Acknowledgements

This material includes content adapted from instructional resources made available by David Silver as part of his Reinforcement Learning course at UCL under CC-BY-NC 4.0.

Refer to https://www.davidsilver.uk/teaching/ for full details.

MONASH University
Information Technology

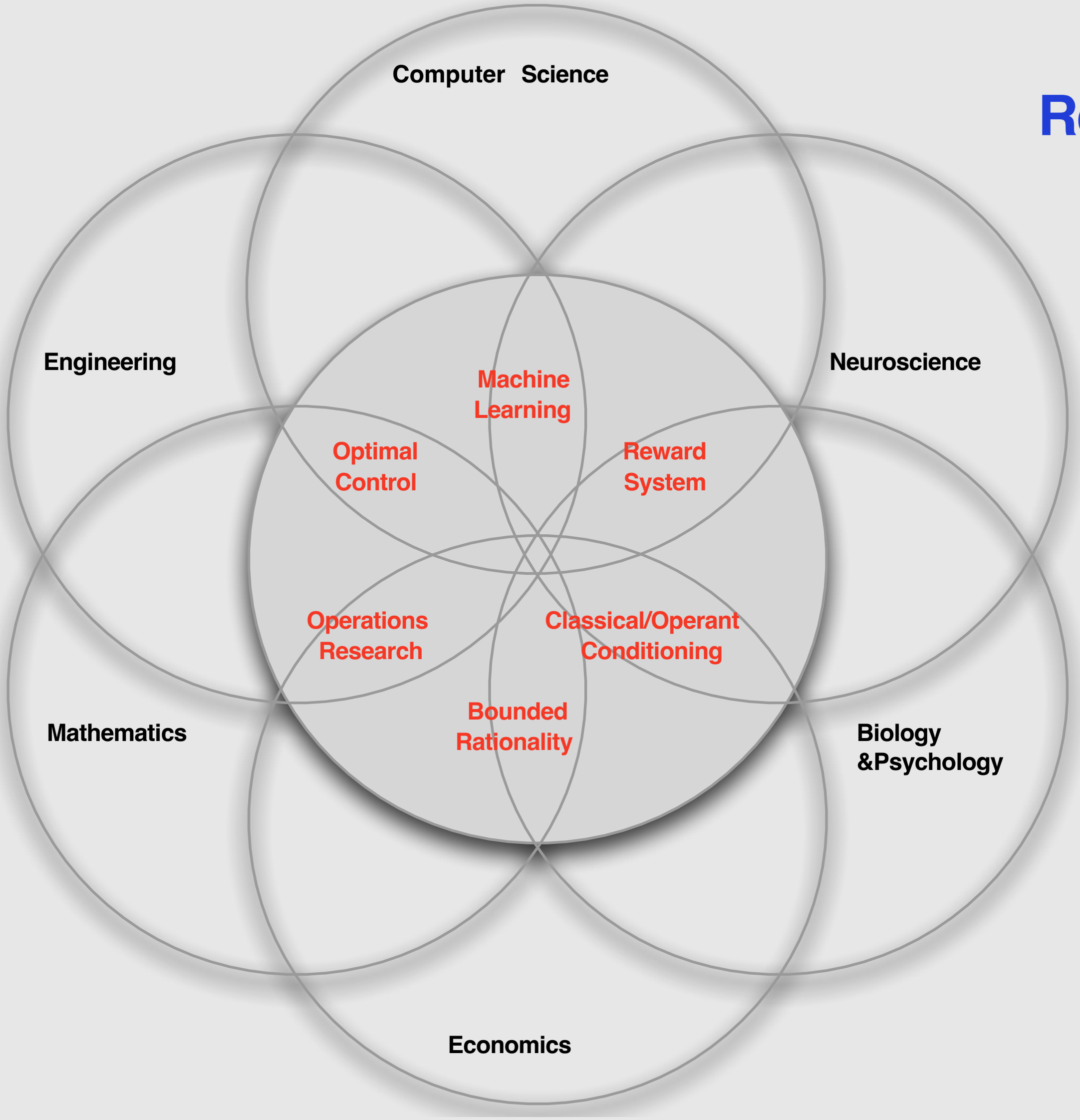
FIT5226

Multi-agent System & Collective Behaviour

Wk 3: Reinforcement Learning - Introduction

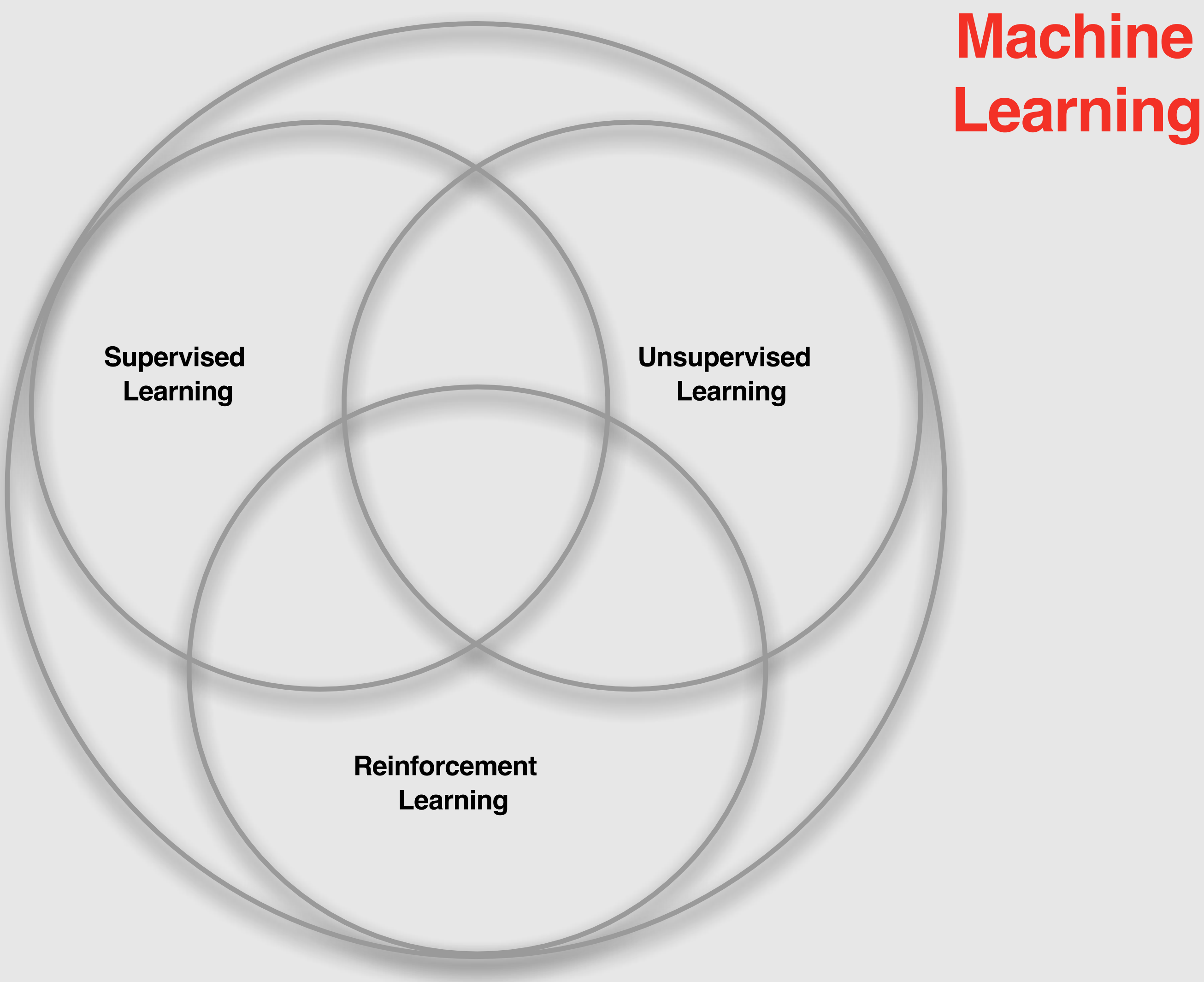
Bernd Meyer, July 2024

Reinforcement Learning

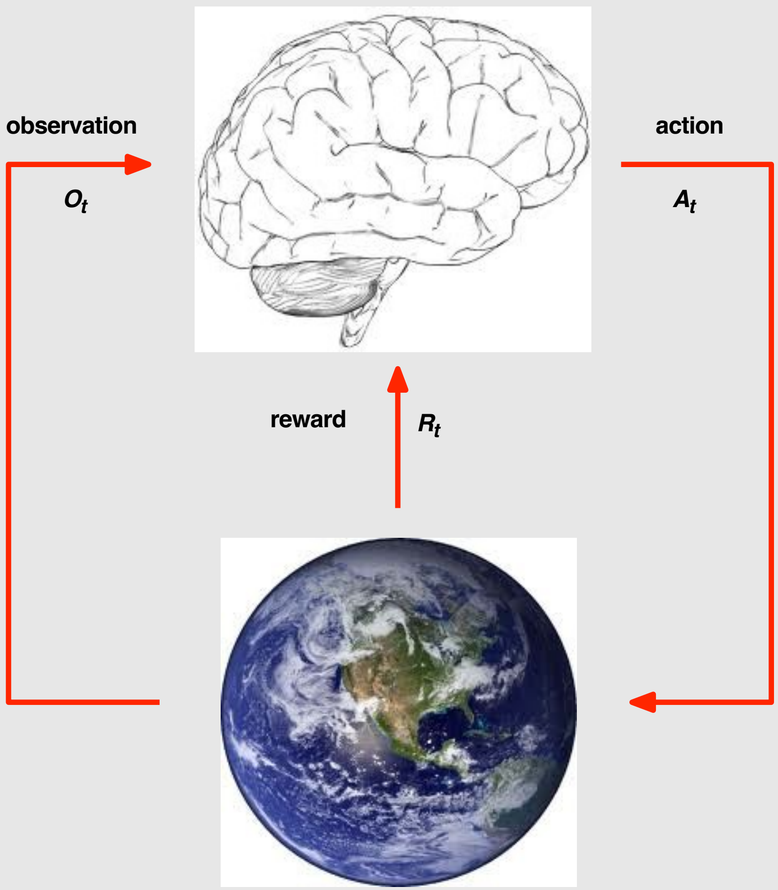


Reinforcement Learning

Reinforcement Learning



RL: Learning by Trial & Error



In each step the agent

- performs an action
- receives a reward
- environment changes

environment is not directly visible to agent

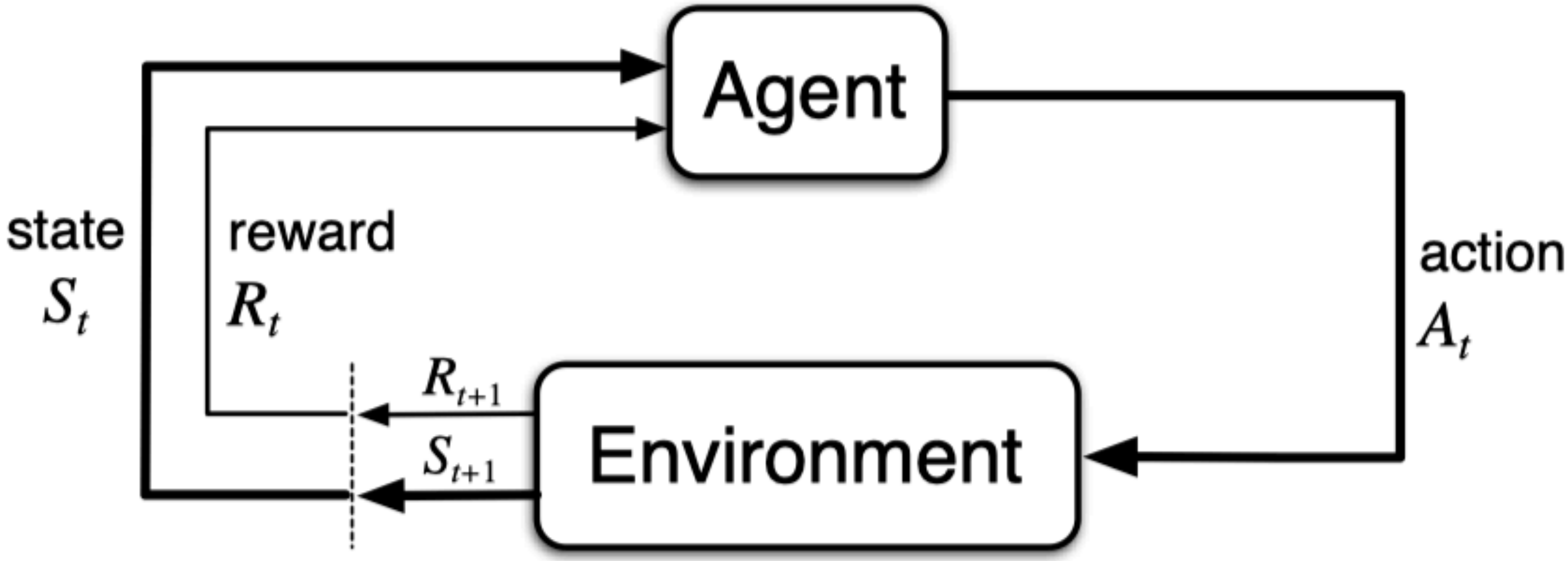
agents learns an internal representation of the environment based on the reward (experience)

Learning the Environment by Trial & Error

- the environment is not known and needs to be learned
- no supervisor knowledge
- environment can only be learned through the reward
- actions change the environment
- from the agent perspective the environment is dynamic and stochastic
  - dynamics captured in a (discrete) "state" of the environment,
  - stochasticity captured in probabilistic state transitions.

- In MAS, the (unknown) actions of others also change the environment.
  - we will initially focus on the single-agent perspective
  - ie. we are treating the others as a part of the (unknown) environment

Action-Cycle



- Time t: agent finds environment in state S_t, picks & executes action A_t
- environment changes in response to A_t from S_t to S_{t+1}
- Time $t+1$: agent finds itself in state S_{t+1} and receives reward R_{t+1}

Markov Property

We are talking about Markov systems.

Reminder: A system is *markov* if for all states

$$P(S_{t+1} \mid S_t) = P(S_{t+1} \mid \ S_1, \ldots, S_t)$$

The future is independent of the past given the present

i.e. history can be forgotten iff the current state is known

Goal of the Agents

the goal of the agent is to

- learn their environment
- while maximising their (cumulative) reward

- Need to balance
  - exploration of the environment
  - exploiting the environment

Exploration vs Exploitation

- Restaurant Choice
- Online Banner Advertisements
- Oil Drilling
- Game Playing

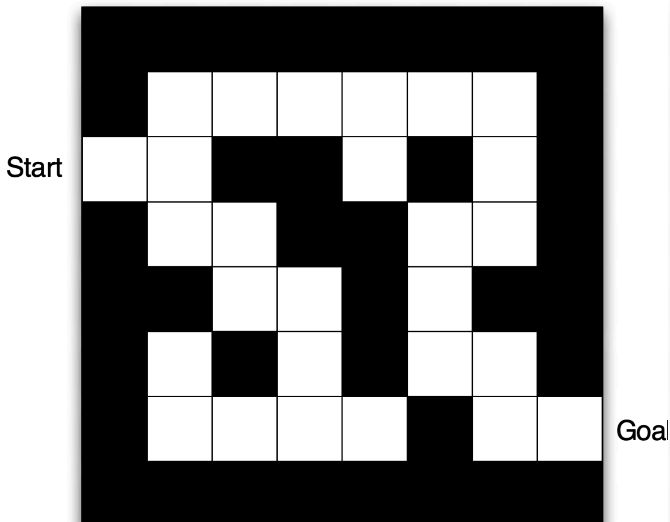
Components of Agents

An RL agent may represent any or all of these explicitly:

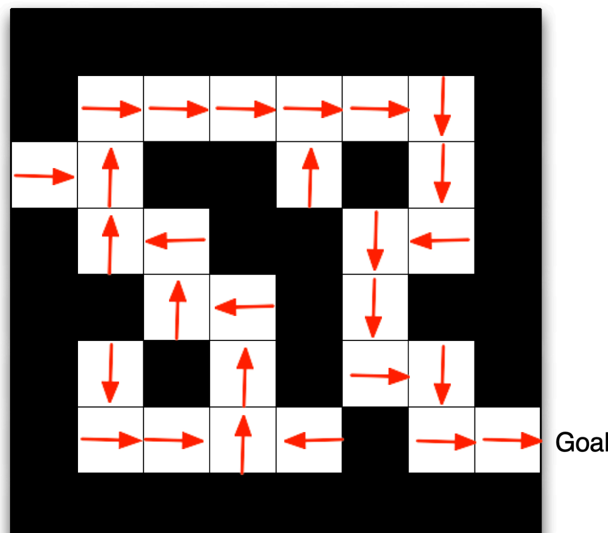
- **Policy**: agent's behaviour/decision function
- **Value function**: how good is each state and/or action
- **Model**: agent's representation of the environment

A First Example: Navigating a Maze

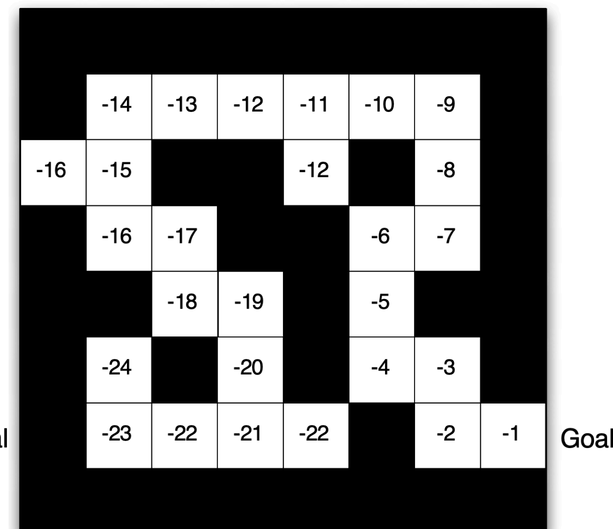
- States: locations (grid cell)
- Rewards: -1 per step
- Actions: up, down, left, right



Environment



Policy



Value Function

Policy

A policy defines the agent's behaviour

It is a map from state to action, e.g.

- Deterministic policy: $a = \pi(s)$

- Stochastic policy:

  $\pi(a \mid s) = P[\text{Action} = a \mid \ \text{state} = s]$

Value Function

The value function predicts the (discounted) future reward in a state given a policy

$$v_\pi(s) = \mathbb{E}_\pi[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots \ \gamma^\infty R_{t+\infty} \mid \ S_t = s]$$

Model

A model captures the environment

In our context, this means a model (*P*, *R*)

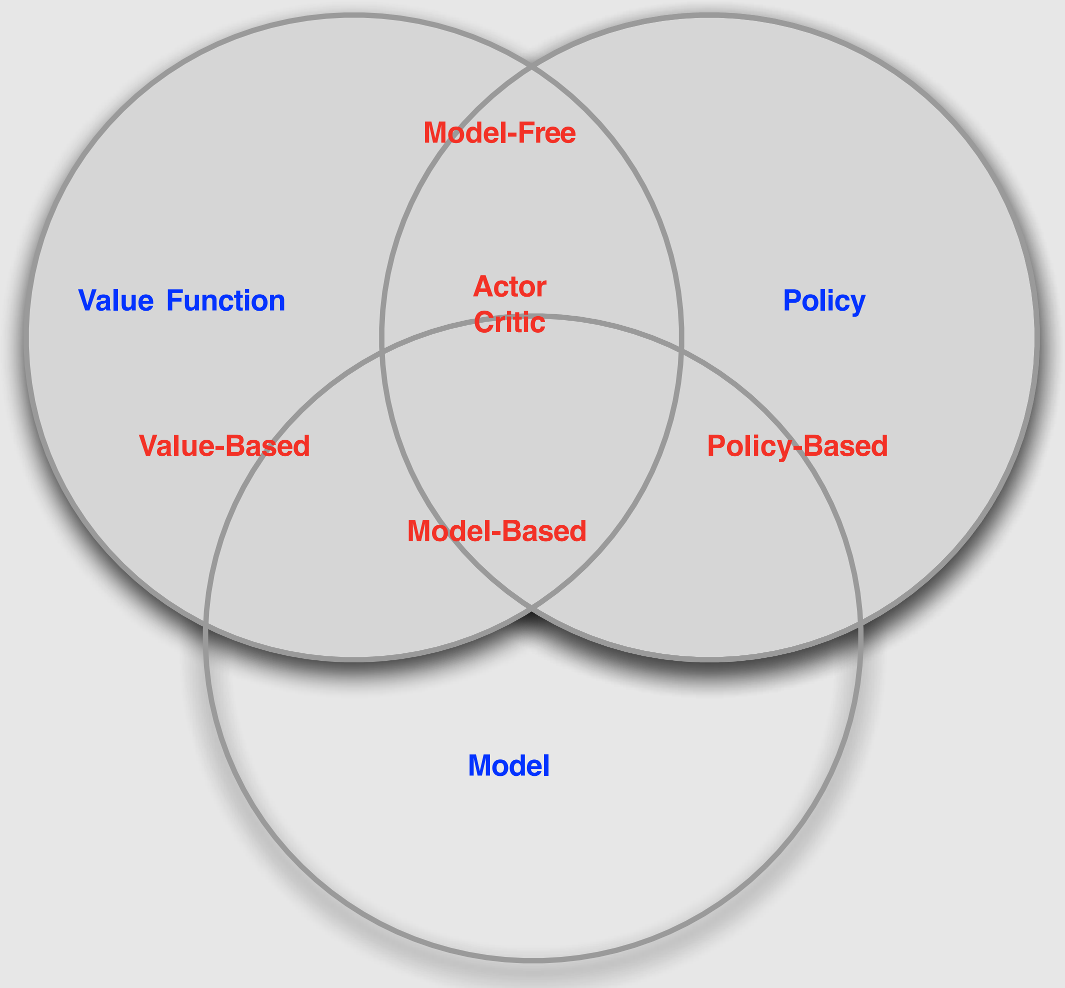
- *P* predicts the next state

$$P^a_{s,s'} = P[s_{t+1} = s' \mid s_t = s, a_t = a]$$

- *R* predicts the next reward

$$R^a_s = \mathbb{E}[R_{t+1} \mid s_t = s, a_t = a]$$

RL Agent Types



Markov Decision Process

Markov decision processes are a formal description of an RL environment

Describes a fully observable environment, i.e. the state completely characterises the process

Special forms, extensions:

- bandits: single-state MDPs
- continuous MDPs (control)
- partially observable environments (POMDP)

Reminder: Markov Chain

A Markov chain (S, P) is a memoryless process given by a sequence of random states

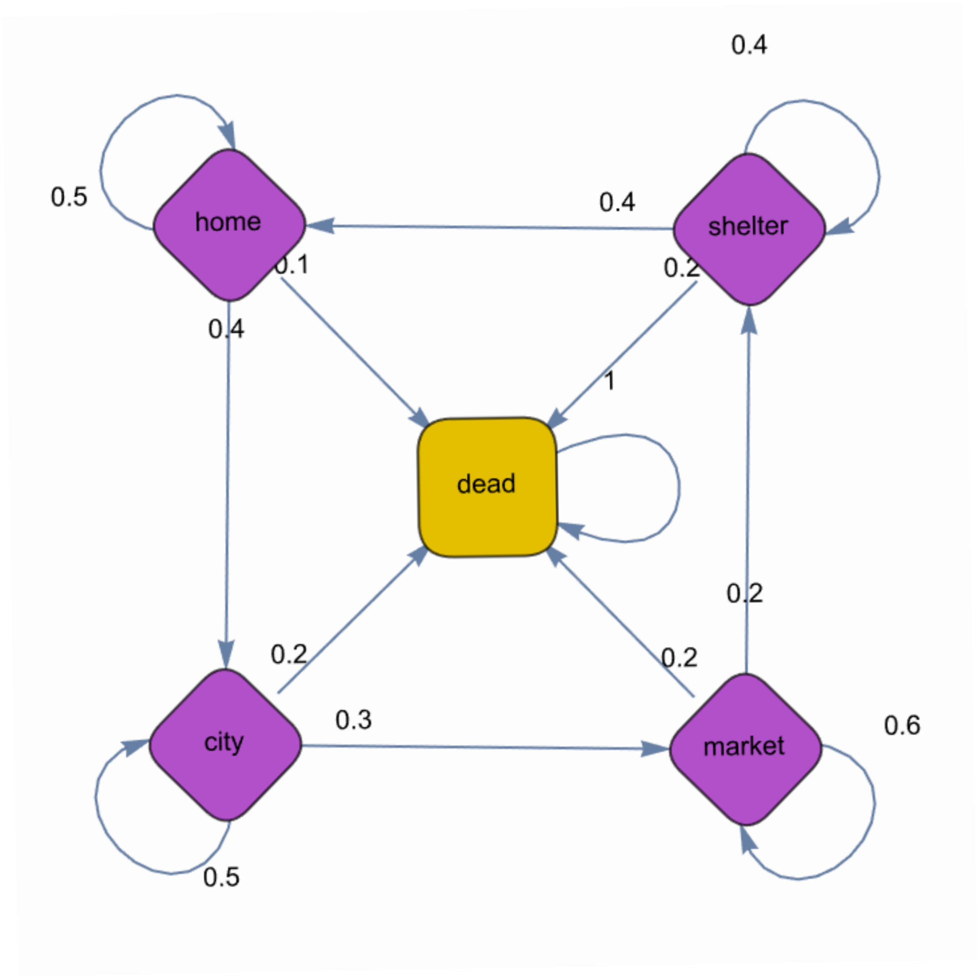
- S is a finite set of states and
- P is a stochastic matrix ie. all row sums are

$$\sum_j P_{i,j} = 1$$

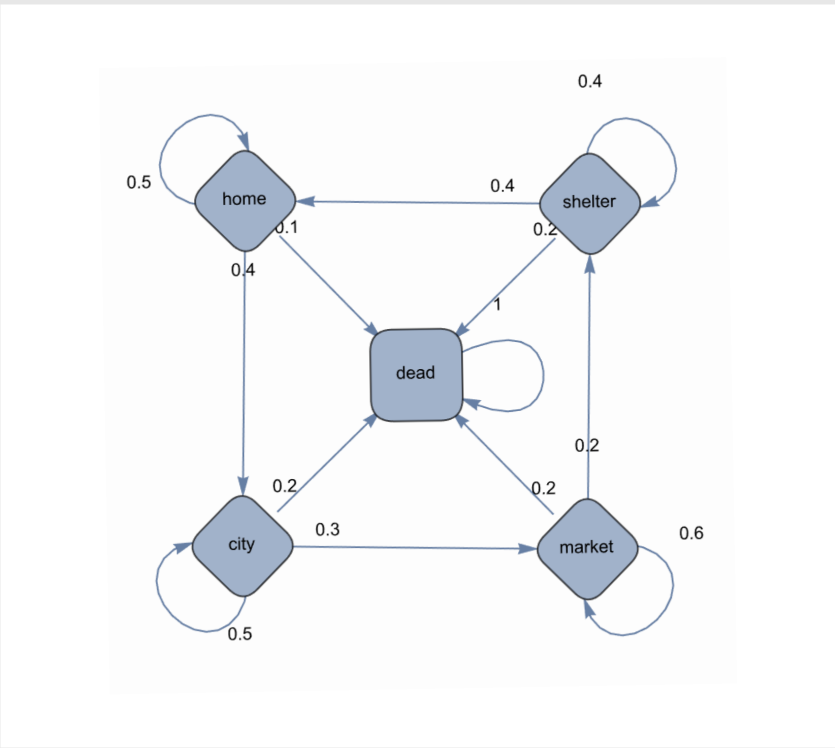
P describes the state transition probabilities, ie

$$P_{s,s'} = P[s_{t+1} = s' \mid \; s_t = s]$$

Example: The 7 lives of cats (MC)



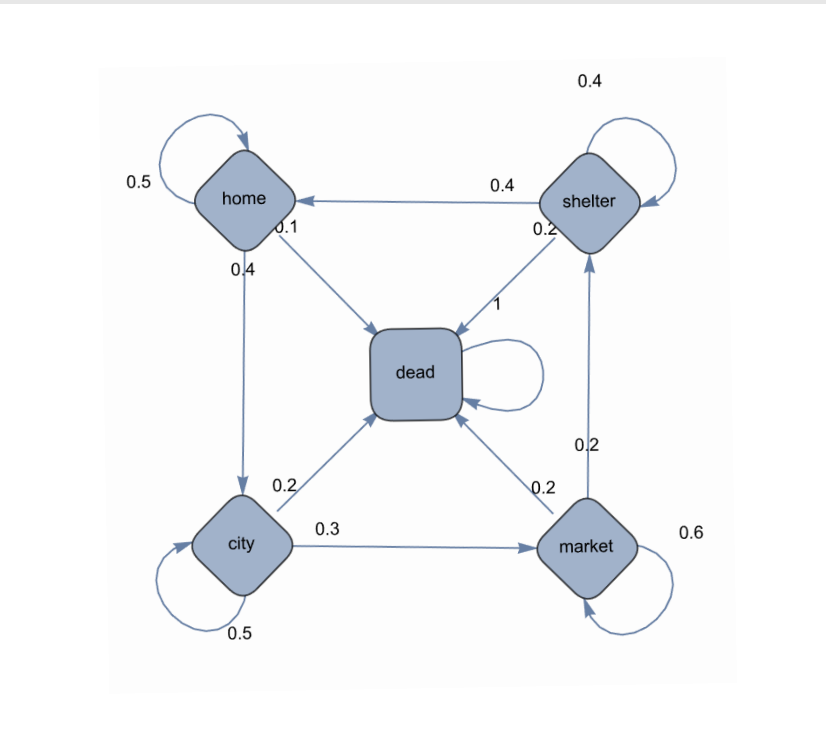
Example: The 7 lives of cats (MC)



Transition Matrix

	h	s	c	m	d
h	0.5	0	0.4	0	0.1
s	0.4	0.4	0	0	0.2
c	0	0	0.5	0.3	0.2
m	0	0.2	0	0.6	0.2
d	0	0	0	0	1

Example: The 7 lives of cats (MC)



Sample episodes for 7LoC Markov Chain starting from S_1 = home

$$S_1, S_2, \ldots, S_T$$

- {home, home, home, home, city, market, market, market, market, dead, dead}
- {home, dead, dead, dead, dead, dead, dead, dead, dead, dead, dead}
- {home, home, city, market, shelter, shelter, shelter, shelter, shelter, shelter, home}

Markov Reward Process

A Markov reward process (S, P, R, γ) is a Markov chain (S, P) with associated rewards

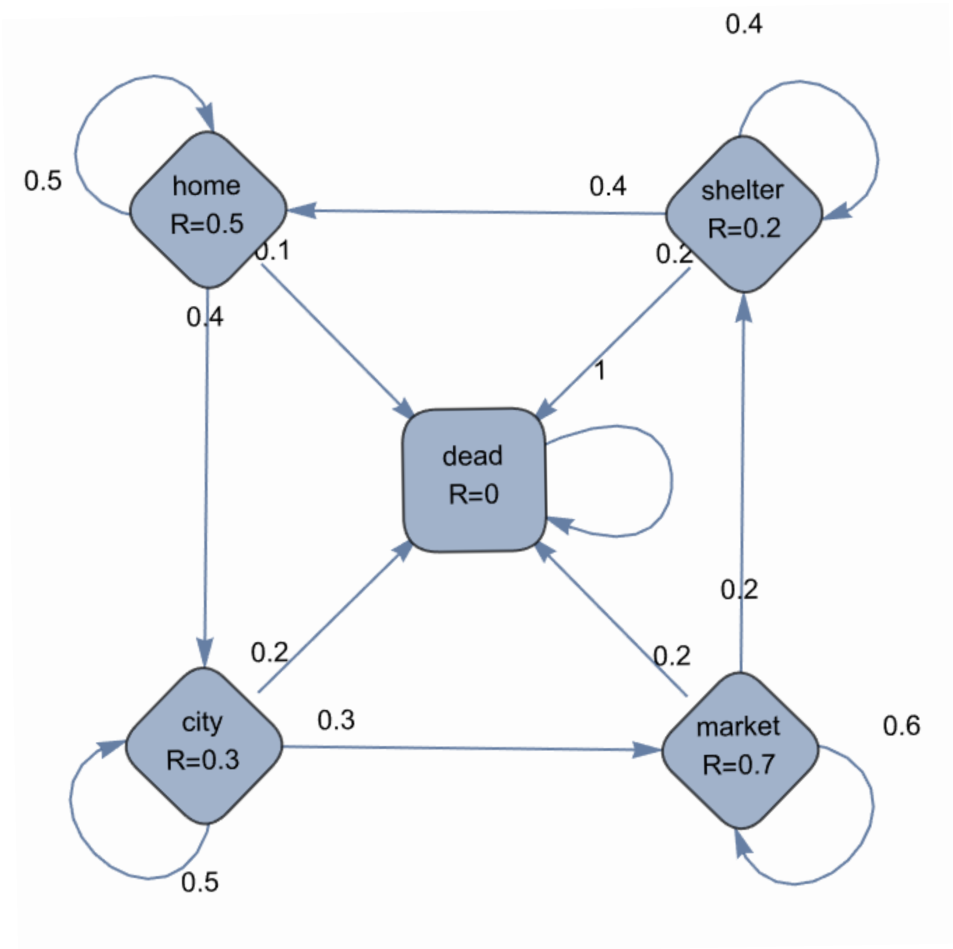
$$R(s) = \mathbb{E}[R_{t+1} \mid \; s_t = s]$$

and reward factor $0 \leq \gamma \in \mathbb{R} \leq 1$

P still describes the state transition probabilities, ie

$$P_{s,s'} = P[s_{t+1} = s' \mid \; s_t = s]$$

Example: The 7 lives of cats (MRP)



Return - don't just look at the instantaneous reward

The return G_t is the discounted sum of all future rewards

$$G_t = \sum_{i=0}^{T} \gamma^i R(s_{t+i+1})$$

immediate reward more important than future reward

$\gamma = 0$ values *only* immediate reward ("myopic")

$\gamma = 1$ does not discount

Discount factors

- ensure that we can handle **infinite episodes** mathematically

- capture some notion of the uncertainty of the future
- are often practically justified (e.g. financial investment)
- reflects psychology:
  children (and people generally) prefer immediate reward!

Value

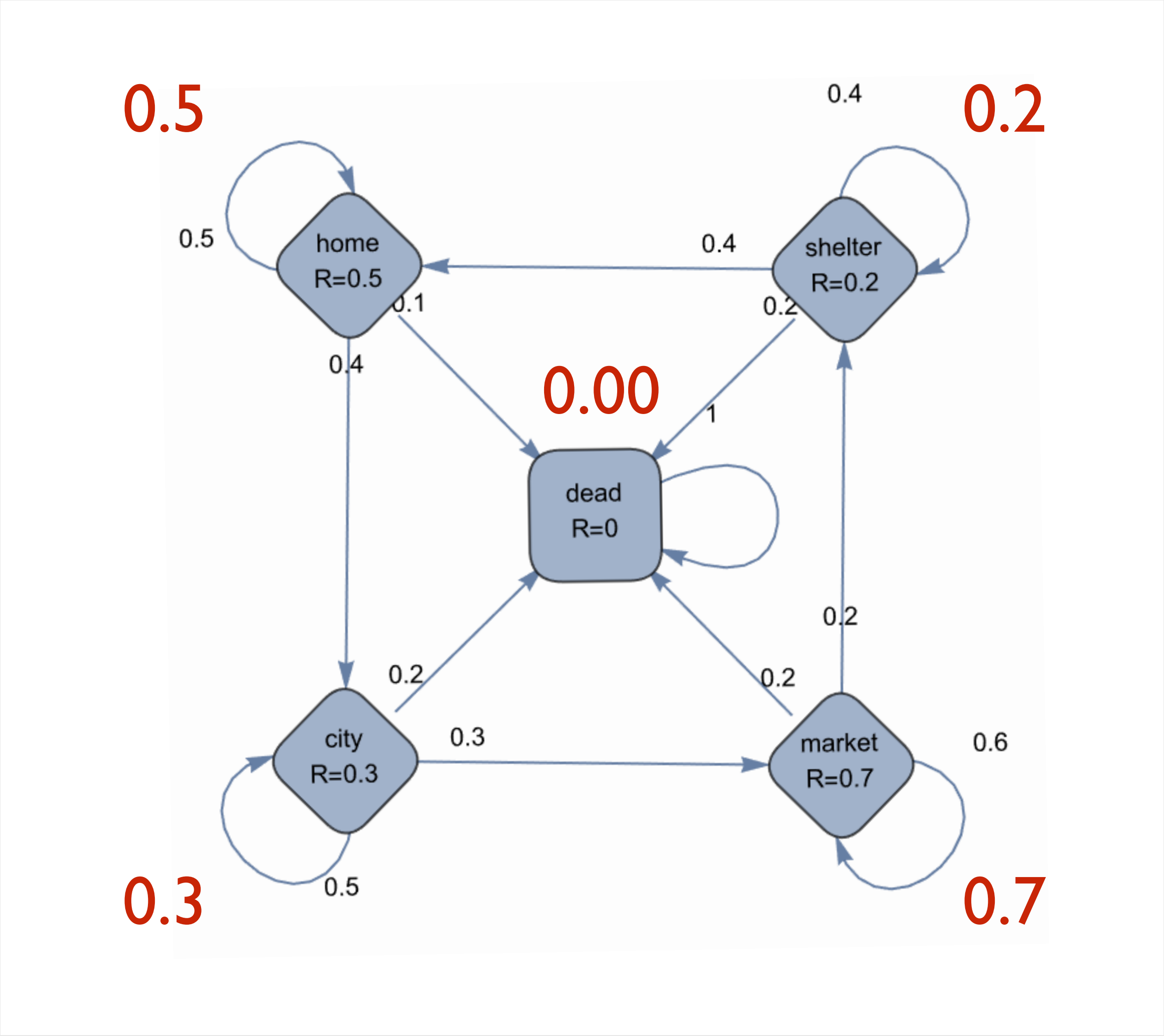
The value of a state is the *expected return* from this state

$$v(s) = \mathbb{E}[G_t \,|\, S_t = s]$$

Reminder: the value function of an agent predicts the future reward in a state given a policy (but for now the transition probabilities are still fixed, there is no policy - later we will learn the best policy.)

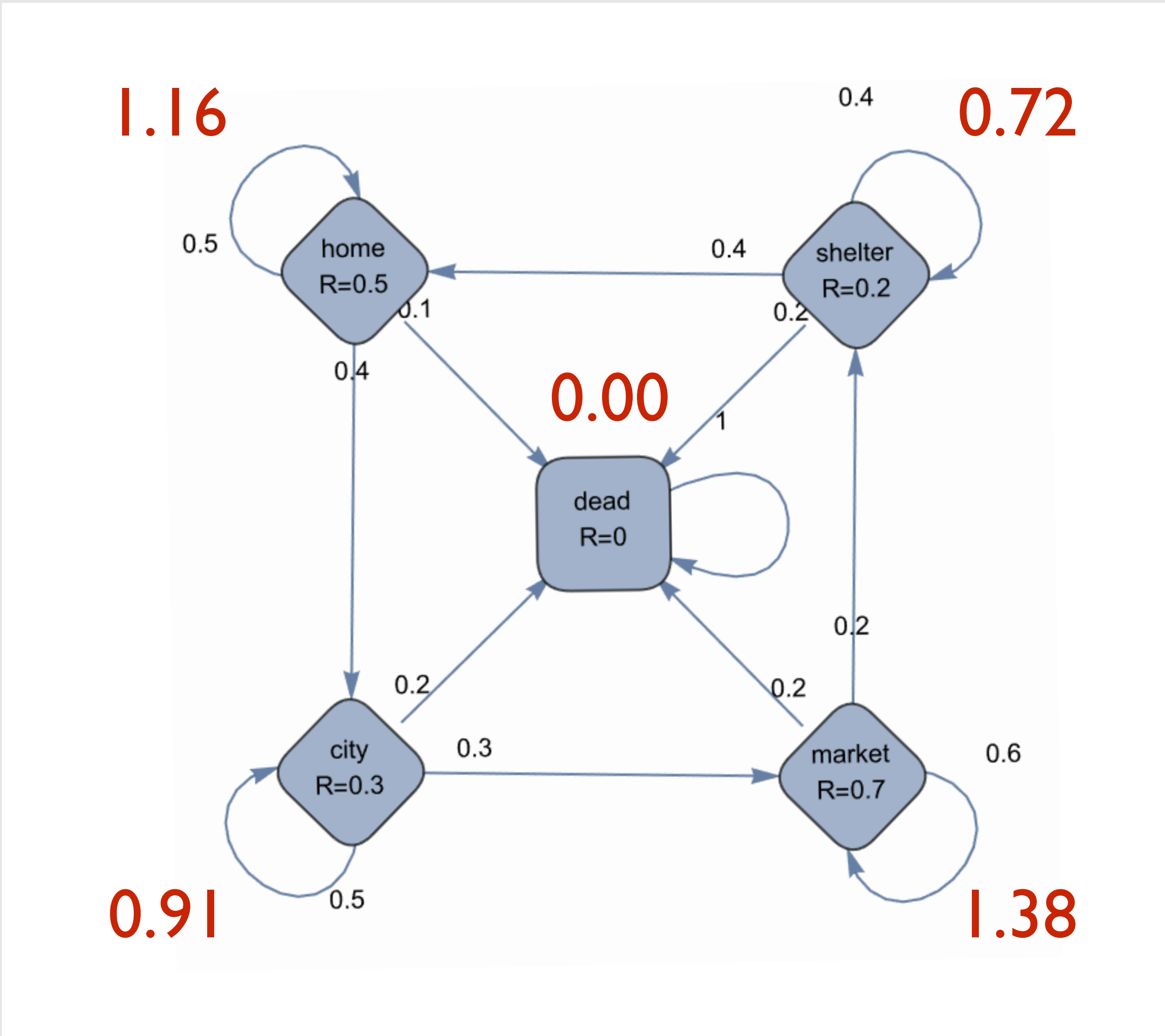
$$v_\pi(s) = \mathbb{E}_\pi[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots \; \gamma^\infty R_{t+\infty} \,|\; S_t = s]$$

Example: The 7 lives of cats (state value)



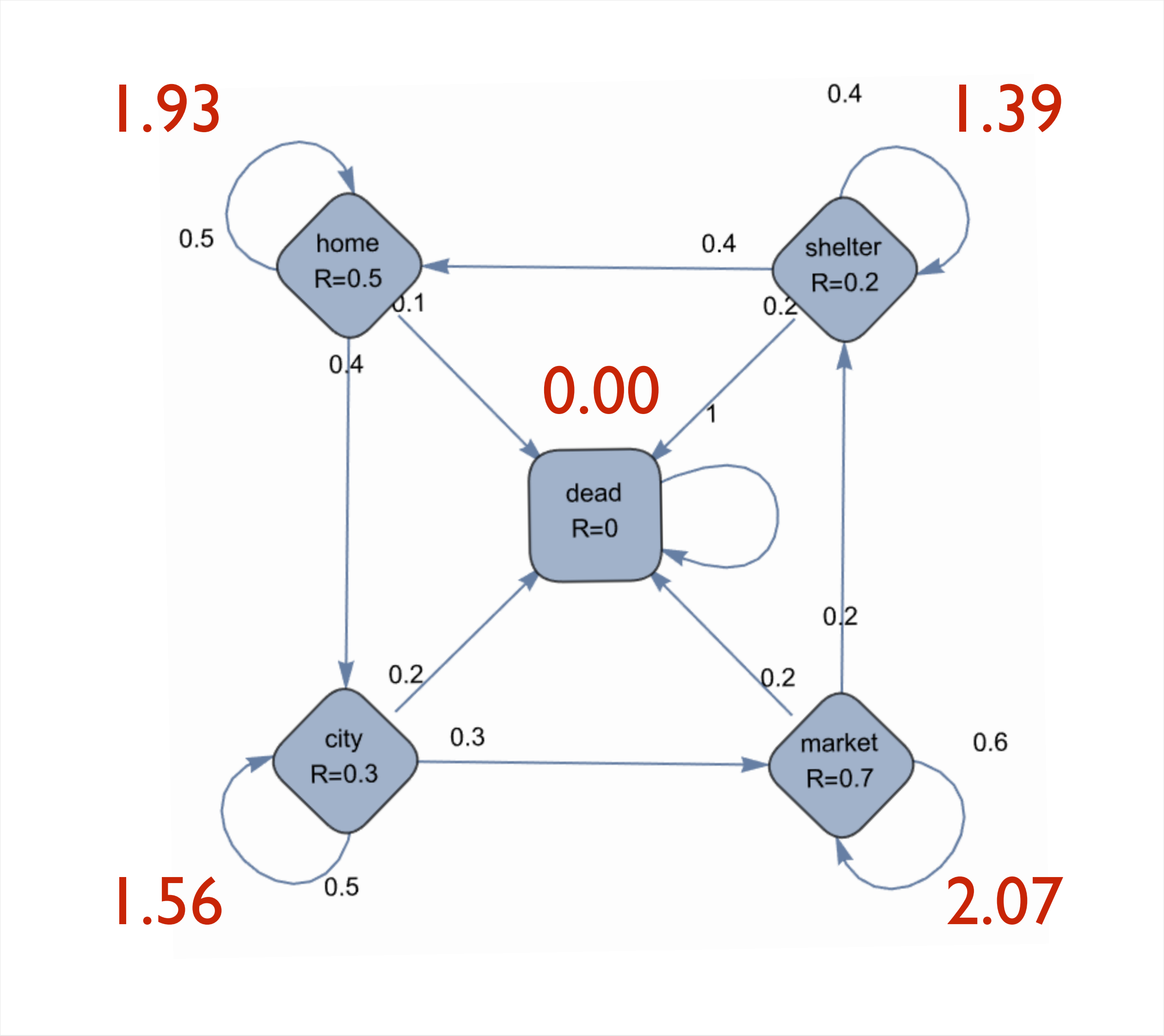
$\gamma = 0$
("myopic")

Example: The 7 lives of cats (state value)



$\gamma = 0.7$

Example: The 7 lives of cats (state value)



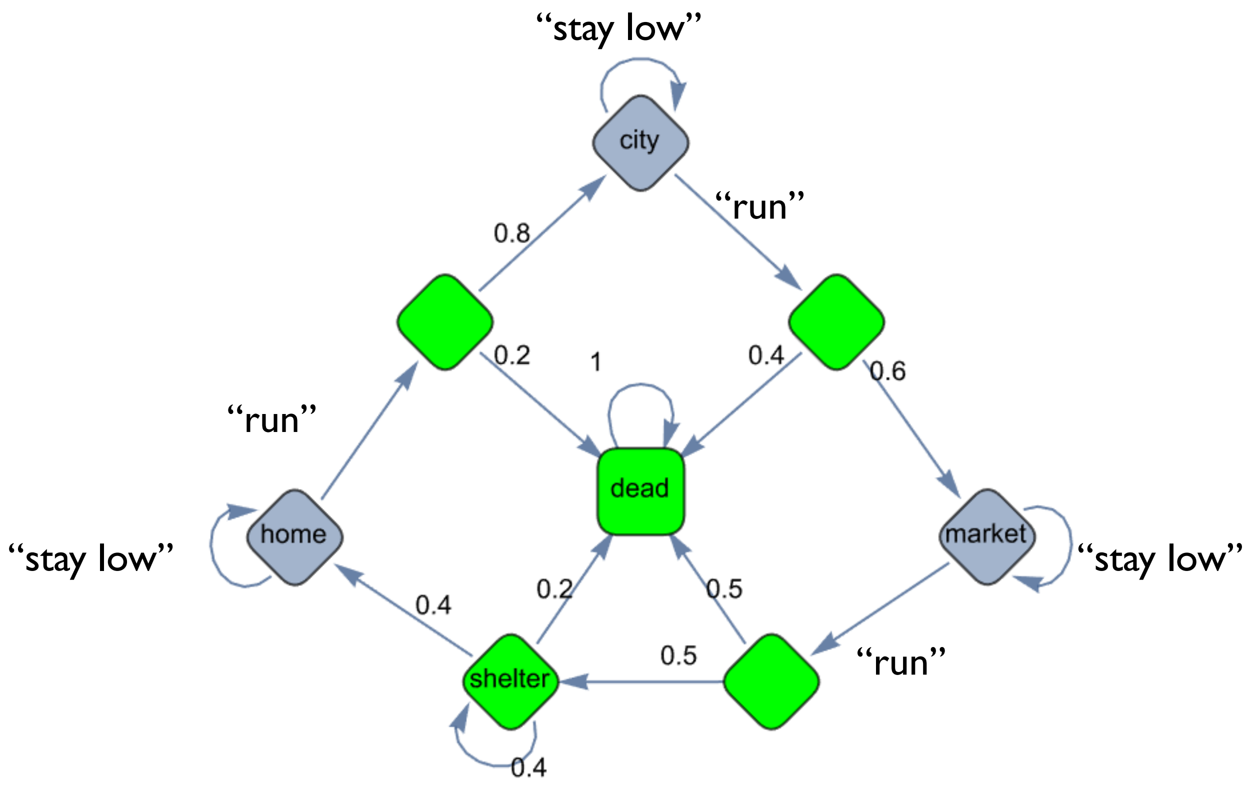
$\gamma = 0.9$

Markov Decision Process (MDP)

A Markov decision process (S, A, P, R, γ) is a Markov reward process (S, P, R, γ) with associated finite set of actions *A*. It consists of

- a finite set of states *S*
- a finite set of actions *A*
- a reward function
  - $R_s^a = \mathbb{E}[R_{t+1} \mid S_t = s, A_t = a]$
- a discount factor $0 \leq \gamma \in \mathbb{R} \leq 1$
- a stochastic matrix P describing state transition
  - $P_{s,s'}^a = P[S_{t+1} \mid S_t = s, A_t = a]$

Example: The 7 lives of cats (MDP)
Should I stay or should I go?



Take home lessons

- Reinforcement learning models how (independent) agents learn about an unknown environment.
- Agents learn by exploring the consequences of their actions (observed through rewards received).
- Actions (can) modify the environment.
- A policy describes how an agent behaves.
- A value function describes how desirable an agent judges a particular environment state to be.
- MRP = MC + rewards
- MDP = MRP with actions determining transition probabilities and rewards
- MDPs are the most fundamental modelling framework for RL