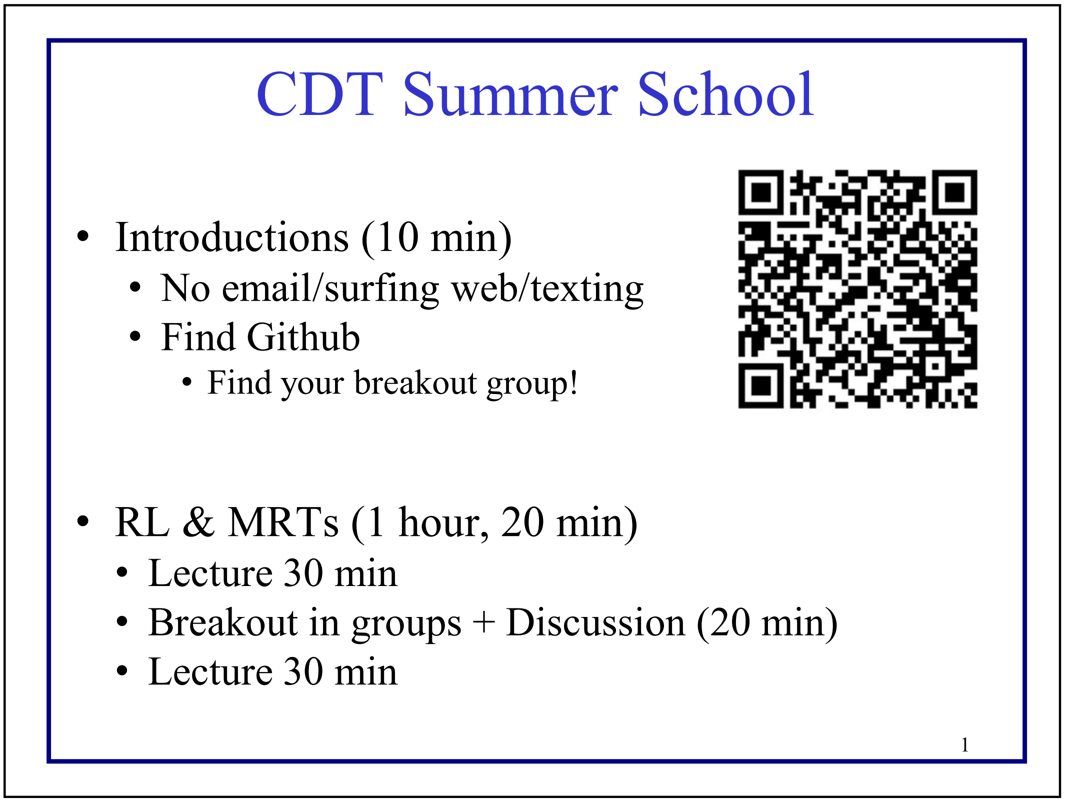
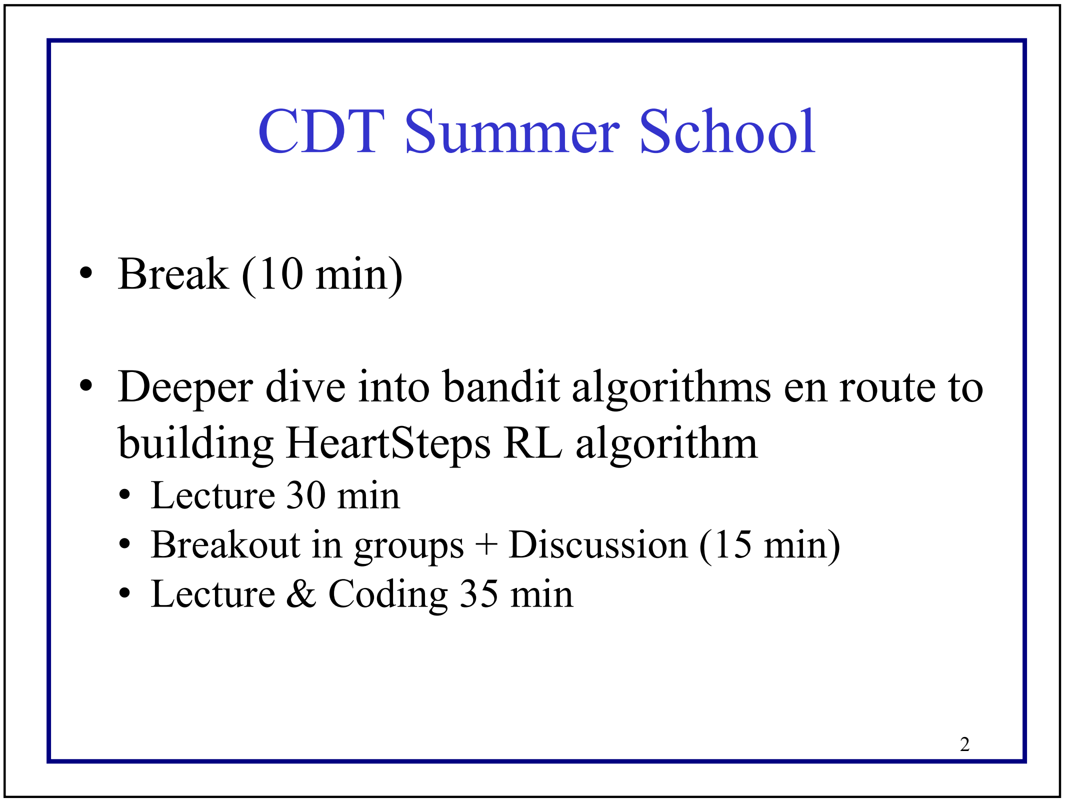


CDT Summer School

- Introductions (10 min)
  - No email/surfing web/texting
  - Find Github
    - Find your breakout group!
- RL & MRTs (1 hour, 20 min)
  - Lecture 30 min
  - Breakout in groups + Discussion (20 min)
  - Lecture 30 min

QR code for https://github.com/StatisticalReinforcementLearningLab/Stat-ML-CDT-2023/tree/main

(Introduce RL and Heartsteps): 3 Hours, 9.30 to 12.30 Monday morning



CDT Summer School

- Break (10 min)
- Deeper dive into bandit algorithms en route to building HeartSteps RL algorithm
  - Lecture 30 min
  - Breakout in groups + Discussion (15 min)
  - Lecture & Coding 35 min

https://github.com/StatisticalReinforcementLearningLab/Stat-ML-CDT-2023/tree/main/precourse

Reinforcement Learning & Micro-Randomization

Susan A Murphy

&

Raaz Dwivedi

some characteristics of digital health in clinical research settings

large no. of stakeholders with differing data needs

need to contribute to behavioral science

low signal to noise ratio with limited data -->intermixing behavioral science with data science

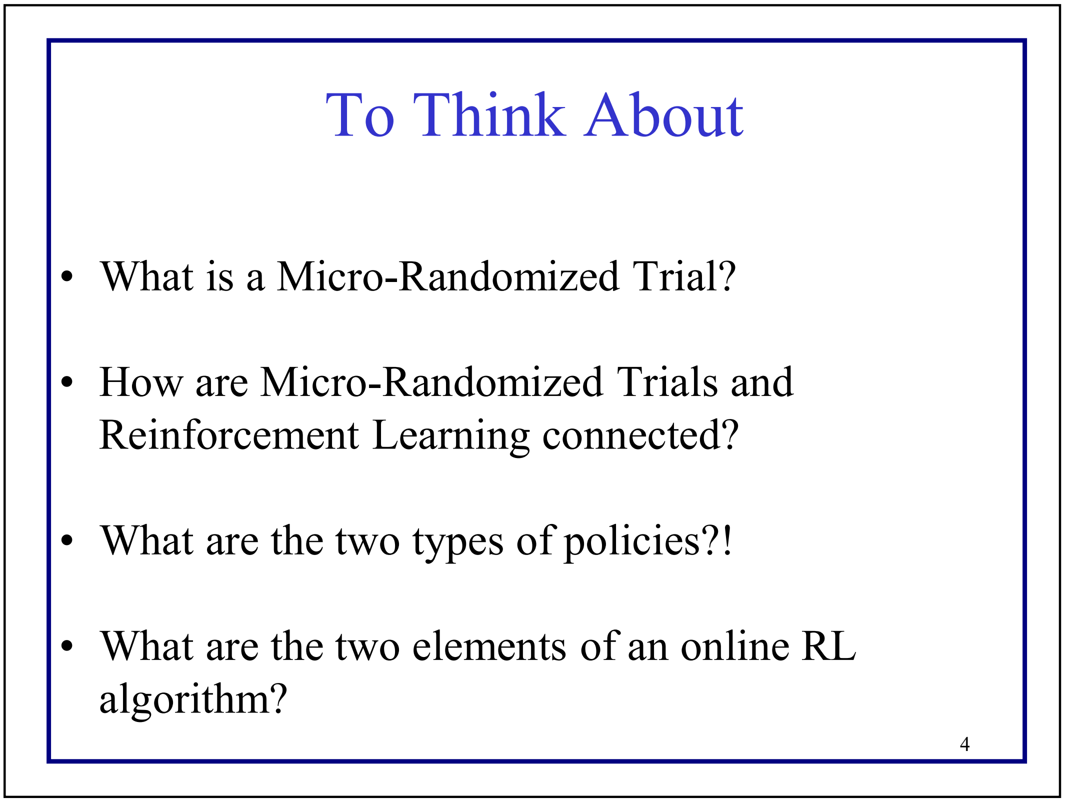
References: (see http://people.seas.harvard.edu/~samurphy/research.html)

Lecture/Practicum/Discussion 1: An introduction to micro-randomized trials, reinforcement learning and bandit algorithms for use in digital intervention development.
References:
1.a Liao,P., Klasnja, P., Tewari, P., Murphy, S.A. Sample Size Calculations for Micro-Randomized Trials in mHealth, Statistics in Medicine. 2016 May 30;35(12):1944-71. [2015 Dec 28 Epub ahead of print] PubMed PMID: 26707831, PMCID PMC4848174.
1.b Russo, D., Van Roy, B., Kazerouni, A., Osband, I., Wen, Z. A Tutorial on Thompson Sampling, https://arxiv.org/abs/1707.02038, Foundations and Trends® in Machine Learning 11 (1), 1-96
1.c Tewari, A. and Murphy, S.A., From Ads to Interventions: Contextual Bandits in Mobile Health, (2017) Mobile Health Sensors, Analytic Methods, and Applications , Springer International Publishing AG 2017, J.M. Rehg et al. (eds.), DOI

10.1007/978-3-319-51394-2_4, pgs 495-518

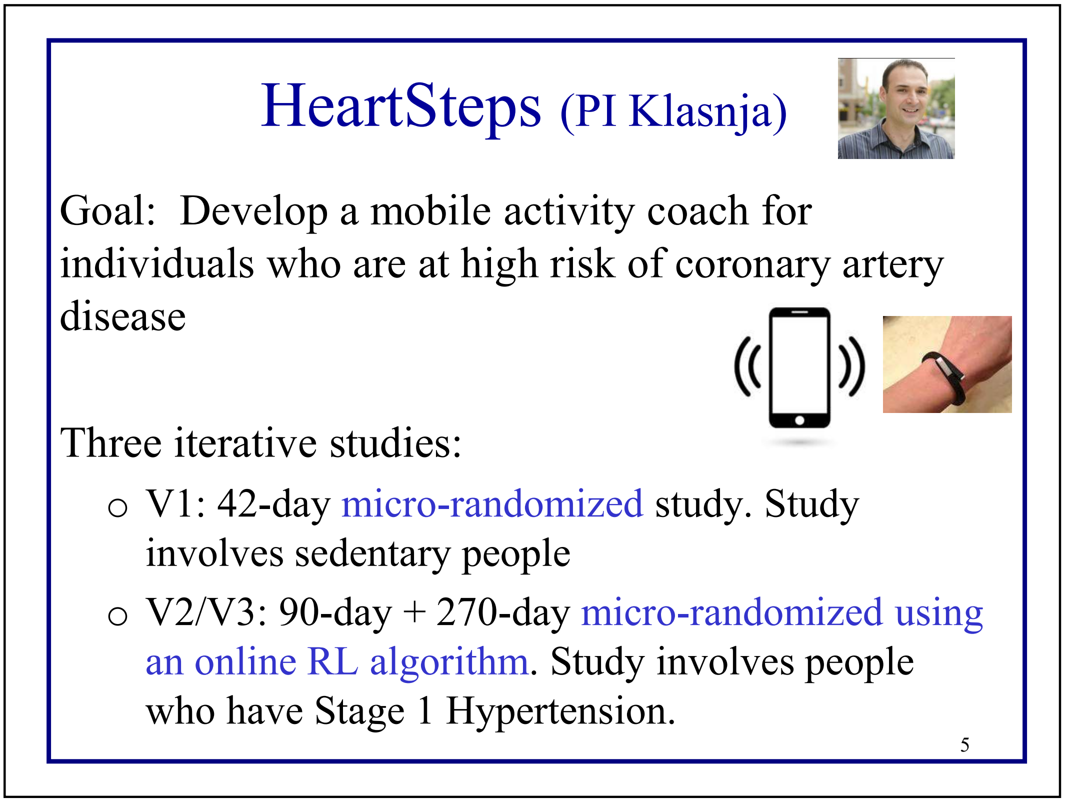
1.d Hadad Vitor, A Hirshberg David, Zhan Ruohan, Wager Stefan, Athey Susan. Confidence intervals for policy evaluation in adaptive experiments, https://www.pnas.org/doi/10.1073/pnas.2014602118

1.e Kelly Zhang, Lucas Janson, Susan Murphy. Inference for batched bandits. Advances in neural information processing systems 33, 9818-9829, 2020 https://arxiv.org/abs/2002.03217



To Think About

- What is a Micro-Randomized Trial?
- How are Micro-Randomized Trials and Reinforcement Learning connected?
- What are the two types of policies?!
- What are the two elements of an online RL algorithm?



HeartSteps (PI Klasnja)

Goal: Develop a mobile activity coach for individuals who are at high risk of coronary artery disease

Three iterative studies:

- V1: 42-day micro-randomized study. Study involves sedentary people
- V2/V3: 90-day + 270-day micro-randomized using an online RL algorithm. Study involves people who have Stage 1 Hypertension.

N=37 in V1   Data at https://github.com/klasnja/HeartStepsV1
N=42 from V2 and N=49 from V3 users—total =91

blood pressure that falls in the stage 1 hypertension range

The systolic pressure is 140 to 159 mm Hg or your diastolic pressure is 90 to 99 mm Hg

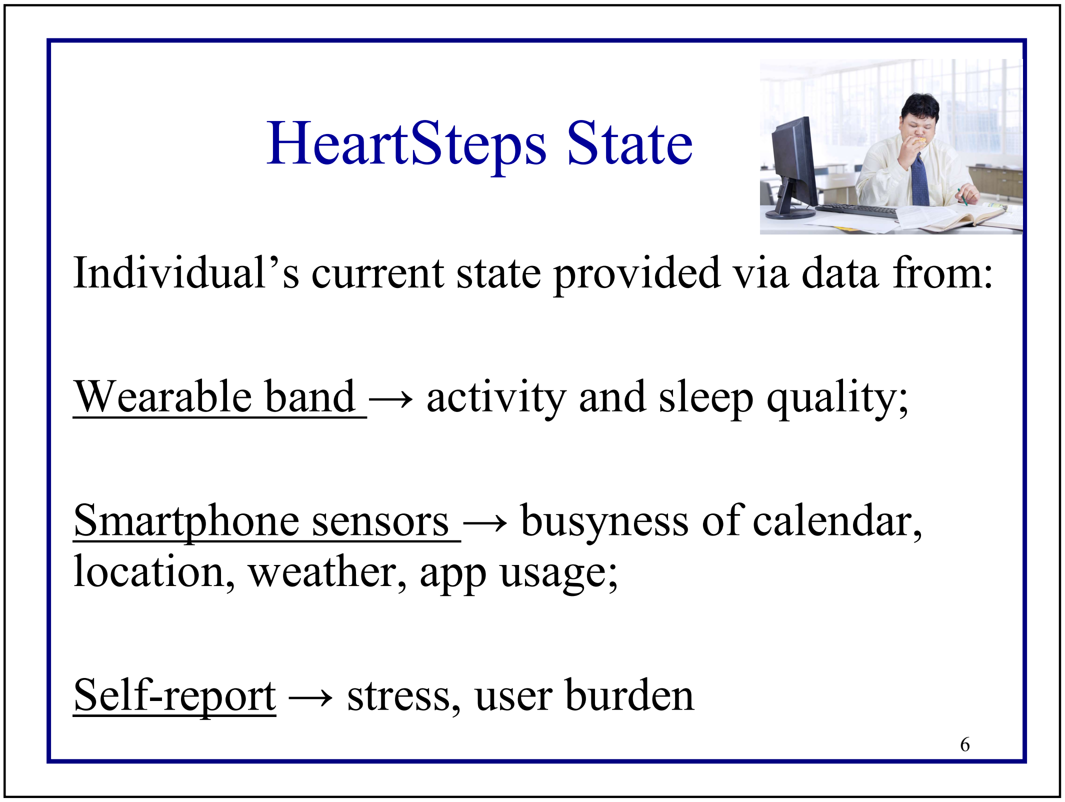
Changing your lifestyle can go a long way toward controlling high blood pressure.

Eating a heart-healthy diet with less salt

Getting regular physical activity

Maintaining a healthy weight or losing weight if you're overweight or obese

Limiting the amount of alcohol you drink



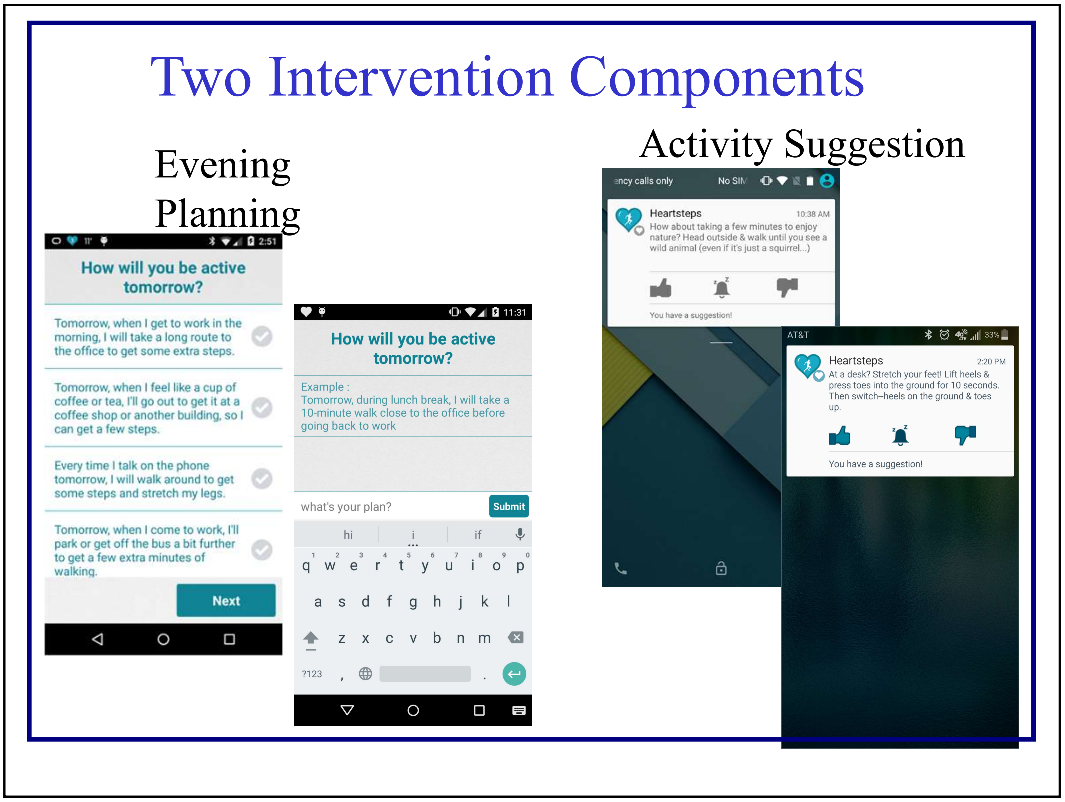
HeartSteps State

Individual’s current state provided via data from:

Wearable band → activity and sleep quality;

Smartphone sensors → busyness of calendar, location, weather, app usage;

Self-report → stress, user burden



Two Intervention Components

Evening Planning

How will you be active tomorrow?

Tomorrow, when I get to work in the morning, I will take a long route to the office to get some extra steps.

Tomorrow, when I feel like a cup of coffee or tea, I'll go out to get it at a coffee shop or another building, so I can get a few steps.

Every time I talk on the phone tomorrow, I will walk around to get some steps and stretch my legs.

Tomorrow, when I come to work, I'll park or get off the bus a bit further to get a few extra minutes of walking.

How will you be active tomorrow?

Example :
Tomorrow, during lunch break, I will take a 10-minute walk close to the office before going back to work

Activity Suggestion

Heartsteps 10:38 AM
How about taking a few minutes to enjoy nature? Head outside & walk until you see a wild animal (even if it's just a squirrel...)

You have a suggestion!

Heartsteps 2:20 PM
At a desk? Stretch your feet! Lift heels & press toes into the ground for 10 seconds. Then switch--heels on the ground & toes up.

You have a suggestion!

Activity suggestions are to help in your automated processing

Evening planning is to help in your reflective processing (controlled behavior)—this was in V1; changed to a weekly intervention in V2/V3

There are many intervention components that make up a mobile health intervention. We only experiment with a few.

Example Intervention components
•Whether to provide an intervention or whether to prompt self-monitoring
•How to deliver an intervention option (via a message on wearable, smartphone notification, SMS)
•"Provide nothing" option

Reminders

Suggestions, tips, motivational messages

Prompts to set goals, complete self-report…

Rewards for goal attainment

Recommendations
Reach out recommendation (contact a friend)
Behavioral strategies (exercise; stay in locations that are supportive of change)
Cognitive strategies (relaxation; reframing)
Motivational messages (reasons for behavior change; barriers for change);

Setting goals; modifying goals
Feedback (often with visualization: fish; flower; garden)
Distractions (game, music, etc.)

Can use sensing and user modeling to determine right delivery time

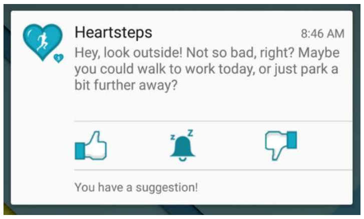
Don’t rely on user’s awareness of times of need or remembering to access

But…

High burden

HeartSteps: Activity Suggestions Component

- 5 time points per day (set according to user's schedule)



Heartsteps 8:46 AM

Hey, look outside! Not so bad, right? Maybe you could walk to work today, or just park a bit further away?

You have a suggestion!

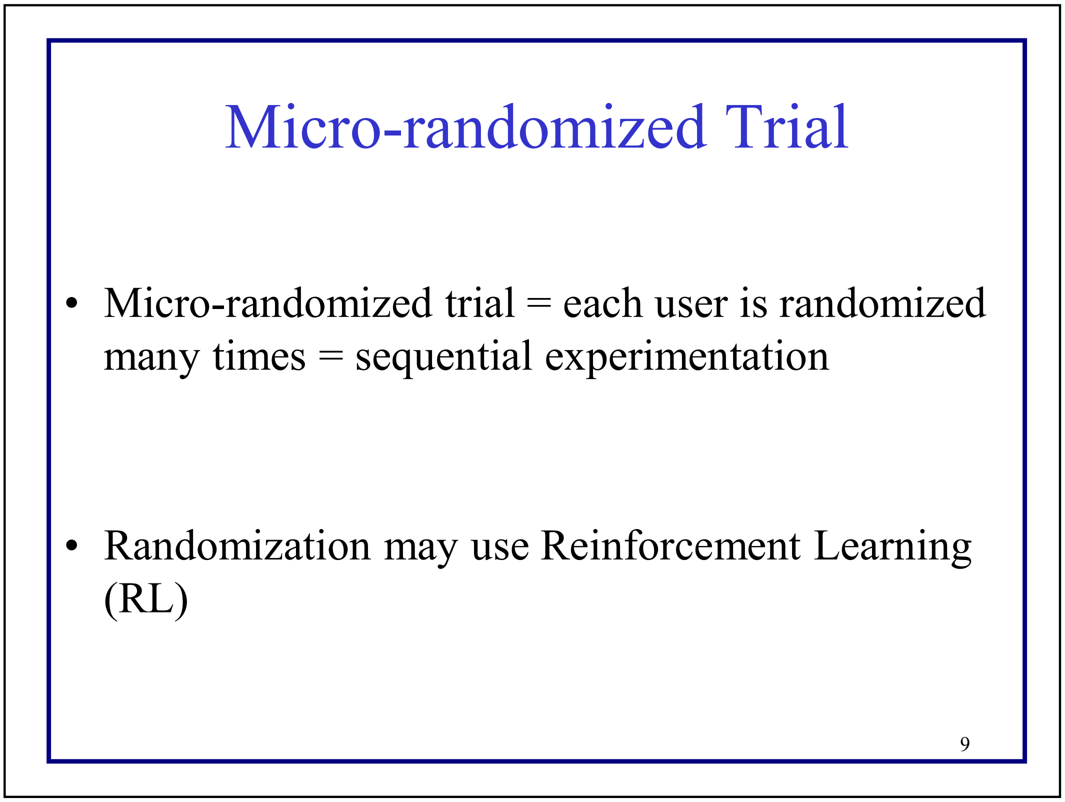
•Treatment (Action): Contextually tailored activity suggestion (deliver or not deliver)

•Outcome (Reward): 30-minute step count following each time point.

One of many components (good morning message, anti-sedentary message, self-monitoring…)

morning, mid-day, mid-afternoon, early evening, after dinner

Reward: Frequently the actions are primarily designed to have a near-term effect on the individual.   E.g. Help then manage current craving/stress, help them manage or be aware of the impact of their social setting on their craving/stress



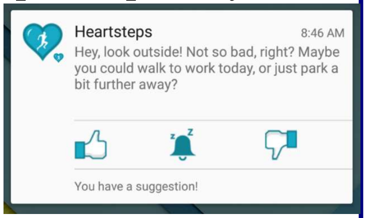
Micro-randomized Trial

- Micro-randomized trial = each user is randomized many times = sequential experimentation

- Randomization may use Reinforcement Learning (RL)

HeartSteps V1

- 210 decision times per user: 5 time points per day



Heartsteps 8:46 AM

Hey, look outside! Not so bad, right? Maybe you could walk to work today, or just park a bit further away?

You have a suggestion!

•Treatment (Action): Contextually tailored activity suggestion (deliver or not deliver); if user is available then randomize with probability 0.6

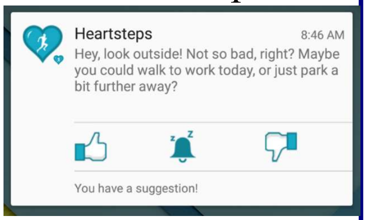
•Outcome (Reward): 30-minute step count following each time point.

Define availability at a decision time (available if definitely not driving a vehicle, not too active recently, has not turned off delivery in HeartSteps settings…)

“micro-randomized” actions in V1
Suggestion tailored to location, weather, day of week, time of day

HeartSteps V2/V3

- 450 or > 1000 decision times per user: 5 time points per day



Heartsteps 8:46 AM

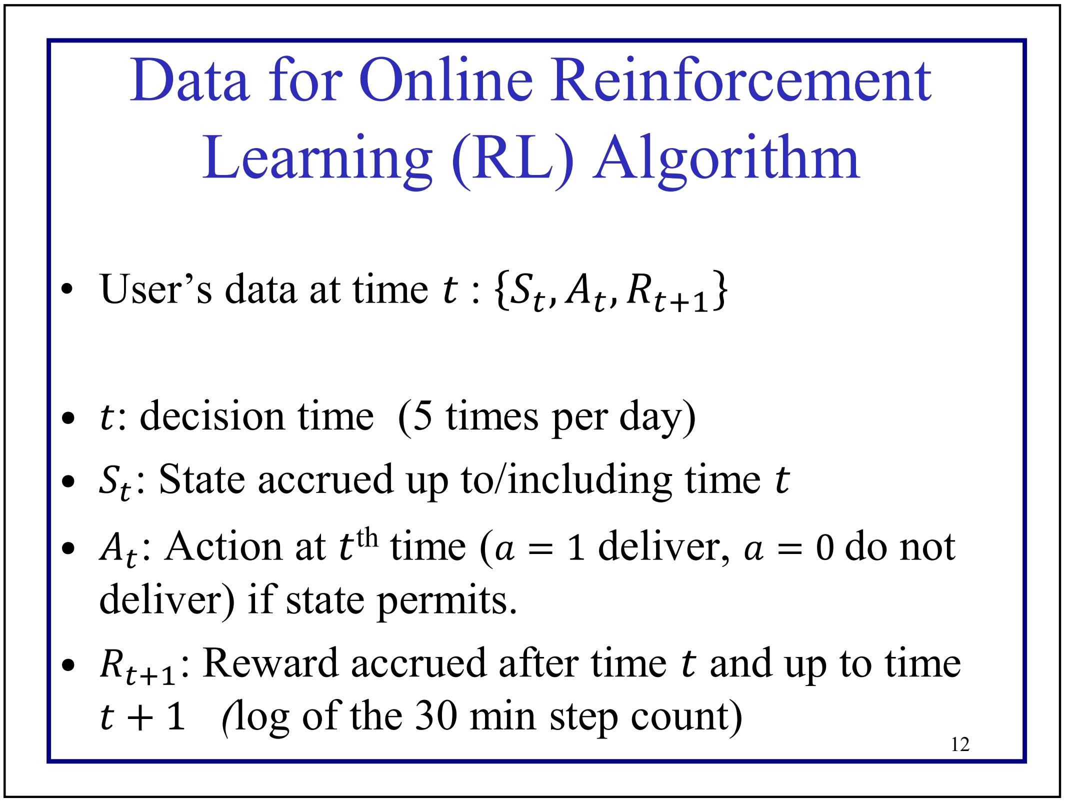
Hey, look outside! Not so bad, right? Maybe you could walk to work today, or just park a bit further away?

You have a suggestion!

•Treatment (Action): Contextually tailored activity suggestion (deliver or not deliver); randomization via an online RL algorithm

•Outcome (Reward): 30-minute step count following each time point.

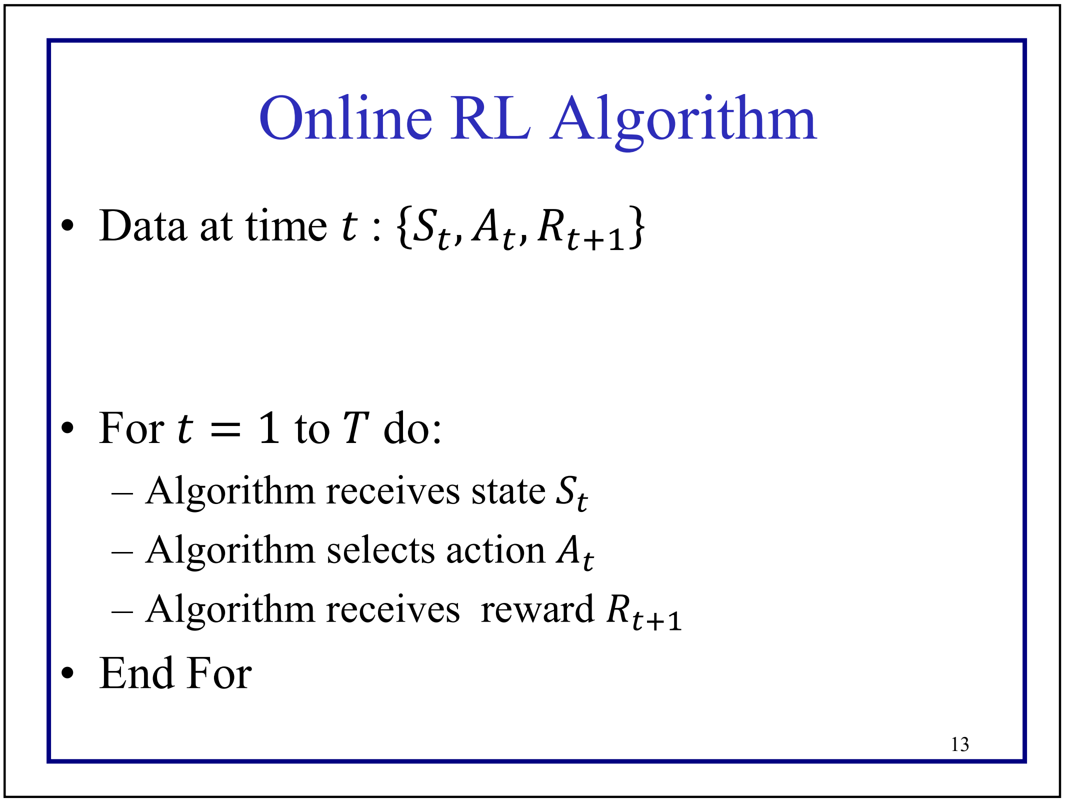
Suggestion tailored to location, weather, day of week, time of day



Data for Online Reinforcement Learning (RL) Algorithm

- User's data at time t : $\{S_t, A_t, R_{t+1}\}$

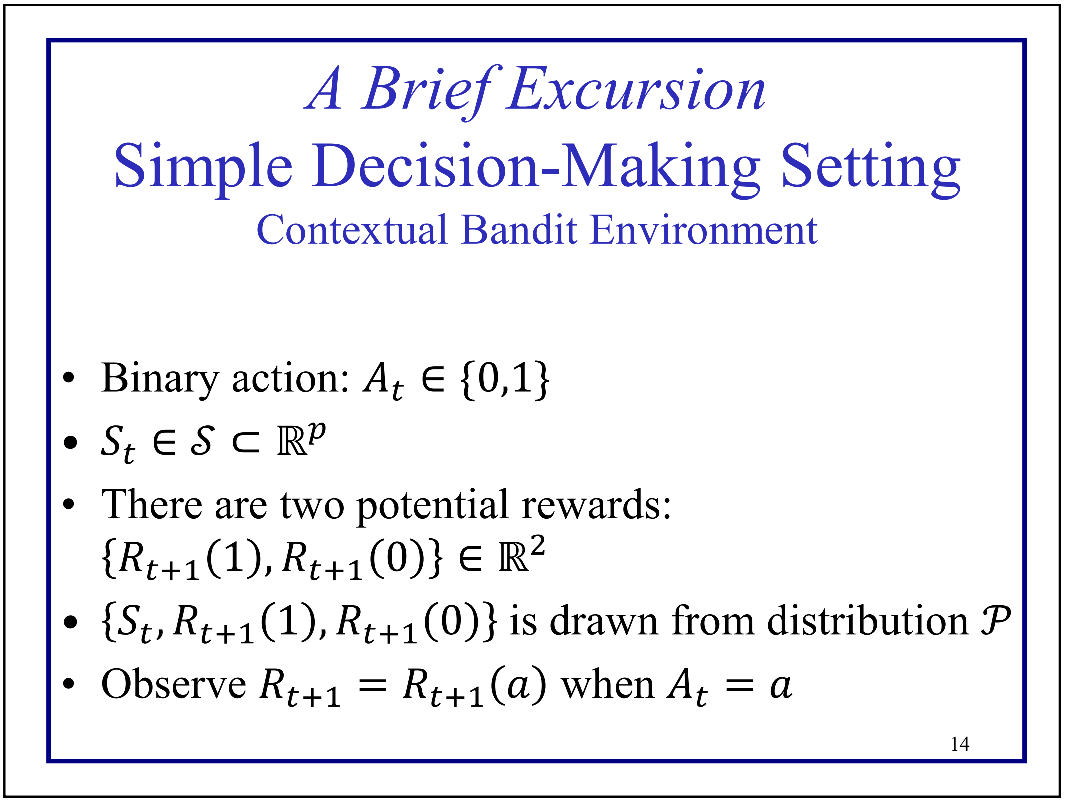
- t: decision time (5 times per day)
- S_t: State accrued up to/including time t
- A_t: Action at t^{th} time ($a = 1$ deliver, $a = 0$ do not deliver) if state permits.
- R_{t+1}: Reward accrued after time t and up to time $t + 1$ (log of the 30 min step count)



Online RL Algorithm

- Data at time t : $\{S_t, A_t, R_{t+1}\}$

- For $t = 1$ to T do:
  - – Algorithm receives state S_t
  - – Algorithm selects action A_t
  - – Algorithm receives reward R_{t+1}
- End For



A Brief Excursion

Simple Decision-Making Setting

Contextual Bandit Environment

- Binary action: $A_t \in \{0,1\}$
- $S_t \in \mathcal{S} \subset \mathbb{R}^p$
- There are two potential rewards: $\{R_{t+1}(1), R_{t+1}(0)\} \in \mathbb{R}^2$
- $\{S_t, R_{t+1}(1), R_{t+1}(0)\}$ is drawn from distribution $\mathcal{P}$
- Observe $R_{t+1} = R_{t+1}(a)$ when $A_t = a$

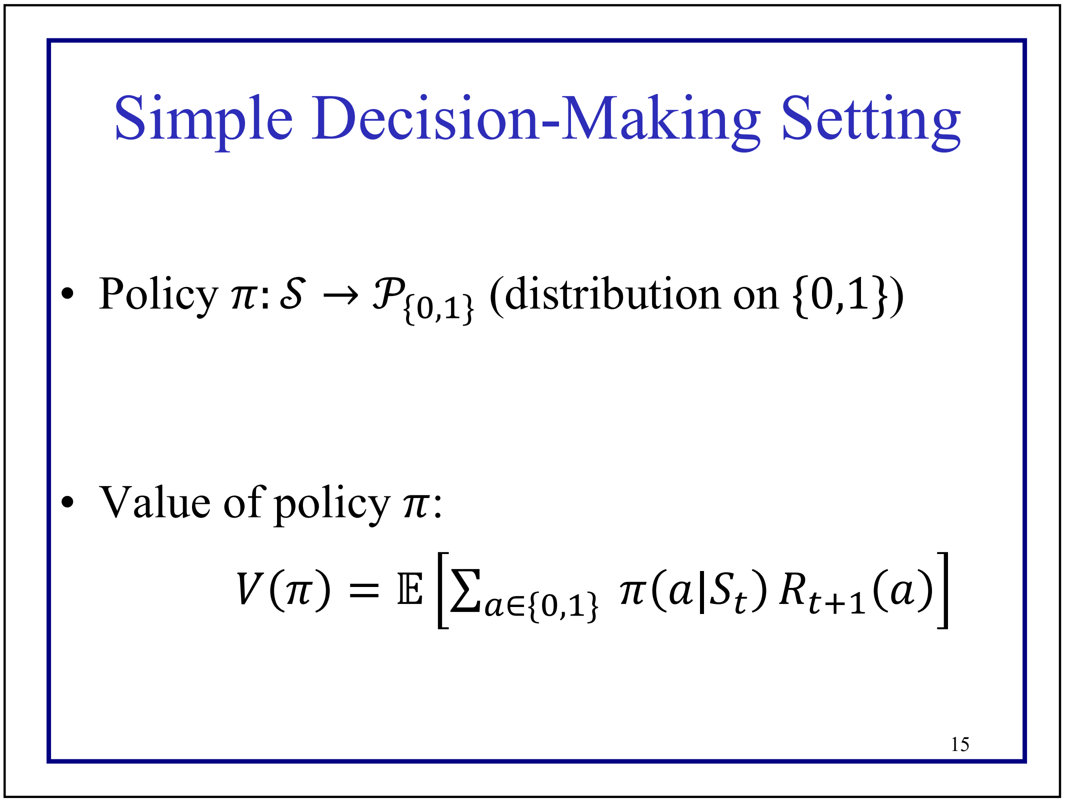
This is often called a "contextual bandit environment" (S_t does not depend on prior actions and the reward, R_{t+1} only depends on time t state, action)

We only get to observe one of the two rewards (one of two labels) depending on which action is selected.

This is what makes RL different from

Supervised ML in which you usually see all labels

Unsupervised ML in which you see no labels



Simple Decision-Making Setting

- Policy $\pi\colon \mathcal{S} \rightarrow \mathcal{P}_{\{0,1\}}$ (distribution on $\{0,1\}$)

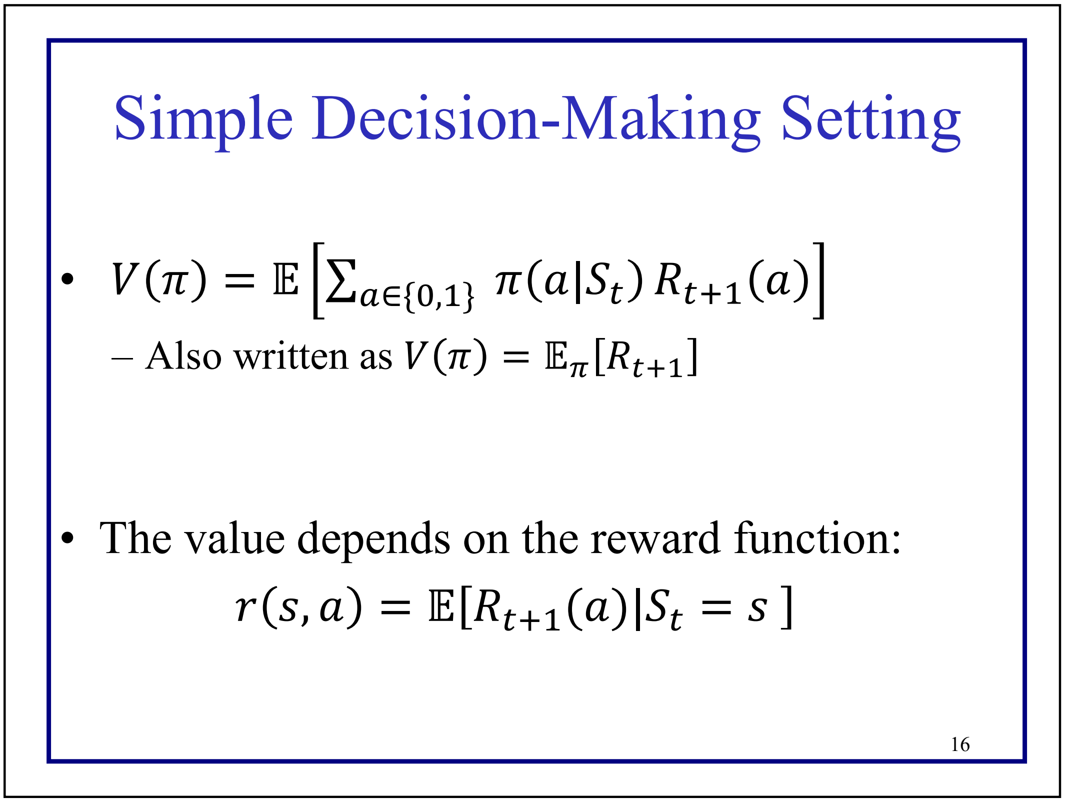
- Value of policy π:

$$V(\pi) = \mathbb{E}\left[\sum_{a\in\{0,1\}} \pi(a|S_t)\, R_{t+1}(a)\right]$$

Policy π may be degenerate (AKA, select a particular action with probability =1 for each state)

$\mathbb{E} == E_{\{S_t, R_{t+1}(1), R_{t+1}(0)\}\sim\mathcal{P}}$

In this simple bandit environment setting people often call the value the expected reward.



Simple Decision-Making Setting

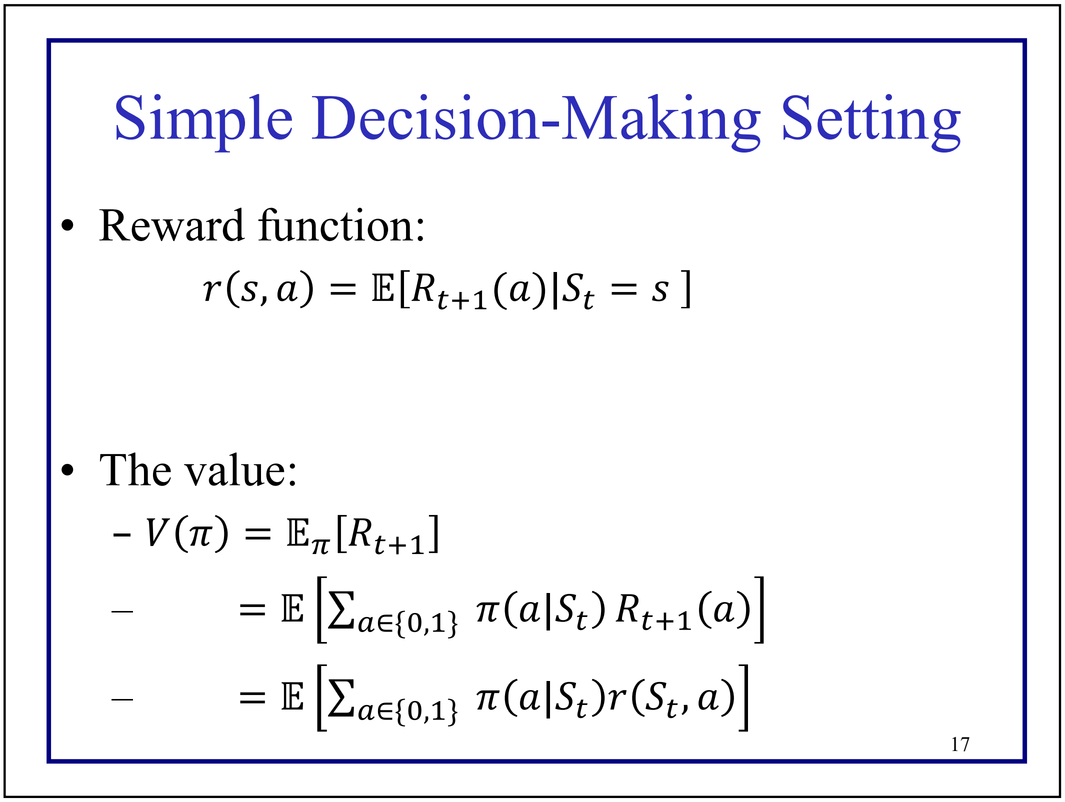
- $V(\pi) = \mathbb{E}\left[\sum_{a\in\{0,1\}} \pi(a|S_t)\, R_{t+1}(a)\right]$
  - Also written as $V(\pi) = \mathbb{E}_{\pi}[R_{t+1}]$

- The value depends on the reward function:

$$r(s,a) = \mathbb{E}[R_{t+1}(a)|S_t = s\ ]$$

Policy π may be degenerate (AKA, select a particular action with probability =1 for each state)

$\mathbb{E} == E_{\{S_t,R_{t+1}(1),R_{t+1}(0)\}\sim\mathcal{P}}$



Simple Decision-Making Setting

- Reward function:

$$r(s,a) = \mathbb{E}[R_{t+1}(a) | S_t = s\ ]$$

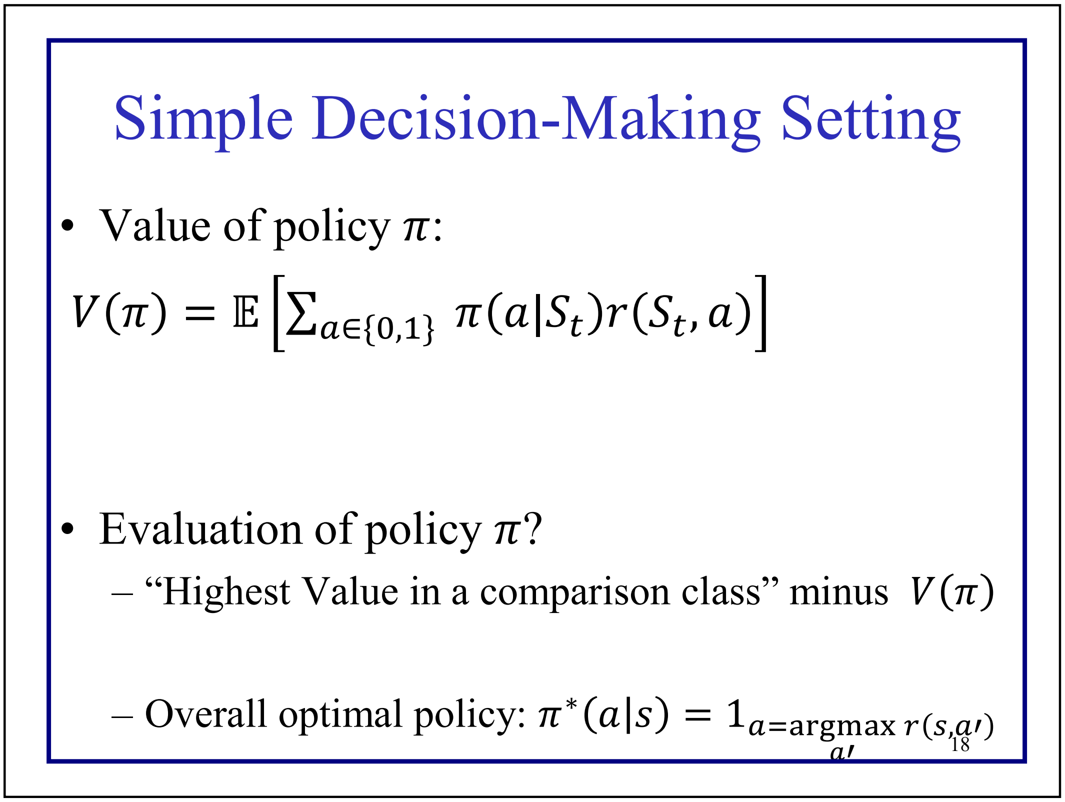
- The value:
  - $V(\pi) = \mathbb{E}_\pi[R_{t+1}]$
  - $\qquad = \mathbb{E}\left[\sum_{a\in\{0,1\}} \pi(a|S_t)\, R_{t+1}(a)\right]$
  - $\qquad = \mathbb{E}\left[\sum_{a\in\{0,1\}} \pi(a|S_t) r(S_t,a)\right]$

$$
\begin{aligned}
V(\pi) &= \mathbb{E}_\pi[R_{t+1}] \\
&= \mathbb{E}\left[\sum_{a\in\{0,1\}} \pi(a|S_t)\, R_{t+1}(a)\right] \\
&= \mathbb{E}\left[\sum_{a\in\{0,1\}} \pi(a|S_t) r(S_t,a)\right] \\
&= \mathbb{E}_\pi[r(S_t,A_t)]
\end{aligned}
$$

In this simple bandit environment setting people often call the value the expected reward.

Recall from slide 13 the algorithm only sees S_t (does not see $R_{t+1}(1)$, $R_{t+1}(0)$, prior to selecting the action A_t) thus conditional on S_t , A_t is independent of $R_{t+1}(1)$, $R_{t+1}(0)$. We obtain:

$$
\begin{aligned}
r(s,a) &= \mathbb{E}[R_{t+1}(a) | S_t = s\ ] \\
&= \mathbb{E}[R_{t+1}(a) || S_t = s, A_t = a] \\
&= \mathbb{E}[R_{t+1} | S_t = s, A_t = a]
\end{aligned}
$$



Simple Decision-Making Setting

- Value of policy π:

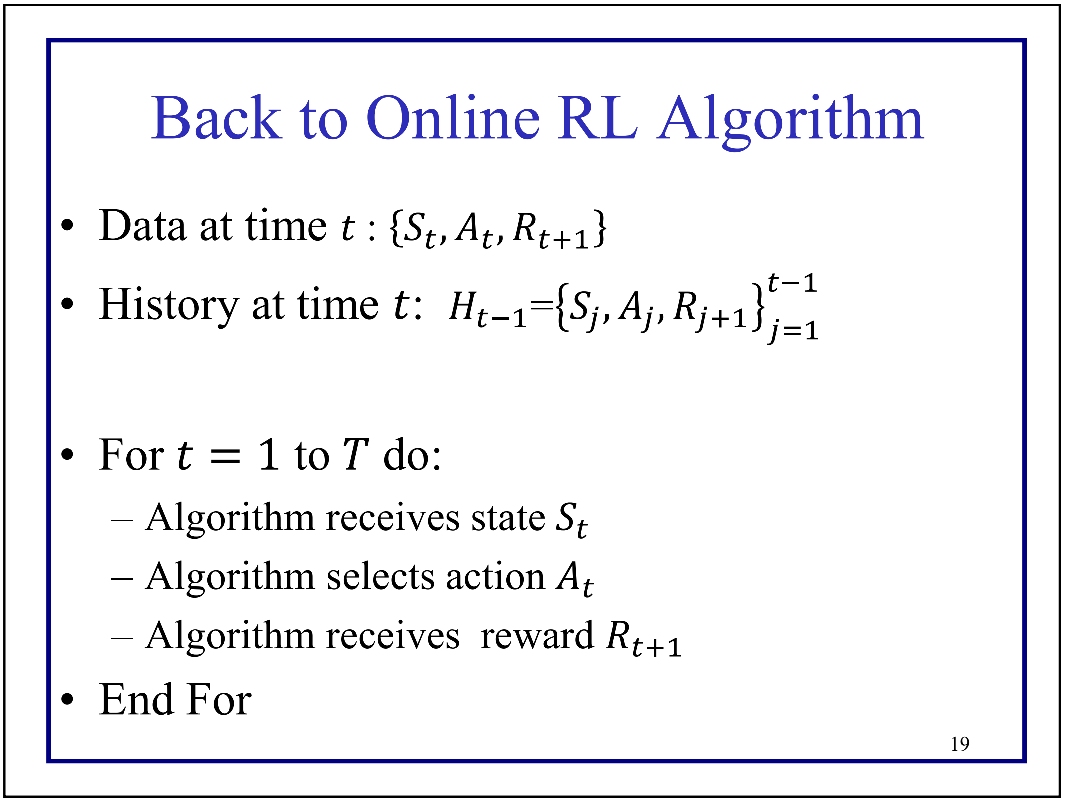
$$V(\pi) = \mathbb{E}\left[\sum_{a\in\{0,1\}} \pi(a|S_t)r(S_t,a)\right]$$

- Evaluation of policy π?
  - "Highest Value in a comparison class" minus $V(\pi)$
  - Overall optimal policy: $\pi^*(a|s) = 1_{a=\underset{a'}{\operatorname{argmax}}\, r(s,a')}$

18

The classical comparison class contains π^*

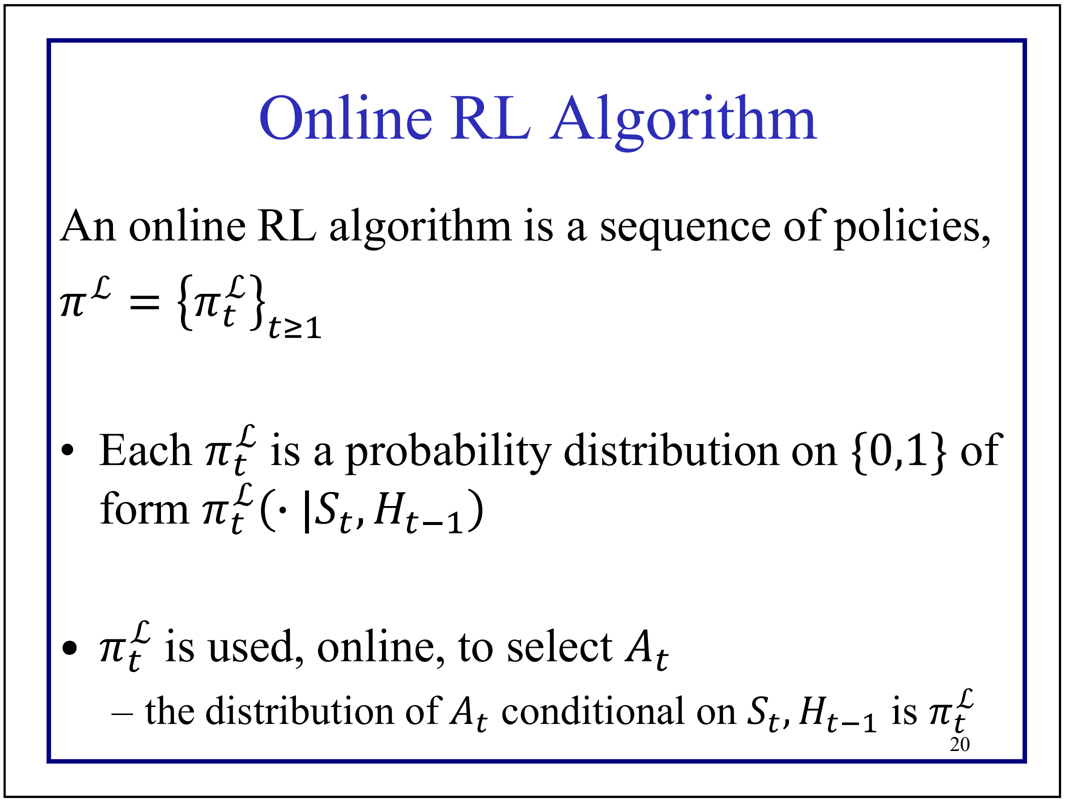
However in real life settings this may not be the case due to constraints that have to be imposed on the set of possible policies.



Back to Online RL Algorithm

- Data at time t : $\{S_t, A_t, R_{t+1}\}$
- History at time t: $H_{t-1}=\{S_j, A_j, R_{j+1}\}_{j=1}^{t-1}$

- For $t = 1$ to T do:
  - – Algorithm receives state S_t
  - – Algorithm selects action A_t
  - – Algorithm receives reward R_{t+1}
- End For



Online RL Algorithm

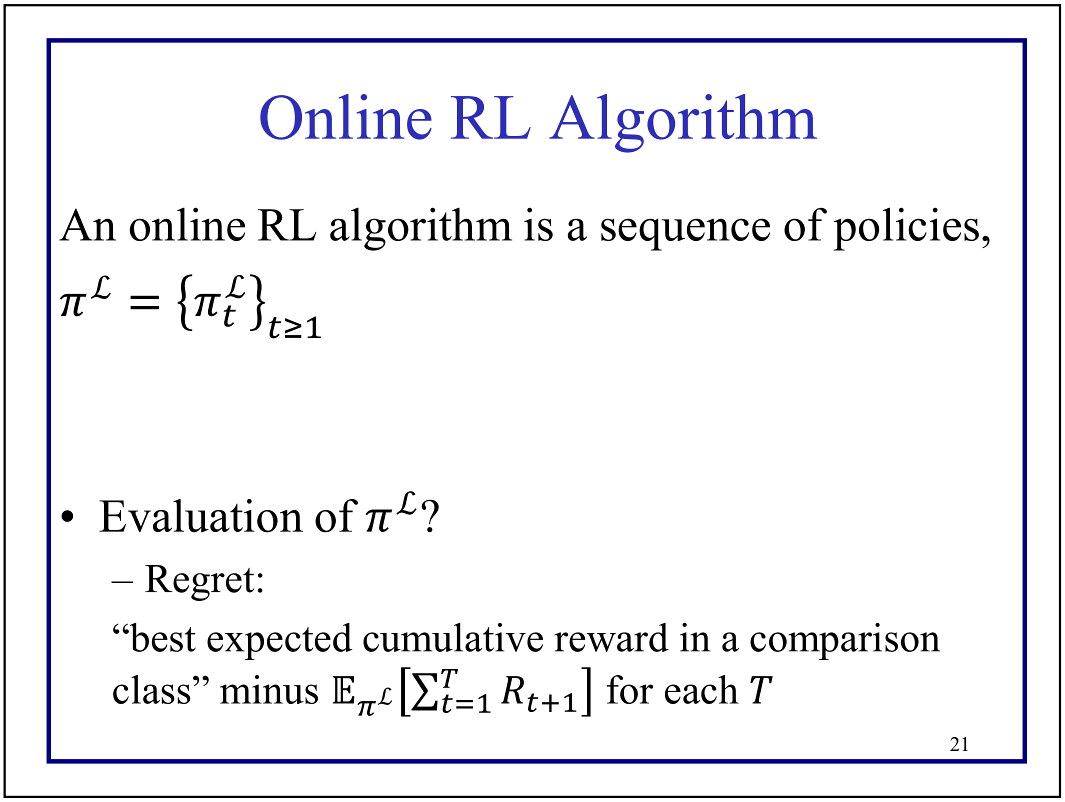
An online RL algorithm is a sequence of policies, $\pi^{\mathcal{L}} = \{\pi_t^{\mathcal{L}}\}_{t\geq 1}$

- Each $\pi_t^{\mathcal{L}}$ is a probability distribution on $\{0,1\}$ of form $\pi_t^{\mathcal{L}}(\cdot \,|S_t, H_{t-1})$

- $\pi_t^{\mathcal{L}}$ is used, online, to select A_t
  - the distribution of A_t conditional on S_t, H_{t-1} is $\pi_t^{\mathcal{L}}$

Use of $\pi_t^{\mathcal{L}}$ to select A_t is micro randomization…

History: H_{t-1}=$\{S_j, A_j, R_{j+1}\}_{j=1}^{t-1}$

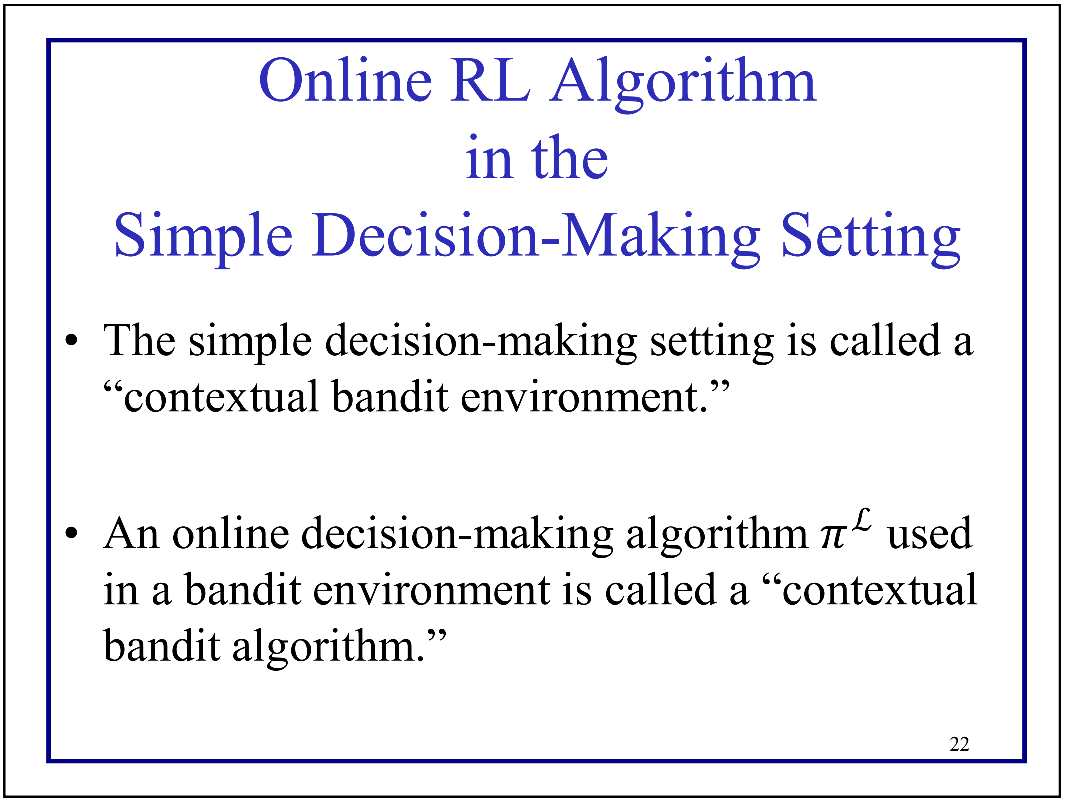


Online RL Algorithm

An online RL algorithm is a sequence of policies, $\pi^{\mathcal{L}} = \{\pi_t^{\mathcal{L}}\}_{t \geq 1}$

- Evaluation of $\pi^{\mathcal{L}}$?
  - Regret:

    "best expected cumulative reward in a comparison class" minus $\mathbb{E}_{\pi^{\mathcal{L}}}\left[\sum_{t=1}^{T} R_{t+1}\right]$ for each T



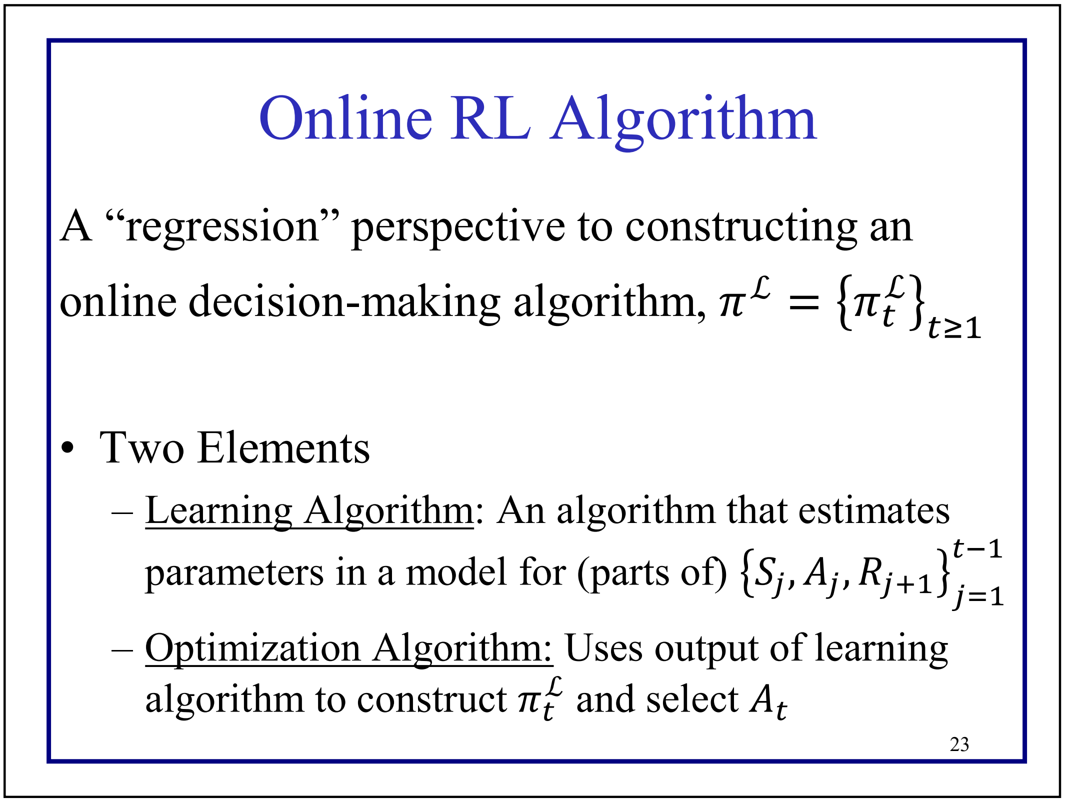
Online RL Algorithm in the Simple Decision-Making Setting

- The simple decision-making setting is called a “contextual bandit environment.”
- An online decision-making algorithm $\pi^{\mathcal{L}}$ used in a bandit environment is called a “contextual bandit algorithm.”

Raaz will discuss a very nice online decision making alg called a “Linear Thompson-Sampling” Algorithm.

For a great tutorial see Sections 1-6 in

Russo, D., Van Roy, B., Kazerouni, A., Osband, I., Wen, Z. A Tutorial on Thompson Sampling, https://arxiv.org/abs/1707.02038, Foundations and Trends® in Machine Learning 11 (1), 1-96



Online RL Algorithm

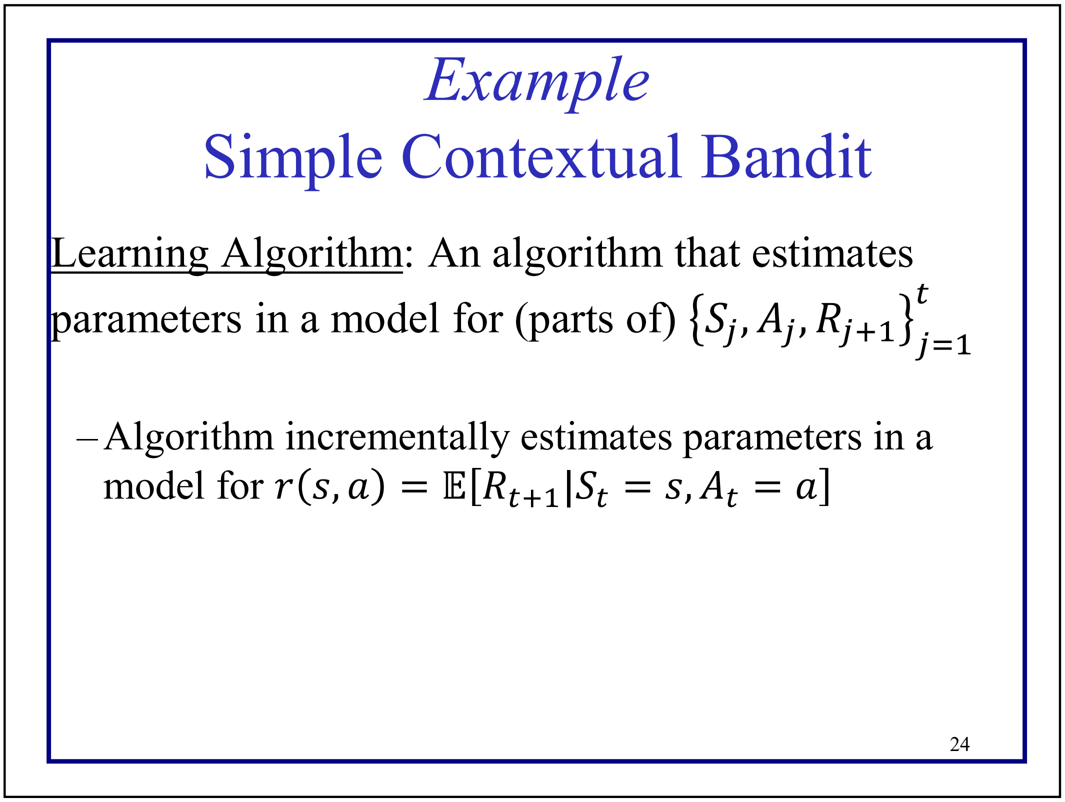
A "regression" perspective to constructing an online decision-making algorithm, $\pi^{\mathcal{L}} = \{\pi_t^{\mathcal{L}}\}_{t \geq 1}$

- Two Elements
  - <u>Learning Algorithm</u>: An algorithm that estimates parameters in a model for (parts of) $\{S_j, A_j, R_{j+1}\}_{j=1}^{t-1}$
  - <u>Optimization Algorithm:</u> Uses output of learning algorithm to construct $\pi_t^{\mathcal{L}}$ and select A_t

In our class we focus on the "regression" perspective. Feels as if we are doing supervised learning even though that is not true…..

The other common perspective is a classification perspective (e.g. policy search)

Many methods combine the two perspectives. Actor-Critic algorithms do this.



Example
Simple Contextual Bandit

Learning Algorithm: An algorithm that estimates parameters in a model for (parts of) $\{S_j, A_j, R_{j+1}\}_{j=1}^{t}$

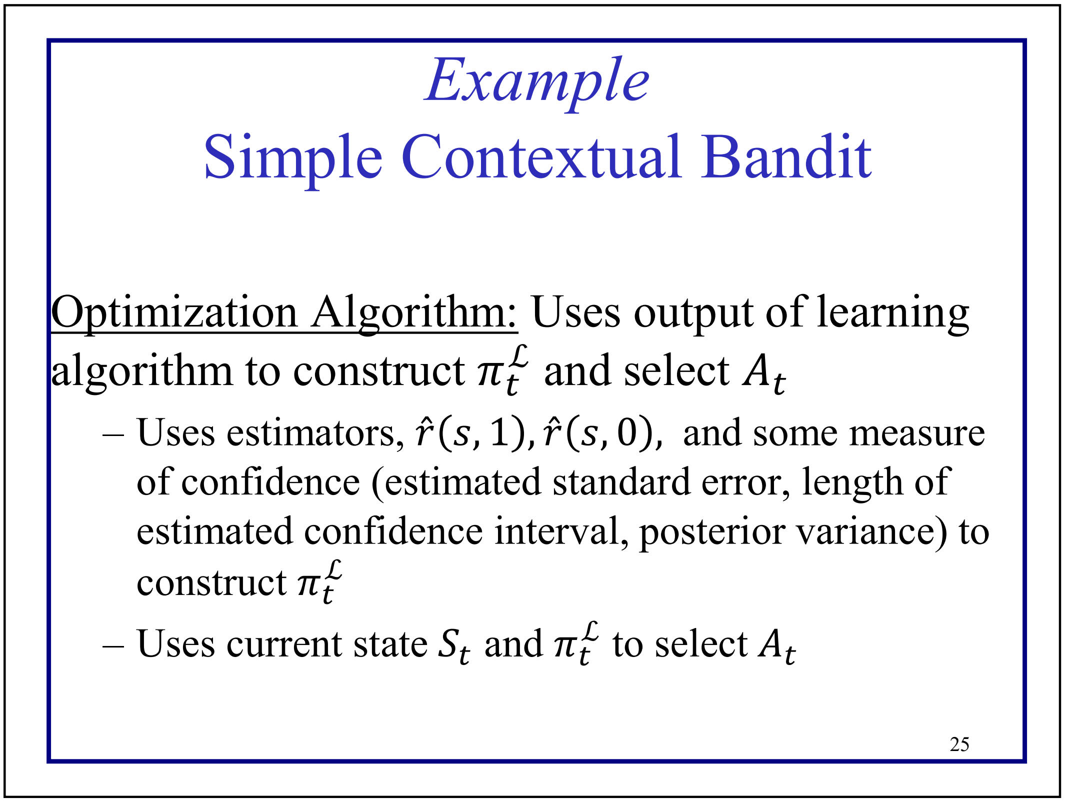
- Algorithm incrementally estimates parameters in a model for $r(s,a) = \mathbb{E}[R_{t+1} | S_t = s, A_t = a]$

R_{t+1} is the target

It is via the following that we turn an RL problem into a regression problem (supervised learning)::::::

Recall from slide 13,19 the algorithm only sees $\mathrm{S_t}$ (does not see $\mathrm{R_{t+1}}(1)$, $\mathrm{R_{t+1}}(0)$, prior to selecting the action $\mathrm{A_t}$) thus conditional on $\mathrm{S_t}$, $\mathrm{A_t}$ is independent of $\mathrm{R_{t+1}}(1)$, $\mathrm{R_{t+1}}(0)$. We obtain:

$$r(s,a) = \mathbb{E}[R_{t+1}(a) | S_t = s\,]$$
$$= \mathbb{E}[R_{t+1}(a) || S_t = s, A_t = a]$$
$$= \mathbb{E}[R_{t+1} | S_t = s, A_t = a]$$



Example
Simple Contextual Bandit

Optimization Algorithm: Uses output of learning algorithm to construct $\pi_t^{\mathcal{L}}$ and select A_t

- Uses estimators, $\hat{r}(s,1), \hat{r}(s,0),$ and some measure of confidence (estimated standard error, length of estimated confidence interval, posterior variance) to construct $\pi_t^{\mathcal{L}}$
- Uses current state S_t and $\pi_t^{\mathcal{L}}$ to select A_t

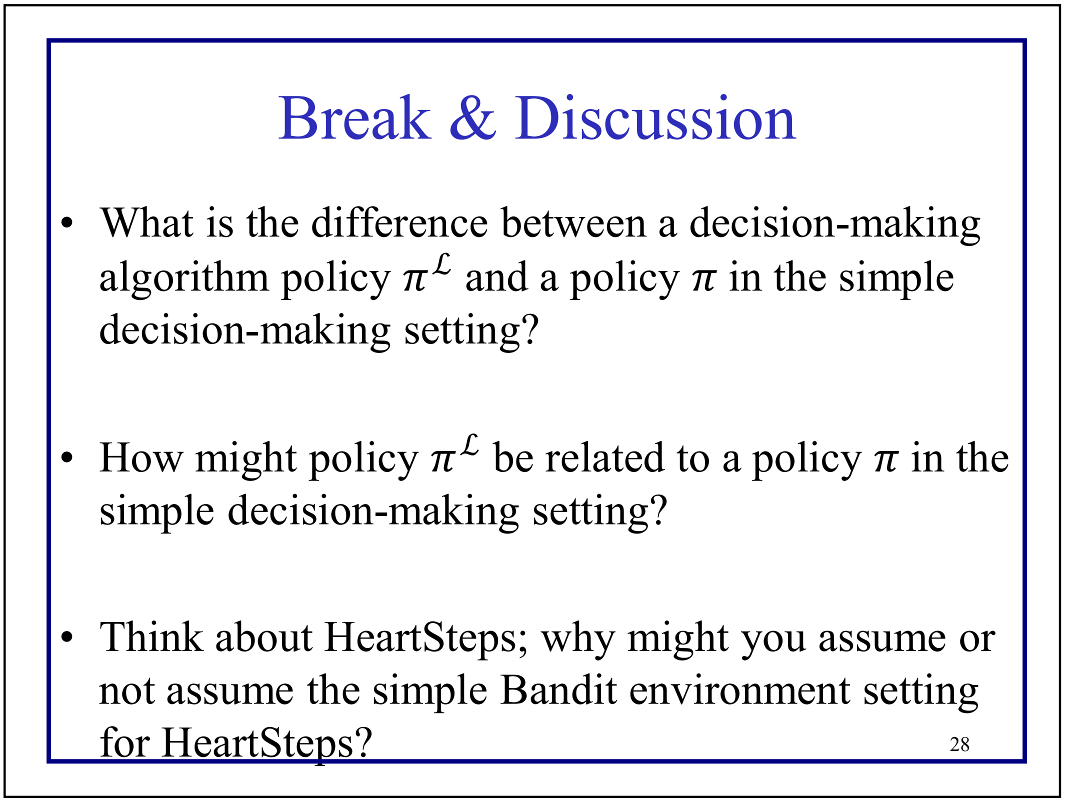
Example
Simple Contextual Bandit

- Learning Algorithm: An algorithm that estimates parameters in a model for (parts of) $\{S_j, A_j, R_{j+1}\}_{j=1}^{t}$
  - Algorithm incrementally estimates parameters in a model for $r(s,a) = \mathbb{E}\left[R_{t+1} | S_t = s, A_t = a\right]$

- Optimization Algorithm: Uses output of learning algorithm to construct $\pi_t^{\mathcal{L}}$ and select A_t
  - Uses estimators, $\hat{r}(s,1), \hat{r}(s,0)$, and some measure of confidence (estimated standard error, length of estimated confidence interval, posterior variance) to construct $\pi_t^{\mathcal{L}}$

RAAZ WILL COVER!!

20 min for Discussion!



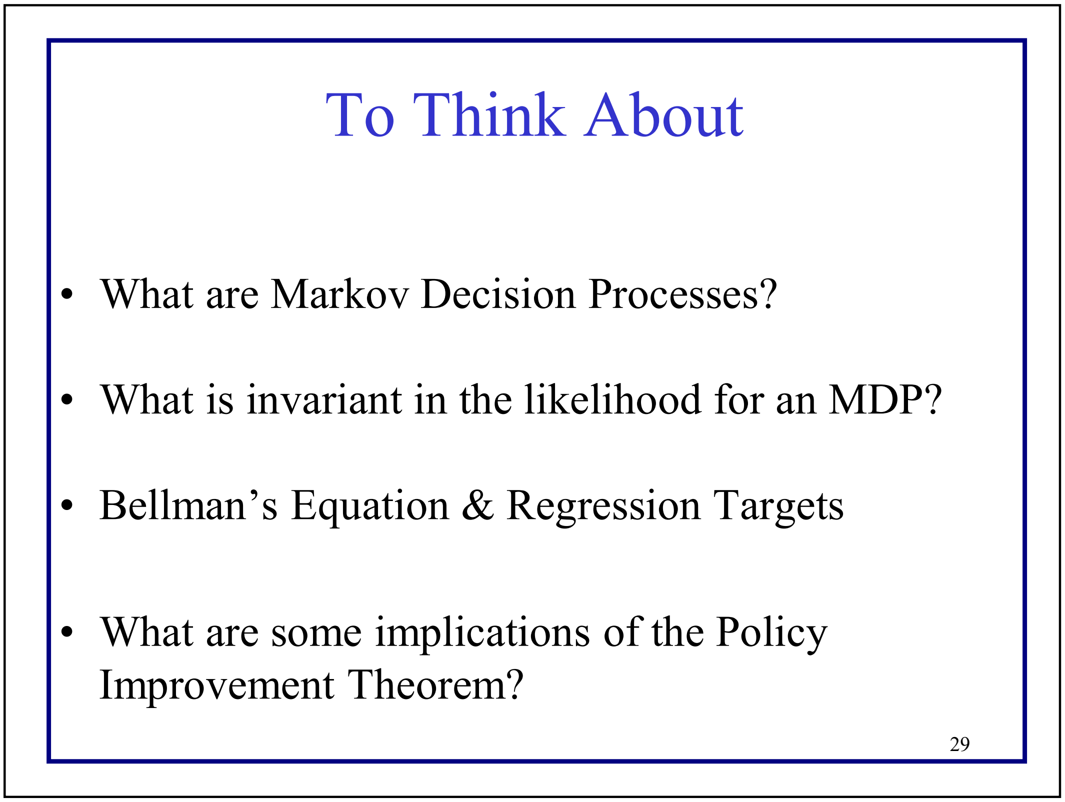
Break & Discussion

- What is the difference between a decision-making algorithm policy $\pi^{\mathcal{L}}$ and a policy π in the simple decision-making setting?

- How might policy $\pi^{\mathcal{L}}$ be related to a policy π in the simple decision-making setting?

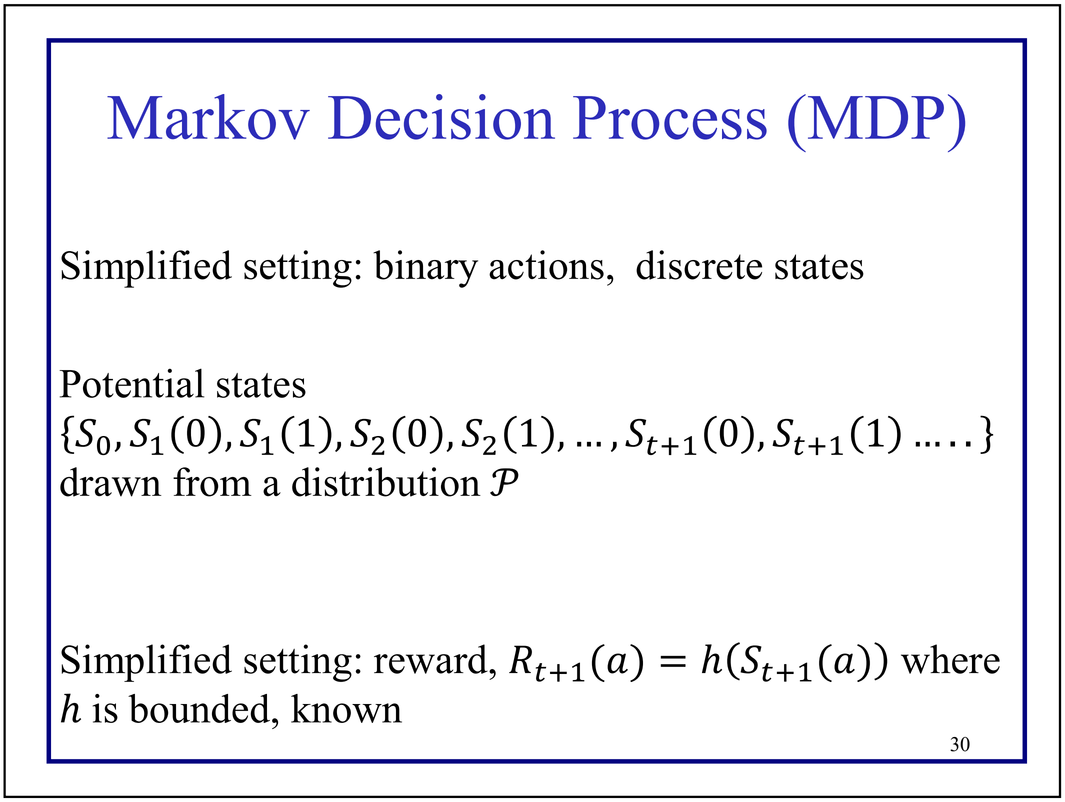
- Think about HeartSteps; why might you assume or not assume the simple Bandit environment setting for HeartSteps?

28



To Think About

- What are Markov Decision Processes?
- What is invariant in the likelihood for an MDP?
- Bellman’s Equation & Regression Targets
- What are some implications of the Policy Improvement Theorem?

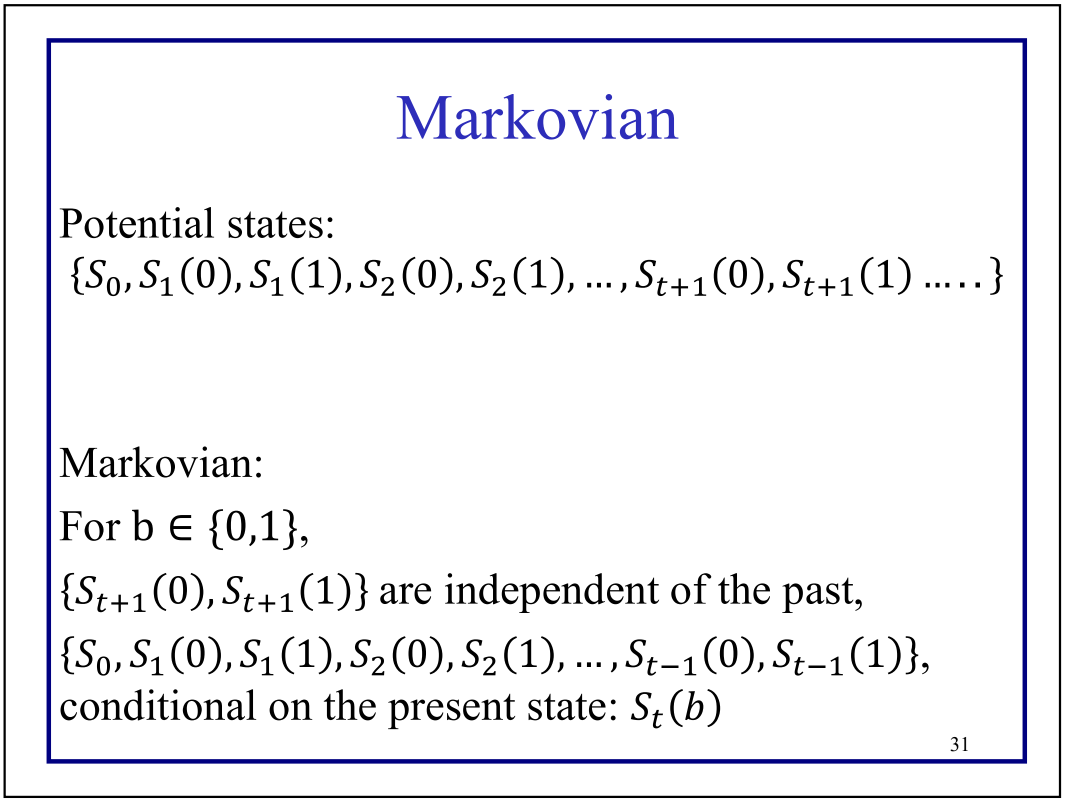


Markov Decision Process (MDP)

Simplified setting: binary actions, discrete states

Potential states
$\{S_0, S_1(0), S_1(1), S_2(0), S_2(1), \ldots, S_{t+1}(0), S_{t+1}(1) \ldots \ldots\}$
drawn from a distribution $\mathcal{P}$

Simplified setting: reward, $R_{t+1}(a) = h(S_{t+1}(a))$ where h is bounded, known



Markovian

Potential states:
$\{S_0, S_1(0), S_1(1), S_2(0), S_2(1), \ldots, S_{t+1}(0), S_{t+1}(1) \ldots\ldots\}$

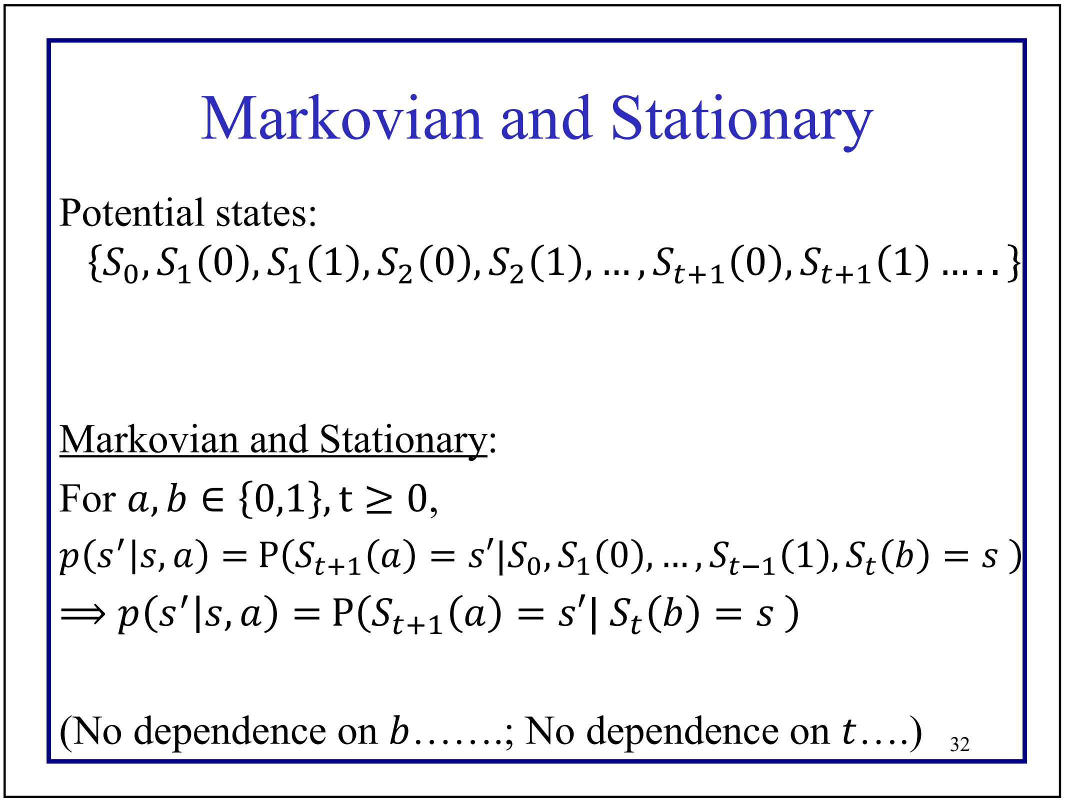
Markovian:

For b ∈ {0,1},

$\{S_{t+1}(0), S_{t+1}(1)\}$ are independent of the past,

$\{S_0, S_1(0), S_1(1), S_2(0), S_2(1), \ldots, S_{t-1}(0), S_{t-1}(1)\}$,
conditional on the present state: $S_t(b)$

Think carefully –what DAG would represent the potential states with the Markovian property?



Markovian and Stationary

Potential states:

$\{S_0, S_1(0), S_1(1), S_2(0), S_2(1), \ldots, S_{t+1}(0), S_{t+1}(1) \ldots\ldots\}$

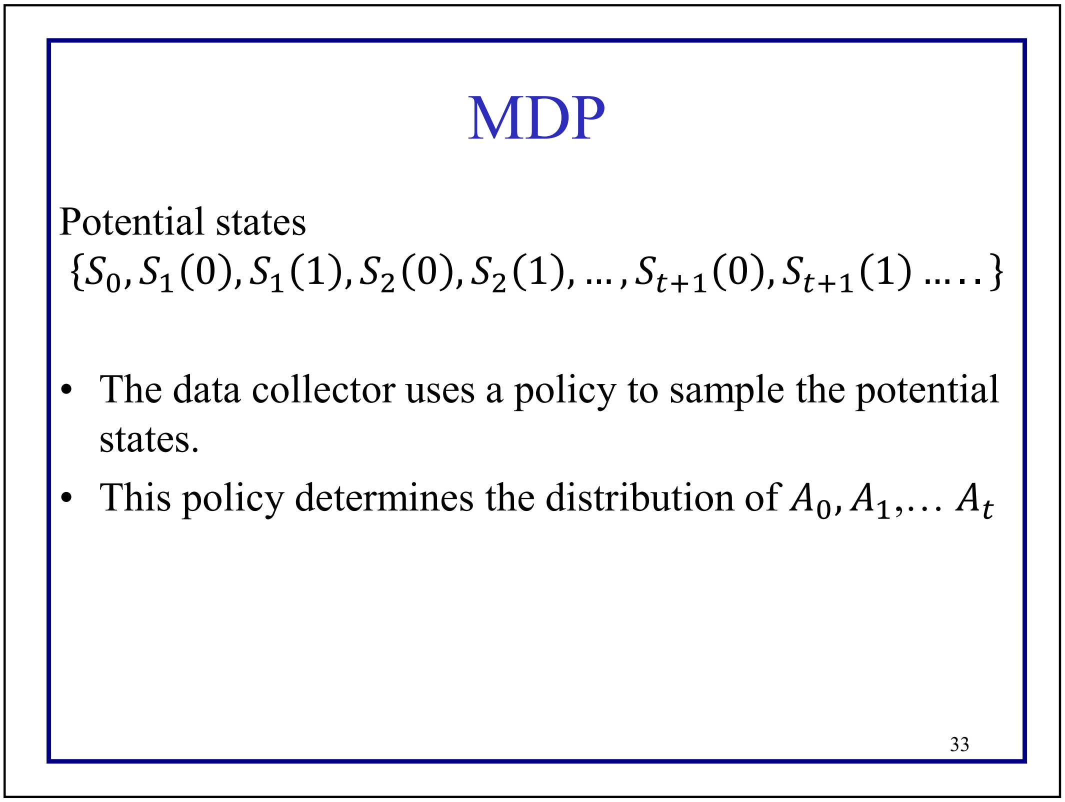
Markovian and Stationary:

For $a, b \in \{0,1\}$, $\mathrm{t} \geq 0$,

$p(s'|s,a) = \mathrm{P}(S_{t+1}(a) = s'|S_0, S_1(0), \ldots, S_{t-1}(1), S_t(b) = s\ )$

$\Longrightarrow p(s'|s,a) = \mathrm{P}(S_{t+1}(a) = s'|\ S_t(b) = s\ )$

(No dependence on b…….; No dependence on t….)

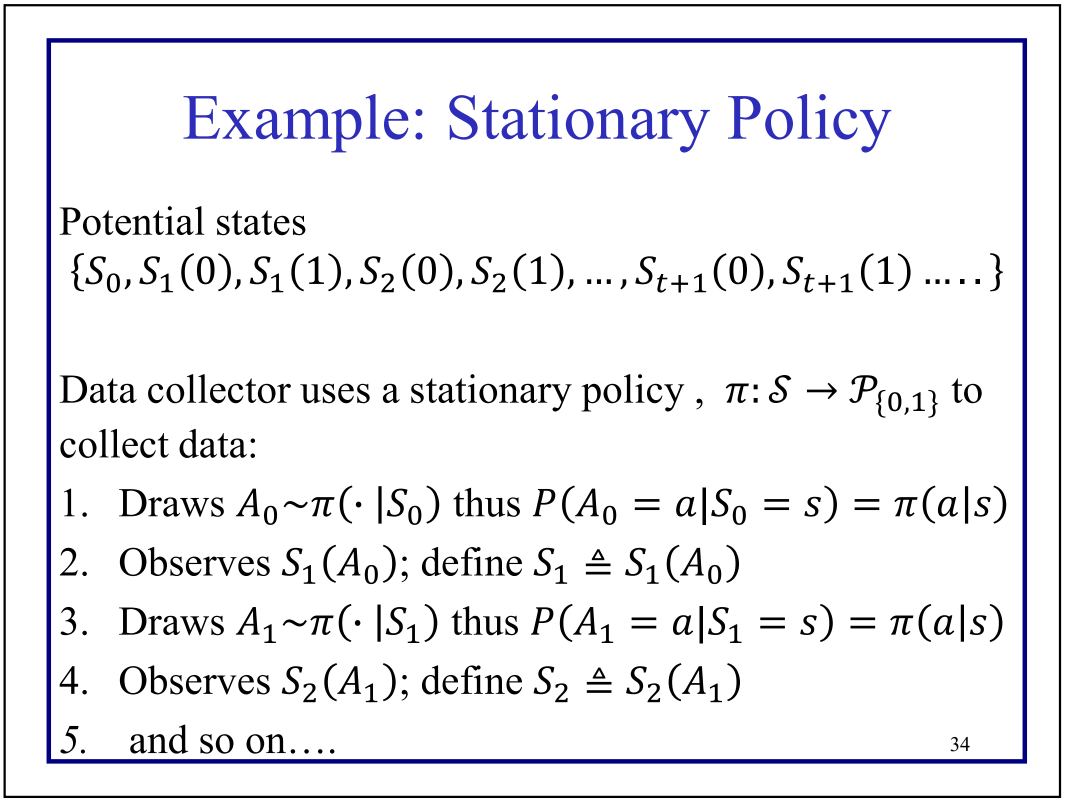


MDP

Potential states
$\{S_0, S_1(0), S_1(1), S_2(0), S_2(1), \ldots, S_{t+1}(0), S_{t+1}(1) \ldots\ldots\}$

- The data collector uses a policy to sample the potential states.
- This policy determines the distribution of $A_0, A_1, \ldots A_t$

Markovian, Stationary policy

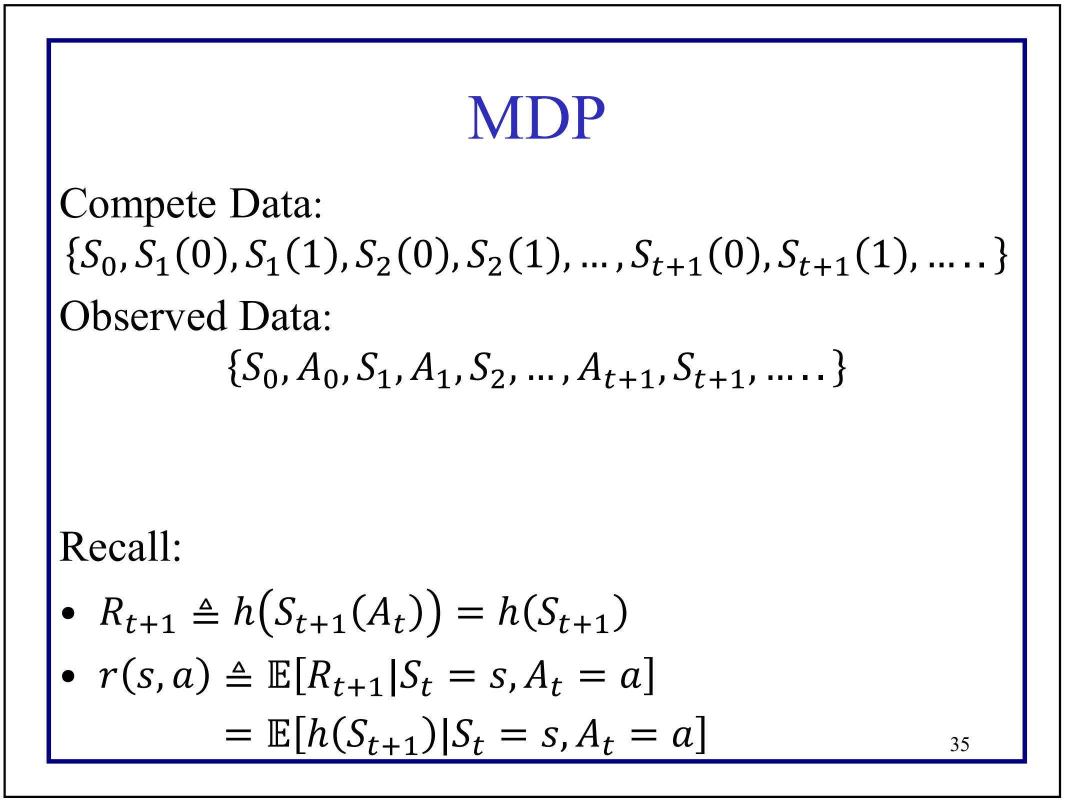


Example: Stationary Policy

Potential states
$\{S_0, S_1(0), S_1(1), S_2(0), S_2(1), \dots, S_{t+1}(0), S_{t+1}(1) \dots \dots\}$

Data collector uses a stationary policy , $\pi: \mathcal{S} \rightarrow \mathcal{P}_{\{0,1\}}$ to collect data:

1. Draws $A_0 \sim \pi(\cdot\,|S_0)$ thus $P(A_0 = a|S_0 = s) = \pi(a|s)$
2. Observes $S_1(A_0)$; define $S_1 \triangleq S_1(A_0)$
3. Draws $A_1 \sim \pi(\cdot\,|S_1)$ thus $P(A_1 = a|S_1 = s) = \pi(a|s)$
4. Observes $S_2(A_1)$; define $S_2 \triangleq S_2(A_1)$
5. and so on….



MDP

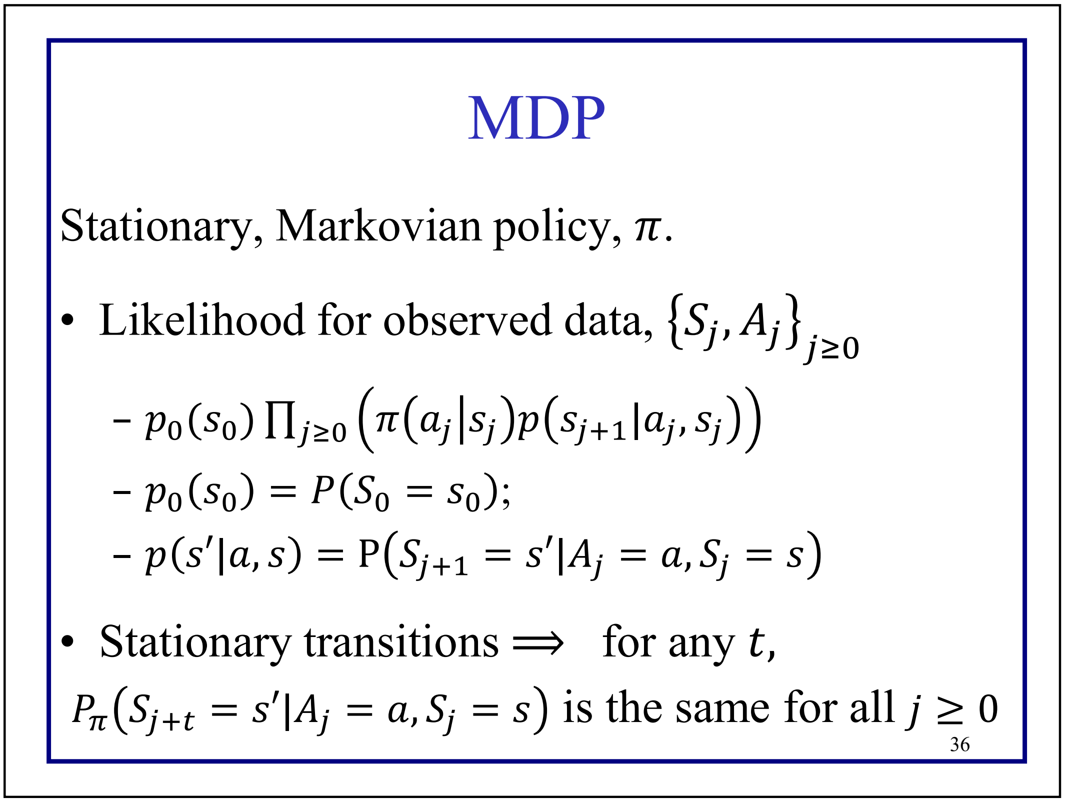
Compete Data:
$\{S_0, S_1(0), S_1(1), S_2(0), S_2(1), \dots, S_{t+1}(0), S_{t+1}(1), \dots\dots\}$

Observed Data:

$$\{S_0, A_0, S_1, A_1, S_2, \dots, A_{t+1}, S_{t+1}, \dots\dots\}$$

Recall:

- $R_{t+1} \triangleq h\big(S_{t+1}(A_t)\big) = h(S_{t+1})$
- $r(s,a) \triangleq \mathbb{E}[R_{t+1} | S_t = s, A_t = a]$
  $= \mathbb{E}[h(S_{t+1}) | S_t = s, A_t = a]$



MDP

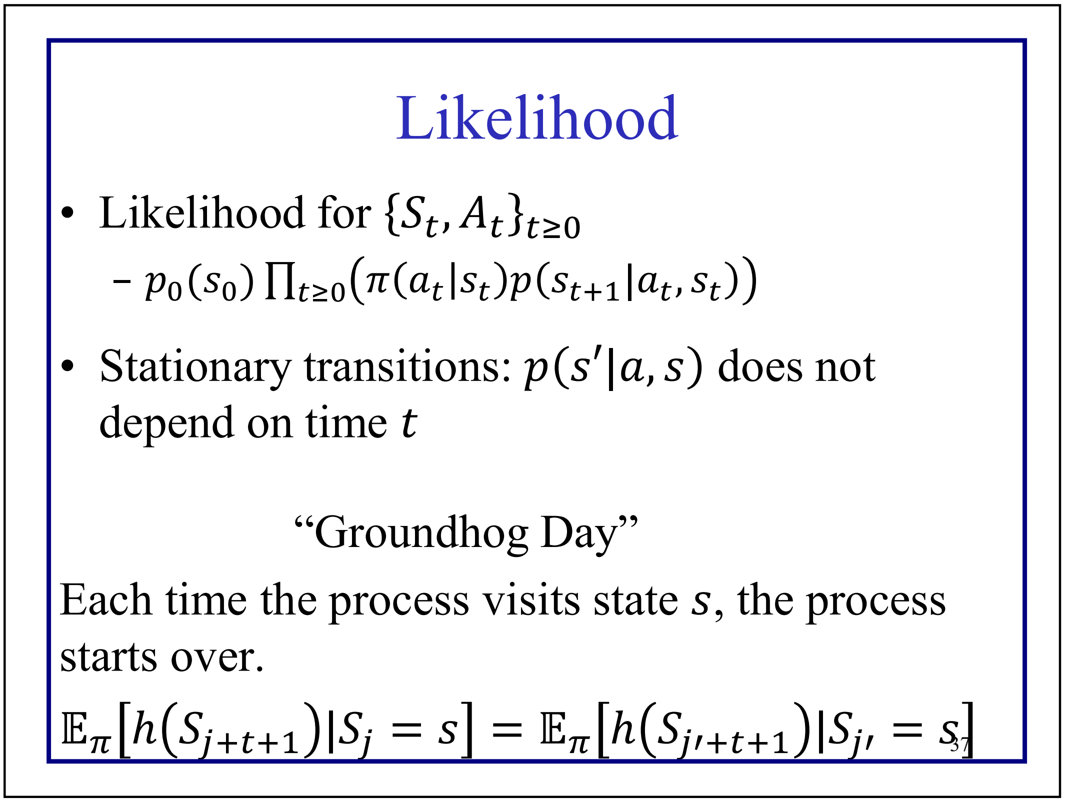
Stationary, Markovian policy, π.

- Likelihood for observed data, $\{S_j, A_j\}_{j\geq 0}$
  - $p_0(s_0)\prod_{j\geq 0}\left(\pi(a_j|s_j)p(s_{j+1}|a_j, s_j)\right)$
  - $p_0(s_0) = P(S_0 = s_0)$;
  - $p(s'|a, s) = \mathrm{P}(S_{j+1} = s'|A_j = a, S_j = s)$
- Stationary transitions $\Longrightarrow$ for any t,

$P_\pi(S_{j+t} = s'|A_j = a, S_j = s)$ is the same for all $j \geq 0$

36

What do we learn by considering this likelihood?



Likelihood

- Likelihood for $\{S_t, A_t\}_{t\geq 0}$
  - $p_0(s_0)\prod_{t\geq 0}\big(\pi(a_t|s_t)p(s_{t+1}|a_t, s_t)\big)$
- Stationary transitions: $p(s'|a, s)$ does not depend on time t

"Groundhog Day"

Each time the process visits state s, the process starts over.

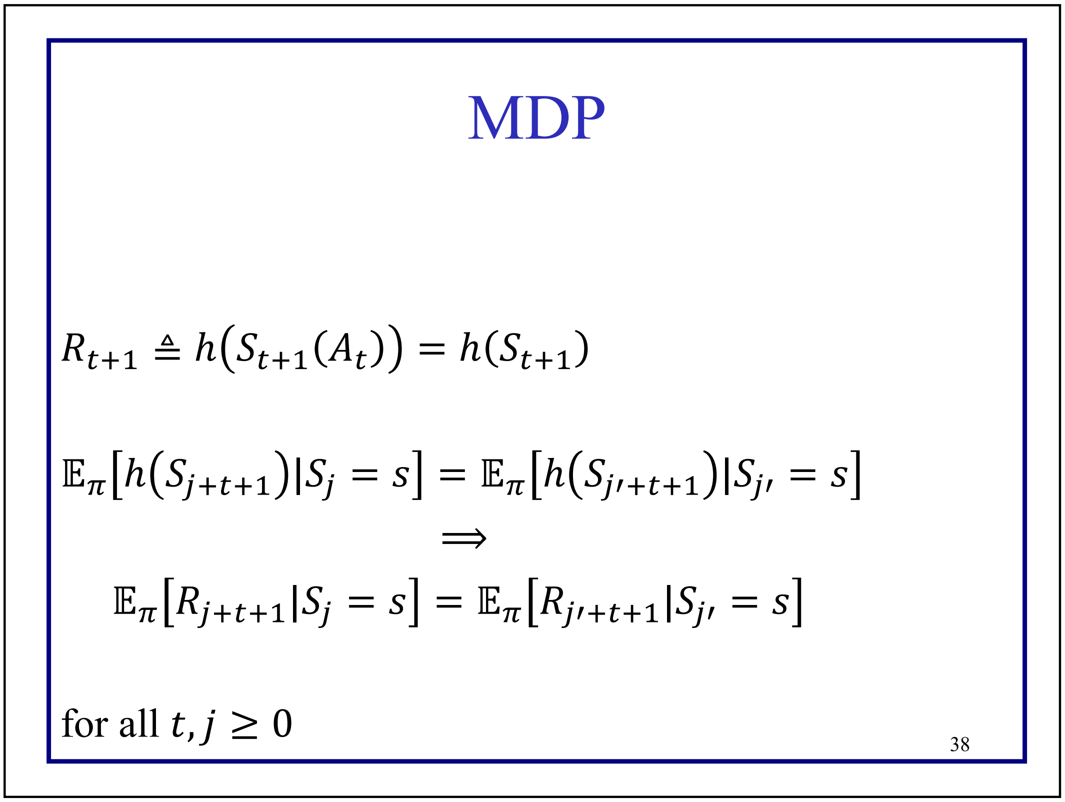
$$\mathbb{E}_\pi\big[h\big(S_{j+t+1}\big)|S_j = s\big] = \mathbb{E}_\pi\big[h\big(S_{j'+t+1}\big)|S_{j'} = s\big]$$

Groundhog Day (1993): Bill Murray stars as a weatherman who gets stuck in a time loop in Punxsutawney, Pennsylvania. He has to relive the same day over and over again until he learns to appreciate life.

Recall

$R_{t+1} \triangleq h\big(S_{t+1}(A_t)\big) = h(S_{t+1})$

$(r(s,a) = \mathbb{E}[R_{t+1}|S_t = s, A_t = a])$



MDP

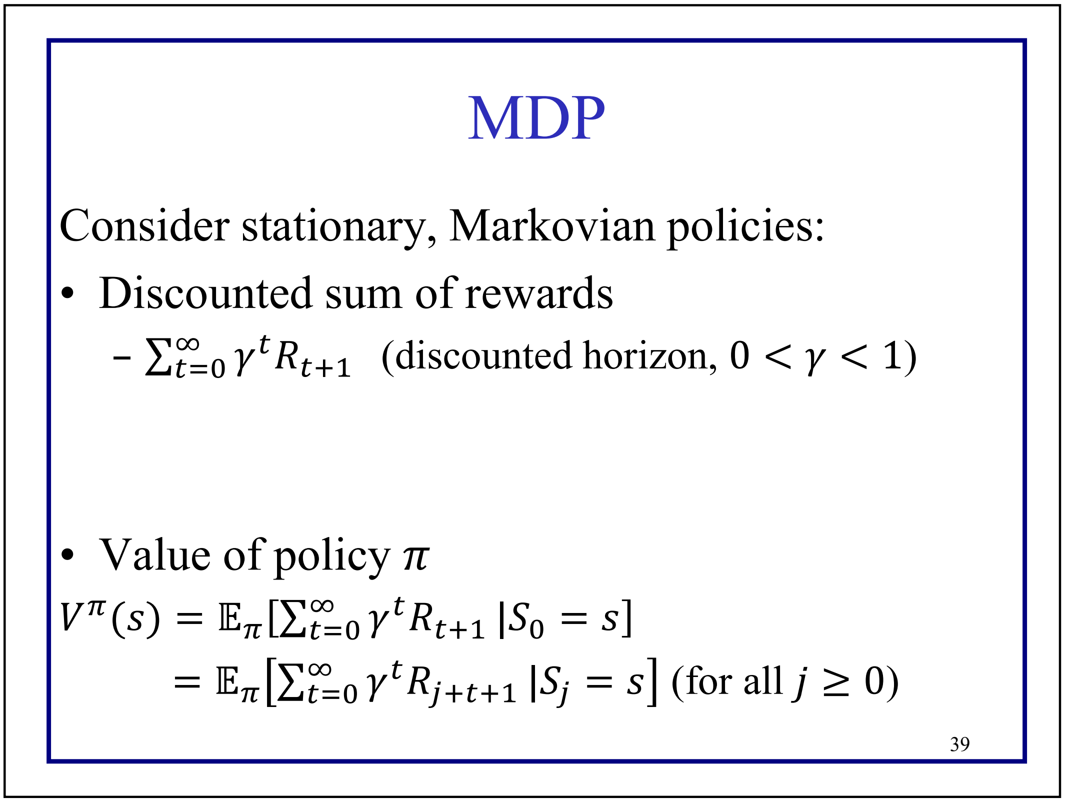
$$R_{t+1} \triangleq h\big(S_{t+1}(A_t)\big) = h(S_{t+1})$$

$$\mathbb{E}_{\pi}\big[h\big(S_{j+t+1}\big)|S_j = s\big] = \mathbb{E}_{\pi}\big[h\big(S_{j'+t+1}\big)|S_{j'} = s\big]$$

$$\Longrightarrow$$

$$\mathbb{E}_{\pi}\big[R_{j+t+1}|S_j = s\big] = \mathbb{E}_{\pi}\big[R_{j'+t+1}|S_{j'} = s\big]$$

for all $t, j \geq 0$



MDP

Consider stationary, Markovian policies:

- Discounted sum of rewards
  - $\sum_{t=0}^{\infty} \gamma^t R_{t+1}$ (discounted horizon, $0 < \gamma < 1$)

- Value of policy π

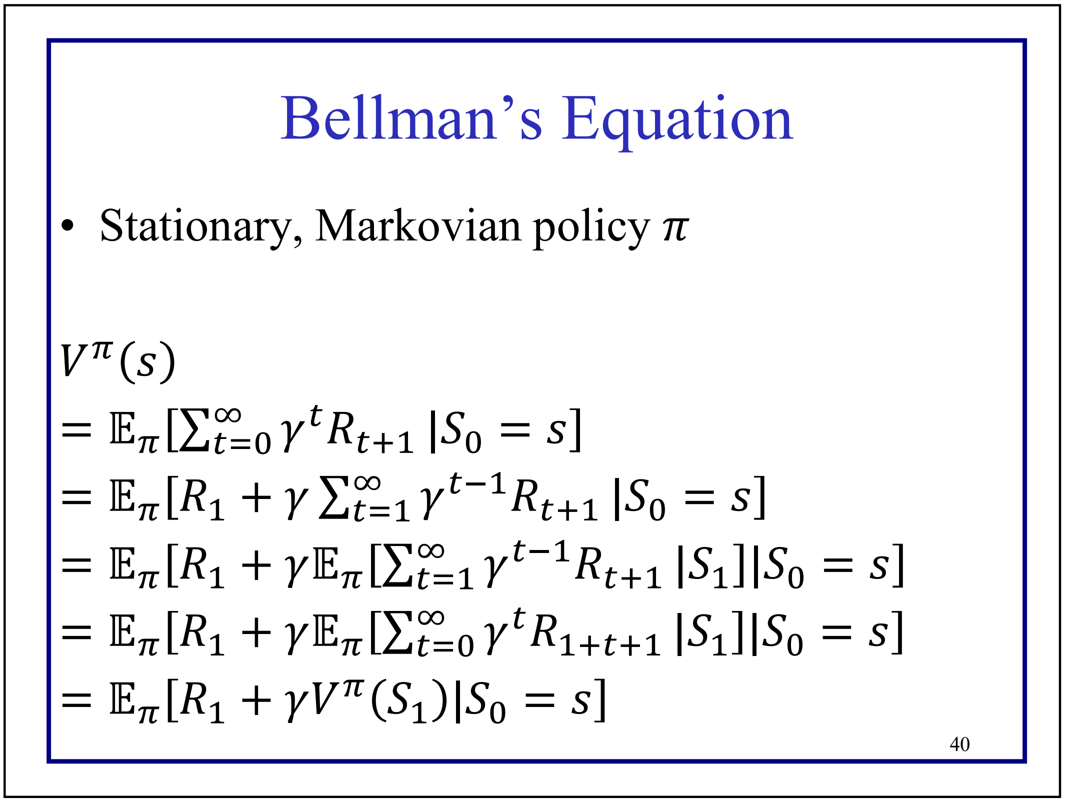
$$V^\pi(s) = \mathbb{E}_\pi[\sum_{t=0}^{\infty} \gamma^t R_{t+1} \, |S_0 = s]$$
$$= \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t R_{j+t+1} \, |S_j = s\right] \text{ (for all } j \geq 0)$$

There are other types of sums of rewards one can consider…

See Puterman, Markov Decision Processes, 2005 Ch. 5, 6 for a proof of why we can restrict attention to stationary, markovian policies.

Note that

$V^\pi(s) = \mathbb{E}_\pi[\sum_{t=0}^{\infty} \gamma^t R_{k+t+1} \, |S_k = s]$ for all k “groundhog day”

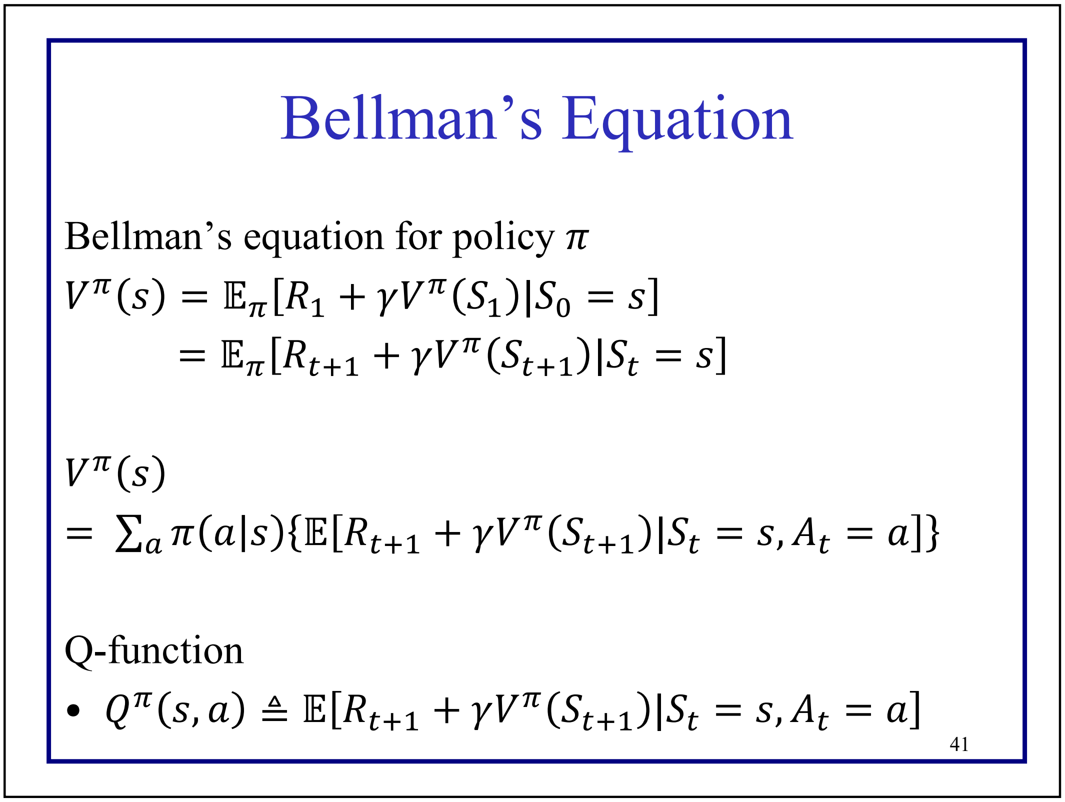


Bellman's Equation

- Stationary, Markovian policy π

$$
\begin{aligned}
&V^\pi(s)\\
&= \mathbb{E}_\pi[\textstyle\sum_{t=0}^\infty \gamma^t R_{t+1} \,|S_0 = s]\\
&= \mathbb{E}_\pi[R_1 + \gamma \textstyle\sum_{t=1}^\infty \gamma^{t-1} R_{t+1} \,|S_0 = s]\\
&= \mathbb{E}_\pi[R_1 + \gamma \mathbb{E}_\pi[\textstyle\sum_{t=1}^\infty \gamma^{t-1} R_{t+1} \,|S_1]|S_0 = s]\\
&= \mathbb{E}_\pi[R_1 + \gamma \mathbb{E}_\pi[\textstyle\sum_{t=0}^\infty \gamma^{t} R_{1+t+1} \,|S_1]|S_0 = s]\\
&= \mathbb{E}_\pi[R_1 + \gamma V^\pi(S_1)|S_0 = s]
\end{aligned}
$$

$V^\pi(s) = \mathbb{E}_\pi[\sum_{t=0}^\infty \gamma^t R_{k+t+1} \,|S_k = s]$ for all k “groundhog day”



Bellman's Equation

Bellman's equation for policy π

$$V^{\pi}(s) = \mathbb{E}_{\pi}[R_1 + \gamma V^{\pi}(S_1)|S_0 = s]$$
$$= \mathbb{E}_{\pi}[R_{t+1} + \gamma V^{\pi}(S_{t+1})|S_t = s]$$

$$V^{\pi}(s)$$
$$= \sum_a \pi(a|s)\{\mathbb{E}[R_{t+1} + \gamma V^{\pi}(S_{t+1})|S_t = s, A_t = a]\}$$

Q-function

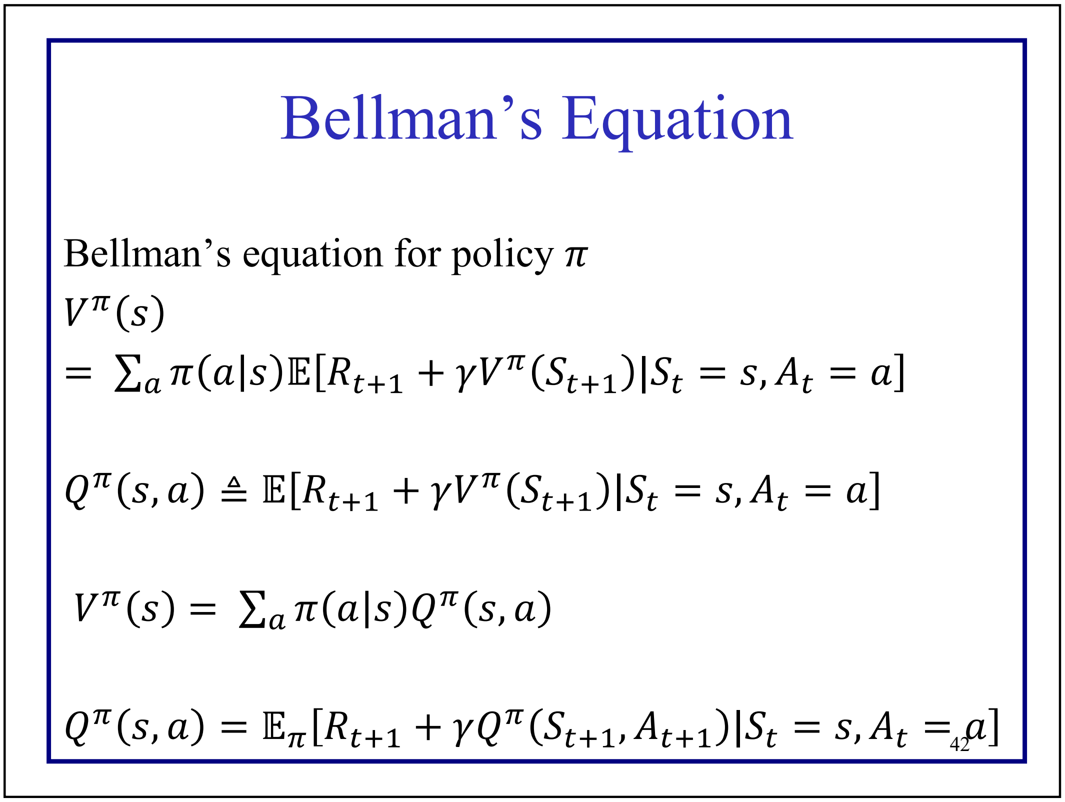
- $Q^{\pi}(s,a) \triangleq \mathbb{E}[R_{t+1} + \gamma V^{\pi}(S_{t+1})|S_t = s, A_t = a]$

Q-function – Quality of action a in state s

Also called the Action-Value function

Q-function – Quality of rewards accrued by first taking action a in state s and then follow policy pi thereafter.

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}[\textstyle\sum_{t=0}^{\infty} \gamma^t R_{t+1} \, |S_0 = s, A_t = a]$$



Bellman’s Equation

Bellman’s equation for policy π

$V^{\pi}(s)$

$= \sum_a \pi(a|s)\mathbb{E}[R_{t+1} + \gamma V^{\pi}(S_{t+1})|S_t = s, A_t = a]$

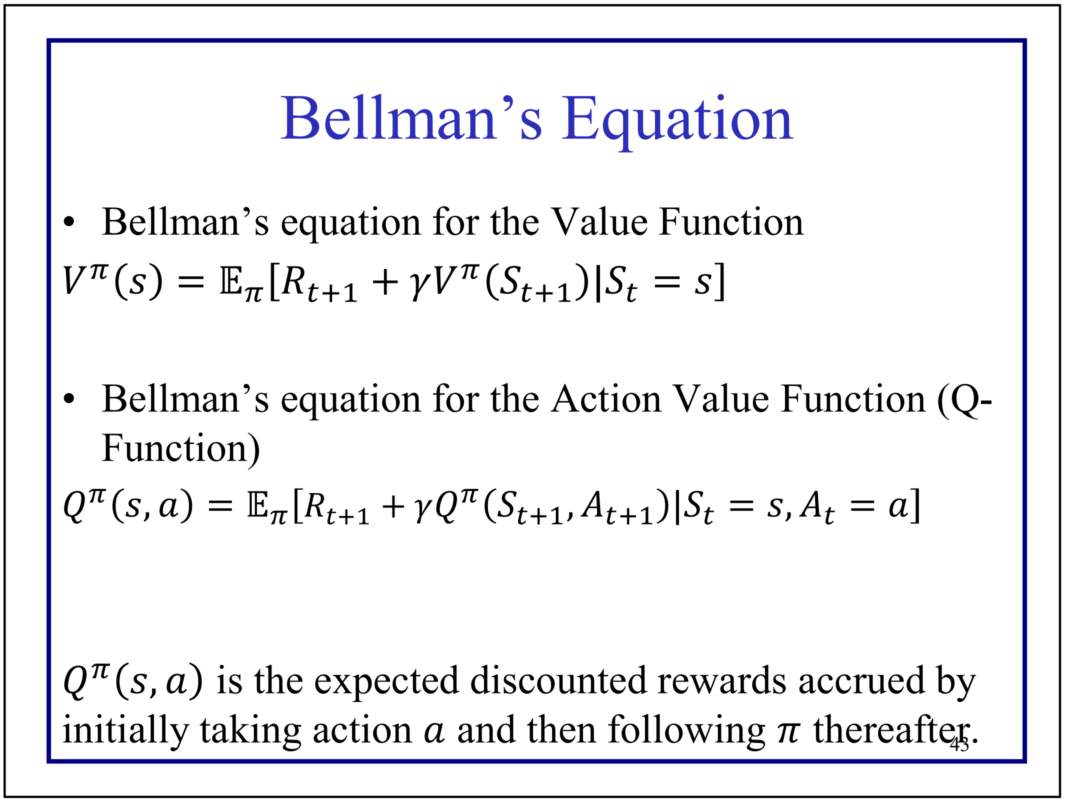
$Q^{\pi}(s,a) \triangleq \mathbb{E}[R_{t+1} + \gamma V^{\pi}(S_{t+1})|S_t = s, A_t = a]$

$V^{\pi}(s) = \sum_a \pi(a|s)Q^{\pi}(s,a)$

$Q^{\pi}(s,a) = \mathbb{E}_{\pi}[R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1})|S_t = s, A_t = a]$

Q-function – Quality of rewards accrued by first taking action a in state s and then follow policy pi thereafter.

$$\begin{aligned} V^{\pi}(s) &= \sum_a \pi(a|s)\mathbb{E}[R_{t+1} + \gamma V^{\pi}(S_{t+1})|S_t = s, A_t = a] \\ &= r(s,a) + \gamma \sum_a \pi(a|s)\mathbb{E}[V^{\pi}(S_{t+1})|S_t = s, A_t = a] \\ &= r(s,a) + \gamma \mathbb{E}_{\pi}[V^{\pi}(S_{t+1})|S_t = s, A_t = a] \end{aligned}$$



Bellman's Equation

- Bellman's equation for the Value Function

$V^{\pi}(s) = \mathbb{E}_{\pi}[R_{t+1} + \gamma V^{\pi}(S_{t+1})|S_t = s]$

- Bellman's equation for the Action Value Function (Q-Function)

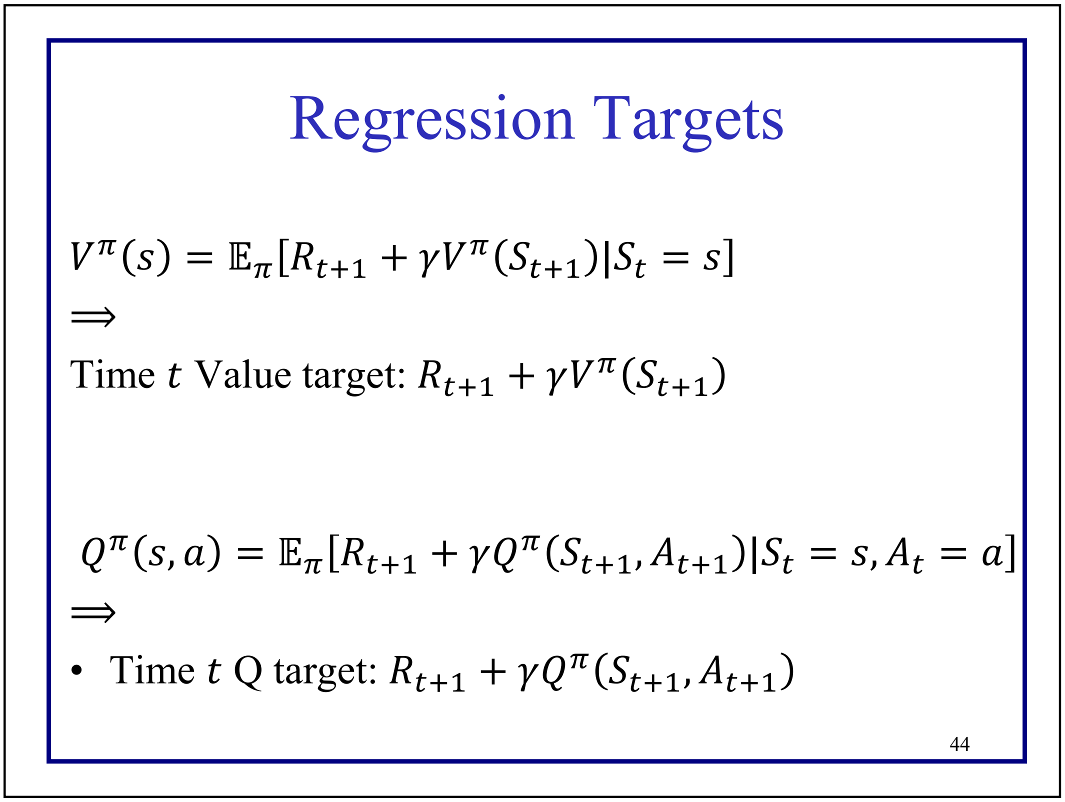
$Q^{\pi}(s,a) = \mathbb{E}_{\pi}[R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1})|S_t = s, A_t = a]$

$Q^{\pi}(s,a)$ is the expected discounted rewards accrued by initially taking action a and then following π thereafter.

Q-function – Quality of rewards accrued by first taking action a in state s and then follow policy pi thereafter.

$$\begin{aligned} V^{\pi}(s) &= \textstyle\sum_a \pi(a|s)\mathbb{E}[R_{t+1} + \gamma V^{\pi}(S_{t+1})|S_t = s, A_t = a] \\ &= r(s,a) + \gamma \textstyle\sum_a \pi(a|s)\mathbb{E}[V^{\pi}(S_{t+1})|S_t = s, A_t = a] \\ &= r(s,a) + \gamma \mathbb{E}_{\pi}[V^{\pi}(S_{t+1})|S_t = s, A_t = a] \end{aligned}$$

$$\begin{aligned} Q^{\pi}(s,a) &= \mathbb{E}_{\pi}[R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1})|S_t = s, A_t = a] \\ &= r(s,a) + \gamma \mathbb{E}[\textstyle\sum_{a'} \pi(a'|s) Q^{\pi}(S_{t+1}, a')\,|S_t = s, A_t = a] \end{aligned}$$



Regression Targets

$V^{\pi}(s) = \mathbb{E}_{\pi}[R_{t+1} + \gamma V^{\pi}(S_{t+1}) | S_t = s]$

$\Longrightarrow$

Time t Value target: $R_{t+1} + \gamma V^{\pi}(S_{t+1})$

$Q^{\pi}(s,a) = \mathbb{E}_{\pi}[R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1}) | S_t = s, A_t = a]$

$\Longrightarrow$

- Time t Q target: $R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1})$

44

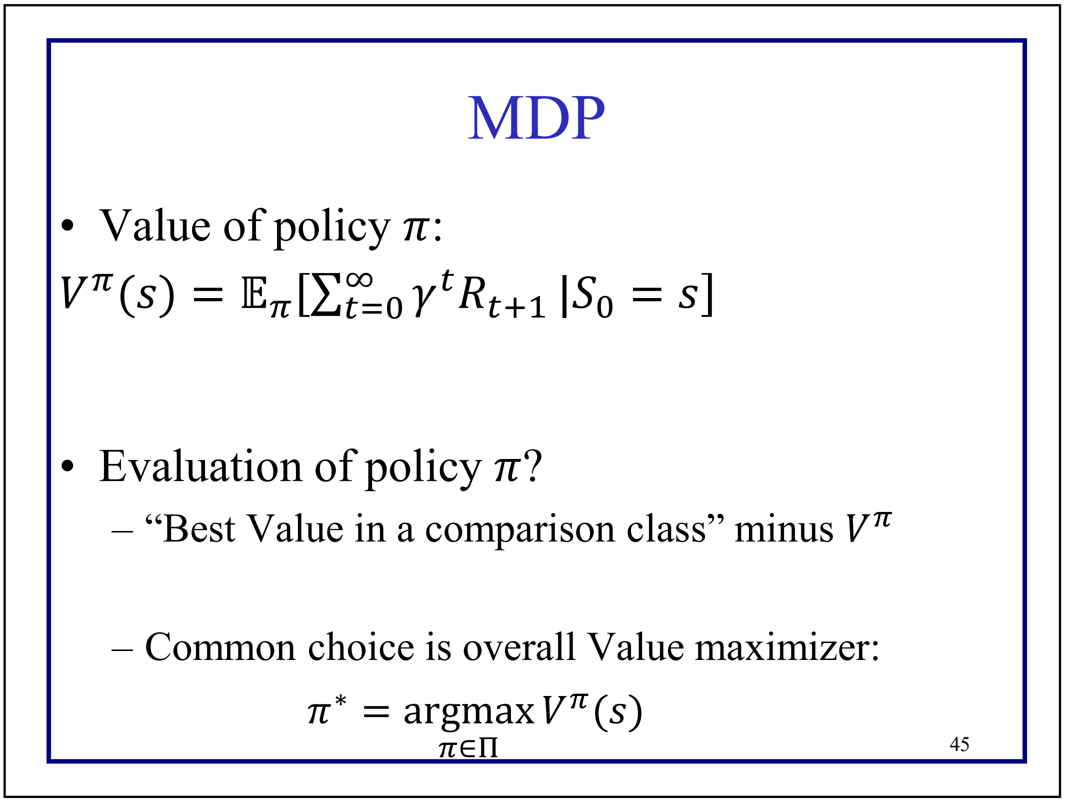
Regression refers to conditional expectations…

These are common targets if you want to learn the value of a particular policy...

Another time t (less noisy) Q target:

$R_{t+1} + \gamma \sum_{a'} \pi(a'|s) Q^{\pi}(S_{t+1}, a')$

Why less noisy?



MDP

- Value of policy π:

$$V^{\pi}(s) = \mathbb{E}_{\pi}[\textstyle\sum_{t=0}^{\infty} \gamma^{t} R_{t+1} \, | S_0 = s]$$

- Evaluation of policy π?
  - "Best Value in a comparison class" minus V^{π}
  - Common choice is overall Value maximizer:

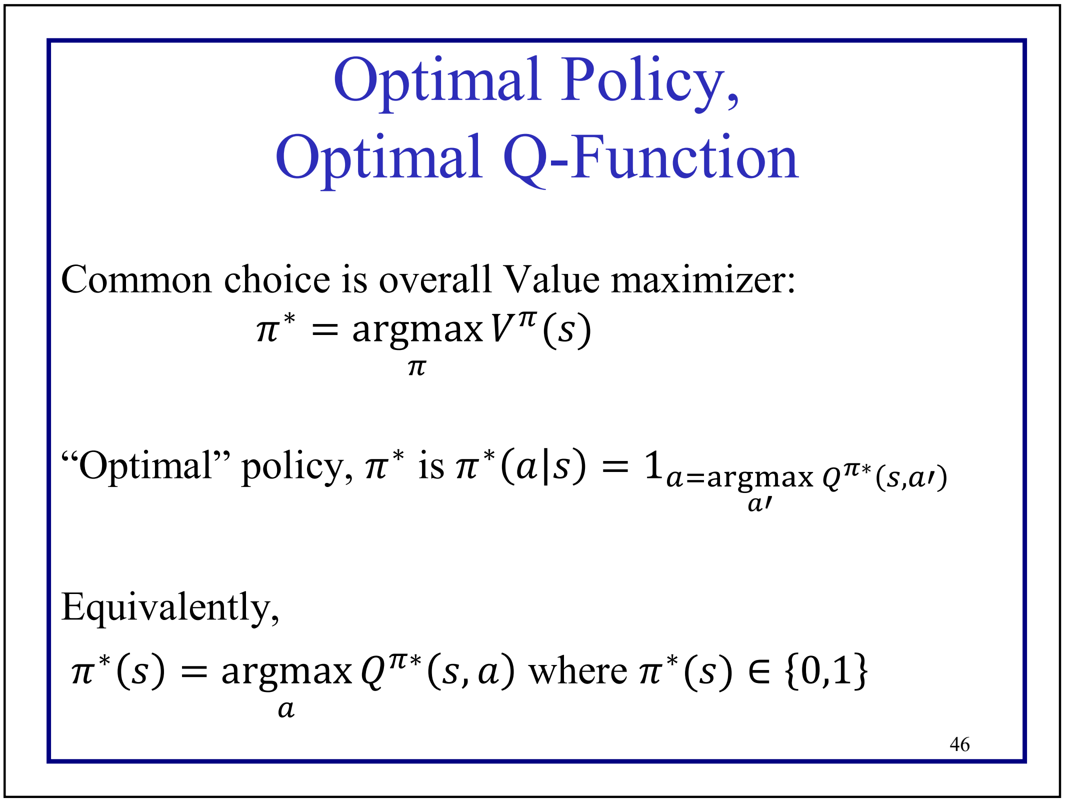
$$\pi^{*} = \underset{\pi \in \Pi}{\operatorname{argmax}} \, V^{\pi}(s)$$

Π is the set of markovian, stationary policies of which there is only $2^{|\mathcal{S}|}$. That is, if there are $|\mathcal{S}|$ discrete states and 2 actions, then the number of possible policies is $2^{|\mathcal{S}|}$; for each state there are two possible actions.

Why does the optimal policy not depend on initial state s?

Why does there exist a markovian, stationary optimal policy in the set of all policies?

See Puterman, Markov Decision Processes, 2005 Ch. 5 (do not need to consider history dependent policy, only consider markovian policies), Ch 6 for a proof of why we can restrict attention to stationary policies and the optimal policy does not depend on initial state.



Optimal Policy, Optimal Q-Function

Common choice is overall Value maximizer:

$$\pi^* = \underset{\pi}{\operatorname{argmax}} V^\pi(s)$$

"Optimal" policy, π^* is $\pi^*(a|s) = 1_{a=\underset{a\prime}{\operatorname{argmax}} Q^{\pi*}(s,a\prime)}$

Equivalently,

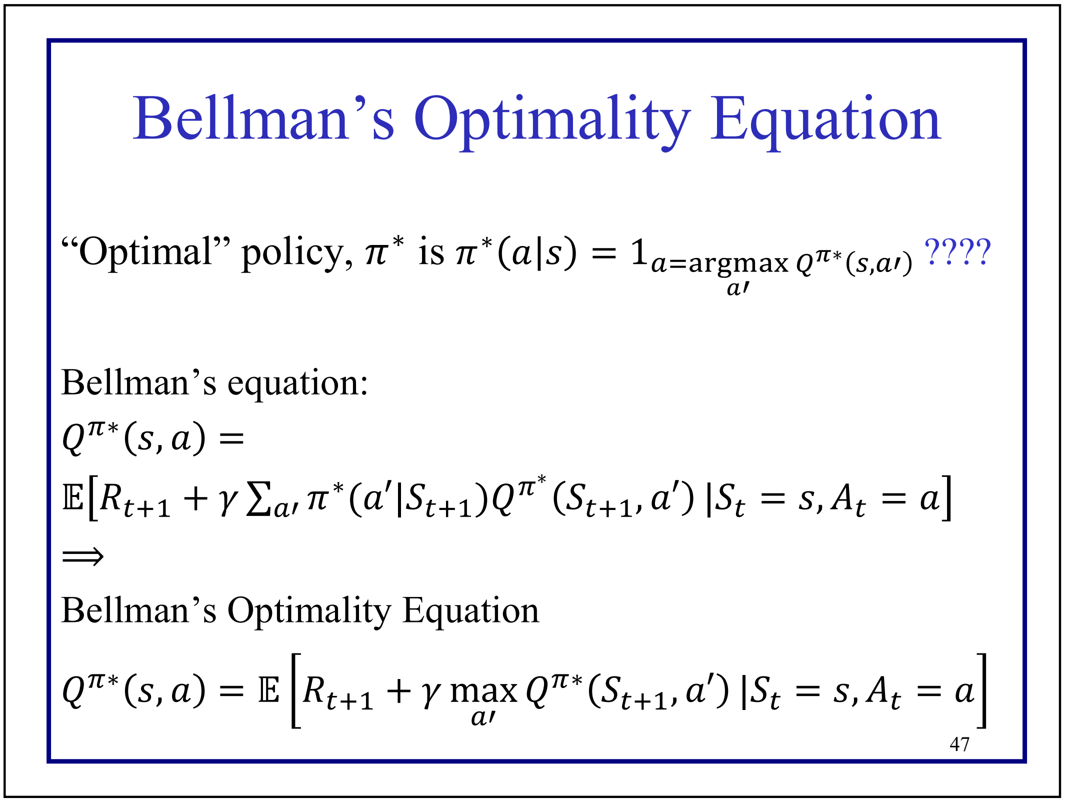
$\pi^*(s) = \underset{a}{\operatorname{argmax}} Q^{\pi*}(s,a)$ where $\pi^*(s) \in \{0,1\}$

46

A common target if you are aiming to learn the optimal policy in an unconstrained stetting.

Optimal policy is degenerate….

Why is $\pi^*(s) = \underset{a}{\operatorname{argmax}} Q^{\pi*}(s,a)$?



Bellman’s Optimality Equation

“Optimal” policy, π^* is $\pi^*(a|s) = 1_{a=\underset{a\prime}{\operatorname{argmax}} Q^{\pi*}(s,a\prime)}$????

Bellman’s equation:

$Q^{\pi*}(s,a) =$

$\mathbb{E}\left[R_{t+1} + \gamma \sum_{a\prime} \pi^*(a'|S_{t+1}) Q^{\pi^*}(S_{t+1},a') \,|S_t = s, A_t = a\right]$

$\Longrightarrow$

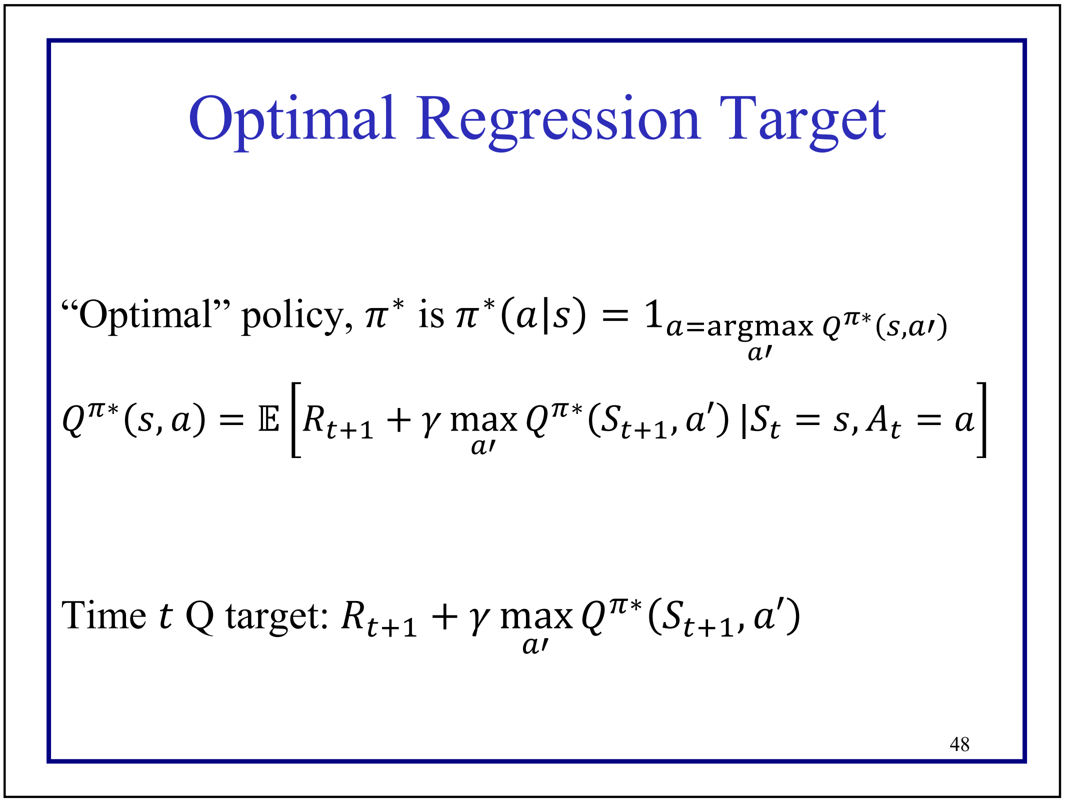
Bellman’s Optimality Equation

$$Q^{\pi*}(s,a) = \mathbb{E}\left[R_{t+1} + \gamma \max_{a\prime} Q^{\pi*}(S_{t+1},a') \,|S_t = s, A_t = a\right]$$

47

Why is $\pi^*(s) = \underset{a}{\operatorname{argmax}} Q^{\pi*}(s,a)$?

We know this due to the policy improvement theorem on the following slides. Policy improvement theorem also implies that there is an optimal policy that is deterministic….



Optimal Regression Target

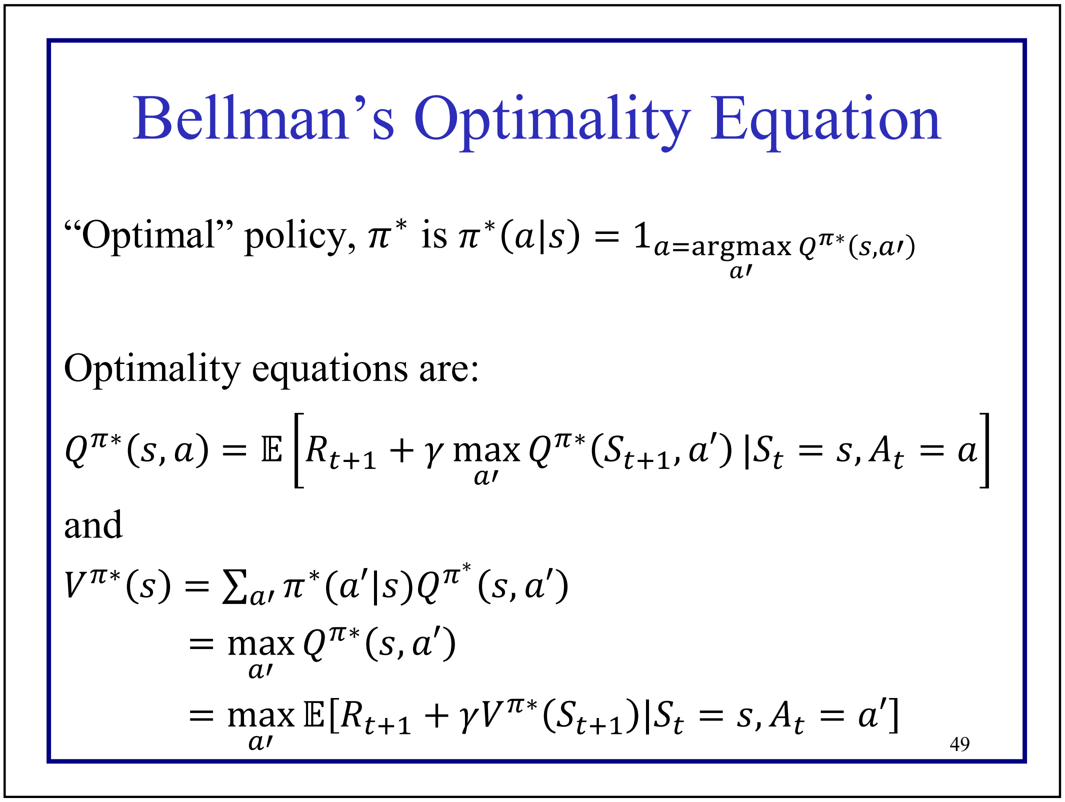
"Optimal" policy, π^* is $\pi^*(a|s) = 1_{a=\underset{a\prime}{\text{argmax}}\, Q^{\pi*}(s,a\prime)}$

$$Q^{\pi*}(s,a) = \mathbb{E}\left[R_{t+1} + \gamma \max_{a\prime} Q^{\pi*}(S_{t+1}, a') \,|S_t = s, A_t = a\right]$$

Time t Q target: $R_{t+1} + \gamma \max_{a\prime} Q^{\pi*}(S_{t+1}, a')$

Why is $\pi^*(s) = \underset{a}{\text{argmax}}\, Q^{\pi*}(s,a)$?

We know this due to the policy improvement theorem on the following slides. Policy improvement theorem also implies that there is an optimal policy that is deterministic….



Bellman’s Optimality Equation

“Optimal” policy, π^* is $\pi^*(a|s) = 1_{a=\underset{a\prime}{\operatorname{argmax}}\, Q^{\pi*}(s,a\prime)}$

Optimality equations are:

$$Q^{\pi*}(s,a) = \mathbb{E}\left[R_{t+1} + \gamma \max_{a\prime} Q^{\pi*}(S_{t+1},a') \,|S_t = s, A_t = a\right]$$

and

$$\begin{aligned} V^{\pi*}(s) &= \textstyle\sum_{a\prime} \pi^*(a'|s) Q^{\pi^*}(s,a') \\ &= \max_{a\prime} Q^{\pi*}(s,a') \\ &= \max_{a\prime} \mathbb{E}[R_{t+1} + \gamma V^{\pi*}(S_{t+1})|S_t = s, A_t = a'] \end{aligned}$$

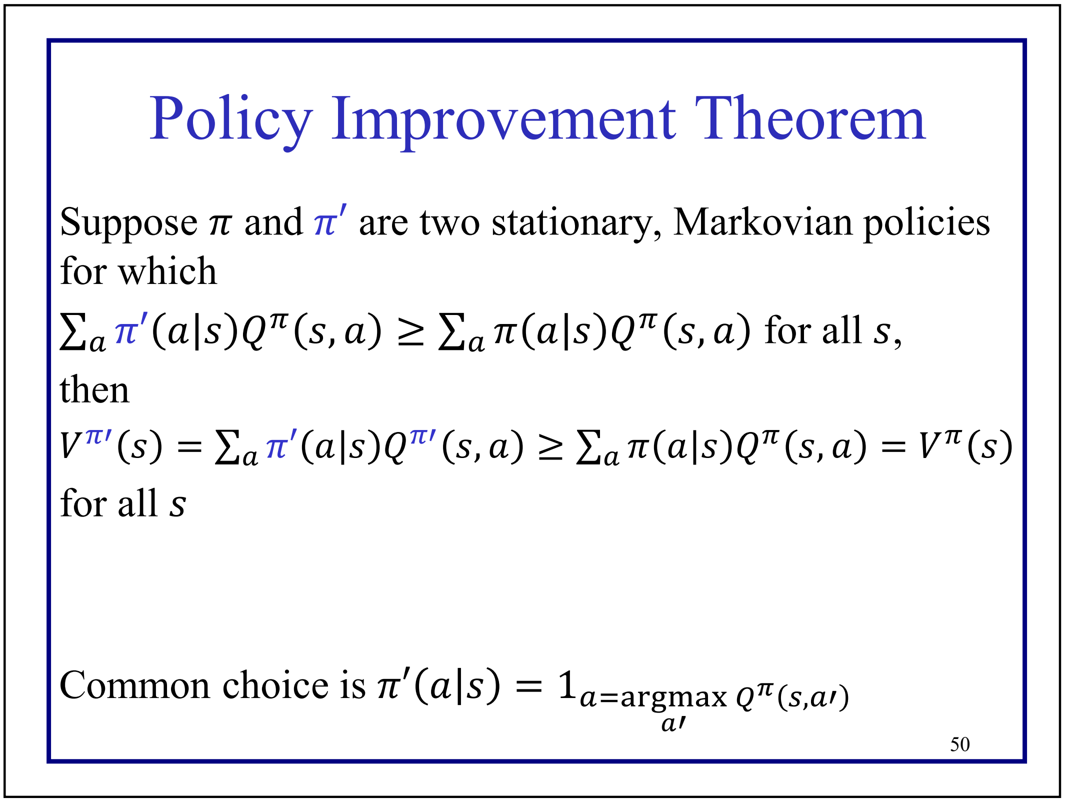
49

Recall

$Q^{\pi}(s,a) \triangleq \mathbb{E}[R_{t+1} + \gamma V^{\pi}(S_{t+1})|S_t = s, A_t = a]$ for all stationary, markovian policies π

Why is $\pi^*(s) = \underset{a}{\operatorname{argmax}}\, Q^{\pi*}(s,a)$?

We know this due to the policy improvement theorem on the following slides. Policy improvement theorem also implies that there is an optimal policy that is deterministic….



Policy Improvement Theorem

Suppose π and π' are two stationary, Markovian policies for which

$\sum_a \pi'(a|s)Q^{\pi}(s,a) \geq \sum_a \pi(a|s)Q^{\pi}(s,a)$ for all s,

then

$V^{\pi'}(s) = \sum_a \pi'(a|s)Q^{\pi'}(s,a) \geq \sum_a \pi(a|s)Q^{\pi}(s,a) = V^{\pi}(s)$

for all s

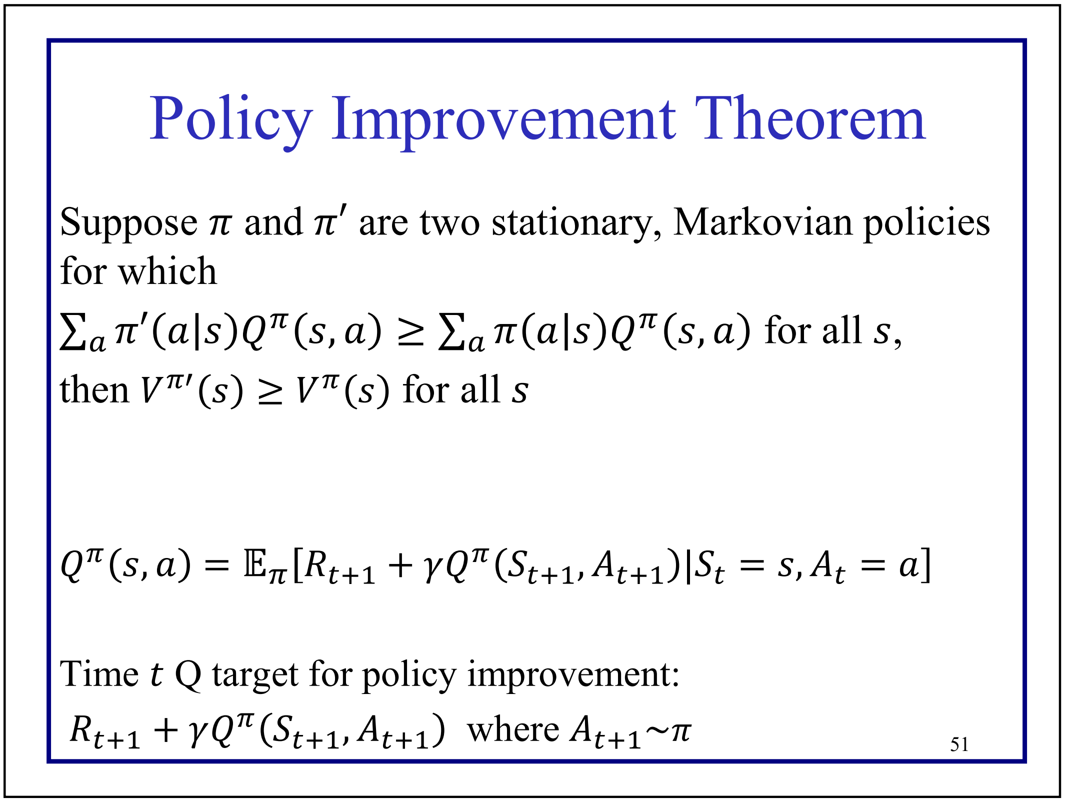
Common choice is $\pi'(a|s) = 1_{a=\underset{a\prime}{\operatorname{argmax}} Q^{\pi}(s,a\prime)}$

50

To see how this implies that $\pi^*(a|s) = 1_{a=\underset{a\prime}{\operatorname{argmax}} Q^{\pi*}(s,a\prime)}$, let π=π^* and consider $\pi'(a|s) = 1_{a=\underset{a\prime}{\operatorname{argmax}} Q^{\pi*}(s,a\prime)}$ aka, put $\pi'(s) = \underset{a}{\operatorname{argmax}}\, Q^{\pi*}(s,a)$,

then the policy improvement theorem implies that $V^{\pi'}(s) \geq V^{\pi^*}(s)$ so π' is an optimal policy.

Chapter 4 in Sutton and Barto



Policy Improvement Theorem

Suppose π and π' are two stationary, Markovian policies for which

$\sum_a \pi'(a|s)Q^{\pi}(s,a) \geq \sum_a \pi(a|s)Q^{\pi}(s,a)$ for all s,

then $V^{\pi'}(s) \geq V^{\pi}(s)$ for all s

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}[R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1})|S_t = s, A_t = a]$$

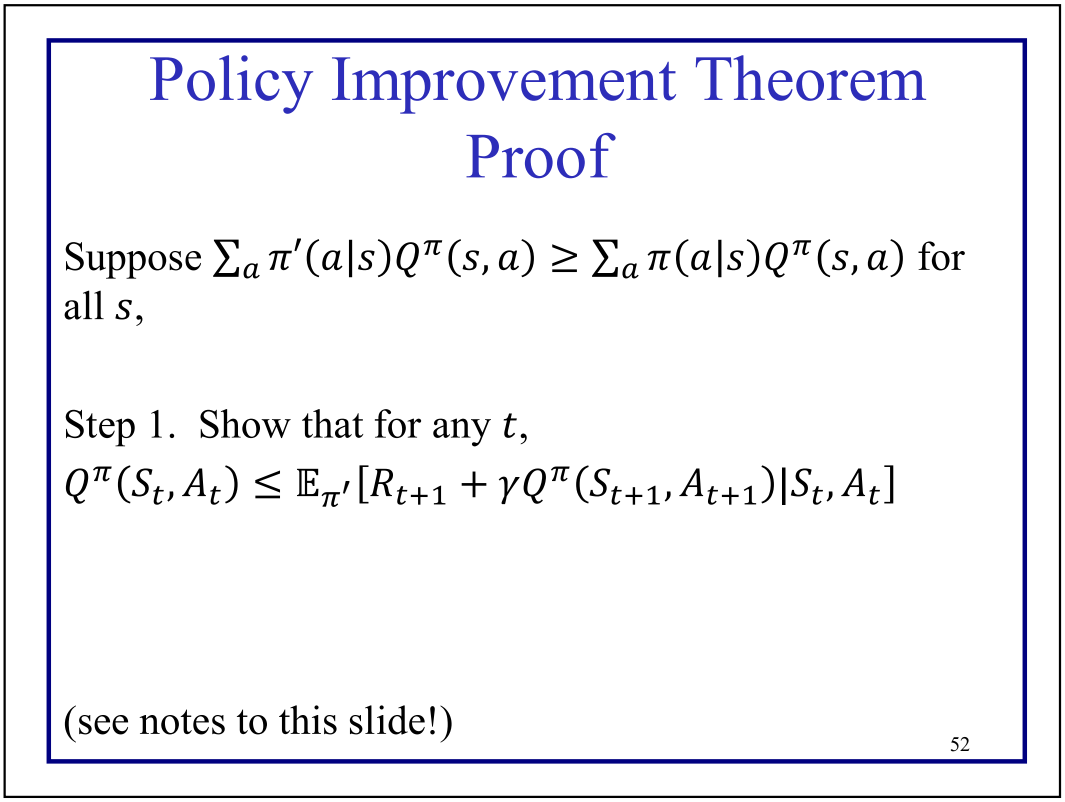
Time t Q target for policy improvement:

$R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1})$ where $A_{t+1} \sim \pi$

Another time t (less noisy) Q target:

$R_{t+1} + \gamma \sum_{a'} \pi(a'|s)Q^{\pi}(S_{t+1}, a')$

Why less noisy?



Policy Improvement Theorem Proof

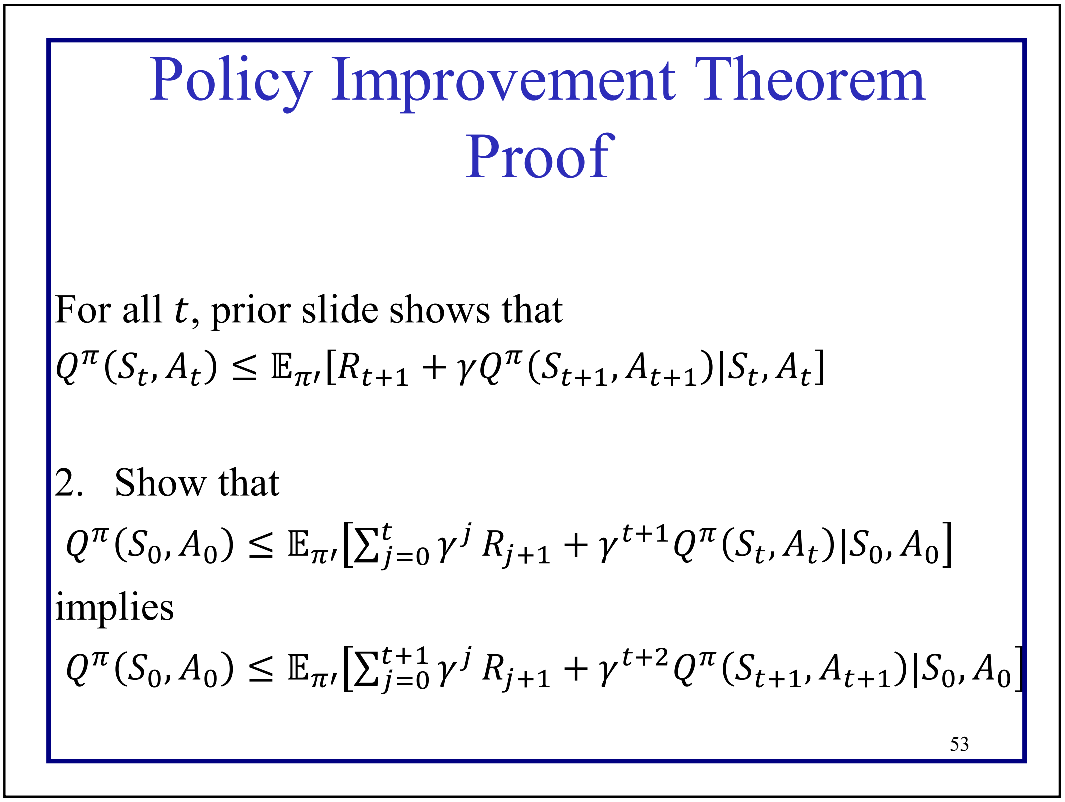
Suppose $\sum_a \pi'(a|s)Q^\pi(s,a) \geq \sum_a \pi(a|s)Q^\pi(s,a)$ for all s,

Step 1. Show that for any t,
$Q^\pi(S_t, A_t) \leq \mathbb{E}_{\pi'}[R_{t+1} + \gamma Q^\pi(S_{t+1}, A_{t+1})|S_t, A_t]$

(see notes to this slide!)

52

$$
\begin{aligned}
Q^\pi(S_t, A_t) &= \mathbb{E}_\pi[R_{t+1} + \gamma Q^\pi(S_{t+1}, A_{t+1})|S_t, A_t] \\
&= \mathbb{E}[R_{t+1} + \gamma \mathbb{E}_\pi[Q^\pi(S_{t+1}, A_{t+1})|S_{t+1}]|S_t, A_t] \\
&= \mathbb{E}[R_{t+1} + \gamma \textstyle\sum_a \pi(a|S_{t+1})Q^\pi(S_{t+1}, a)\ |S_t, A_t] \\
&\leq \mathbb{E}[R_{t+1} + \gamma \textstyle\sum_a \pi'(a|S_{t+1})Q^\pi(S_{t+1}, a)\ |S_t, A_t] \\
&= \mathbb{E}[R_{t+1} + \gamma \mathbb{E}_{\pi'}[Q^\pi(S_{t+1}, A_{t+1})|S_{t+1}]|S_t, A_t] \\
&= \mathbb{E}_{\pi'}[R_{t+1} + \gamma Q^\pi(S_{t+1}, A_{t+1})|S_t, A_t]
\end{aligned}
$$



Policy Improvement Theorem Proof

For all t, prior slide shows that
$Q^{\pi}(S_t, A_t) \leq \mathbb{E}_{\pi'}[R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1}) | S_t, A_t]$

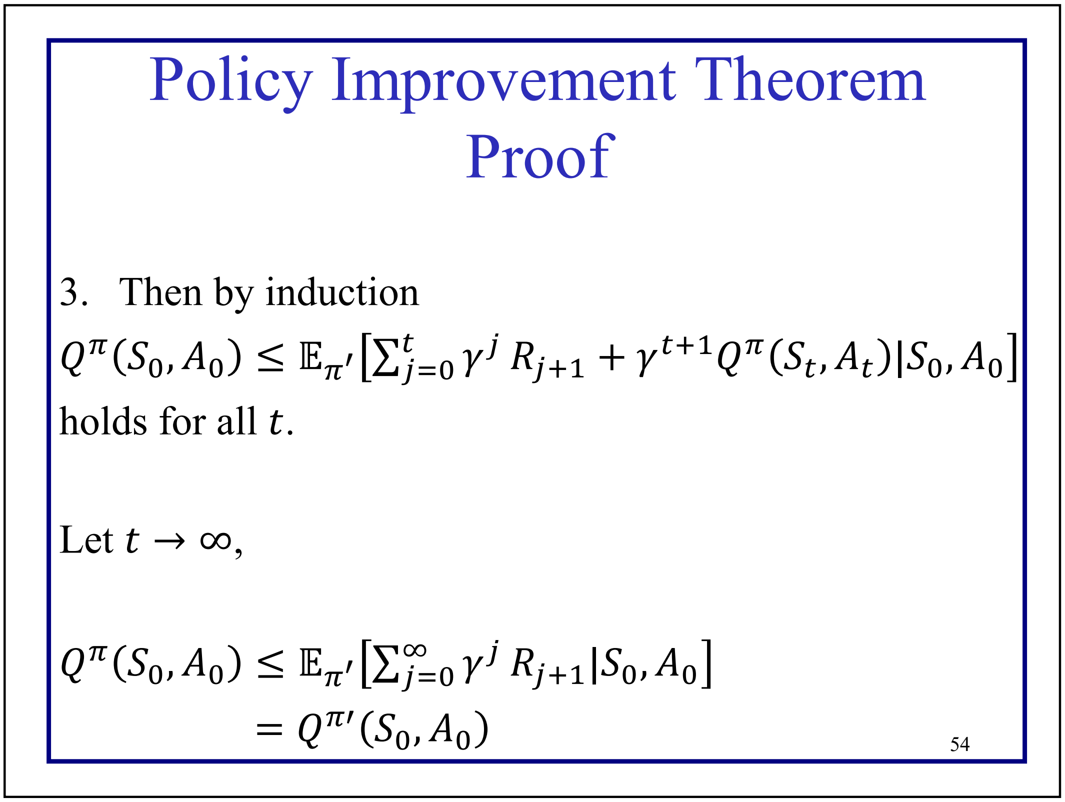
2. Show that

$$Q^{\pi}(S_0, A_0) \leq \mathbb{E}_{\pi'}\left[\sum_{j=0}^{t} \gamma^j R_{j+1} + \gamma^{t+1} Q^{\pi}(S_t, A_t) | S_0, A_0\right]$$

implies

$$Q^{\pi}(S_0, A_0) \leq \mathbb{E}_{\pi'}\left[\sum_{j=0}^{t+1} \gamma^j R_{j+1} + \gamma^{t+2} Q^{\pi}(S_{t+1}, A_{t+1}) | S_0, A_0\right]$$

To obtain 2. just use the result from prior slide: $Q^{\pi}(S_t, A_t) \leq \mathbb{E}_{\pi'}[R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1}) | S_t, A_t]$ inside the expectation on the right-hand side.



Policy Improvement Theorem Proof

3. Then by induction

$$Q^{\pi}(S_0, A_0) \le \mathbb{E}_{\pi'}\left[\sum_{j=0}^{t} \gamma^{j} R_{j+1} + \gamma^{t+1} Q^{\pi}(S_t, A_t) | S_0, A_0\right]$$

holds for all t.

Let $t \to \infty$,

$$Q^{\pi}(S_0, A_0) \le \mathbb{E}_{\pi'}\left[\sum_{j=0}^{\infty} \gamma^{j} R_{j+1} | S_0, A_0\right]$$
$$= Q^{\pi'}(S_0, A_0)$$

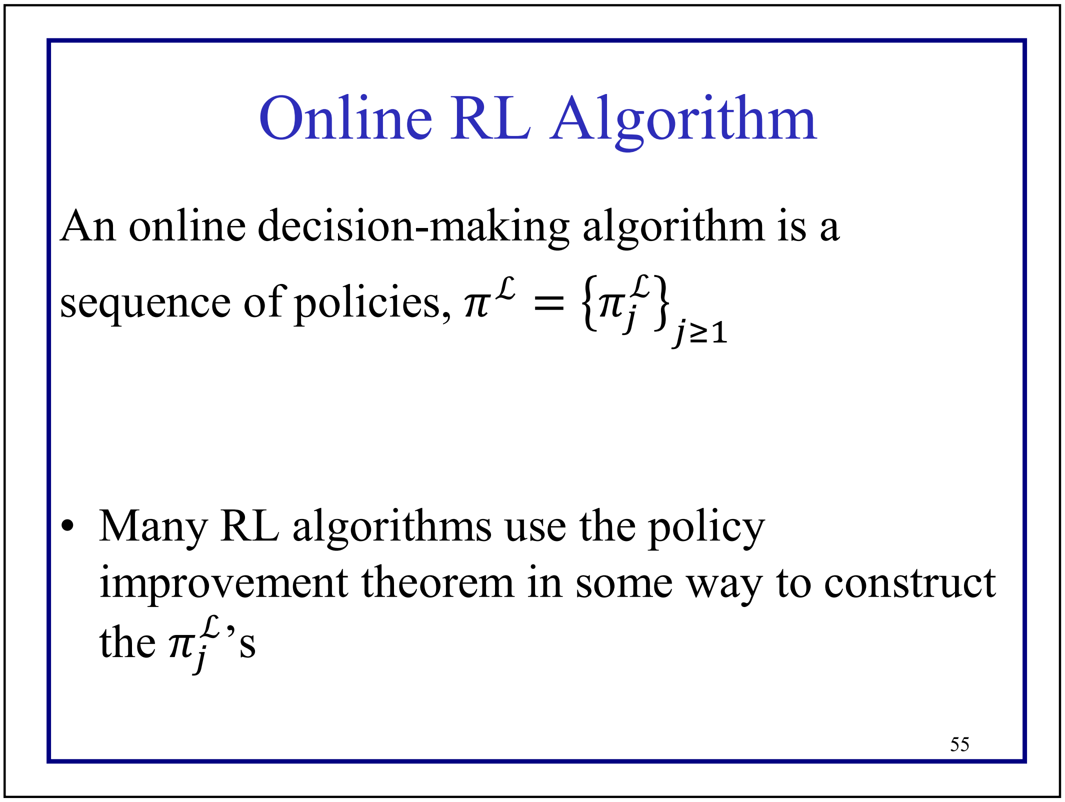
R_{j+1} is a.s. bounded and γ is in (0,1). Thus the limit $t \to \infty$ can be taken inside of the expectation.

Policy improvement theorem also states that if,

$\sum_a \pi'(a|s) Q^{\pi}(s,a) \ge \sum_a \pi(a|s) Q^{\pi}(s,a)$ for all s, with a strict inequality for one state

Then

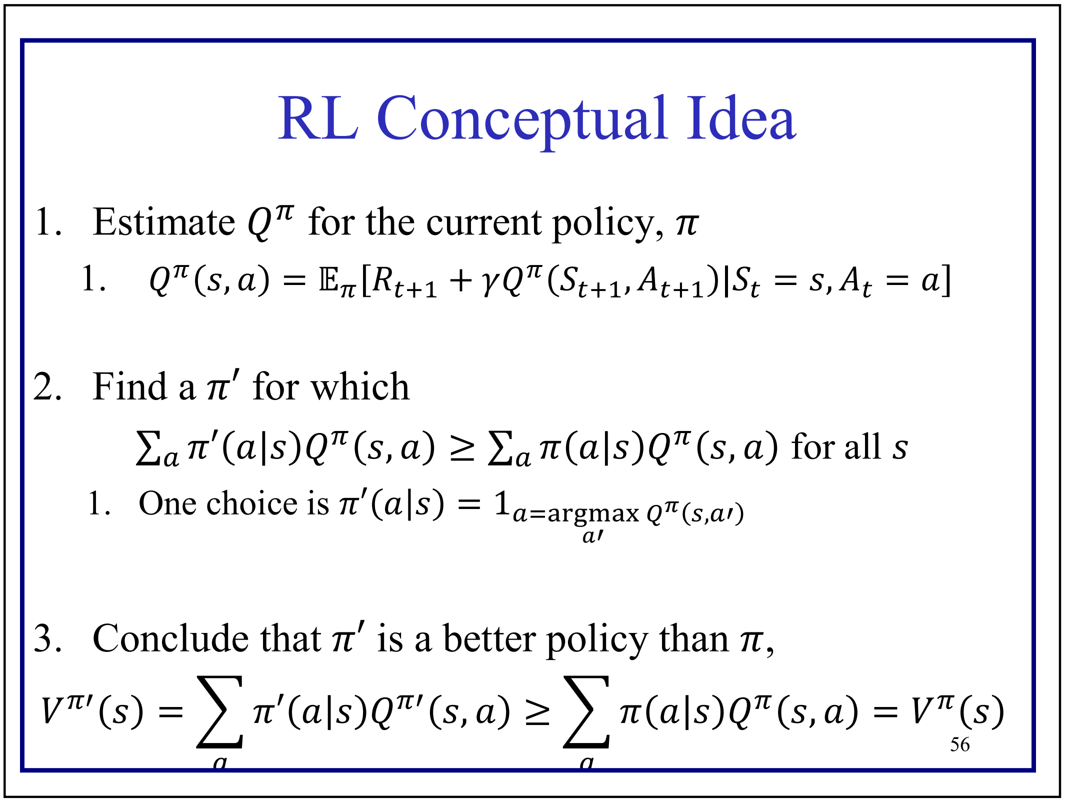
$V^{\pi}(s) \le V^{\pi'}(s)$ for all states, with a strict inequality for at least one state.



Online RL Algorithm

An online decision-making algorithm is a sequence of policies, $\pi^{\mathcal{L}} = \left\{\pi_j^{\mathcal{L}}\right\}_{j\geq 1}$

- Many RL algorithms use the policy improvement theorem in some way to construct the $\pi_j^{\mathcal{L}}$'s



RL Conceptual Idea

1. Estimate Q^{π} for the current policy, π
   1. $Q^{\pi}(s,a) = \mathbb{E}_{\pi}[R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1}) | S_t = s, A_t = a]$

2. Find a π' for which

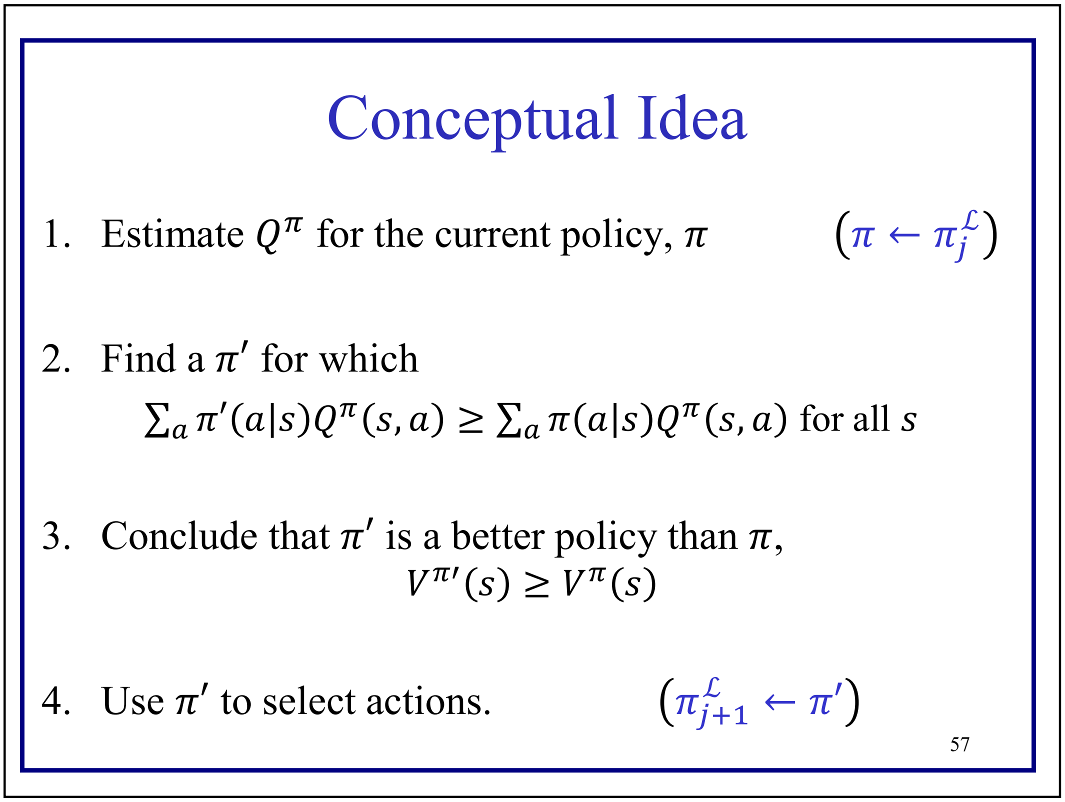
   $\sum_a \pi'(a|s) Q^{\pi}(s,a) \geq \sum_a \pi(a|s) Q^{\pi}(s,a)$ for all s
   1. One choice is $\pi'(a|s) = 1_{a=\underset{a'}{\operatorname{argmax}}\, Q^{\pi}(s,a')}$

3. Conclude that π' is a better policy than π,

$$V^{\pi'}(s) = \sum_a \pi'(a|s) Q^{\pi'}(s,a) \geq \sum_a \pi(a|s) Q^{\pi}(s,a) = V^{\pi}(s)$$

56

Time t Q target for policy improvement: $R_{t+1} + \gamma Q^{\pi}(S_{t+1}, A_{t+1})$ where $A_{t+1} \sim \pi$



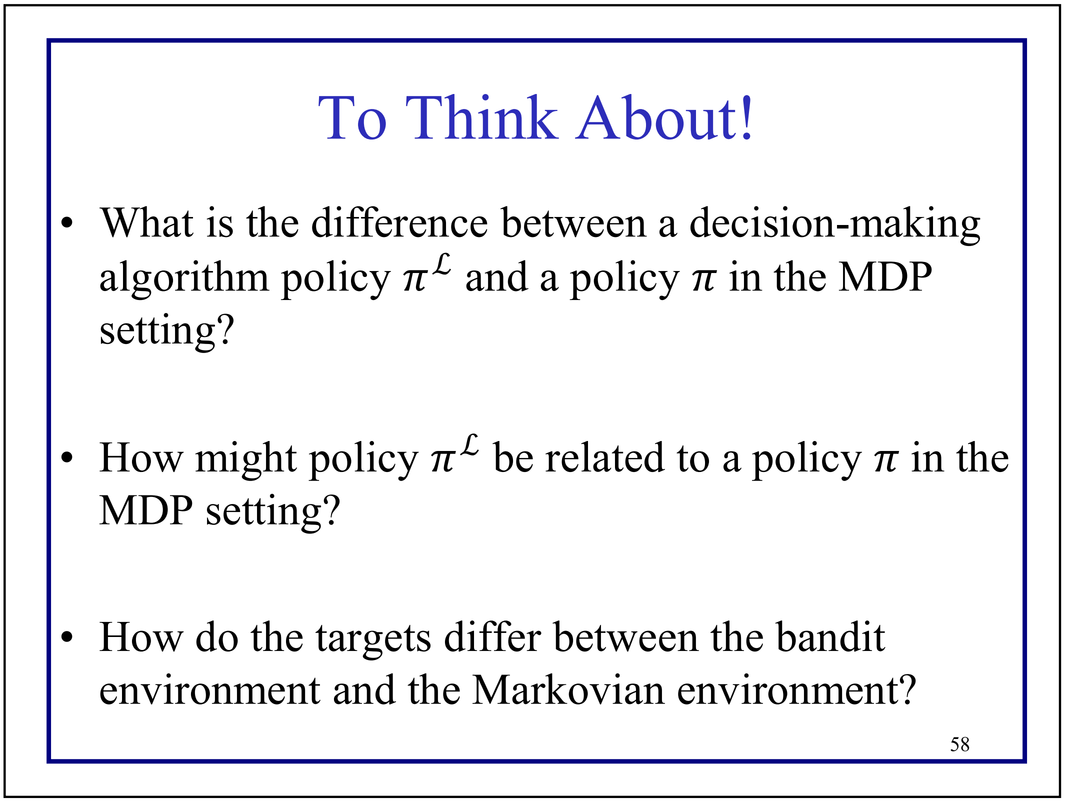
Conceptual Idea

1. Estimate Q^{π} for the current policy, π $\left(\pi \leftarrow \pi_j^{\mathcal{L}}\right)$

2. Find a π' for which
   $\sum_a \pi'(a|s)Q^{\pi}(s,a) \geq \sum_a \pi(a|s)Q^{\pi}(s,a)$ for all s

3. Conclude that π' is a better policy than π,
   $V^{\pi'}(s) \geq V^{\pi}(s)$

4. Use π' to select actions. $\left(\pi_{j+1}^{\mathcal{L}} \leftarrow \pi'\right)$



To Think About!

- What is the difference between a decision-making algorithm policy $\pi^{\mathcal{L}}$ and a policy π in the MDP setting?
- How might policy $\pi^{\mathcal{L}}$ be related to a policy π in the MDP setting?
- How do the targets differ between the bandit environment and the Markovian environment?

58