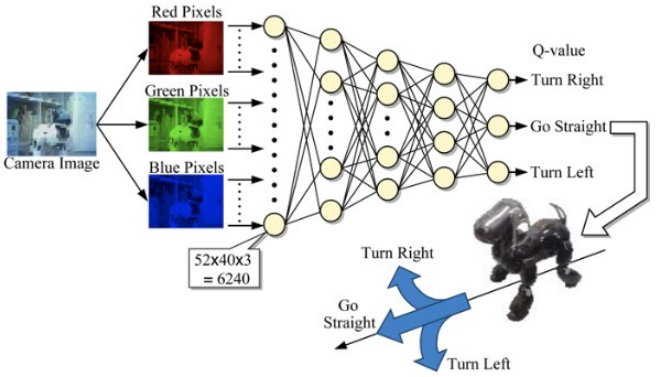
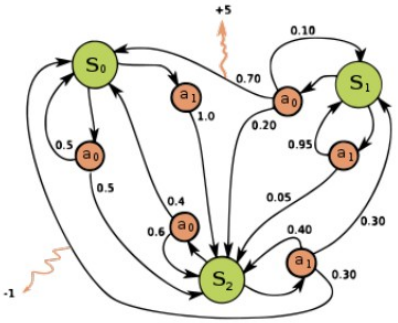
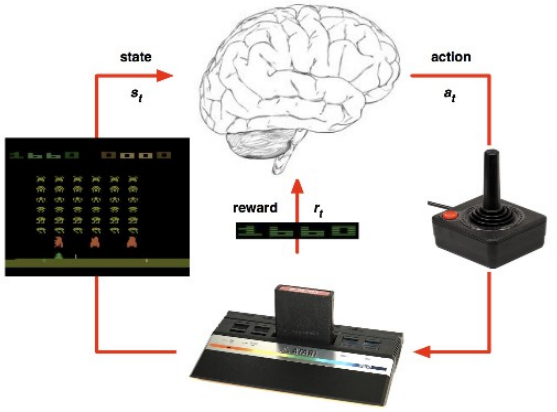
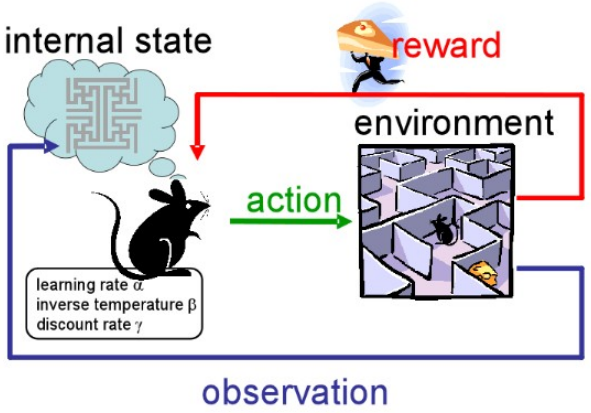
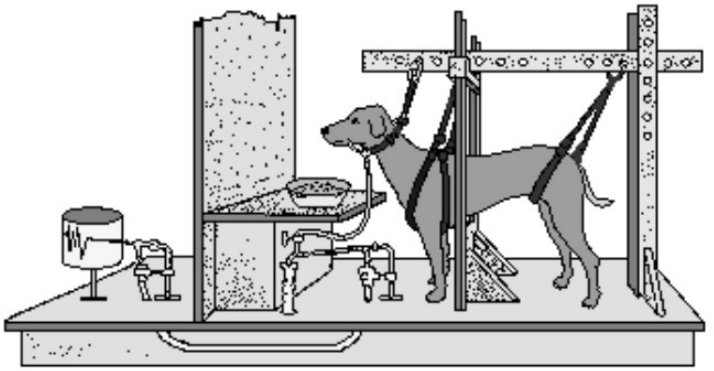
COMP 138: Reinforcement Learning











Instructor: Jivko Sinapov
Webpage: https://www.eecs.tufts.edu/~jsinapov/teaching/comp150_RL_Fall2020/

Announcements

Reading Assignment

- Chapter 12 of SB
- Research Article – see canvas
- Responses should discuss both readings
- You get extra credit for answering others' questions!

Reading Assignment

Sutton, Richard S., Doina Precup, and Satinder Singh.

"Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning."

Artificial intelligence 112.1-2 (1999): 181-211.

Project Proposal

- Due this Friday

We’re hiring!

- Project Title: “Science of Artificial Intelligence and Learning for Open-world Novelty”
- TLDR: how can AI agents deal with dynamically changing (“open”) worlds?
- Project expected to run for 2+ years, involves 3+ PhD students, several MS and undergraduates + partners at ASU
- If interested, send me an email with resume and we’ll start the conversation


DigMinecraft.com


RYANOPOLY
A PROPERTY TRADING GAME DESIGNED BY DANI
COMMUNITY CHEST
CHANCE
GO
WORK
A WORK EMAIL
JUST VISITING


pygame window
Server: 32 FPS
Client: 60 FPS
Vehicle: Nissan Patrol
Map: Town03
Simulation time: 8:06:10
Speed: 6 km/h
Heading: -86° NW
Location: (-77.8, 28.5)
Height: 0 m
Throttle:
Steer:
Brake:
Reverse:
Hand brake:
Manual:
Gear: 2
Collision
Number of vehicles: 101
Nearby vehicles:
24m Volkswagen T2
24m Mercedes-Benz Coupe
26m Audi Tt
30m Harley-Davidson Low R.
34m Bmw Isetta
44m Toyota Prius
44m Audi Tt
44m Bmw Grandtourer
67m Kawasaki Ninja
68m Seat Leon
72m Dodge Charger Police
73m Volkswagen T2
76m Yamaha Yzf
CARLA

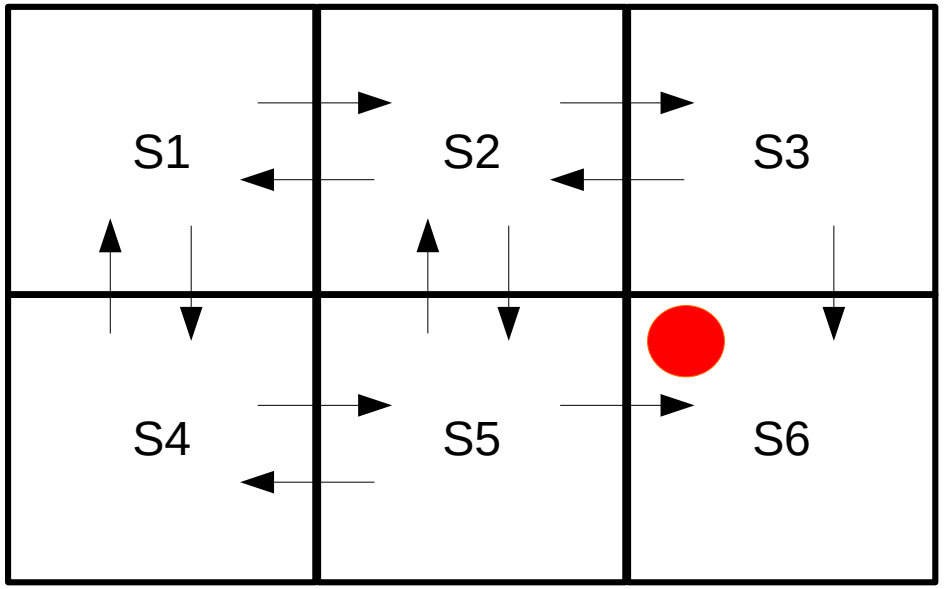
Midterm Evaluations

- At end of class

Today: Control with Function Approximation

$$Q(s,a) = \sum_{i=1}^{n} f_i(s,a) w_i$$

The limitations of Tabular Methods



+ 100 reward for getting to S6
0 for all other transitions

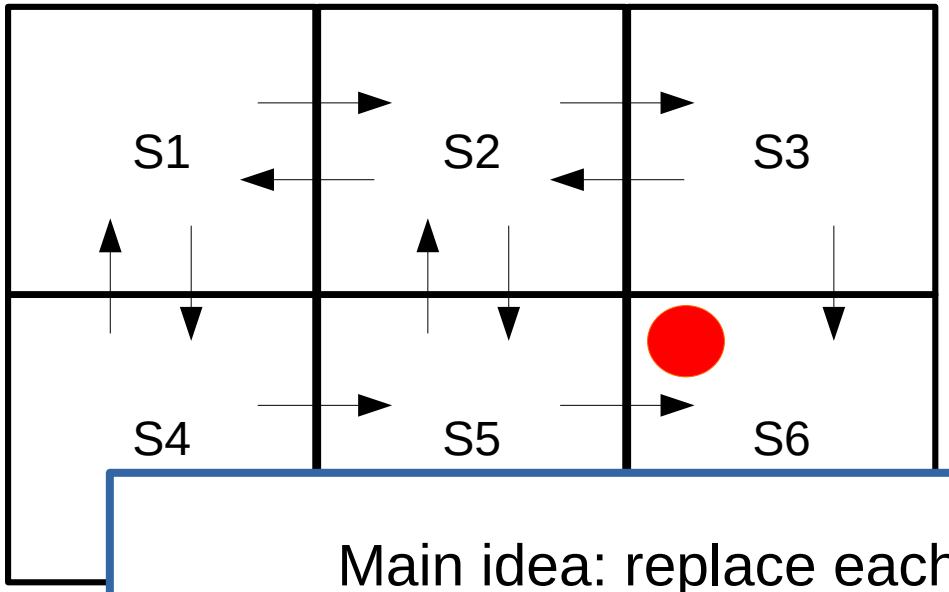
Update rule upon executing action a, ending up in state s' and observing reward r :

$$Q(s , a)=r +\gamma \max a' Q(s' ,a ')$$

γ = 0.5 (discount factor)

Q-Table

S1	right	25
S1	down	25
S2	right	50
S2	left	12.5
S2	down	50
S3	left	25
S3	down	100
S4	up	12.5
S4	right	50
S5	left	25
S5	up	25
S5	right	100



Main idea: replace each state-action pair with a feature vector

+ 1
0 for all other transitions

Q-Table

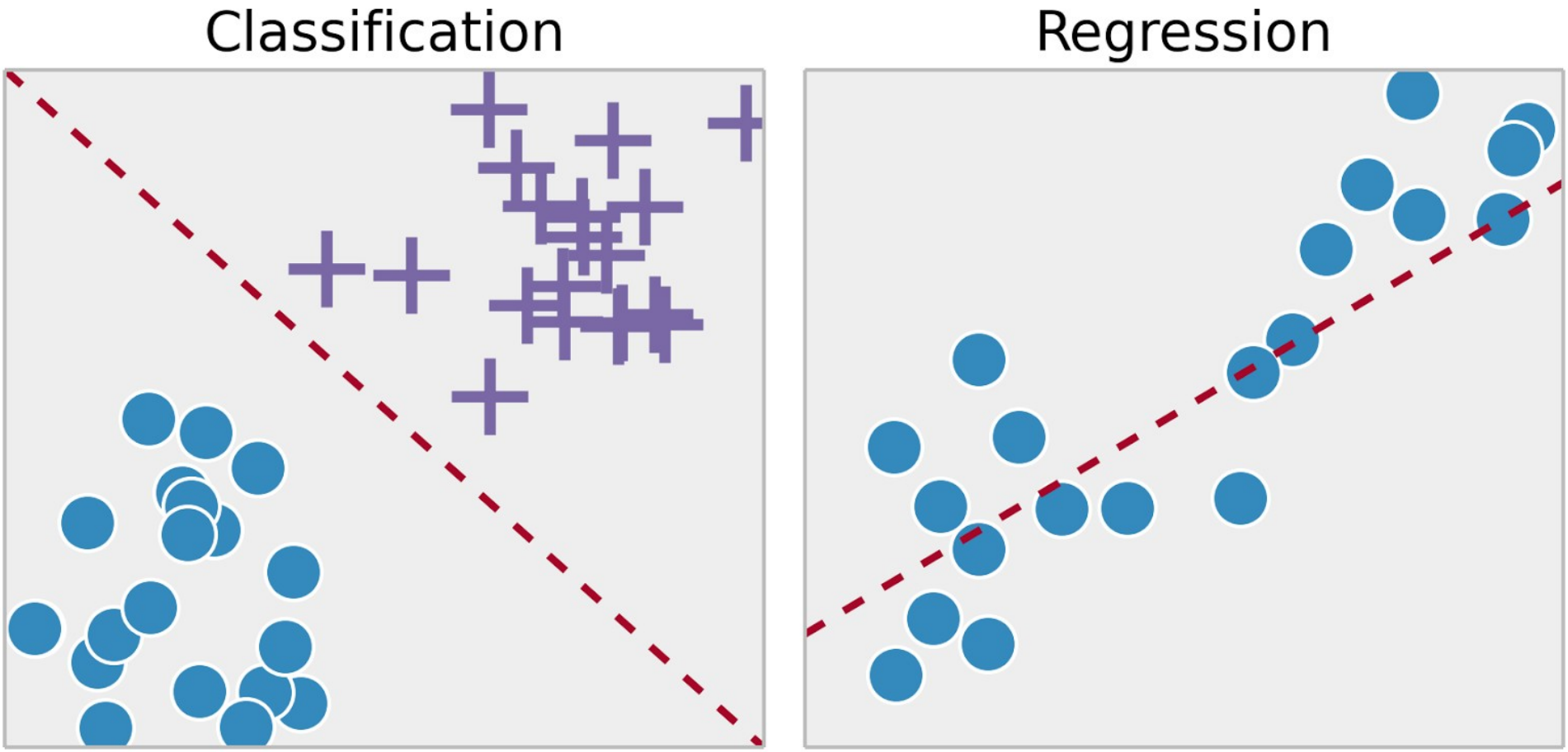
S1	right	25
S1	down	25
S2	right	50
S2	left	12.5
S2	down	50
S3	left	25
0.5, -0.7, 0.2, …, 0.9		100
S4	up	12.5
S4	right	50
S5	left	25
S5	up	25
S5	right	100

Update rule upon executing action a, ending up in state s' and observing reward r :

$$Q(s, a) = r + \gamma \max_{a'} Q(s', a')$$

$\gamma = 0.5$ (discount factor)

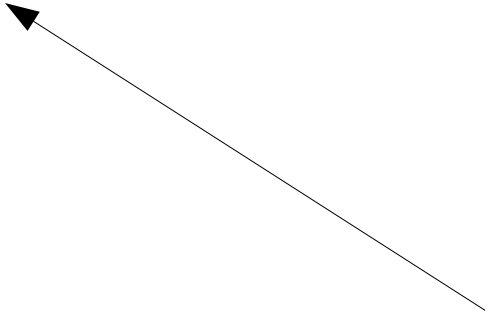
Connection to Supervised ML



[https://cdn-images-1.medium.com/max/1600/1*ASYpFfDh7XnreU-ygqXonw.png]

Linear Q-Function Approximation

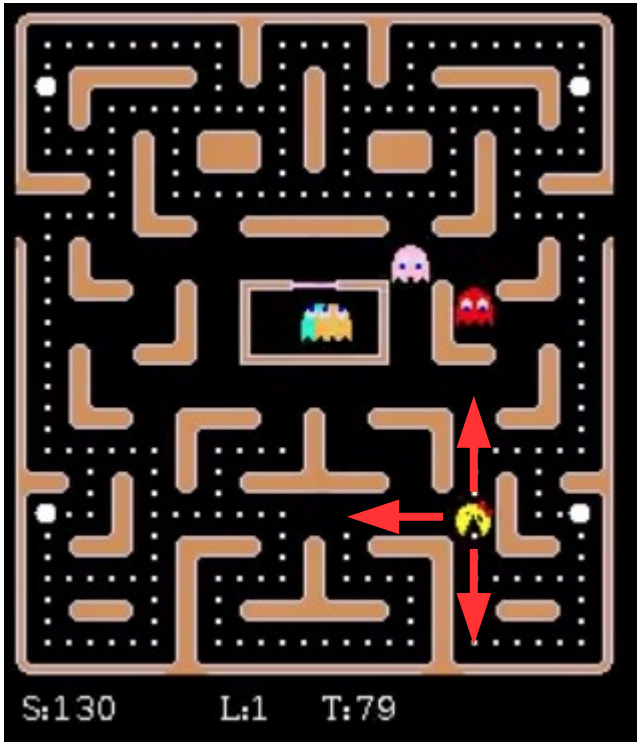
$$Q^*(s,a) = \mathcal{R}(s,a) + \gamma \sum_{s'} \mathcal{P}(s'|s,a) \max_{a'} Q^*(s',a')$$



$$w_1 * x_1 + w_2 * x_2 + \ldots + w_n * x_n$$

$$Q(s,a) = \sum_{i=1}^{n} f_i(s,a) w_i$$

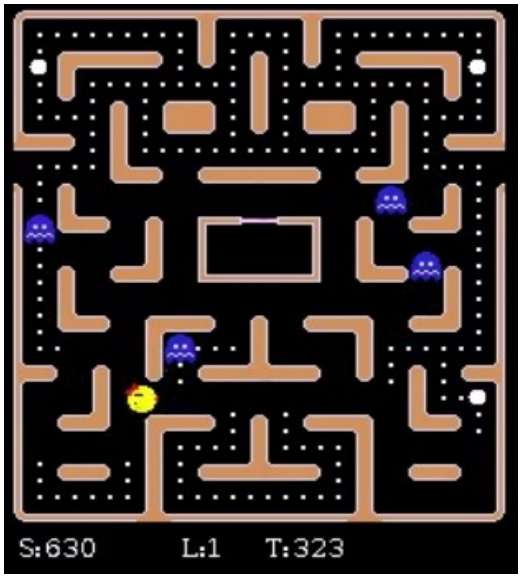
Example: Ms. Pac-man

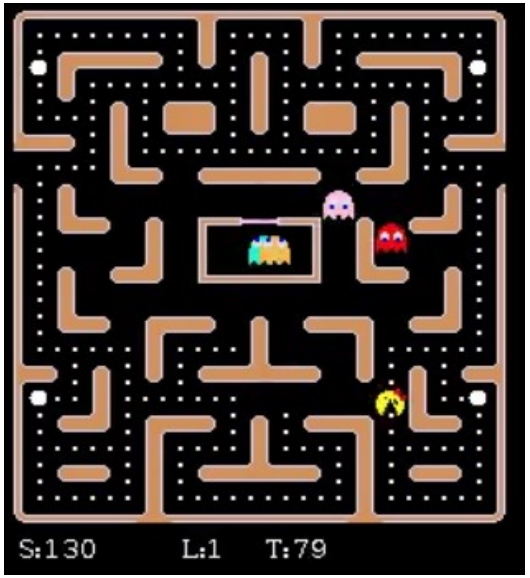


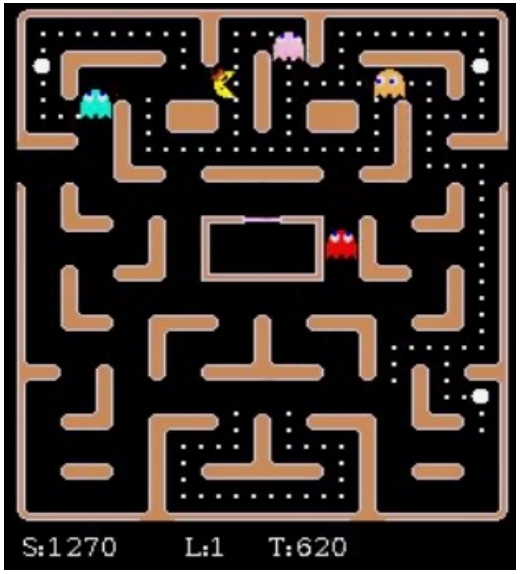
The problem: for a given action and the current configuration, compute a fixed-length feature vector

Each feature must have some semantic “meaning”

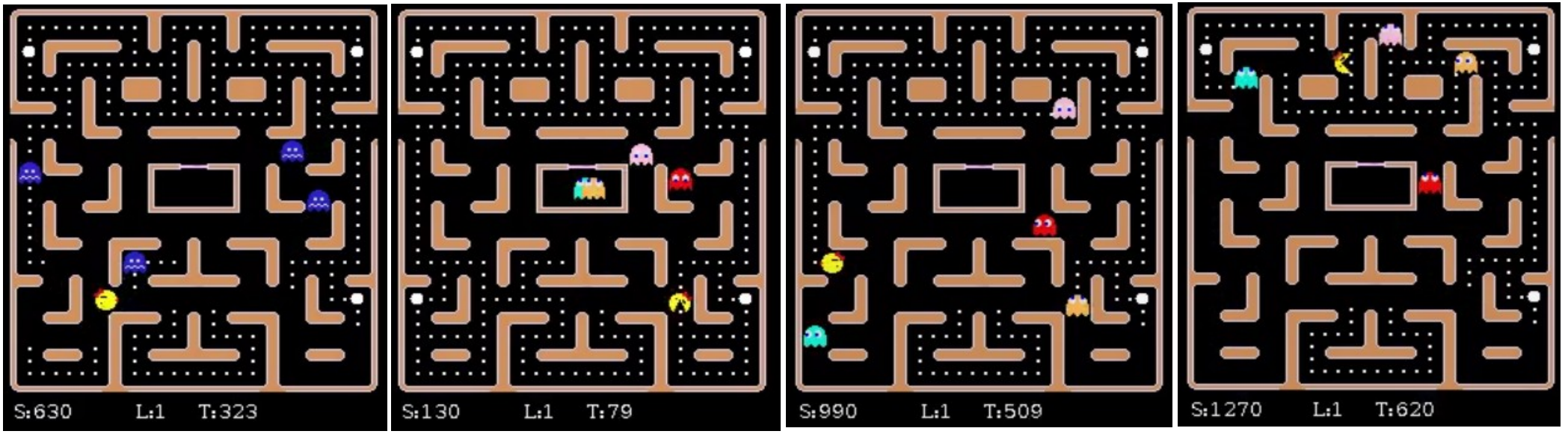
Example Configurations







Small group activity: feature engineering



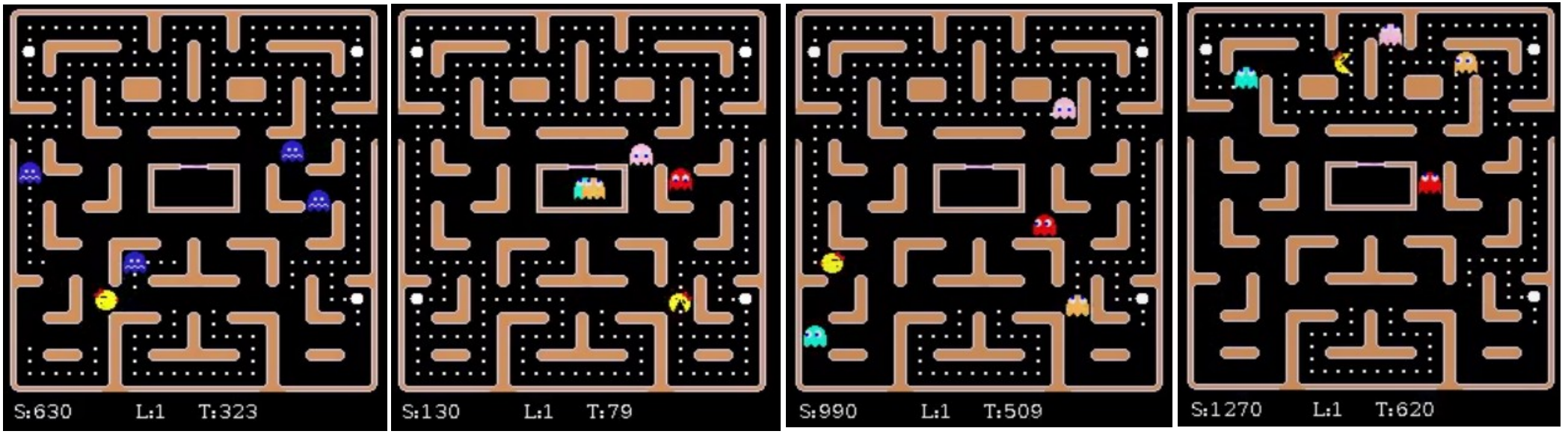
Be the feature engineer: given a configuration and a cardinal direction, design the feature types that describe how the world “looks like” in that direction; assume you have access to the underlying game simulator; the board itself is a graph with nodes and edges and for each node, you know whether there is a pill, power, pill, a ghost, and its state (edible or not, direction of movement)

Example feature: $x_{ghost\text{-}k}$ = 0.0 if no ghost is present up to *K* nodes towards the action’s direction and 1.0 otherwise

Be as precise as possible!

Assume linear q-function approximation – can you come up with an initial set of weights given the semantics of the features you designed?

Discussion – what did you come up with?



Be the feature engineer: given a configuration and a cardinal direction, design the feature types that describe how the world “looks like” in that direction; assume you have access to the underlying game simulator; the board itself is a graph with nodes and edges and for each node, you know whether there is a pill, power, pill, a ghost, and its state (edible or not, direction of movement)

Example feature: $x_{ghost\text{-}k}$ = 0.0 if no ghost is present up to *K* nodes towards the direction and 1.0 otherwise

Be as precise as possible!

Assume linear q-function approximation – can you come up with an initial set of weights given the semantics of the features you designed?

Example Q-Learning Update with Function Approximation

Non-linear Function Approximation

Paper Discussion

“I was a little confused about how the replay memory works with the agent as the deep Q-learning runs. What is the difference between this and bootstrapping?”

- Noah

Paper Discussion

“How would CNN architecture complexity compare to modern games where there is alot more things going on visually than in the atari games?”

- Frederick

Paper Discussion

“A question that I have about the paper is that since a state is defined by a sequence of actions and observations, how do we determine the length of such a sequence? Does it make more sense to set the length to a fixed value or to have it vary dynamically?”

- Martin

Paper Discussion

“I guess the next step is then how far can we take the model-free model, and whether this sort of neural network could surpass the performance of RL algorithms in any other problems?”

- Jonathan

Paper Discussion

“Q: This paper has been written in 2013, what have been the changes in the state-of-the-art methods that take as input only the raw pixels?”

- Camelia

Reading Discussion

Reading Discussion

“… so does being able to see n-steps ahead allow for the algorithm to be less likely to get stuck in the local minimums?”

- Courtney

Reading Discussion

“Q: Could you explain a bit more in detail how they use grid-tilings in example 10.1?

Q: Could you go over the concepts of quality of a policy, r(π), and ergodicity? The way the authors explain them in section 10.3 is quite confusing for me.”

- Camelia

Reading Discussion

“Could you go a bit more in depth on differential semi-gradient sarsa and how it's different from regular sarsa? The book is very brief on this algorithm.”

- James

Project Breakout

- Finalize domain / problem you want to address
- Decide on individual responsibilities
- Write questions for me

Midterm Evaluations

THE END