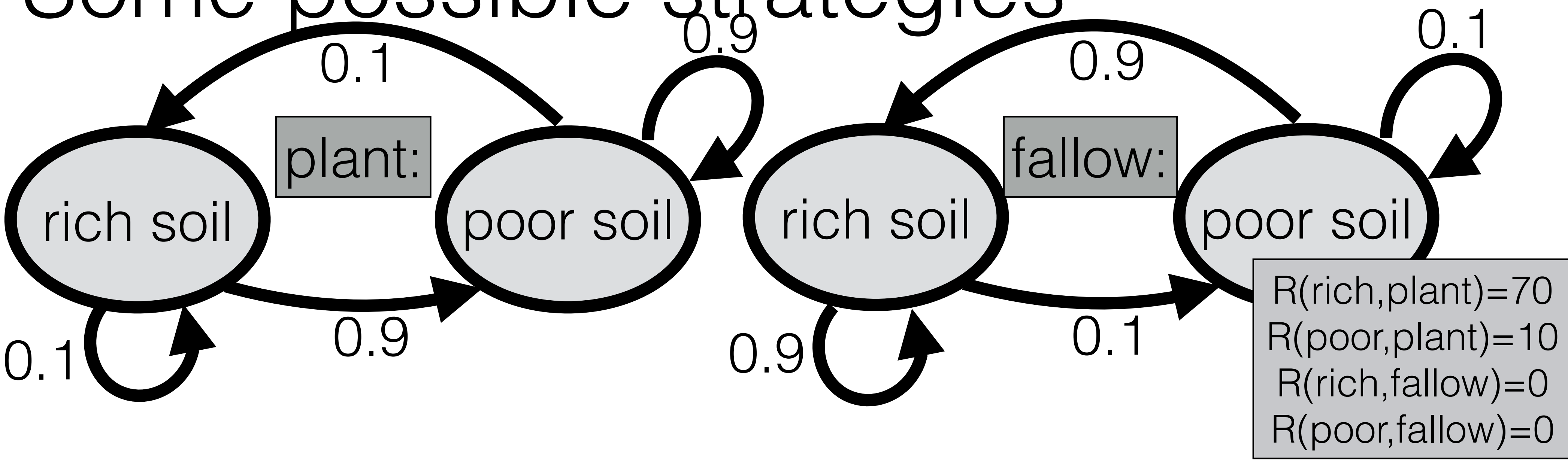
Reinforcement Learning

Prof. Tamara Broderick

Edited From 6.036 Fall21 Offering

Some possible strategies

Example (ii)



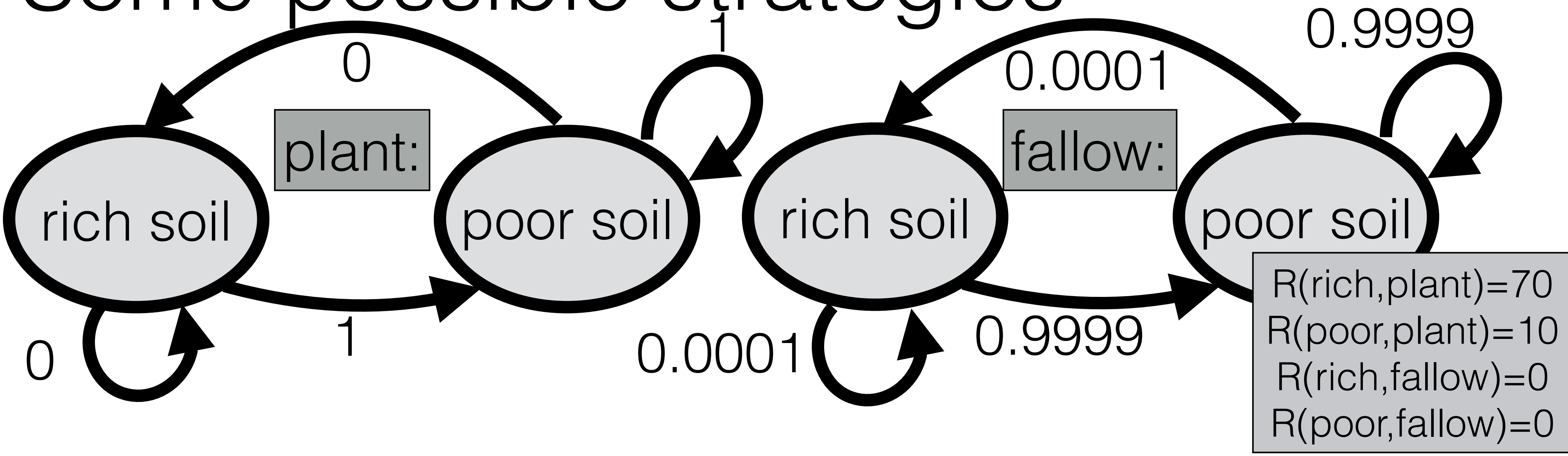
- Strategy A: always try actions uniformly at random
  - E.g. $s = \text{poor}$

    $a = \text{fallow}, s = \text{rich}, r = 0$

    $a = \text{plant}, s = \text{poor}, r = 70$
- Strategy B: after a few moves, choose a policy (e.g. whatever seems best so far) and commit to it
  - E.g. from here: if rich, plant & if poor, fallow
- What could go wrong with each strategy?

Some possible strategies

Example (iii)

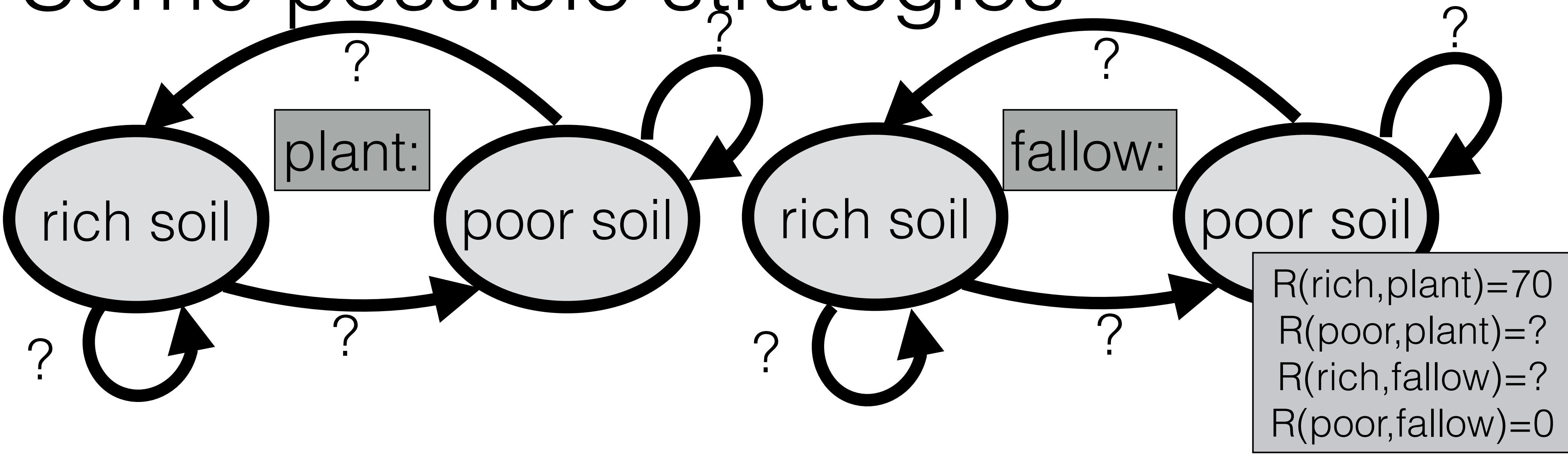


- Strategy A: always try actions uniformly at random
  - E.g. $s = \text{poor}$

$$a = \text{fallow}, s = \text{rich}, r = 0$$
$$a = \text{plant}, s = \text{poor}, r = 70$$

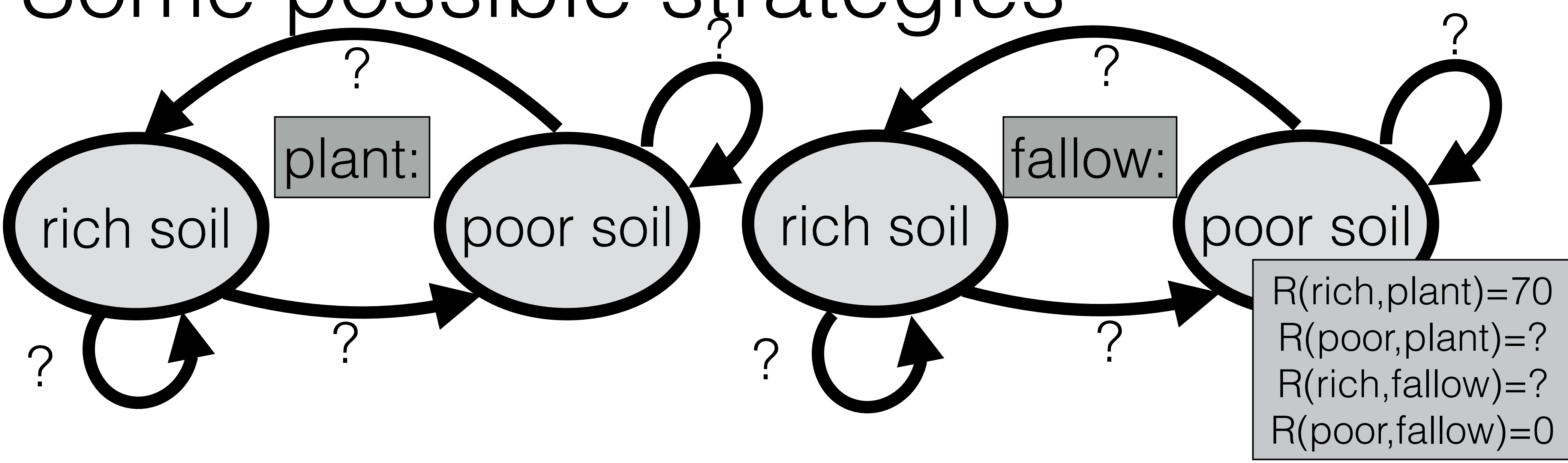
- Strategy B: after a few moves, choose a policy (e.g. whatever seems best so far) and commit to it
  - E.g. from here: if rich, plant & if poor, fallow
- What could go wrong with each strategy?

Some possible strategies



- Strategy A: always try actions uniformly at random
  - E.g. $s = \text{poor}$
    $a = \text{fallow}, s = \text{rich}, r = 0$
    $a = \text{plant}, s = \text{poor}, r = 70$
- Strategy B: after a few moves, choose a policy (e.g. whatever seems best so far) and commit to it
  - E.g. from here: if rich, plant & if poor, fallow
- What could go wrong with each strategy?
- What are the benefits of each strategy?

Some possible strategies

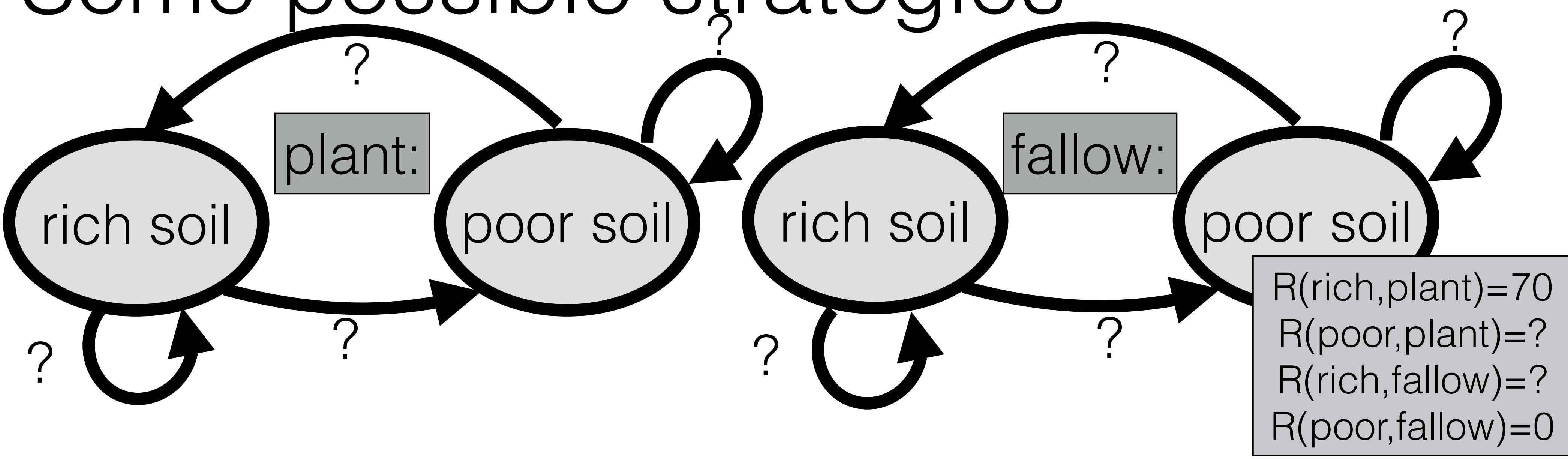


- Strategy A: always try actions uniformly at random
  - E.g. $s = \text{poor}$
    $a = \text{fallow}, s = \text{rich}, r = 0$
    $a = \text{plant}, s = \text{poor}, r = 70$

Focused on exploring

- Strategy B: after a few moves, choose a policy (e.g. whatever seems best so far) and commit to it
  - E.g. from here: if rich, plant & if poor, fallow
- What could go wrong with each strategy?
- What are the benefits of each strategy?

Some possible strategies



- Strategy A: always try actions uniformly at random
  - E.g. $s = \text{poor}$
    $a = \text{fallow}, s = \text{rich}, r = 0$
    $a = \text{plant}, s = \text{poor}, r = 70$

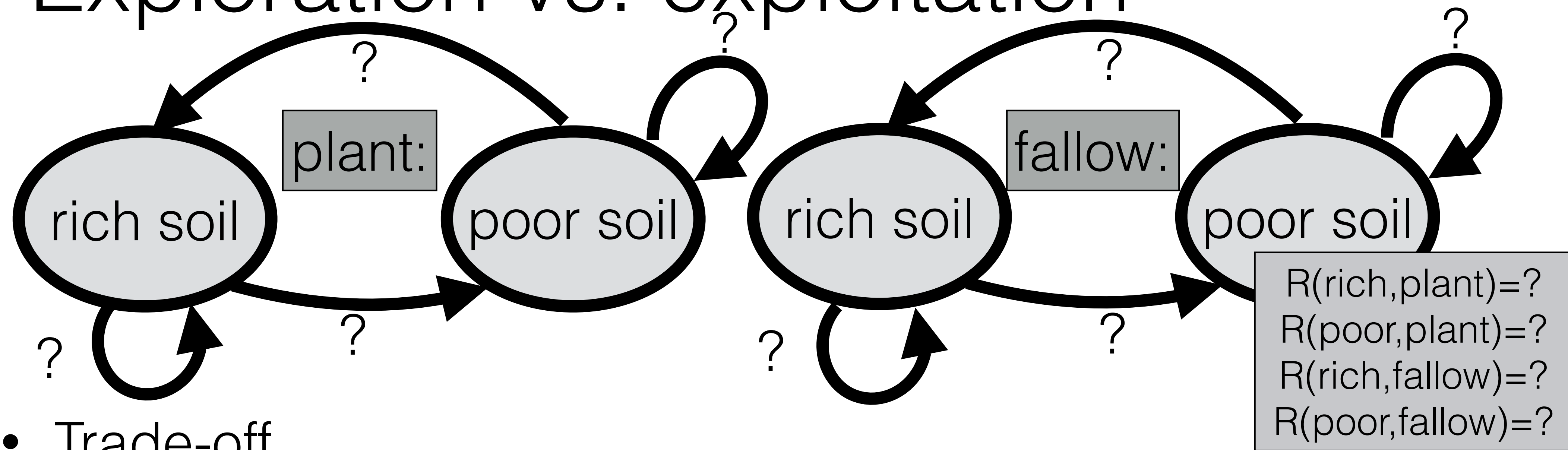
Focused on exploring

- Strategy B: after a few moves, choose a policy (e.g. whatever seems best so far) and commit to it
  - E.g. from here: if rich, plant & if poor, fallow

Focused on exploiting

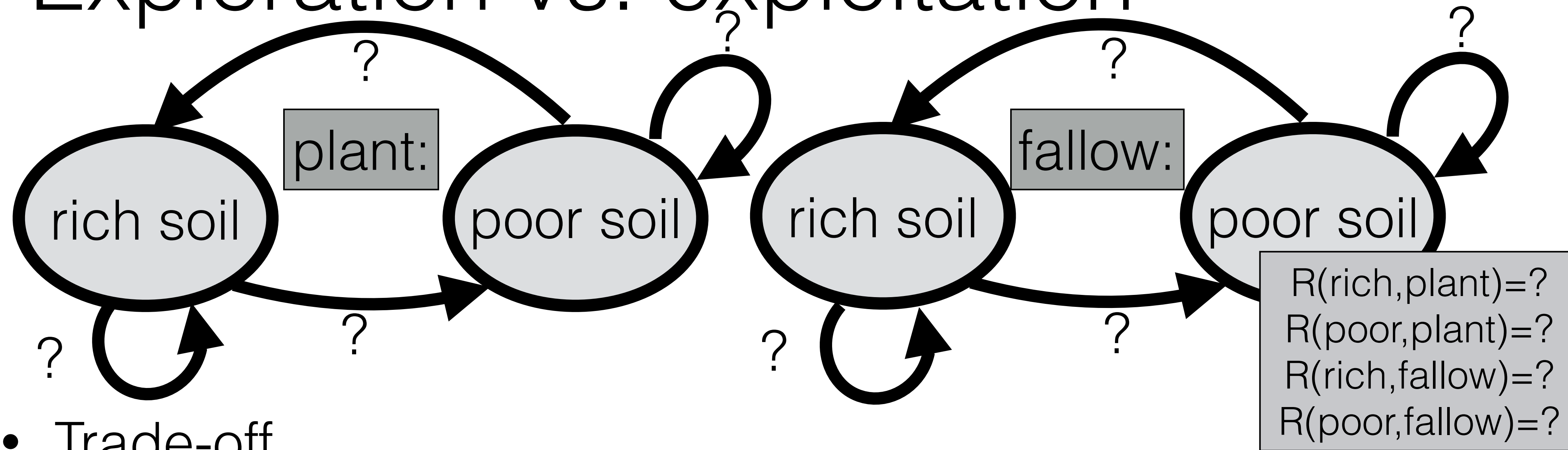
- What could go wrong with each strategy?
- What are the benefits of each strategy?

Exploration vs. exploitation



- Trade-off
  - **Exploration**: the more we explore, the better we understand the world (e.g. *T* and *R*)
  - **Exploitation**: based on what we know about the world, we can take actions with the aim to get highest reward
- One option (not the only one!): **ϵ-greedy strategy**
  - With probability 1-ϵ , exploit
  - With probability ϵ , choose an action uniformly at random

Exploration vs. exploitation

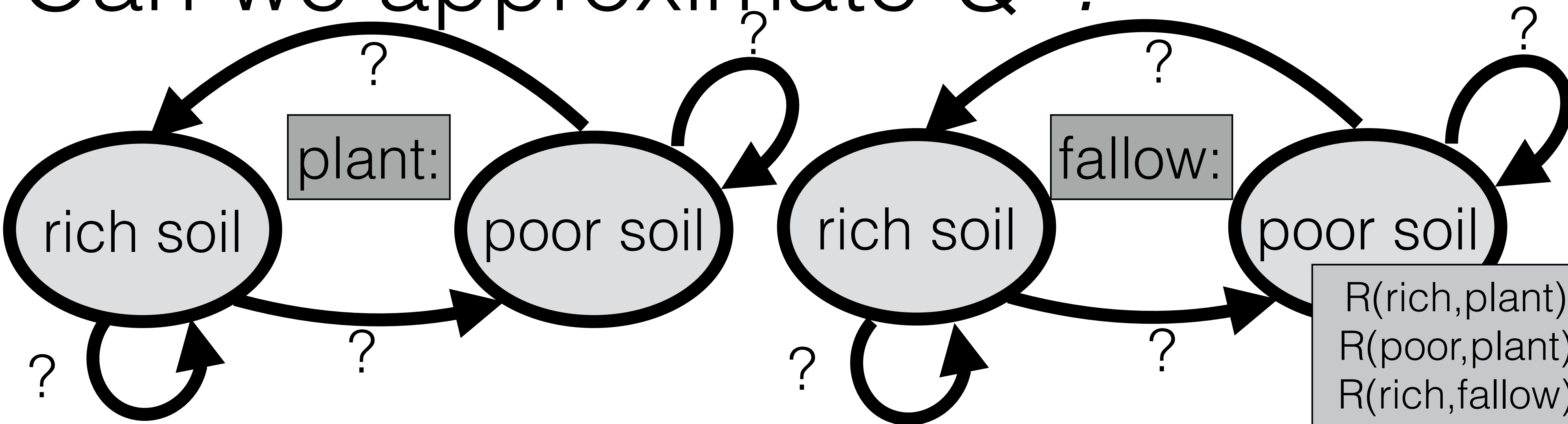


- Trade-off
  - **Exploration**: the more we explore, the better we understand the world (e.g. T and R)
  - **Exploitation**: based on what we know about the world, we can take actions with the aim to get highest reward
- One option (not the only one!): **ϵ-greedy strategy**
  - With probability 1-ϵ , exploit
  - With probability ϵ , choose an action uniformly at random

Need to specify how!

- Consider infinite horizon. If we had Q^*, we could exploit.
  - Idea: estimate Q^* from the observations ("data") so far.

Can we approximate Q^*?



R(rich,plant)=?
R(poor,plant)=?
R(rich,fallow)=?
R(poor,fallow)=?

- Option 1: Estimate transition model *T* and reward function *R*

Data at step *t*: $s^{(t)}, a^{(t)}, r^{(t)}, s^{(t+1)}$

`Initialize` $s^{(1)} = s_0$

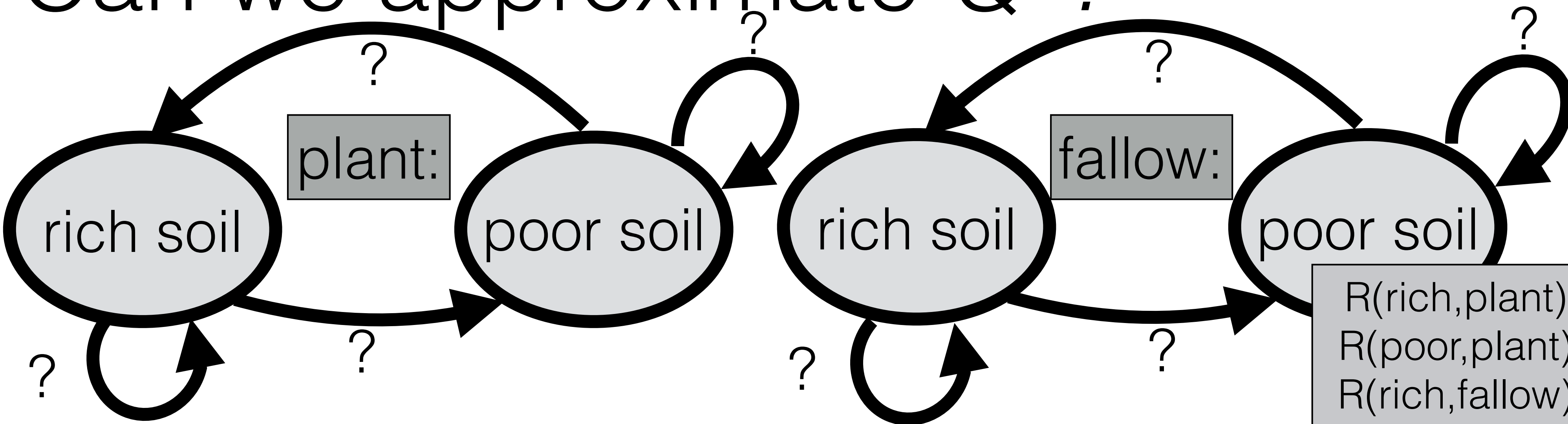
`Initialize: any` s, a, s': $\hat{T}(s, a, s') = \frac{1}{|S|}; \hat{R}(s, a) = 0; Q$

`for` `t = 1, 2, 3, …`

  $a^{(t)}$ `= select_action(`$s^{(t)}, Q$ `)`

E.g. ϵ-**greedy**

Can we approximate Q^*?



R(rich,plant)=?
R(poor,plant)=?
R(rich,fallow)=?
R(poor,fallow)=?

- Option 1: Estimate transition model T and reward function R

Data at step t: $s^{(t)}, a^{(t)}, r^{(t)}, s^{(t+1)}$

Initialize $s^{(1)} = s_0$

Initialize: any s, a, s': $\hat{T}(s,a,s') = \frac{1}{|\mathcal{S}|}$; $\hat{R}(s,a) = 0$; Q

for t = 1, 2, 3, …

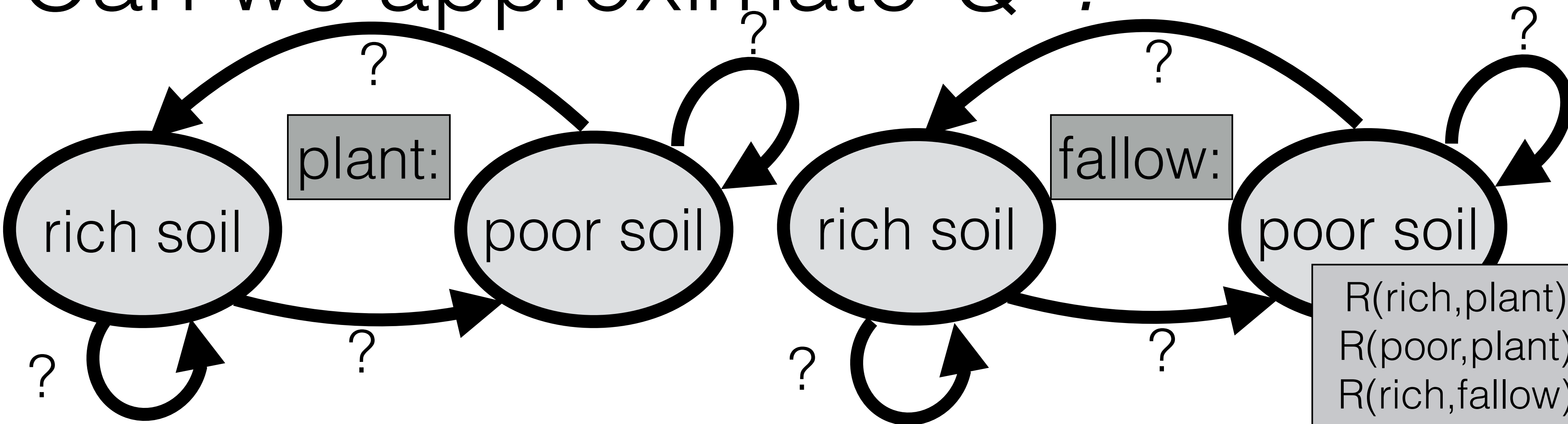
  $a^{(t)}$ = select_action($s^{(t)}, Q$) **E.g. ϵ-greedy**

  $r^{(t)}, s^{(t+1)}$ = execute($a^{(t)}$)

  $\hat{R}(s^{(t)}, a^{(t)}) = r^{(t)}$

  Each s,a,s′: $\hat{T}(s,a,s') = \frac{1+\sum_{i=1}^{t} \mathbf{1}\{s^{(i)}=s, a^{(i)}=a, s^{(i+1)}=s'\}}{|\mathcal{S}|+\sum_{i=1}^{t} \mathbf{1}\{s^{(i)}=s, a^{(i)}=a\}}$

Can we approximate Q^*?



R(rich,plant)=?
R(poor,plant)=?
R(rich,fallow)=?
R(poor,fallow)=?

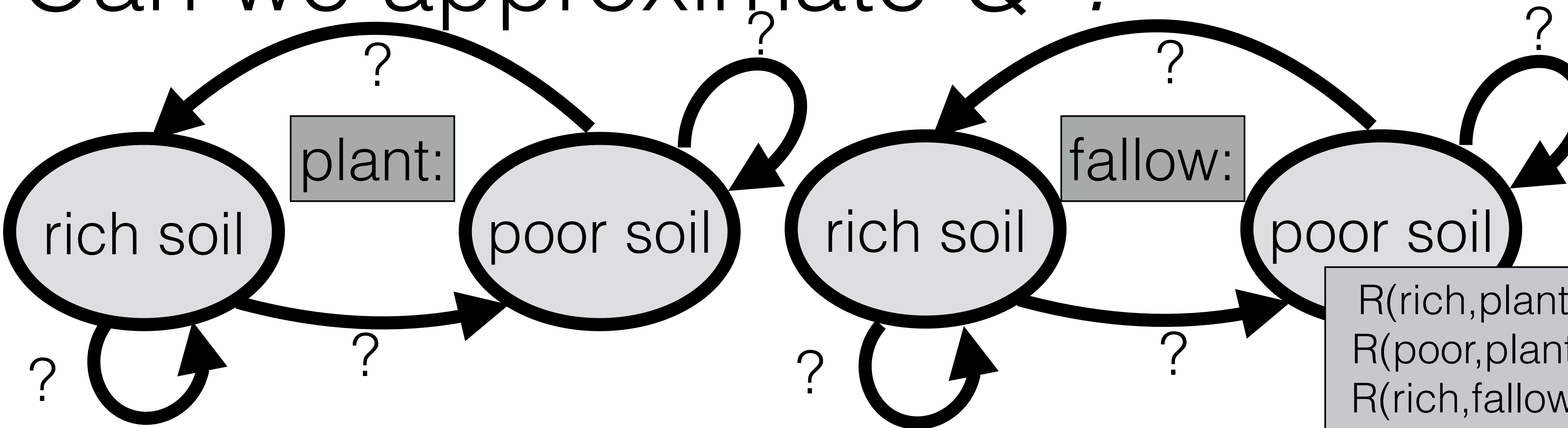
- Option 1: Estimate transition model *T* and reward function *R*

Data at step *t*: $s^{(t)}, a^{(t)}, r^{(t)}, s^{(t+1)}$

Initialize $s^{(1)} = s_0$
Initialize: any s, a, s': $\hat{T}(s,a,s') = \frac{1}{|\mathcal{S}|}$; $\hat{R}(s,a) = 0$; Q
for t = 1, 2, 3, …
  $a^{(t)}$ = select_action($s^{(t)}, Q$)
  $r^{(t)}, s^{(t+1)}$ = execute($a^{(t)}$)
  $\hat{R}(s^{(t)}, a^{(t)}) = r^{(t)}$
  Each s,a,s': $\hat{T}(s,a,s') = \frac{1+\sum_{i=1}^{t} \mathbf{1}\{s^{(i)}=s, a^{(i)}=a, s^{(i+1)}=s'\}}{|\mathcal{S}|+\sum_{i=1}^{t} \mathbf{1}\{s^{(i)}=s, a^{(i)}=a\}}$

E.g. ϵ-**greedy**

Can we approximate Q^*?



R(rich,plant)=?
R(poor,plant)=?
R(rich,fallow)=?
R(poor,fallow)=?

- Option 1: Estimate transition model *T* and reward function *R*

Data at step *t*: $s^{(t)}, a^{(t)}, r^{(t)}, s^{(t+1)}$

Initialize $s^{(1)} = s_0$

Initialize: any s, a, s': $\hat{T}(s,a,s') = \frac{1}{|\mathcal{S}|}$; $\hat{R}(s,a) = 0$; Q

for t = 1, 2, 3, …

  $a^{(t)}$ = select_action($s^{(t)}, Q$)

E.g. ϵ**-greedy**

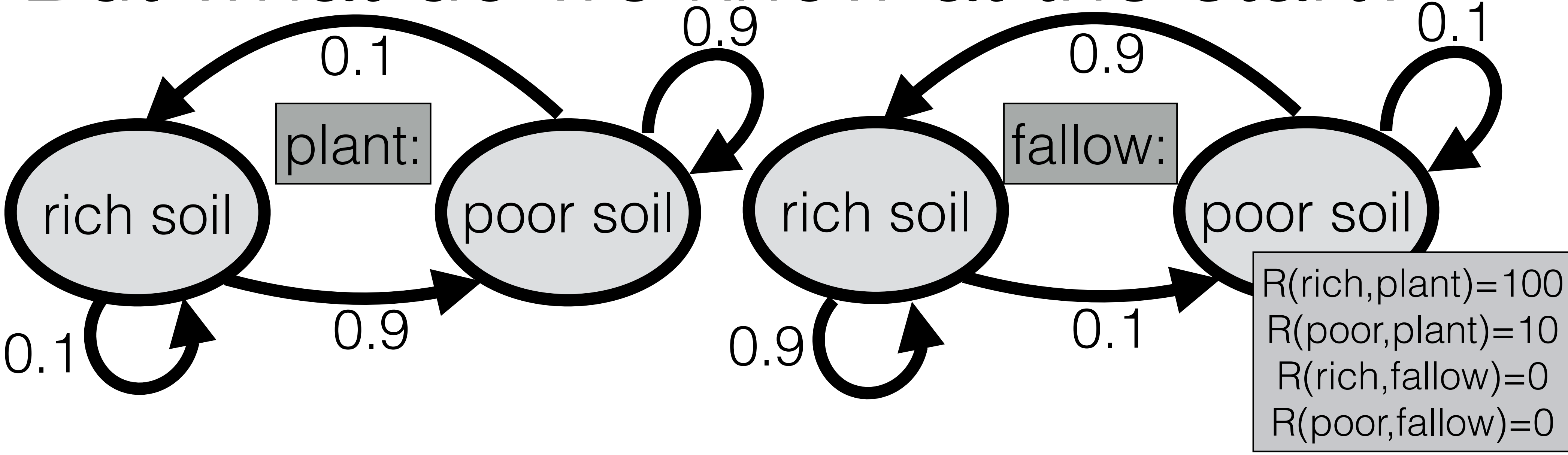
  $r^{(t)}, s^{(t+1)}$ = execute($a^{(t)}$)

  $\hat{R}(s^{(t)}, a^{(t)}) = r^{(t)}$

  Each s,a,s': $\hat{T}(s,a,s') = \frac{1+\sum_{i=1}^{t} \mathbf{1}\{s^{(i)}=s, a^{(i)}=a, s^{(i+1)}=s'\}}{|\mathcal{S}|+\sum_{i=1}^{t} \mathbf{1}\{s^{(i)}=s, a^{(i)}=a\}}$

  Q = infinite-horizon-value-iteration($\hat{R}, \hat{T}$)

But what do we know at the start?



- General goal: Find a policy to maximize expected reward.
- Up to this point: Assume we know full Markov decision process (MDP).
  - We figure out best policy and use it from the start.
- But we often *don't* know the transition model *T* or reward function *R* before we start.
- Next: Assume we do know the states, actions, and discount. But we don't know *T* or *R*.
  - Find a sequence of actions to maximize expected reward.