# Intelligent Interactive Systems

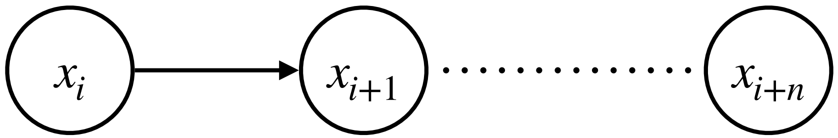
## Markov Decision Processes

Mark Lee, University of Birmingham

Original slides by

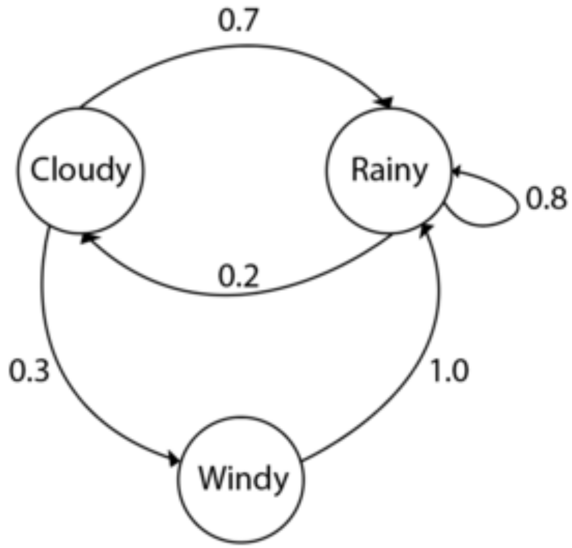
**Andrew Howes, University of Birmingham and Aalto University**

# Markov chains



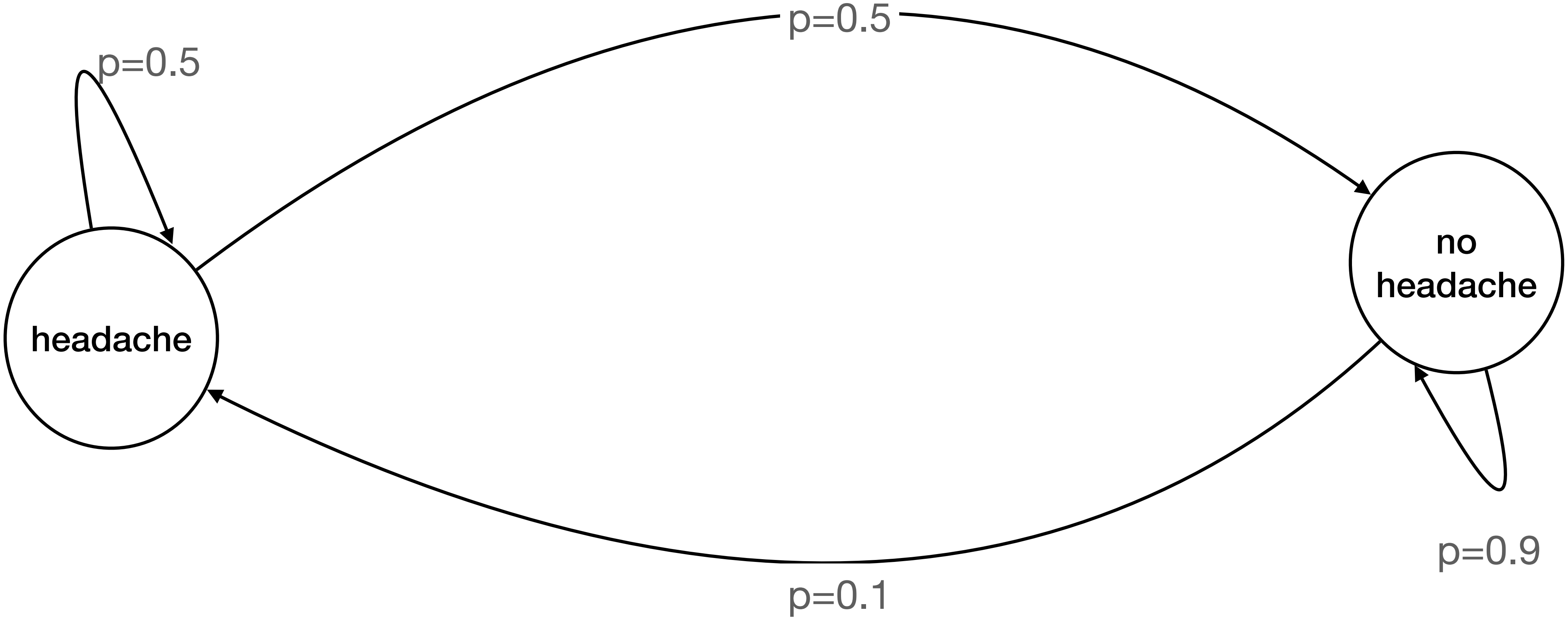

$$x_{i+1} = x_i + N(0,\sigma)$$

# Markov chains can be used to model a wide range of stochastic systems

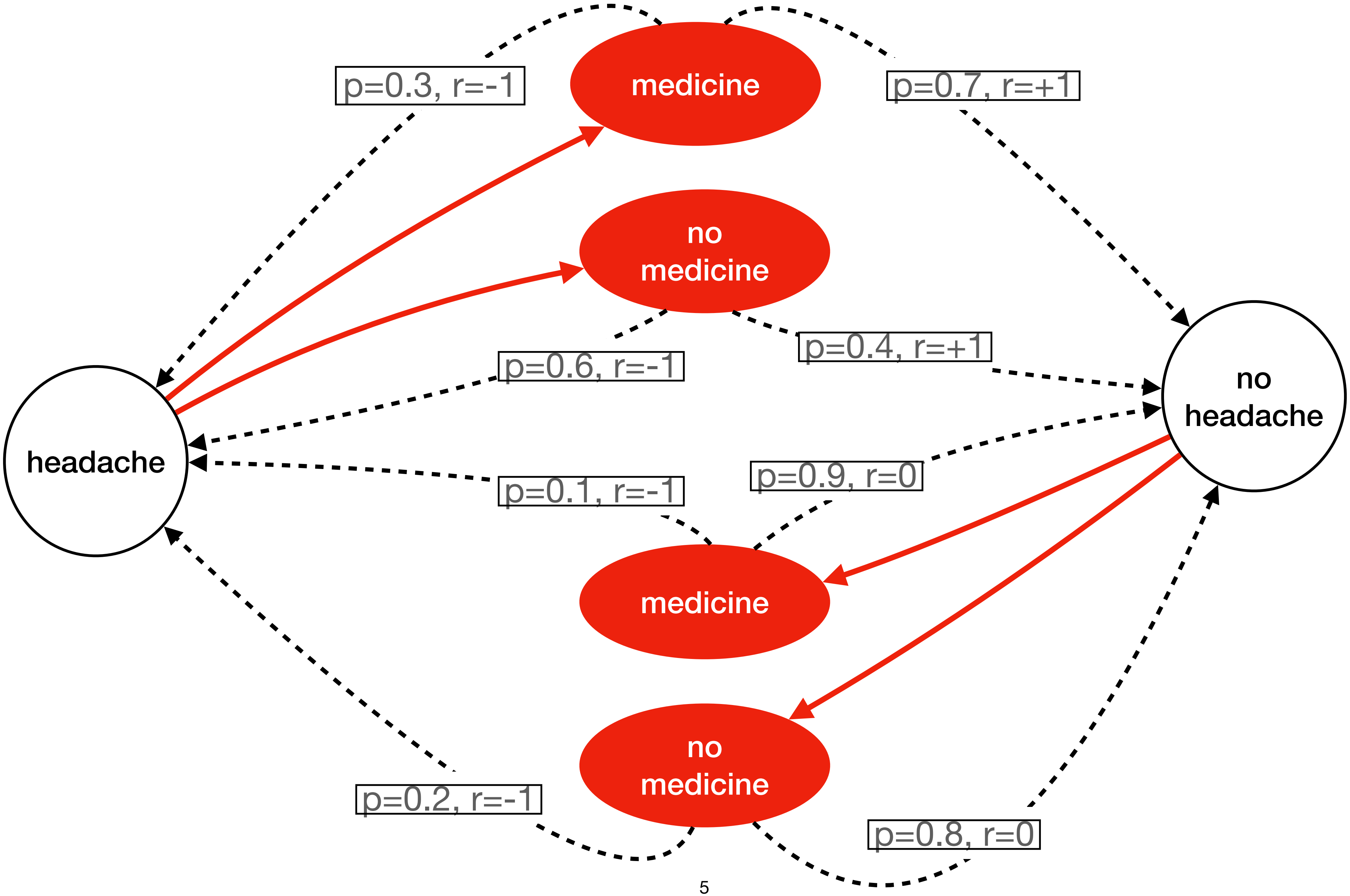



$$
\begin{array}{l c}
 & \begin{array}{ccc}\text{Cloudy} & \text{Rainy} & \text{Windy}\end{array} \\
\begin{array}{l}\text{Cloudy} \\ \text{Rainy} \\ \text{Windy}\end{array} & \begin{bmatrix} 0.0 & 0.7 & 0.3 \\ 0.2 & 0.8 & 0.0 \\ 0.0 & 1.0 & 0.0 \end{bmatrix}
\end{array}
$$

# Markov chains



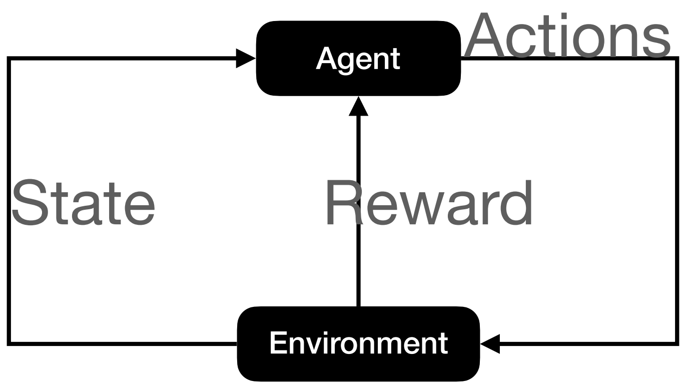
# MDP: Rewards



medicine
no medicine
medicine
no medicine
headache
no headache
p=0.3, r=-1
p=0.7, r=+1
p=0.4, r=+1
p=0.6, r=-1
p=0.9, r=0
p=0.1, r=-1
p=0.2, r=-1
p=0.8, r=0

# Markov Decision Process

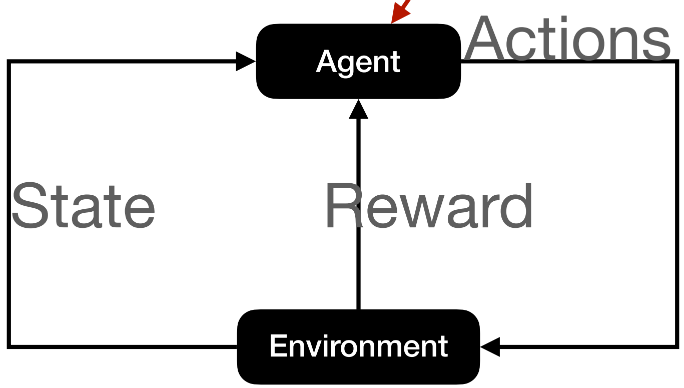
- $S$ is a set of states.
- $A$ is a set of actions
- State transition function. The new state is $s'$ with probability conditioned on the previous state $s$ and the action $a$:
    - $P(s' \mid s, a)$
- Rewards. A scalar reward is received depending on the transition from $s$ to $s'$:
    - $r = R(s', s)$

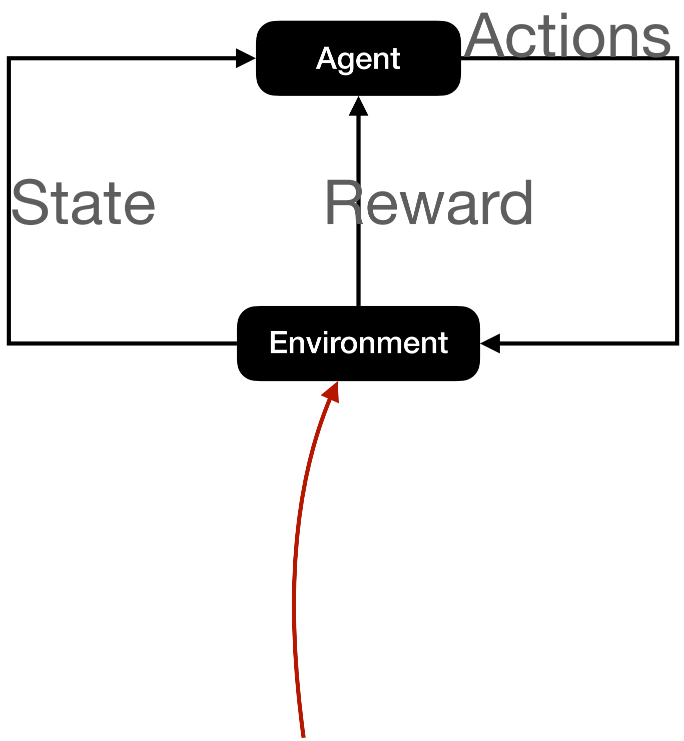

Agent
Actions
State
Reward
Environment

Intelligent Interactive System

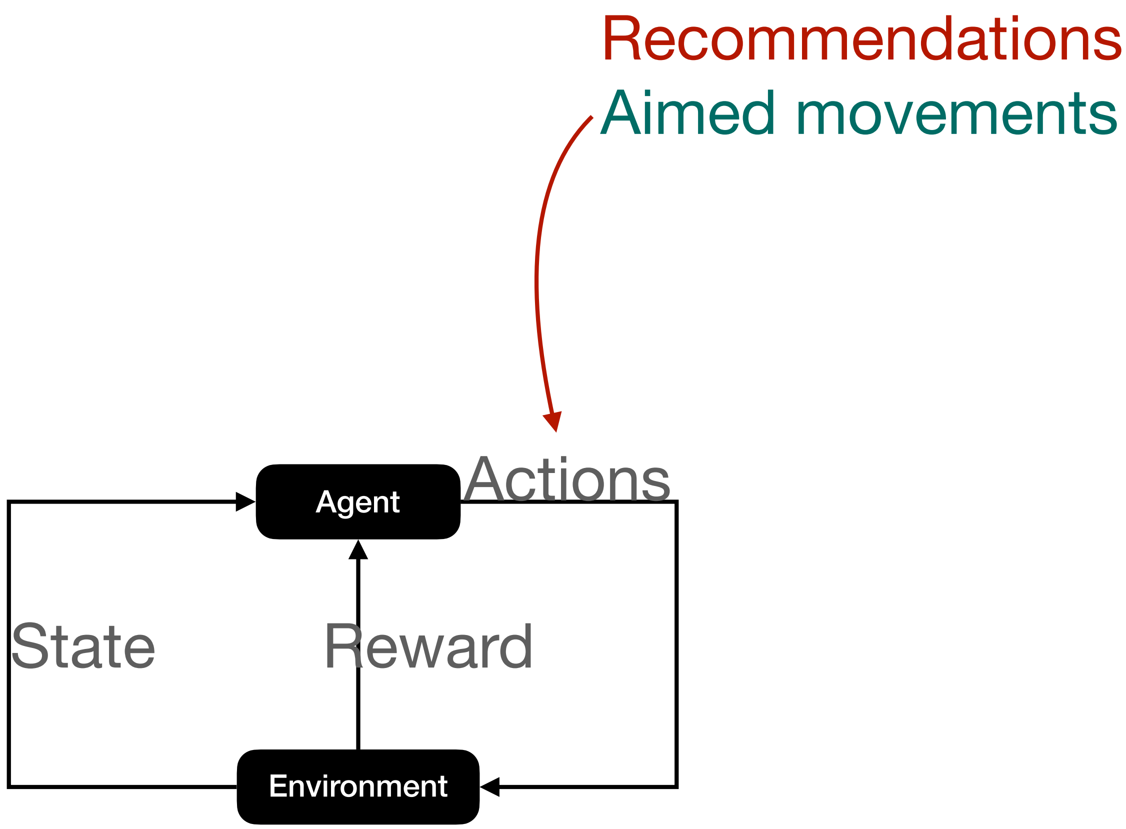
or

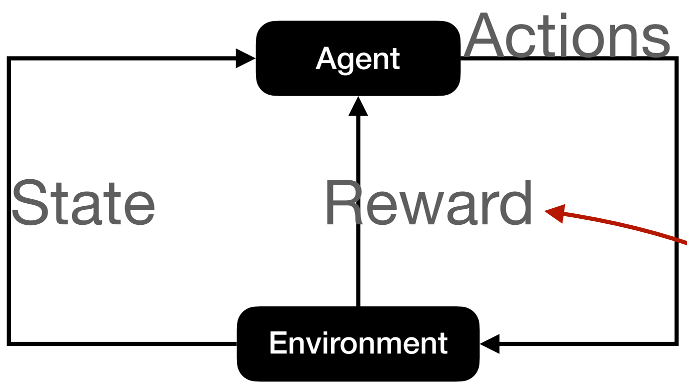
A model of a human



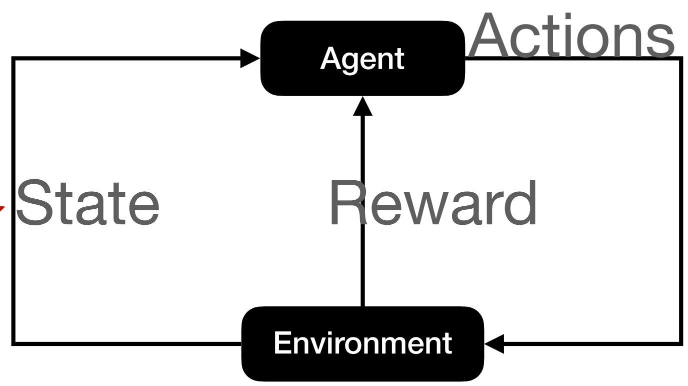



The state transition function


Recommendations
Aimed movements
Agent
Actions
State
Reward
Environment



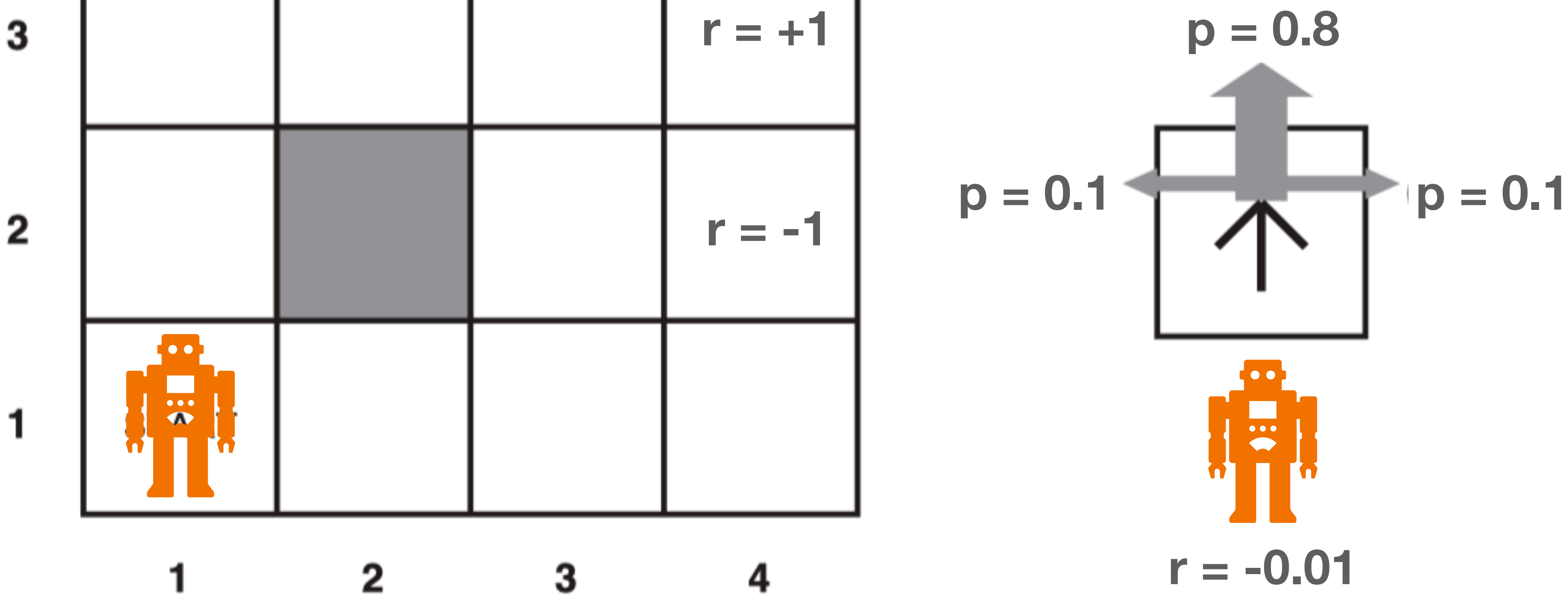

Outcomes
Expected Value
Subjective Expected Utility
Prospect Theory



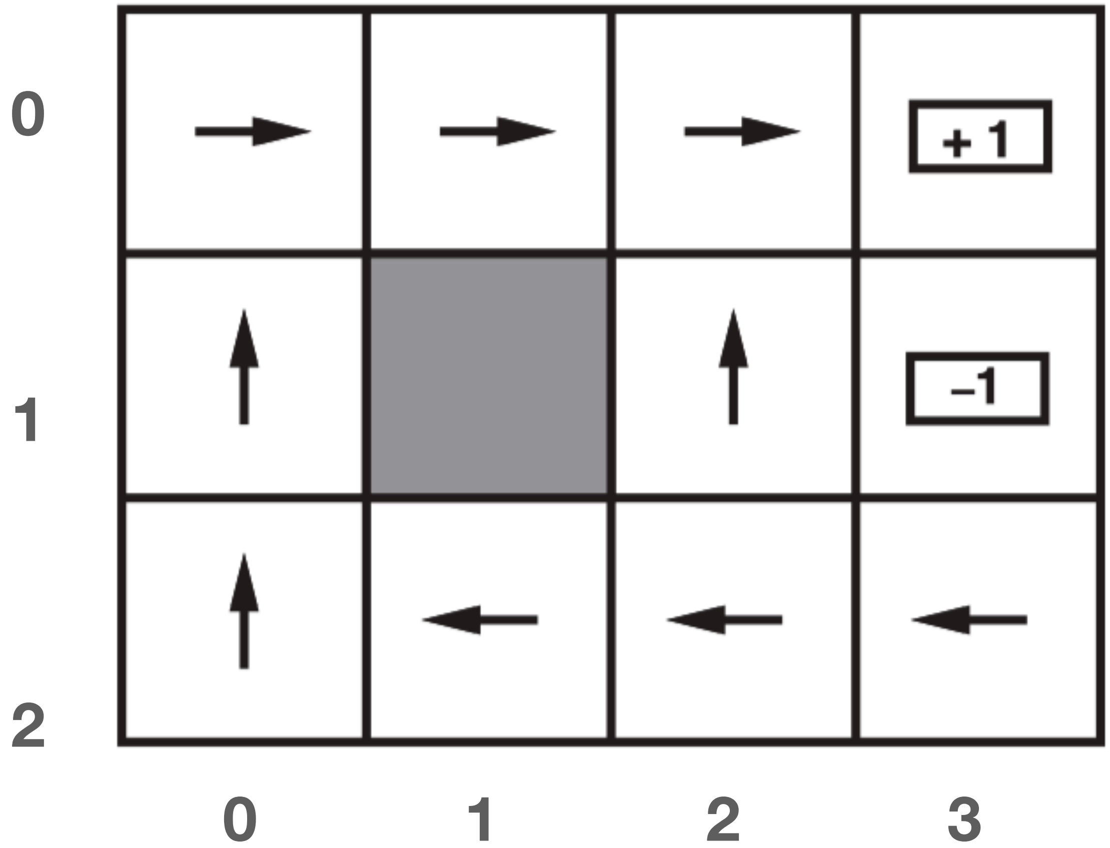

Driving
Chatting
Shopping
Memory
Emotions

# Design an MDP for a two state, two action human decision problem of your choosing.

# Design an MDP for a two state, two action intelligent interactive system.


3
2
1
r = +1
r = -1
1
2
3
4
p = 0.8
p = 0.1
p = 0.1
r = -0.01

# An Optimal Policy $\pi*$



# The Bellman Equation for MDPs

$$V^{\pi*}(s) = \max_a \left\{ R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^{\pi*}(s') \right\}.$$

- $V^{\pi^*}(s)$ is the value of state $s$ assuming the optimal policy $\pi$.
- It is defined as the maximum of the expected values of all actions $a$ from state $s$.
- The expected value of an action has two parts, the reward defined by $R(s,a)$ and the discounted $\gamma$ sum of all possible expected outcome values $V^{\pi^*}(s')$ if $a$ is chosen assuming that the optimal policy continues to determine future actions selections.

$$
\begin{aligned}
U(1,1) = -0.01 + \gamma \ \max[\ & 0.8U(1,2) + 0.1U(2,1) + 0.1U(1,1), && (Up)\\
& 0.9U(1,1) + 0.1U(1,2), && (Left)\\
& 0.9U(1,1) + 0.1U(2,1), && (Down)\\
& 0.8U(2,1) + 0.1U(1,2) + 0.1U(1,1)\ ]. && (Right)
\end{aligned}
$$

# Algorithms

- Value iteration
- Policy iteration
- Q-learning
- Reinforcement Learning (RL)
- Deep Reinforcement Learning (DRL)
- Proximal Policy Optimisation (PPO)
- etc.

# Value Iteration

**function** VALUE-ITERATION($mdp, \epsilon$) **returns** a utility function
**inputs**: $mdp$, an MDP with states $S$, actions $A(s)$, transition model $P(s' \mid s, a)$,
rewards $R(s)$, discount $\gamma$
$\epsilon$, the maximum error allowed in the utility of any state
**local variables**: $U$, $U'$, vectors of utilities for states in $S$, initially zero
$\delta$, the maximum change in the utility of any state in an iteration

- **Inputs**: a MDP withs state $S$, action $A(s)$, transition model $P(s'|s,a)$, rewards $R(s)$, discount $\gamma$
- **repeat**:
  - $U \leftarrow U'; \delta \leftarrow 0$
  - **for each state** $s$ in $S$ do:
    - $U'[s] \leftarrow R[s] + \gamma \max_{a \in A(s)} \sum_{s'} P(s'|s,a)U[s']$
    - **if** $U'[s] - U[s] > \delta$ **then** $\delta \leftarrow |U'[s] - U[s]|$
- **until** $\delta < \epsilon(1-\gamma)/\gamma$
- **return** $U$

---

The initial U is:

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | +1 |
| 0 | WALL | 0 | −1 |
| 0 | 0 | 0 | 0 |

During the value iteration:

| | | | |
|---|---|---|---|
| −0.01 | −0.01 | 0.782 | +1 |
| −0.01 | WALL | −0.01 | −1 |
| −0.01 | −0.01 | −0.01 | −0.01 |

| | | | |
|---|---|---|---|
| −0.01 | 0.607 | 0.858 | +1 |
| −0.01 | WALL | 0.509 | −1 |
| −0.01 | −0.01 | −0.01 | −0.01 |

| | | | |
|---|---|---|---|
| 0.892 | 0.921 | 0.945 | +1 |
| 0.863 | WALL | 0.711 | −1 |
| 0.815 | 0.754 | 0.685 | 0.456 |

| | | | |
|---|---|---|---|
| 0.893 | 0.921 | 0.946 | +1 |
| 0.867 | WALL | 0.714 | −1 |
| 0.829 | 0.785 | 0.726 | 0.478 |

| | | | |
|---|---|---|---|
| 0.894 | 0.921 | 0.946 | +1 |
| 0.869 | WALL | 0.721 | −1 |
| 0.837 | 0.802 | 0.754 | 0.513 |

| | | | |
|---|---|---|---|
| 0.903 | 0.930 | 0.954 | +1 |
| 0.879 | WALL | 0.789 | −1 |
| 0.853 | 0.830 | 0.805 | 0.639 |

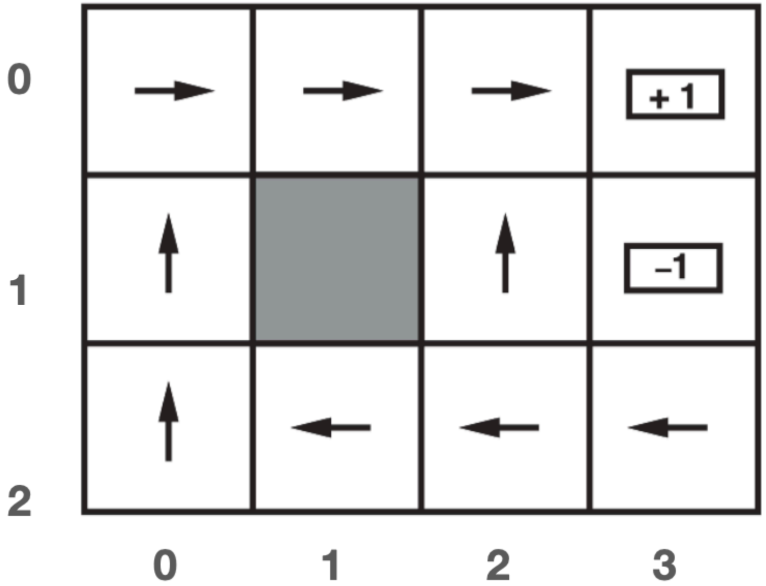

| | | | |
|---|---|---|---|
| 0.903 | 0.930 | 0.954 | +1 |
| 0.879 | WALL | 0.789 | −1 |
| 0.853 | 0.830 | 0.805 | 0.639 |

| | | | |
|---|---|---|---|
| 0.903 | 0.930 | 0.954 | +1 |
| 0.879 | WALL | 0.789 | −1 |
| 0.853 | 0.830 | 0.805 | 0.639 |

The optimal policy is:

| | | | |
|---|---|---|---|
| Right | Right | Right | +1 |
| Up | WALL | Left | −1 |
| Up | Left | Left | Down |

# Optimal Policy $\pi*$ ?




```
The optimal policy is:

| Right | Right | Right | +1    |
| Up    | WALL  | Left  | –1    |
| Up    | Left  | Left  | Down  |
```

# Discuss

# Summary

- Markov Chains cannot represent decisions.
- If we want to model sequential human decision processes then we need Markov Decision Processes (MDPs).
- MPDs define a decision problem in terms of a set of states, a set of actions, a transition function and a reward function.
- The selection of actions represent intentions but the outcomes of those intentions can be uncertain.
- The Bellman equation defines the value of a decision problem in terms of the immediate reward and the sum of all future rewards.
- Value iteration is one algorithm that can be used to find the optimal policy.
- Policies can sometimes be counter-intuitive.

# Reading


- Alagoz, O., Hsu, H., Schaefer, A. J., & Roberts, M. S. (2010). Markov decision processes: a tool for sequential decision making under uncertainty. Medical Decision Making, 30(4), 474-483.


**Further Reading**


- Oh, S. H., Lee, S. J., Noh, J., & Mo, J. (2021). Optimal treatment recommendations for diabetes patients using the Markov decision process along with the South Korean electronic health records. Scientific reports, 11(1), 6920.

- Ma, S., Guo, J., Zeng, S., Che, H., & Pan, X. (2022). Modeling eye movement in dynamic interactive tasks for maximizing situation awareness based on Markov decision process. Scientific Reports, 12(1), 13298.

# Thank you!