CS 747, Autumn 2020: Week 9, Lecture 1

Shivaram Kalyanakrishnan

Department of Computer Science and Engineering
Indian Institute of Technology Bombay

Autumn 2020

Question from Last Week

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 3, s_1, 1, s_\top$

(Let T denote the number of episodes.)

Question from Last Week

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 3, s_1, 1, s_\top$

(Let T denote the number of episodes.)

- Is $\lim_{T \to \infty} \hat{V}^T_{\text{First-visit}} = V^\pi$?

Question from Last Week

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 3, s_1, 1, s_\top$

(Let T denote the number of episodes.)

- Is $\lim_{T\to\infty} \hat{V}^T_{\text{First-visit}} = V^\pi$? Yes.

Question from Last Week

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 3, s_1, 1, s_\top$

(Let T denote the number of episodes.)

- Is $\lim_{T \to \infty} \hat{V}^T_{\text{First-visit}} = V^\pi$? Yes.
- Is $\lim_{T \to \infty} \hat{V}^T_{\text{Every-visit}} = V^\pi$?

Question from Last Week

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top.$
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top.$
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top.$
Episode 4: $s_3, 1, s_\top.$
Episode 5: $s_2, 3, s_2, 3, s_1, 1, s_\top$

(Let T denote the number of episodes.)

- Is $\lim_{T\to\infty} \hat{V}^T_{\text{First-visit}} = V^\pi$? Yes.
- Is $\lim_{T\to\infty} \hat{V}^T_{\text{Every-visit}} = V^\pi$? Yes.

Question from Last Week

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 3, s_1, 1, s_\top$

(Let T denote the number of episodes.)

- Is $\lim_{T\to\infty} \hat{V}^T_{\text{First-visit}} = V^\pi$? Yes.
- Is $\lim_{T\to\infty} \hat{V}^T_{\text{Every-visit}} = V^\pi$? Yes.
- Is $\lim_{T\to\infty} \hat{V}^T_{\text{Second-visit}} = V^\pi$?

Question from Last Week

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 3, s_1, 1, s_\top$

(Let T denote the number of episodes.)

- Is $\lim_{T \to \infty} \hat{V}^T_{\text{First-visit}} = V^\pi$? Yes.
- Is $\lim_{T \to \infty} \hat{V}^T_{\text{Every-visit}} = V^\pi$? Yes.
- Is $\lim_{T \to \infty} \hat{V}^T_{\text{Second-visit}} = V^\pi$? Yes.

Question from Last Week

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 3, s_1, 1, s_\top$.

(Let T denote the number of episodes.)

- Is $\lim_{T\to\infty} \hat{V}^T_{\text{First-visit}} = V^\pi$? Yes.
- Is $\lim_{T\to\infty} \hat{V}^T_{\text{Every-visit}} = V^\pi$? Yes.
- Is $\lim_{T\to\infty} \hat{V}^T_{\text{Second-visit}} = V^\pi$? Yes.
- Is $\lim_{T\to\infty} \hat{V}^T_{\text{Last-visit}} = V^\pi$?

Question from Last Week

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 3, s_1, 1, s_\top$.

(Let T denote the number of episodes.)

- Is $\lim_{T\to\infty} \hat{V}^T_{\text{First-visit}} = V^\pi$? Yes.
- Is $\lim_{T\to\infty} \hat{V}^T_{\text{Every-visit}} = V^\pi$? Yes.
- Is $\lim_{T\to\infty} \hat{V}^T_{\text{Second-visit}} = V^\pi$? Yes.
- Is $\lim_{T\to\infty} \hat{V}^T_{\text{Last-visit}} = V^\pi$? No.

Reinforcement Learning

1. Least-squares and Maximum likelihood estimators
2. On-line implementation of First-visit MC
3. TD(0) algorithm
4. Convergence of Batch TD(0)
5. Control with TD learning

Reinforcement Learning

1. Least-squares and Maximum likelihood estimators
2. On-line implementation of First-visit MC
3. TD(0) algorithm
4. Convergence of Batch TD(0)
5. Control with TD learning

Estimate p

- You have two coins.

Coin 1

Coin 2

Estimate p

- You have two coins.
- You are told that the probability of a head (1-reward) for Coin 1 is $p \in [0, 0.5]$, and that for Coin 2 is $2p$.

Coin 1

$\mathbb{P}\{\text{heads}\} = p$

Coin 2

$\mathbb{P}\{\text{heads}\} = 2p$

Estimate p

- You have two coins.
- You are told that the probability of a head (1-reward) for Coin 1 is $p \in [0, 0.5]$, and that for Coin 2 is $2p$.
- Hence the corresponding probabilities of a tail (0-reward) are $1 - p$ and $1 - 2p$, respectively.

Coin 1

$\mathbb{P}\{\text{heads}\} = p$

Coin 2

$\mathbb{P}\{\text{heads}\} = 2p$

Estimate p

- You have two coins.
- You are told that the probability of a head (1-reward) for Coin 1 is $p \in [0, 0.5]$, and that for Coin 2 is $2p$.
- Hence the corresponding probabilities of a tail (0-reward) are $1 - p$ and $1 - 2p$, respectively.
- You toss each coin once and see these outcomes.

Coin 1

$\mathbb{P}\{\text{heads}\} = p$

Outcome = 1

Coin 2

$\mathbb{P}\{\text{heads}\} = 2p$

Outcome = 0

Estimate p

- You have two coins.
- You are told that the probability of a head (1-reward) for Coin 1 is $p \in [0, 0.5]$, and that for Coin 2 is $2p$.
- Hence the corresponding probabilities of a tail (0-reward) are $1 - p$ and $1 - 2p$, respectively.
- You toss each coin once and see these outcomes.

Coin 1	Coin 2
	
$\mathbb{P}\{\text{heads}\} = p$	$\mathbb{P}\{\text{heads}\} = 2p$
Outcome = 1	Outcome = 0

What is your estimate of p (call it $\hat{p}$)?

Two Common Estimates

- **Least-squares estimate.**

  For $q \in [0, 0.5]$,

$$SE(q) = (q - 1)^2 + (2q - 0)^2.$$

$$\hat{p}_{LS} \stackrel{\text{def}}{=} \operatorname*{argmin}_{q \in [0, 0.5]} SE(q) = 0.2.$$

Two Common Estimates

- **Least-squares estimate.**
  For $q \in [0, 0.5]$,

$$SE(q) = (q - 1)^2 + (2q - 0)^2.$$

$$\hat{p}_{LS} \stackrel{\text{def}}{=} \operatorname*{argmin}_{q \in [0,0.5]} SE(q) = 0.2.$$

- **Maximum likelihood estimate.**
  For $q \in [0, 0.5]$,

$$L(q) = q(1 - 2q).$$

$$\hat{p}_{ML} \stackrel{\text{def}}{=} \operatorname*{argmax}_{q \in [0,0.5]} L(q) = 0.25.$$

Two Common Estimates

- **Least-squares estimate.**
  For $q \in [0, 0.5]$,
  $$SE(q) = (q - 1)^2 + (2q - 0)^2.$$
  $$\hat{p}_{LS} \stackrel{\text{def}}{=} \operatorname*{argmin}_{q \in [0,0.5]} SE(q) = 0.2.$$
- **Maximum likelihood estimate.**
  For $q \in [0, 0.5]$,
  $$L(q) = q(1 - 2q).$$
  $$\hat{p}_{ML} \stackrel{\text{def}}{=} \operatorname*{argmax}_{q \in [0,0.5]} L(q) = 0.25.$$
- Which estimate is "correct"?

Two Common Estimates

- **Least-squares estimate.**
  For $q \in [0, 0.5]$,

$$SE(q) = (q - 1)^2 + (2q - 0)^2.$$

$$\hat{p}_{LS} \stackrel{\text{def}}{=} \underset{q \in [0,0.5]}{\operatorname{argmin}} SE(q) = 0.2.$$

- **Maximum likelihood estimate.**
  For $q \in [0, 0.5]$,

$$L(q) = q(1 - 2q).$$

$$\hat{p}_{ML} \stackrel{\text{def}}{=} \underset{q \in [0,0.5]}{\operatorname{argmax}} L(q) = 0.25.$$

- Which estimate is "correct"? Neither!

Two Common Estimates

- **Least-squares estimate.**
  For $q \in [0, 0.5]$,
  $$SE(q) = (q - 1)^2 + (2q - 0)^2.$$
  $$\hat{p}_{LS} \stackrel{\text{def}}{=} \operatorname*{argmin}_{q \in [0,0.5]} SE(q) = 0.2.$$
- **Maximum likelihood estimate.**
  For $q \in [0, 0.5]$,
  $$L(q) = q(1 - 2q).$$
  $$\hat{p}_{ML} \stackrel{\text{def}}{=} \operatorname*{argmax}_{q \in [0,0.5]} L(q) = 0.25.$$
- Which estimate is "correct"? Neither!
- Which estimate is more useful?

Two Common Estimates

- **Least-squares estimate.**
  For $q \in [0, 0.5]$,

$$SE(q) = (q-1)^2 + (2q-0)^2.$$

$$\hat{p}_{LS} \stackrel{\text{def}}{=} \underset{q \in [0,0.5]}{\operatorname{argmin}} SE(q) = 0.2.$$

- **Maximum likelihood estimate.**
  For $q \in [0, 0.5]$,

$$L(q) = q(1-2q).$$

$$\hat{p}_{ML} \stackrel{\text{def}}{=} \underset{q \in [0,0.5]}{\operatorname{argmax}} L(q) = 0.25.$$

- Which estimate is "correct"? Neither!
- Which estimate is more useful? Depends on the use!

Two Common Estimates

- **Least-squares estimate.**
  For $q \in [0, 0.5]$,
  $$SE(q) = (q - 1)^2 + (2q - 0)^2.$$
  $$\hat{p}_{LS} \stackrel{\text{def}}{=} \operatorname*{argmin}_{q \in [0,0.5]} SE(q) = 0.2.$$
- **Maximum likelihood estimate.**
  For $q \in [0, 0.5]$,
  $$L(q) = q(1 - 2q).$$
  $$\hat{p}_{ML} \stackrel{\text{def}}{=} \operatorname*{argmax}_{q \in [0,0.5]} L(q) = 0.25.$$
- Which estimate is "correct"? Neither!
- Which estimate is more useful? Depends on the use!
- Note that there are other estimates, too.

Reinforcement Learning

1. Least-squares and Maximum likelihood estimators
2. On-line implementation of First-visit MC
3. TD(0) algorithm
4. Convergence of Batch TD(0)
5. Control with TD learning

First-visit MC Again

- Assume episodic task with S = $\{s_1, s_2, s_3\}$; following π.
- Say we start each episode with state s (for illustration s_2).

Episode 1: $s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_2, 3, s_2, 3, s_1, 1, s_\top$

First-visit MC Again

- Assume episodic task with S = $\{s_1, s_2, s_3\}$; following π.
- Say we start each episode with state s (for illustration s_2).

Episode 1: $s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_2, 3, s_2, 3, s_1, 1, s_\top$

- $\hat{V}^1 = G(s_2, 1, 1) = 4$.

First-visit MC Again

- Assume episodic task with S = $\{s_1, s_2, s_3\}$; following π.
- Say we start each episode with state s (for illustration s_2).

Episode 1: $s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_2, 3, s_2, 3, s_1, 1, s_\top$

- $\hat{V}^1 = G(s_2, 1, 1) = 4$.
- $\hat{V}^2 = \frac{1}{2}\{G(s_2, 1, 1) + G(s_2, 2, 1)\} = 5.5$.

First-visit MC Again

- Assume episodic task with S = $\{s_1, s_2, s_3\}$; following π.
- Say we start each episode with state s (for illustration s_2).

Episode 1: $s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_2, 3, s_2, 3, s_1, 1, s_\top$.

- $\hat{V}^1 = G(s_2, 1, 1) = 4$.
- $\hat{V}^2 = \frac{1}{2}\{G(s_2, 1, 1) + G(s_2, 2, 1)\} = 5.5$.
- $\hat{V}^3 = \frac{1}{3}\{G(s_2, 1, 1) + G(s_2, 2, 1) + G(s_2, 3, 1)\} \approx 6.33$.

First-visit MC Again

- Assume episodic task with S = $\{s_1, s_2, s_3\}$; following π.
- Say we start each episode with state s (for illustration s_2).

Episode 1: $s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_2, 3, s_2, 3, s_1, 1, s_\top$.

- $\hat{V}^1 = G(s_2, 1, 1) = 4$.
- $\hat{V}^2 = \frac{1}{2}\{G(s_2, 1, 1) + G(s_2, 2, 1)\} = 5.5$.
- $\hat{V}^3 = \frac{1}{3}\{G(s_2, 1, 1) + G(s_2, 2, 1) + G(s_2, 3, 1)\} \approx 6.33$.
- In general, for $t \geq 1$:

$$\hat{V}^t(s) = \frac{1}{t}\sum_{i=1}^{t} G(s, i, 1).$$

An On-line Implementation

$$\hat{V}^t(s) = \frac{1}{t}\sum_{i=1}^{t} G(s, t, 1)$$

An On-line Implementation

$$\hat{V}^t(s) = \frac{1}{t}\sum_{i=1}^{t} G(s, t, 1)$$
$$= \frac{1}{t}\left(\sum_{i=1}^{t-1} G(s, i, 1) + G(s, t, 1)\right)$$

An On-line Implementation

$$\begin{aligned}
\hat{V}^t(s) &= \frac{1}{t}\sum_{i=1}^{t} G(s, t, 1) \\
&= \frac{1}{t}\left(\sum_{i=1}^{t-1} G(s, i, 1) + G(s, t, 1)\right) \\
&= \frac{1}{t}\Big((t-1)\hat{V}^{t-1}(s) + G(s, t, 1)\Big)
\end{aligned}$$

An On-line Implementation

$$
\begin{aligned}
\hat{V}^{t}(s) &= \frac{1}{t}\sum_{i=1}^{t} G(s, t, 1) \\
&= \frac{1}{t}\left(\sum_{i=1}^{t-1} G(s, i, 1) + G(s, t, 1)\right) \\
&= \frac{1}{t}\Big((t-1)\hat{V}^{t-1}(s) + G(s, t, 1)\Big) \\
&= (1-\alpha_t)\hat{V}^{t-1}(s) + \alpha_t G(s, t, 1) \text{ for } \alpha_t = \frac{1}{t}.
\end{aligned}
$$

An On-line Implementation

$$
\begin{aligned}
\hat{V}^t(s) &= \frac{1}{t}\sum_{i=1}^{t} G(s,t,1) \\
&= \frac{1}{t}\left(\sum_{i=1}^{t-1} G(s,i,1) + G(s,t,1)\right) \\
&= \frac{1}{t}\left((t-1)\hat{V}^{t-1}(s) + G(s,t,1)\right) \\
&= (1-\alpha_t)\hat{V}^{t-1}(s) + \alpha_t G(s,t,1) \text{ for } \alpha_t = \frac{1}{t}.
\end{aligned}
$$

- We already know that $\lim_{t\to\infty} \hat{V}^t(s) = V^\pi(s)$.

An On-line Implementation

$$\begin{aligned}
\hat{V}^t(s) &= \frac{1}{t}\sum_{i=1}^{t} G(s,t,1) \\
&= \frac{1}{t}\left(\sum_{i=1}^{t-1} G(s,i,1) + G(s,t,1)\right) \\
&= \frac{1}{t}\left((t-1)\hat{V}^{t-1}(s) + G(s,t,1)\right) \\
&= (1-\alpha_t)\hat{V}^{t-1}(s) + \alpha_t G(s,t,1) \text{ for } \alpha_t = \frac{1}{t}.
\end{aligned}$$

- We already know that $\lim_{t\to\infty} \hat{V}^t(s) = V^\pi(s)$.
- Will we get convergence to $V^\pi(s)$ for other choices for α_t?

Stochastic Approximation

- Result due to Robbins and Monro (1951).

Stochastic Approximation

- Result due to Robbins and Monro (1951).
- Let the sequence $(\alpha_t)_{t \geq 1}$ satisfy
  - $\sum_{t=1}^{\infty} \alpha_t = \infty$.
  - $\sum_{t=1}^{\infty} (\alpha_t)^2 < \infty$.

Stochastic Approximation

- Result due to Robbins and Monro (1951).
- Let the sequence $(\alpha_t)_{t \geq 1}$ satisfy
  - $\sum_{t=1}^{\infty} \alpha_t = \infty$.
  - $\sum_{t=1}^{\infty} (\alpha_t)^2 < \infty$.
- For $t \geq 1$, set

$$\hat{V}^t(s) \leftarrow (1 - \alpha_t)\hat{V}^{t-1}(s) + \alpha_t G(s, t, 1).$$

Stochastic Approximation

- Result due to Robbins and Monro (1951).
- Let the sequence $(\alpha_t)_{t \geq 1}$ satisfy
  - $\sum_{t=1}^{\infty} \alpha_t = \infty$.
  - $\sum_{t=1}^{\infty} (\alpha_t)^2 < \infty$.
- For $t \geq 1$, set

$$\hat{V}^t(s) \leftarrow (1 - \alpha_t)\hat{V}^{t-1}(s) + \alpha_t G(s, t, 1).$$

- Then $\lim_{t \to \infty} \hat{V}^t(s) = V^\pi(s)$.

Stochastic Approximation

- Result due to Robbins and Monro (1951).
- Let the sequence $(\alpha_t)_{t \geq 1}$ satisfy
  - $\sum_{t=1}^{\infty} \alpha_t = \infty$.
  - $\sum_{t=1}^{\infty} (\alpha_t)^2 < \infty$.
- For $t \geq 1$, set

$$\hat{V}^t(s) \leftarrow (1 - \alpha_t)\hat{V}^{t-1}(s) + \alpha_t G(s, t, 1).$$

- Then $\lim_{t \to \infty} \hat{V}^t(s) = V^{\pi}(s)$.
- $(\alpha_t)_{t \geq 1}$ is the "learning rate" or "step size".

Stochastic Approximation

- Result due to Robbins and Monro (1951).
- Let the sequence $(\alpha_t)_{t \geq 1}$ satisfy
  - $\sum_{t=1}^{\infty} \alpha_t = \infty$.
  - $\sum_{t=1}^{\infty} (\alpha_t)^2 < \infty$.
- For $t \geq 1$, set

$$\hat{V}^t(s) \leftarrow (1 - \alpha_t)\hat{V}^{t-1}(s) + \alpha_t G(s, t, 1).$$

- Then $\lim_{t \to \infty} \hat{V}^t(s) = V^\pi(s)$.

- $(\alpha_t)_{t \geq 1}$ is the "learning rate" or "step size".
- Must be large enough, as well as small enough!

Stochastic Approximation

- Result due to Robbins and Monro (1951).
- Let the sequence $(\alpha_t)_{t \geq 1}$ satisfy
  - $\sum_{t=1}^{\infty} \alpha_t = \infty$.
  - $\sum_{t=1}^{\infty} (\alpha_t)^2 < \infty$.
- For $t \geq 1$, set

$$\hat{V}^t(s) \leftarrow (1 - \alpha_t)\hat{V}^{t-1}(s) + \alpha_t G(s, t, 1).$$

- Then $\lim_{t \to \infty} \hat{V}^t(s) = V^\pi(s)$.

- $(\alpha_t)_{t \geq 1}$ is the "learning rate" or "step size".
- Must be large enough, as well as small enough!
- No need to store all previous episodes; t and $\hat{V}^t$ suffice.

Reinforcement Learning

1. Least-squares and Maximum likelihood estimators
2. On-line implementation of First-visit MC
3. TD(0) algorithm
4. Convergence of Batch TD(λ)
5. Control with TD learning

Bootstrapping

- Suppose $\hat{V}^t$ is our current estimate of state-values.

Bootstrapping

- Suppose $\hat{V}^t$ is our current estimate of state-values.
- Say we generate this episode.

$$\boxed{s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top.}$$

Bootstrapping

- Suppose $\hat{V}^t$ is our current estimate of state-values.
- Say we generate this episode.

$$\boxed{s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top.}$$

- At what point of time can we update our estimate $\hat{V}^t(s_2)$?

Bootstrapping

- Suppose $\hat{V}^t$ is our current estimate of state-values.
- Say we generate this episode.

$$\boxed{s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top.}$$

- At what point of time can we update our estimate $\hat{V}^t(s_2)$?

- With MC methods, we would wait for $s_\top$, and then update $\hat{V}^{t+1}(s_2) \leftarrow \hat{V}^t(s_2)(1 - \alpha_{t+1}) + \alpha_{t+1} M$, where $M = 2 + \gamma \cdot 1 + \gamma^2 \cdot 1 + \gamma^3 \cdot 2 + \gamma^4 \cdot 1$.

Bootstrapping

- Suppose $\hat{V}^t$ is our current estimate of state-values.
- Say we generate this episode.

$$s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top.$$

- At what point of time can we update our estimate $\hat{V}^t(s_2)$?

- With MC methods, we would wait for $s_\top$, and then update $\hat{V}^{t+1}(s_2) \leftarrow \hat{V}^t(s_2)(1 - \alpha_{t+1}) + \alpha_{t+1}M$, where $M = 2 + \gamma \cdot 1 + \gamma^2 \cdot 1 + \gamma^3 \cdot 2 + \gamma^4 \cdot 1$.

- Instead, how about this update as soon as we see s_3? $\hat{V}^{t+1}(s_2) \leftarrow \hat{V}^t(s_2)(1 - \alpha_{t+1}) + \alpha_{t+1}B$, where $B = 2 + \gamma\hat{V}^t(s_3)$.

Bootstrapping

- Suppose $\hat{V}^t$ is our current estimate of state-values.
- Say we generate this episode.

$$\boxed{s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top.}$$

- At what point of time can we update our estimate $\hat{V}^t(s_2)$?

- With MC methods, we would wait for $s_\top$, and then update
$\hat{V}^{t+1}(s_2) \leftarrow \hat{V}^t(s_2)(1 - \alpha_{t+1}) + \alpha_{t+1} M$, where
$M = 2 + \gamma \cdot 1 + \gamma^2 \cdot 1 + \gamma^3 \cdot 2 + \gamma^4 \cdot 1$. Monte Carlo estimate.

- Instead, how about this update as soon as we see s_3?
$\hat{V}^{t+1}(s_2) \leftarrow \hat{V}^t(s_2)(1 - \alpha_{t+1}) + \alpha_{t+1} B$, where
$B = 2 + \gamma \hat{V}^t(s_3)$. Bootstrapped estimate.

Temporal Difference Learning: TD(0)

Assume policy to be evaluated is π.
Initialise $\hat{V}^0$ arbitrarily.
Assume that the agent is born in state s^0.

For $t = 0, 1, 2, \ldots$:
  Take action $a^t \sim \pi(s^t)$.
  Obtain reward r^t, next state s^{t+1}.
  $\hat{V}^{t+1}(s^t) \leftarrow \hat{V}^t(s^t) + \alpha_{t+1}\{r^t + \gamma \hat{V}^t(s^{t+1}) - \hat{V}^t(s^t)\}$.
  For $s \in S \setminus \{s^t\}$: $\hat{V}^{t+1}(s) \leftarrow \hat{V}^t(s)$. //Often left implicit.

Temporal Difference Learning: TD(0)

Assume policy to be evaluated is π.
Initialise $\hat{V}^0$ arbitrarily.
Assume that the agent is born in state s^0.

For $t = 0, 1, 2, \ldots$:
    Take action $a^t \sim \pi(s^t)$.
    Obtain reward r^t, next state s^{t+1}.
    $\hat{V}^{t+1}(s^t) \leftarrow \hat{V}^t(s^t) + \alpha_{t+1}\{r^t + \gamma \hat{V}^t(s^{t+1}) - \hat{V}^t(s^t)\}$.
    For $s \in S \setminus \{s^t\}$: $\hat{V}^{t+1}(s) \leftarrow \hat{V}^t(s)$. //Often left implicit.

- $\hat{V}^t(s^t)$: current estimate; $r^t + \gamma \hat{V}^t(s^{t+1})$: new estimate.
- $r^t + \gamma \hat{V}^t(s^{t+1}) - \hat{V}^t(s^t)$: temporal difference prediction error.
- α_{t+1}: learning rate.

Temporal Difference Learning: TD(0)

Assume policy to be evaluated is π.
Initialise $\hat{V}^0$ arbitrarily.
Assume that the agent is born in state s^0.

For $t = 0, 1, 2, \ldots$:
 Take action $a^t \sim \pi(s^t)$.
 Obtain reward r^t, next state s^{t+1}.
 $\hat{V}^{t+1}(s^t) \leftarrow \hat{V}^t(s^t) + \alpha_{t+1}\{r^t + \gamma \hat{V}^t(s^{t+1}) - \hat{V}^t(s^t)\}$.
 For $s \in S \setminus \{s^t\}$: $\hat{V}^{t+1}(s) \leftarrow \hat{V}^t(s)$. //Often left implicit.

- $\hat{V}^t(s^t)$: current estimate; $r^t + \gamma \hat{V}^t(s^{t+1})$: new estimate.
- $r^t + \gamma \hat{V}^t(s^{t+1}) - \hat{V}^t(s^t)$: temporal difference prediction error.
- α_{t+1}: learning rate.
- Under standard conditions, $\lim_{t \to \infty} \hat{V}^t = V^\pi$.

Temporal Difference Learning: TD(0)

Assume policy to be evaluated is π.
Initialise $\hat{V}^0$ arbitrarily.
Assume that the agent is born in state s^0.

For $t = 0, 1, 2, \ldots$:
  Take action $a^t \sim \pi(s^t)$.
  Obtain reward r^t, next state s^{t+1}.
  $\hat{V}^{t+1}(s^t) \leftarrow \hat{V}^t(s^t) + \alpha_{t+1}\{r^t + \gamma \hat{V}^t(s^{t+1}) - \hat{V}^t(s^t)\}$.
  For $s \in S \setminus \{s^t\}$: $\hat{V}^{t+1}(s) \leftarrow \hat{V}^t(s)$. //Often left implicit.

- $\hat{V}^t(s^t)$: current estimate; $r^t + \gamma \hat{V}^t(s^{t+1})$: new estimate.
- $r^t + \gamma \hat{V}^t(s^{t+1}) - \hat{V}^t(s^t)$: temporal difference prediction error.
- α_{t+1}: learning rate.
- Under standard conditions, $\lim_{t \to \infty} \hat{V}^t = V^\pi$.
- In episodic tasks, keep $\hat{V}^t(s_\top)$ fixed at 0 (no updating).

Reinforcement Learning

1. Least-squares and Maximum likelihood estimators
2. On-line implementation of First-visit MC
3. TD(0) algorithm
4. Convergence of Batch TD(0)
5. Control with TD learning

First-visit MC Estimate

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.

- Recall that for $s \in S$,

$$\hat{V}^T_{\text{First-visit}}(s) = \frac{\sum_{i=1}^{T} G(s, i, 1)}{\sum_{i=1}^{T} \mathbf{1}(s, i, 1)}.$$

First-visit MC Estimate

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.

- Recall that for $s \in S$,

$$\hat{V}^T_{\text{First-visit}}(s) = \frac{\sum_{i=1}^{T} G(s, i, 1)}{\sum_{i=1}^{T} \mathbf{1}(s, i, 1)}.$$

- For $s \in S$, $V : S \to \mathbb{R}$, define

$$Error_{\text{First}}(V, s) \stackrel{\text{def}}{=} \sum_{i=1}^{T} \mathbf{1}(s, i, 1) \left(V(s) - G(s, i, 1)\right)^2 .$$

First-visit MC Estimate

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.

- Recall that for $s \in S$,

$$\hat{V}^T_{\text{First-visit}}(s) = \frac{\sum_{i=1}^{T} G(s, i, 1)}{\sum_{i=1}^{T} \mathbf{1}(s, i, 1)}.$$

- For $s \in S$, $V : S \to \mathbb{R}$, define

$$Error_{\text{First}}(V, s) \stackrel{\text{def}}{=} \sum_{i=1}^{T} \mathbf{1}(s, i, 1) \left(V(s) - G(s, i, 1) \right)^2.$$

- Observe that for $s \in S$, $\hat{V}^T_{\text{First-visit}}(s) = \operatorname{argmin}_V Error_{\text{First}}(V, s)$.

Every-visit MC Estimate

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.

- Recall that for $s \in S$,

$$\hat{V}^T_{\text{Every-visit}}(s) = \frac{\sum_{i=1}^{T} \sum_{j=1}^{\infty} G(s, i, j)}{\sum_{i=1}^{T} \sum_{j=1}^{\infty} \mathbf{1}(s, i, j)}.$$

Every-visit MC Estimate

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.

- Recall that for $s \in S$,

$$\hat{V}^T_{\text{Every-visit}}(s) = \frac{\sum_{i=1}^{T} \sum_{j=1}^{\infty} G(s,i,j)}{\sum_{i=1}^{T} \sum_{j=1}^{\infty} \mathbf{1}(s,i,j)}.$$

- For $s \in S$, $V : S \to \mathbb{R}$, define

$$Error_{\text{Every}}(V, s) \stackrel{\text{def}}{=} \sum_{i=1}^{T} \sum_{j=1}^{\infty} \mathbf{1}(s,i,j)\,(V(s) - G(s,i,j))^2 .$$

Every-visit MC Estimate

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.

- Recall that for $s \in S$,

$$\hat{V}^T_{\text{Every-visit}}(s) = \frac{\sum_{i=1}^{T} \sum_{j=1}^{\infty} G(s, i, j)}{\sum_{i=1}^{T} \sum_{j=1}^{\infty} \mathbf{1}(s, i, j)}.$$

- For $s \in S$, $V : S \to \mathbb{R}$, define

$$Error_{\text{Every}}(V, s) \stackrel{\text{def}}{=} \sum_{i=1}^{T} \sum_{j=1}^{\infty} \mathbf{1}(s, i, j) \left(V(s) - G(s, i, j) \right)^2.$$

- Observe for $s \in S$, $\hat{V}^T_{\text{Every-visit}}(s) = \text{argmin}_V \, Error_{\text{Every}}(V, s)$.

Batch TD(0) Estimate

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.

- After any finite T episodes, the estimate of $TD(0)$ will depend on the initial estimate V^0.
- To "forget" V^0, run the T collected episodes over and over again, and make TD(0) updates.

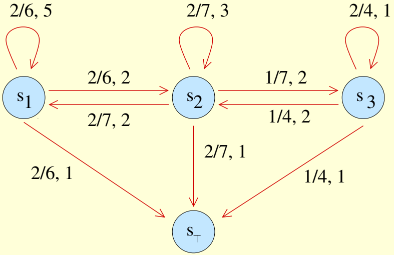
Batch TD(0) Estimate

Episode 1
Episode 2
Episode 3
Episode 4
Episode 5
Episode 6 (= Episode 1)
Episode 7 (= Episode 2)
Episode 8 (= Episode 3)
Episode 9 (= Episode 4)
Episode 10 (= Episode 5)
Episode 11 (= Episode 1)
Episode 12 (= Episode 2)
$\vdots$

- Anneal the learning rate as usual ($\alpha_t = \frac{1}{t}$).
- $\lim_{t\to\infty} V^t$ will not depend on $\hat{V}^0$.
- It only depends on T episodes of real data.
- Refer to $\lim_{t\to\infty} \hat{V}^t$ as $\hat{V}^T_{\text{Batch-TD(0)}}$.
- Can we conclude something relevant about $\hat{V}^T_{\text{Batch-TD(0)}}$?

Batch TD(0) Estimate

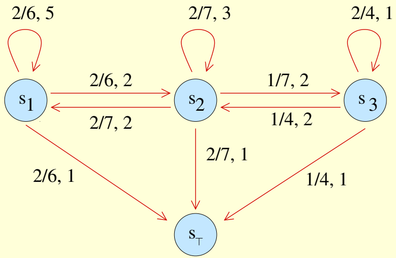
Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.



- Let M_{MLE} be the MDP $(S, A, \hat{T}, \hat{R}, \gamma)$ with the highest likelihood of generating this data (true T, R unknown).

Batch TD(0) Estimate

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.



- Let M_{MLE} be the MDP $(S, A, \hat{T}, \hat{R}, \gamma)$ with the highest likelihood of generating this data (true T, R unknown).
- $\hat{V}^T_{\text{Batch-TD(0)}}$ is the same as V^π on M_{MLE}!

Comparison

- Data.

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.

- Estimates.

	s_1	s_2	s_3
$\hat{V}^T_{\text{First-visit}}$	7.33	6.25	3
$\hat{V}^T_{\text{Every-visit}}$	5.83	4.29	3.25
$\hat{V}^T_{\text{Batch-TD(0)}}$	7.5	7	6

Comparison

- Data.

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.

- Estimates.

	s_1	s_2	s_3
$\hat{V}^T_{\text{First-visit}}$	7.33	6.25	3
$\hat{V}^T_{\text{Every-visit}}$	5.83	4.29	3.25
$\hat{V}^T_{\text{Batch-TD(0)}}$	7.5	7	6

- Which estimate is "correct"? Which is more useful?
- Is it recommended to bootstrap or not?

Comparison

- Data.

Episode 1: $s_1, 5, s_1, 2, s_2, 3, s_2, 1, s_\top$.
Episode 2: $s_2, 2, s_3, 1, s_3, 1, s_3, 2, s_2, 1, s_\top$.
Episode 3: $s_1, 2, s_2, 2, s_1, 5, s_1, 1, s_\top$.
Episode 4: $s_3, 1, s_\top$.
Episode 5: $s_2, 3, s_2, 2, s_1, 1, s_\top$.

- Estimates.

	s_1	s_2	s_3
$\hat{V}^T_{\text{First-visit}}$	7.33	6.25	3
$\hat{V}^T_{\text{Every-visit}}$	5.83	4.29	3.25
$\hat{V}^T_{\text{Batch-TD(0)}}$	7.5	7	6

- Which estimate is "correct"? Which is more useful?
- Is it recommended to bootstrap or not?
- Usually a "middle path" works best. Coming up next week!

Reinforcement Learning

1. Least-squares and Maximum likelihood estimators
2. On-line implementation of First-visit MC
3. TD(0) algorithm
4. Convergence of Batch TD(0)
5. Control with TD learning

Sketch

1. Maintain action value function estimate $\hat{Q}^t : S \times A \to \mathbb{R}$ for $t \geq 0$, initialised arbitrarily.

   We would like to get $\hat{Q}^t$ to converge to $Q^\star$.

Sketch

1. Maintain action value function estimate $\hat{Q}^t : S \times A \to \mathbb{R}$ for $t \geq 0$, initialised arbitrarily.

   We would like to get $\hat{Q}^t$ to converge to $Q^\star$.

2. Follow policy π^t at time step $t \geq 0$, for example one that is ϵ_t-greedy with respect to $\hat{Q}^t$.

   Set ϵ_t to ensure infinite exploration of every state-action pair and also being greedy in the limit.

Sketch

1. Maintain action value function estimate $\hat{Q}^t : S \times A \to \mathbb{R}$ for $t \geq 0$, initialised arbitrarily.
   We would like to get $\hat{Q}^t$ to converge to $Q^\star$.

2. Follow policy π^t at time step $t \geq 0$, for example one that is ϵ_t-greedy with respect to $\hat{Q}^t$.
   Set ϵ_t to ensure infinite exploration of every state-action pair and also being greedy in the limit.

3. Every transition (s^t, a^t, r^t, s^{t+1}) conveys information about the underlying MDP. Update $\hat{Q}^t$ based on the transition.
   Can use TD learning (suitably adapted) to make the update.

Sketch

1. Maintain action value function estimate $\hat{Q}^t : S \times A \to \mathbb{R}$ for $t \geq 0$, initialised arbitrarily.
   We would like to get $\hat{Q}^t$ to converge to $Q^\star$.

2. Follow policy π^t at time step $t \geq 0$, for example one that is ϵ_t-greedy with respect to $\hat{Q}^t$.
   Set ϵ_t to ensure infinite exploration of every state-action pair and also being greedy in the limit.

3. Every transition (s^t, a^t, r^t, s^{t+1}) conveys information about the underlying MDP. Update $\hat{Q}^t$ based on the transition.
   Can use TD learning (suitably adapted) to make the update.
   We see three different update rules.

Three Control Algorithms

- From state s^t, action taken is $a^t \sim \pi^t(s^t)$.

Three Control Algorithms

- From state s^t, action taken is $a^t \sim \pi^t(s^t)$.
- Update made to $\hat{Q}^t$ after observing transition s^t, a^t, r^t, s^{t+1}:

$$\hat{Q}^{t+1}(s^t, a^t) \leftarrow \hat{Q}^t(s^t, a^t) + \alpha_{t+1}\{\text{Target} - \hat{Q}^t(s_t, a^t)\}.$$

Three Control Algorithms

- From state s^t, action taken is $a^t \sim \pi^t(s^t)$.
- Update made to $\hat{Q}^t$ after observing transition s^t, a^t, r^t, s^{t+1}:

$$\hat{Q}^{t+1}(s^t, a^t) \leftarrow \hat{Q}^t(s^t, a^t) + \alpha_{t+1}\{\text{Target} - \hat{Q}^t(s_t, a^t)\}.$$

Q-learning: $\text{Target} = r^t + \gamma \max_{a \in A} \hat{Q}^t(s^{t+1}, a)$.

Sarsa: $\text{Target} = r^t + \gamma \hat{Q}^t(s^{t+1}, a^{t+1})$.

Expected Sarsa: $\text{Target} = r^t + \gamma \sum_{a \in A} \pi^t(s^{t+1}, a)\hat{Q}^t(s^{t+1}, a)$.

Three Control Algorithms

- From state s^t, action taken is $a^t \sim \pi^t(s^t)$.
- Update made to $\hat{Q}^t$ after observing transition s^t, a^t, r^t, s^{t+1}:

$$\hat{Q}^{t+1}(s^t, a^t) \leftarrow \hat{Q}^t(s^t, a^t) + \alpha_{t+1}\{\text{Target} - \hat{Q}^t(s_t, a^t)\}.$$

Q-learning: Target $= r^t + \gamma \max_{a \in A} \hat{Q}^t(s^{t+1}, a)$.

Sarsa: Target $= r^t + \gamma \hat{Q}^t(s^{t+1}, a^{t+1})$.

Expected Sarsa: Target $= r^t + \gamma \sum_{a \in A} \pi^t(s^{t+1}, a) \hat{Q}^t(s^{t+1}, a)$.

- Q-learning's update is off-policy; the other two are on-policy.
- $\lim_{t \to \infty} \hat{Q}^t = Q^\star$ for all three if π^t is ϵ_t-greedy w.r.t. $\hat{Q}^t$.
- If $\pi^t = \pi$ (time-invariant) and it still visits every state-action pair infinitely often, then $\lim_{t \to \infty} \hat{Q}^t$ is Q^π for Sarsa and Expected Sarsa, but is $Q^\star$ for Q-learning!

Temporal Difference Learning: Review

- Temporal difference (TD) learning is at the heart of RL.
- An instance of on-line learning (computationally cheap updates after each interaction).
- Applies to both prediction and control.
- Q-learning, Sarsa, Expected Sarsa are all model-free (use $\theta(|S||A|)$-sized memory); can still be optimal in the limit.
- Bootstrapping exploits the underlying Markovian structure, which Monte Carlo methods ignore.
- The TD(λ) family of algorithms, $\lambda \in [0, 1]$, allows for controlling the extent of bootstrapping: $\lambda = 0$ implements "full bootstrapping" and $\lambda = 1$ is "no bootstrapping."

Temporal Difference Learning: Review

- Temporal difference (TD) learning is at the heart of RL.
- An instance of on-line learning (computationally cheap updates after each interaction).
- Applies to both prediction and control.
- Q-learning, Sarsa, Expected Sarsa are all model-free (use $\theta(|S||A|)$-sized memory); can still be optimal in the limit.
- Bootstrapping exploits the underlying Markovian structure, which Monte Carlo methods ignore.
- The TD(λ) family of algorithms, $\lambda \in [0, 1]$, allows for controlling the extent of bootstrapping: $\lambda = 0$ implements "full bootstrapping" and $\lambda = 1$ is "no bootstrapping."

  Coming up next week.