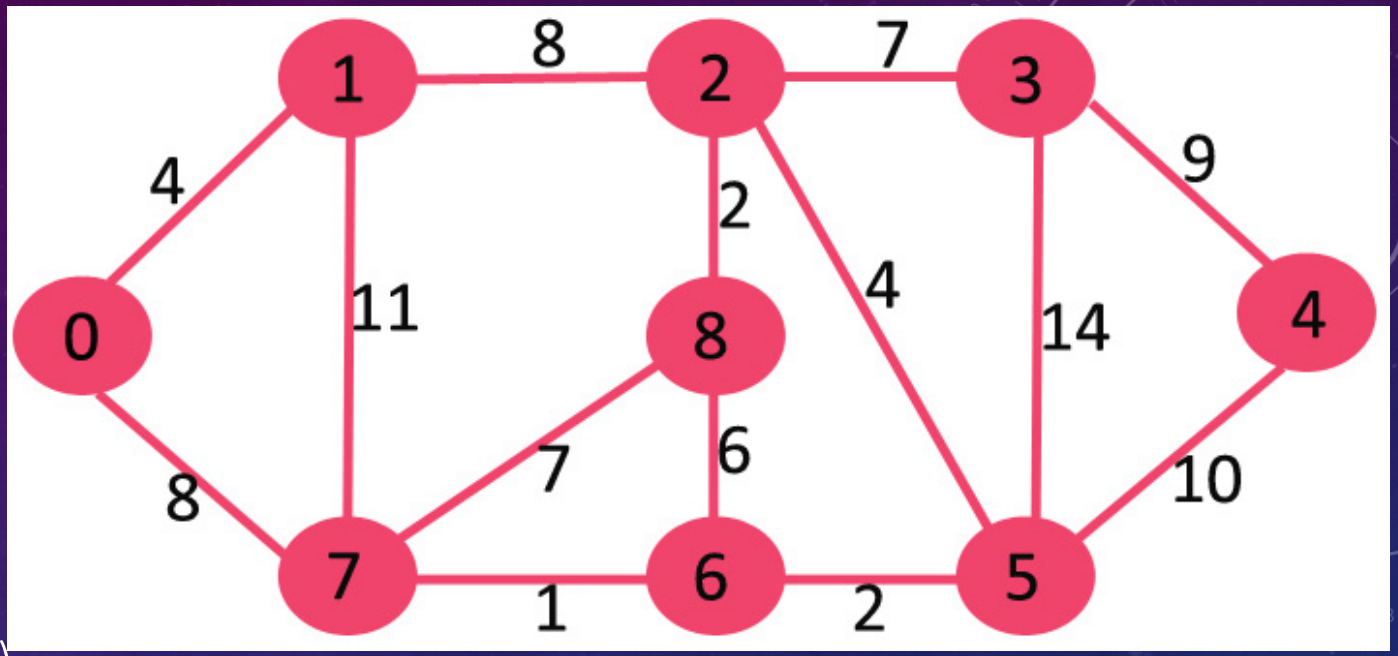


REDEFINE THE PROBLEM AS A GRAPH

- Graph G has S states and T transitions
- Maximize reward from A to F
- States denote states of problem
- Transitions denote reward/penalty



DIJKSTRA'S ALGORITHM



- Start at source state
- Initialize distance as infinity (source = 0)
- Choose smallest state not already chosen and travel along its transitions. If transition results in smaller distance to those vertices, update with smaller number.
- Repeat until all states have been reached

THE SOLUTION

- Single source shortest path = optimal solution for reinforcement learning
- States are trivial to determine based on the problem
- How do we determine the transition values?

TRANSITIONS

- Does the transition exist? (Can we get from one state to the next state?)
- How valuable is the transition?
  - Leads to dead end = penalty
  - Leads to success = reward
- Train over many many examples

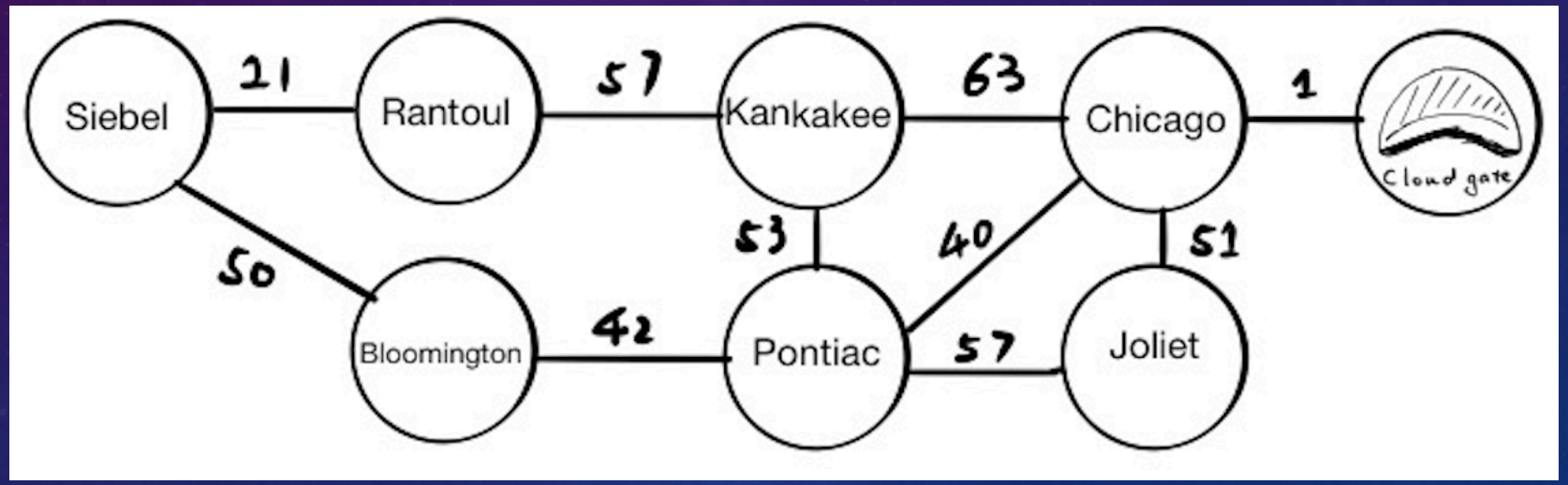
STORING YOUR GRAPH

- One idea: 2D List with entries being transition values
- Another idea: List of Linked Lists with Linked List storing transitions and values
- To "save" your training, format into txt or csv using File I/O

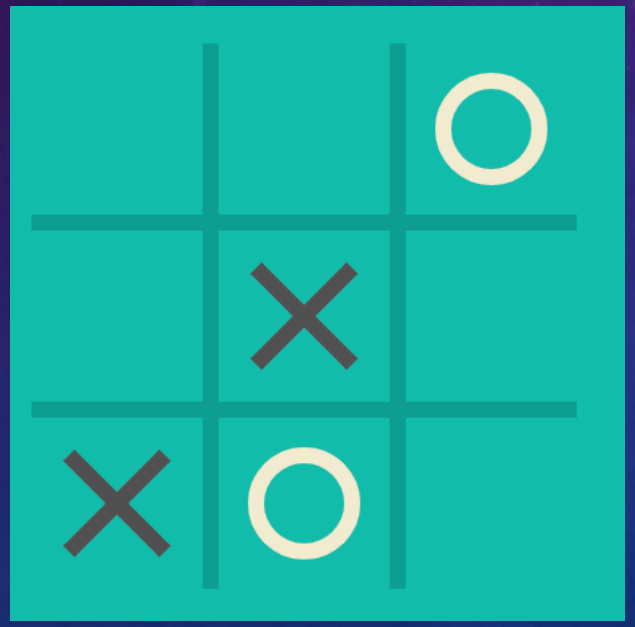
USE CASES

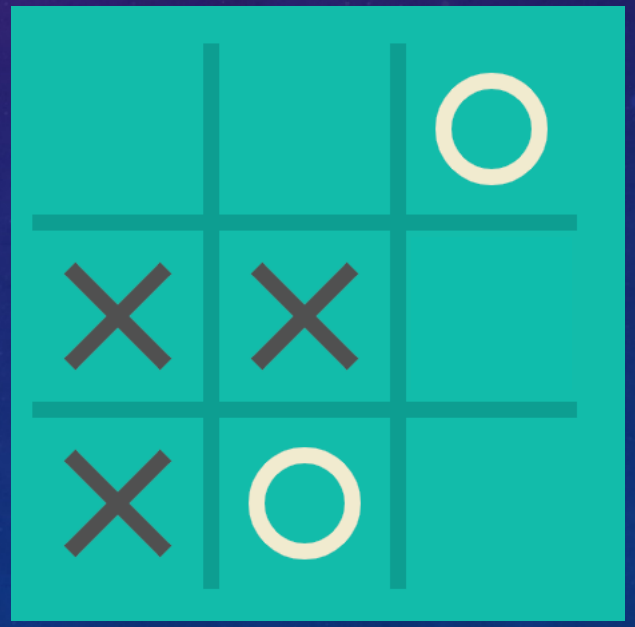
- Many classic games, puzzles, etc.
- Functioning Robotics
- Anything with finite states

USE DIJKSTRA'S ALGORITHM TO DETERMINE THE OPTIMAL PATH YIELDING THE MAXIMUM REWARD:



IDENTIFY THE NUMBER OF STATES A TIC-TAC-TOE BOARD HAS. (UPPER BOUND IS FINE). THEN, IDENTIFY A PLAUSIBLE TRANSITION REWARD/PENALTY FROM LEFT TO RIGHT:





PROJECT SPRINT 2 WEEK 1

- Wrap up project sprint 1; perform sprint review and retrospective
- Perform Daily Standup in groups (ideally, document this)
- Project Time! :)