Lecture 3: Model-Free Policy Evaluation: Policy Evaluation Without Knowing How the World Works[1]

Emma Brunskill

CS234 Reinforcement Learning

Winter 2019

[1]Material builds on structure from David SIlver's Lecture 4: Model-Free Prediction. Other resources: Sutton and Barto Jan 1 2018 draft Chapter/Sections: 5.1; 5.5; 6.1-6.3

Today's Plan

前回まで：
Markov reward と　decision process

- Last Time:
  - Markov reward / decision processes
  - Policy evaluation & control when have true model (of how the world works)
- **Today**
  - **Policy evaluation without known dynamics & reward models**
- Next Time:
  - Control when don't have a model of how the world works

今日はモデルフリー型の方策評価の話

世界がどう動くかわからないモデルベース型の場合は次回
そうではない？

This Lecture: Policy Evaluation

今日の講義でやること

MDP（マルコフ決定過程）モデルがわからない場合の期待リターンを推定する

- Estimating the expected return of a particular policy if don't have access to true MDP models
- Dynamic programming
  動的計画法
- Monte Carlo policy evaluation
  - Policy evaluation when don't have a model of how the world work
    - Given on-policy samples
      モンテカルロ推定
- Temporal Difference (TD)
  TD法
- Metrics to evaluate and compare algorithms

評価のアルゴリズムについて

Recall

復習

- Definition of Return, G_t (for a MRP)
  - Discounted sum of rewards from time step t to horizon

値引き合計

$$G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \cdots$$

- Definition of State Value Function, $V^\pi(s)$
  - Expected return from starting in state s under policy π

$$V^\pi(s) = \mathbb{E}_\pi[G_t | s_t = s] = \mathbb{E}_\pi[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \cdots | s_t = s]$$

- Definition of State-Action Value Function, $Q^\pi(s, a)$
  - Expected return from starting in state s, taking action a and then following policy π

$$\begin{aligned} Q^\pi(s, a) &= \mathbb{E}_\pi[G_t | s_t = s, a_t = a] \\ &= \mathbb{E}_\pi[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \cdots | s_t = s, a_t = a] \end{aligned}$$

Dynamic Programming for Policy Evaluation

動的計画法について

- Initialize $V_0^\pi(s) = 0$ for all s
- For $k = 1$ until convergence
  - For all s in S

$$V_k^\pi(s) = r(s, \pi(s)) + \gamma \sum_{s' \in S} p(s'|s, \pi(s)) V_{k-1}^\pi(s')$$

Dynamic Programming for Policy π, Value Evaluation

- Initialize $V_0^\pi(s) = 0$ for all s
- For $k = 1$ until convergence
  - For all s in S

$$V_k^\pi(s) = r(s, \pi(s)) + \gamma \sum_{s' \in S} p(s'|s, \pi(s)) V_{k-1}^\pi(s')$$

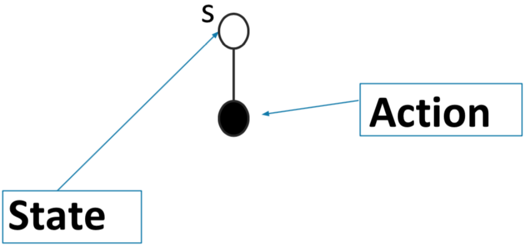
- $V_k^\pi(s)$ is exact value of k-horizon value of state s under policy π
- $V_k^\pi(s)$ is an estimate of infinite horizon value of state s under policy π

V^pi(s)の間違いでは？

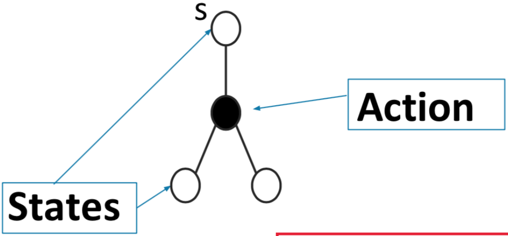
$$V^\pi(s) = \mathbb{E}_\pi[G_t|s_t = s] \approx \mathbb{E}_\pi[r_t + \gamma V_{k-1}|s_t = s]$$

なんでイコールではない？
→最終的に収束するから

Dynamic Programming Policy Evaluation $V^\pi(s) \leftarrow \mathbb{E}_\pi[r_t + \gamma V_{k-1} | s_t = s]$

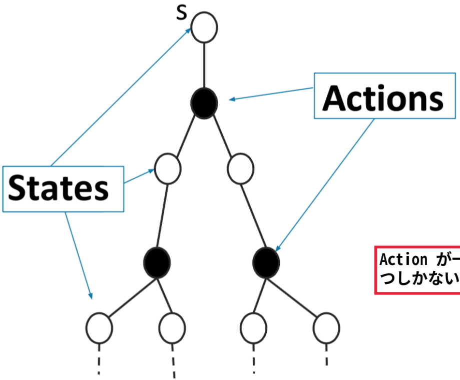


Dynamic Programming Policy Evaluation $V^\pi(s) \leftarrow \mathbb{E}_\pi[r_t + \gamma V_{k-1} | s_t = s]$



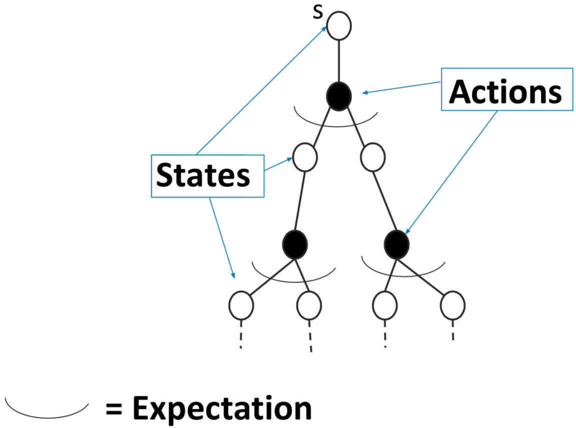
Action をとることで state が変化する

Dynamic Programming Policy Evaluation $V^\pi(s) \leftarrow \mathbb{E}_\pi[r_t + \gamma V_{k-1} | s_t = s]$



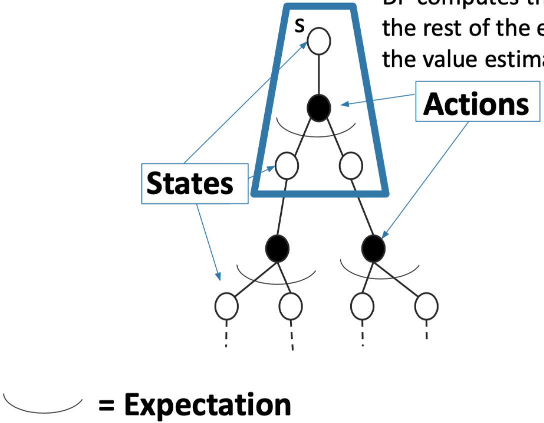
Action が一つしかないのは、action1
つしかない方策piの評価を行うため

Dynamic Programming Policy Evaluation $V^\pi(s) \leftarrow \mathbb{E}_\pi[r_t + \gamma V_{k-1} | s_t = s]$



Dynamic Programming Policy Evaluation $V^{\pi}(s) \leftarrow \mathbb{E}_{\pi}[r_t + \gamma V_{k-1} | s_t = s]$

bootstrap：外部の入力を必要とせずに実行される自己開始型のプロセス



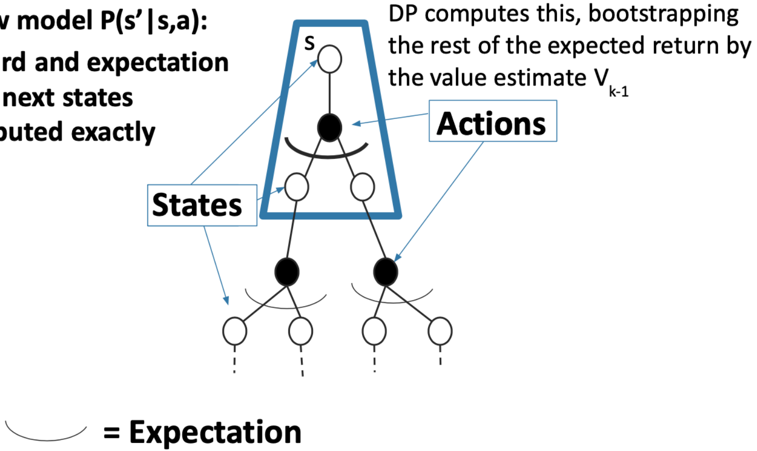
DP computes this, **bootstrapping** the rest of the expected return by the value estimate V_{k-1}

- Bootstrapping: Update for V uses an estimate

Dynamic Programming Policy Evaluation $V^\pi(s) \leftarrow \mathbb{E}_\pi[r_t + \gamma V_{k-1} | s_t = s]$

Know model P(s'|s,a):

reward and expectation over next states computed exactly



- Bootstrapping: Update for V uses an estimate

Policy Evaluation: $V^{\pi}(s) = \mathbb{E}_{\pi}[G_t | s_t = s]$

- $G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \cdots$ in MDP M under policy π
- Dynamic Programming
  - $V^{\pi}(s) \approx \mathbb{E}_{\pi}[r_t + \gamma V_{k-1} | s_t = s]$
  - **Requires model of MDP M**
  - Bootstraps future return using value estimate
  - Requires Markov assumption: bootstrapping regardless of history
- What if don't know dynamics model P and/ or reward model R?
- **Today: Policy evaluation without a model**
  - Given data and/or ability to interact in the environment
  - Efficiently compute a good estimate of a policy π

今日の議題：モデルフリー型の方策評価

This Lecture Overview: Policy Evaluation

- Dynamic Programming
- **Evaluating the quality of an estimator**
- **Monte Carlo policy evaluation**
    - Policy evaluation when don't know dynamics and/or reward model
        - Given on policy samples
- Temporal Difference (TD)
- Metrics to evaluate and compare algorithms

Monte Carlo (MC) Policy Evaluation

- $G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \cdots$ in MDP M under policy π
- $V^\pi(s) = \mathbb{E}_{T \sim \pi}[G_t | s_t = s]$
  - Expectation over trajectories T generated by following π
- Simple idea: Value = mean return
- If trajectories are all finite, sample set of trajectories & average returns

Tは方策piによって生み出される軌跡
（訪れる一連の状態のことでいい？）

Tが有限の場合は、シンプルに平均を
とってreturnを決めましょう

Monte Carlo (MC) Policy Evaluation

モンテカルロ方策評価

- If trajectories are all finite, sample set of trajectories & average returns
- Does not require MDP dynamics/rewards

マルコフ決定過程である必要なし！
環境からリターンをサンプリングできるだけでいい

- No bootstrapping

V_k-1 を覚えておく必要なし

- Does not assume state is Markov
- Can **only** be applied to episodic MDPs
  - Averaging over returns from a complete episo
  - Requires each episode to terminate

先生の例：
空港までの所要時間を考える際、いつも同じ方策（高速道路使う）ででた結果の平均値（100分）は推定値として妥当だよね

一連のepisodeに関するデータのみ経験できる必要がある
「人生は一度きり」

Monte Carlo (MC) On Policy Evaluation

- Aim: estimate $V^\pi(s)$ given episodes generated under policy π
  - $s_1, a_1, r_1, s_2, a_2, r_2, \ldots$ where the actions are sampled from π
- $G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \cdots$ in MDP M under policy π
- $V^\pi(s) = \mathbb{E}_\pi[G_t | s_t = s]$
- MC computes empirical mean return
- Often do this in an incremental fashion
  - After each episode, update estimate of V^π

経験平均リターンの計算

First-Visit Monte Carlo (MC) On Policy Evaluation

最初にsに訪れた時のreturnのみ考えるモンテカルロ法

Initialize $N(s) = 0$, $G(s) = 0 \ \forall s \in S$
Loop

- Sample episode $i = s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$
- Define $G_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \cdots \gamma^{T_i - 1} r_{i,T_i}$ as return from time step t onwards in ith episode
- For each state s visited in episode i
  - For **first** time t that state s is visited in episode i
    - Increment counter of total first visits: $N(s) = N(s) + 1$
    - Increment total return $G(s) = G(s) + G_{i,t}$
    - Update estimate $V^\pi(s) = G(s)/N(s)$

以降

Bias, Variance and MSE

推定方策V^pi(s)をどうやって評価するかの指標の話

θによってパラメータ設定され、観測データから分布P(x¦θ)を決定するモデルについて考える

- Consider a statistical model that is parameterized by θ and that determines a probability distribution over observed data $P(x|\theta)$
- Consider a statistic $\hat{\theta}$ that provides an estimate of θ and is a function of observed data x
  - E.g. for a Gaussian distribution with known variance, the average of a set of i.i.d data points is an estimate of the mean of the Gaussian
- Definition: the bias of an estimator $\hat{\theta}$ is:

$$Bias_\theta(\hat{\theta}) = \mathbb{E}_{x|\theta}[\hat{\theta}] - \theta$$

bias：真の値からのずれ

- Definition: the variance of an estimator $\hat{\theta}$ is:

$$Var(\hat{\theta}) = \mathbb{E}_{x|\theta}[(\hat{\theta} - \mathbb{E}[\hat{\theta}])^2]$$

variance：推定値の分散

- Definition: mean squared error (MSE) of an estimator $\hat{\theta}$ is:

$$MSE(\hat{\theta}) = Var(\hat{\theta}) + Bias_\theta(\hat{\theta})^2$$

First-Visit Monte Carlo (MC) On Policy Evaluation

Initialize $N(s) = 0$, $G(s) = 0 \ \forall s \in S$
Loop

- Sample episode $i = s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$
- Define $G_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \cdots \gamma^{T_i - 1} r_{i,T_i}$ as return from time step t onwards in ith episode
- For each state s visited in episode i
  - For **first** time t that state s is visited in episode i
    - Increment counter of total first visits: $N(s) = N(s) + 1$
    - Increment total return $G(s) = G(s) + G_{i,t}$
    - Update estimate $V^\pi(s) = G(s)/N(s)$

Properties:

- V^π estimator is an unbiased estimator of true $\mathbb{E}_\pi[G_t | s_t = s]$
- By law of large numbers, as $N(s) \to \infty$, $V^\pi(s) \to \mathbb{E}_\pi[G_t | s_t = s]$

大数の法則から、推定値V^piは真の値に収束する

Every-Visit Monte Carlo (MC) On Policy Evaluation

一つのepisode中でsに訪れた時全てのreturnを考慮するモンテカルロ法

Initialize $N(s) = 0, \; G(s) = 0 \; \forall s \in S$
Loop

- Sample episode $i = s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$
- Define $G_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \cdots \gamma^{T_i-1} r_{i,T_i}$ as return from time step t onwards in ith episode
- For each state s visited in episode i
  - For **every** time t that state s is visited in episode i
    - Increment counter of total first visits: $N(s) = N(s) + 1$
    - Increment total return $G(s) = G(s) + G_{i,t}$
    - Update estimate $V^{\pi}(s) = G(s)/N(s)$

Every-Visit Monte Carlo (MC) On Policy Evaluation

Initialize $N(s) = 0,\ G(s) = 0\ \forall s \in S$

Loop

- Sample episode $i = s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \dots, s_{i,T_i}$
- Define $G_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \cdots \gamma^{T_i - 1} r_{i,T_i}$ as return from time step t onwards in ith episode
- For each state s visited in episode i
  - For **every** time t that state s is visited in episode i
    - Increment counter of total first visits: $N(s) = N(s) + 1$
    - Increment total return $G(s) = G(s) + G_{i,t}$
    - Update estimate $V^\pi(s) = G(s)/N(s)$

同じところに訪れたらindependentでなくなるからunbiasedになる
→なんで？
同じデータ使うと創刊に引っ張られてbiasが生まれる...?

Properties:

- V^π every-vist MC estimator is an **biased** e
- But consistent estimator and often has better MSE

first-visitと違いV^piにバイアスがかかるが、推定値はconsistentになりMSEはよくなる。データ数多いから？

Incremental Monte Carlo (MC) On Policy Evaluation

After each episode $i = s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots$

- Define $G_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \cdots$ as return from time step t onwards in ith episode
- For state s visited at time step t in episode i
  - Increment counter of total first visits: $N(s) = N(s) + 1$
  - Update estimate

$$V^\pi(s) = V^\pi(s)\frac{N(s)-1}{N(s)} + \frac{G_{i,t}}{N(s)} = V^\pi(s) + \frac{1}{N(s)}(G_{i,t} - V^\pi(s))$$

増分モンテカルロ法

Incremental Monte Carlo (MC) On Policy Evaluation, Running Mean

Initialize $N(s) = 0$, $G(s) = 0$ $\forall s \in S$
Loop

- Sample episode $i = s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$
- Define $G_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \cdots \gamma^{T_i - 1} r_{i,T_i}$ as return from time step t onwards in ith episode
- For state s visited at time step t in episode i
  - Increment counter of total first visits: $N(s) = N(s) + 1$
  - Update estimate

$$V^\pi(s) = V^\pi(s) + \alpha(G_{i,t} - V^\pi(s))$$

- $\alpha = \frac{1}{N(s)}$: identical to every visit MC
- $\alpha > \frac{1}{N(s)}$: forget older data, helpful for non-stationary domains

オンライン推薦システムみたいに、状況が変わる場合に経験的にこの方法が採用される

Check Your Understanding: MC On Policy Evaluation

Initialize $N(s) = 0$, $G(s) = 0$ $\forall s \in S$
Loop

- Sample episode $i = s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$
- $G_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \cdots \gamma^{T_i - 1} r_{i,T_i}$
- For each state s visited in episode i
  - For **first or every** time t that state s is visited in episode i
    - $N(s) = N(s) + 1$, $G(s) = G(s) + G_{i,t}$
    - Update estimate $V^\pi(s) = G(s)/N(s)$

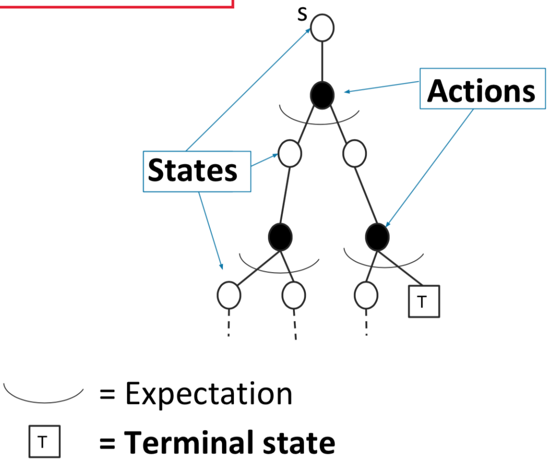
Example:

- Mars rover: R = [1 0 0 0 0 0 +10] for any action
- $\pi(s) = a_1$ $\forall s$, $\gamma = 1$. any action from s_1 and s_7 terminates episode
- Trajectory = (s_3, a_1, 0, s_2, a_1, 0, s_2, a_1, 0, s_1, a_1, 1, terminal)
- First visit MC estimate of V of each state?
  V = [1 1 1 0 0 0 0]
- Every visit MC estimate of s_2?
  $V(s_2) = 1$

MC Policy Evaluation

$$V^\pi(s) = V^\pi(s) + \alpha(G_{i,t} - V^\pi(s))$$

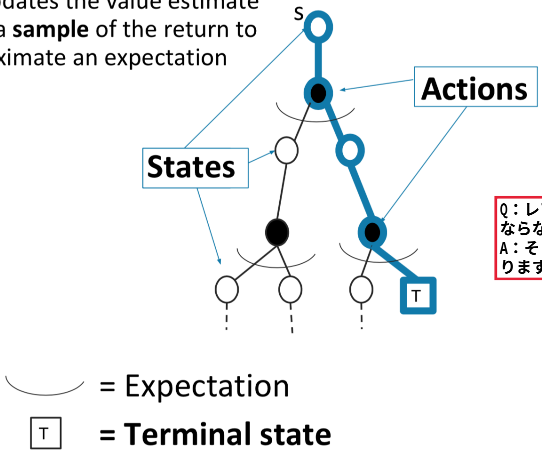
平均の取り方について



MC Policy Evaluation

$$V^{\pi}(s) = V^{\pi}(s) + \alpha(G_{i,t} - V^{\pi}(s))$$

MC updates the value estimate using a **sample** of the return to approximate an expectation



Q：レアな状態の場合推定としてよくならないんじゃないか？
A：そうですね。high variance になります

Monte Carlo (MC) Policy Evaluation Key Limitations

推定値はふつうvarianceが高い
減らすためには大量のデータ必要

- Generally high variance estimator
  - Reducing variance can require a lot of data
- Requires episodic settings
  - Episode must end before data from that episode can be used to update the value function

episode が終わったデータのみ、価値
観数の更新に使える

Monte Carlo (MC) Policy Evaluation Summary

- Aim: estimate $V^\pi(s)$ given episodes generated under policy π
  - $s_1, a_1, r_1, s_2, a_2, r_2, \ldots$ where the actions are sampled from π
- $G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \cdots$ in MDP M under policy π
- $V^\pi(s) = \mathbb{E}_\pi[G_t | s_t = s]$
- Simple: Estimates expectation by empirical average (given episodes sampled from policy of interest) or reweighted empirical average (importance sampling)
- Updates value estimate by using a **sample** of return to approximate the expectation
- No bootstrapping
- Converges to true value under some (generally mild) assumptions

This Lecture: Policy Evaluation

- Estimating the expected return of a particular policy if don't have access to true MDP models
- Dynamic programming
- Monte Carlo policy evaluation
  - Policy evaluation when don't have a model of how the world work
    - Given on-policy samples
- **Temporal Difference (TD)**
- Metrics to evaluate and compare algorithms

Temporal Difference Learning

TD法について

- "If one had to identify one idea as central and novel to reinforcement learning, it would undoubtedly be temporal-difference (TD) learning." – Sutton and Barto 2017
- Combination of Monte Carlo & dynamic programming methods
- Model-free
- **Bootstraps and samples**
- Can be used in episodic or infinite-horizon non-episodic settings
- Immediately updates estimate of V after each (s, a, r, s') tuple

episodeが終了したデータでも終了していないデータでも使用可能

Vの更新に使うのは
現在の状態 s
アクション a
return r
次の状態 s'
のタプル

Temporal Difference Learning for Estimating V

目的：方策piのもとでの価値観数V^piの推定

- Aim: estimate $V^\pi(s)$ given episodes generated under policy π
- $G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \cdots$ in MDP M under policy π
- $V^\pi(s) = \mathbb{E}_\pi[G_t | s_t = s]$
- Recall Bellman operator (if know MDP models)

$$B^\pi V(s) = r(s, \pi(s)) + \gamma \sum_{s' \in S} p(s'|s, \pi(s)) V(s')$$

- In incremental every-visit MC, update estimate using 1 sample of return (for the current ith episode)

$$V^\pi(s) = V^\pi(s) + \alpha(G_{i,t} - V^\pi(s))$$

- Insight: have an estimate of V^π, use to estimate expected return

$$V^\pi(s) = V^\pi(s) + \alpha([r_t + \gamma V^\pi(s_{t+1})] - V^\pi(s))$$

期待return Gの推定値を更新に使いましょう

Temporal Difference [$TD(0)$] Learning

- Aim: estimate $V^\pi(s)$ given episodes generated under policy π
  - $s_1, a_1, r_1, s_2, a_2, r_2, \ldots$ where the actions are sampled from π
- Simplest TD learning: update value towards estimated value

$$V^\pi(s_t) = V^\pi(s_t) + \alpha(\underbrace{[r_t + \gamma V^\pi(s_{t+1})]}_{\text{TD target}} - V^\pi(s_t))$$

↑k+1が添え字

↑kが添え字

- TD error:

$$\delta_t = r_t + \gamma V^\pi(s_{t+1}) - V^\pi(s_t)$$

- Can immediately update value estimate after (s, a, r, s') tuple
- Don't need episodic setting

terminate する必要なし！

Temporal Difference [$TD(0)$] Learning Algorithm

Input: α
Initialize $V^\pi(s) = 0$, $\forall s \in S$
Loop

- Sample **tuple** (s_t, a_t, r_t, s_{t+1})
- $V^\pi(s_t) = V^\pi(s_t) + \alpha(\underbrace{[r_t + \gamma V^\pi(s_{t+1})]}_{\text{TD target}} - V^\pi(s_t))$

Check Your Understanding: TD Learning

Input: α
Initialize $V^\pi(s) = 0$, $\forall s \in S$
Loop

Q：前回やったQ（行動価値関数）と似てるように見えるけど何が違うの？
A：TD学習は方策を固定します

- Sample **tuple** (s_t, a_t, r_t, s_{t+1})
- $V^\pi(s_t) = V^\pi(s_t) + \alpha(\underbrace{[r_t + \gamma V^\pi(s_{t+1})]}_{\text{TD target}} - V^\pi(s_t))$

α=1 なら、過去の推定は全く気にしないようにする

Example:

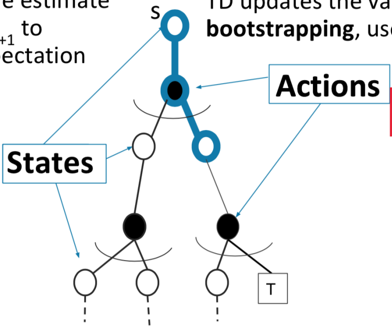
- Mars rover: R = [1 0 0 0 0 0 +10] for any action
- $\pi(s) = a_1$ $\forall s$, $\gamma = 1$. any action from s_1 and s_7 terminates episode
- Trajectory = $(s_3, a_1, 0, s_2, a_1, 0, s_2, a_1, 0, s_1, a_1, 1, \text{terminal})$
- First visit MC estimate of V of each state? [1 1 1 0 0 0 0]
- Every visit MC estimate of V of $s2$? 1
- TD estimate of all states (init at 0) with $\alpha = 1$?
  V = [1 0 0 0 0 0 0 0]

Temporal Difference Policy Evaluation

$$V^{\pi}(s_t) = V^{\pi}(s_t) + \alpha([r_t + \gamma V^{\pi}(s_{t+1})] - V^{\pi}(s_t))$$

TD updates the value estimate using a **sample** of s_{t+1} to approximate an expectation

TD updates the value estimate by **bootstrapping**, uses estimate of $V(s_{t+1})$



次の状態での価値V(s_t+1)を使うのがbootstrapping？

= Expectation

T = **Terminal state**

どうしてTD法がモンテカルロ法と動的計画法のハイブリッドかみたいな話をここでしていたが、よくわからなかった

This Lecture: Policy Evaluation

- Estimating the expected return of a particular policy if don't have access to true MDP models
- Dynamic programming
- Monte Carlo policy evaluation
  - Policy evaluation when don't have a model of how the world work
    - Given on-policy samples
    - Given off-policy samples
- Temporal Difference (TD)
- **Metrics to evaluate and compare algorithms**

Check Your Understanding: For Dynamic Programming, MC and TD Methods, Which Properties Hold?

どの性質を満たしているかの確認
DP, MC, TDごとに

現在のドメインのモデルがない場合に
使用可能か

- Usable when no models of current domain
- Handles continuing (non-episodic) domains episodic でないドメインを扱えるか
- Handles Non-Markovian domains マルコフ性を保持しないドメインを扱えるか
- Converges to true value in limit [1]
- Unbiased estimate of value

制限内で真の値に収束するか（マルコフ性は仮定する）

推定値にバイアスはないか

[1] For tabular representations of value function. More on this in later lectures

Some Important Properties to Evaluate Policy Evaluation Algorithms

Q：モデルフリー型でどのアルゴリズムでも真の値に収束しないものってある？
A：ある。関数近似した場合とか
今は全状態の価値観数を計算できているが、状態数が多くなった場合に関数近似を行う

- Usable when no models of current domain
  - DP: No MC: Yes TD: Yes
- Handles continuing (non-episodic) domains
  - DP: Yes MC: No TD: Yes
- Handles Non-Markovian domains
  - DP: No MC: Yes TD: No
- Converges to true value in limit [2]
  - DP: Yes MC: Yes TD: Yes
- Unbiased estimate of value
  - DP: NA MC: Yes TD: No

Q：TDが unbiased でない理由は？
A：推定値を計算するのに次の状態の価値を使用するから、これがよくない
真のV^piの値を用いない（推定値で代用している）から

consistent estimator かどうか

ここでMCがYesになっているけど、これはfirst-visitの時のみ

Q：最後の二つの違いって？
A：unbiasedは、データ数に制限があるときに真の値に収束するか
consistencyは、データが無限にあるときに、真の値に収束するか否か

[2]For tabular representations of value function. More on this in later lectures

Some Important Properties to Evaluate Model-free Policy Evaluation Algorithms

モデルフリー型方策を評価するアルゴリズムで大事なことは3つ

- Bias/variance characteristics
- Data efficiency
- Computational efficiency

Bias/Variance of Model-free Policy Evaluation Algorithms

- Return G_t is an unbiased estimate of $V^\pi(s_t)$
- TD target $[r_t + \gamma V^\pi(s_{t+1})]$ is a biased estimate of $V^\pi(s_t)$
- But often much lower variance than a single return G_t
- Return function of multi-step sequence of random actions, states & rewards
- TD target only has one random action, reward and next state
- MC
  - Unbiased
  - High variance
  - Consistent (converges to true) even with function approximation
- TD
  - Some bias
  - Lower variance
  - TD(0) converges to true value with tabular representation
  - TD(0) does not always converge with function approximation

s_1	s_2	s_3	s_4	s_5	s_6	s_7
$R(s_1) = +1$ *Okay Field Site*	$R(s_2) = 0$	$R(s_3) = 0$	$R(s_4) = 0$ 	$R(s_5) = 0$	$R(s_6) = 0$	$R(s_7) = +10$ *Fantastic Field Site*

- Mars rover: R $=$ [1 0 0 0 0 0 $+$10] for any action
- $\pi(s) = a_1 \ \forall s$, $\gamma = 1$. any action from s_1 and s_7 terminates episode
- Trajectory $= (s_3, a_1, 0, s_2, a_1, 0, s_2, a_1, 0, s_1, a_1, 1, \text{terminal})$
- First visit MC estimate of V of each state? [1 1 1 0 0 0 0]
- Every visit MC estimate of V of s_2? 1
- TD estimate of all states (init at 0) with $\alpha = 1$ is [1 0 0 0 0 0 0]
- TD(0) only uses a data point (s, a, r, s') once
- Monte Carlo takes entire return from s to end of episode

TD(0)は一つのデータ組(s,a,r,s')しか使用しないのに対し、モンテカルロはスタートからepisodeの最後までを考慮する（からunbiased？）

Batch MC and TD

- Batch (Offline) solution for finite dataset
  - Given set of K episodes
  - Repeatedly sample an episode from K
  - Apply MC or TD(0) to the sampled episode
- What do MC and TD(0) converge to?

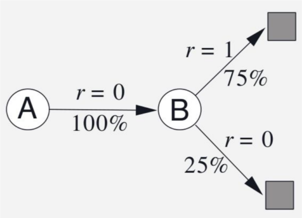
MC, TD(0)は何に収束するか

AB Example: (Ex. 6.4, Sutton & Barto, 2018

αは十分小さいものとする

- Two states A, B with $\gamma = 1$
- Given 8 episodes of experience:
  - $A, 0, B, 0$
  - $B, 1$ (observed 6 times)
  - $B, 0$
- What are $V(A), V(B)$?

AB Example: (Ex. 6.4, Sutton & Barto, 2018)



- Two states A, B with $\gamma = 1$
- Given 8 episodes of experience:
  - $A, 0, B, 0$
  - $B, 1$ (observed 6 times)
  - $B, 0$
- $V(B) = 0.75$ by TD or MC
- What about $V(A)$?

V(B)は,MCだと6/8,TDだと同じく6/8に収束する

Batch MC and TD: Converges

- Monte Carlo in batch setting converges to min MSE (mean squared error)
  - Minimize loss with respect to observed returns
  - In AB example, $V(A) = 0$

MCはAを含むepisodeでreturnが0なので、V(A)=0

- TD(0) converges to DP policy V^π for the MDP with the maximum likelihood model estimates
  - Maximum likelihood Markov decision process model

P(B|A)=1
V(B)=3/4
r(A)=0

$$\hat{P}(s'|s,a) = \frac{1}{N(s,a)} \sum_{k=1}^{K} \sum_{t=1}^{L_k - 1} \mathbb{1}(s_{k,t} = s, a_{k,t} = a, s_{k,t+1} = s')$$

$$\hat{r}(s,a) = \frac{1}{N(s,a)} \sum_{k=1}^{K} \sum_{t=1}^{L_k - 1} \mathbb{1}(s_{k,t} = s, a_{k,t} = a) r_{t,k}$$

  - Compute V^π using this model
  - In AB example, $V(A) = 0.75$

TDだとV(B)の推定値をもとにしてV(A)を推定する。今回はγ=1,V(B)=3/4 なので、V(A)=3/4

Some Important Properties to Evaluate Model-free Policy Evaluation Algorithms

どっちのほうがいいのか？

- Data efficiency & Computational efficiency
- In simplest TD, use (s, a, r, s') once to update $V(s)$
  - $O(1)$ operation per update
  - In an episode of length L, $O(L)$
- In MC have to wait till episode finishes, then also $O(L)$
- MC can be more data efficient than simple TD
- But TD exploits Markov structure
  - If in Markov domain, leveraging this is helpful

このスライドの説明だとTDぼろ負けの
ように見えるが...

Alternative: Certainty Equivalence V^π MLE MDP Model Estimates

- Model-based option for policy evaluation without true models
- After each (s, a, r, s') tuple
  - Recompute maximum likelihood MDP model for (s, a)

$$\hat{P}(s'|s,a) = \frac{1}{N(s,a)} \sum_{k=1}^{K} \sum_{t=1}^{L_k-1} \mathbb{1}(s_{k,t} = s, a_{k,t} = a, s_{k,t+1} = s')$$

$$\hat{r}(s,a) = \frac{1}{N(s,a)} \sum_{k=1}^{K} \sum_{t=1}^{L_k-1} \mathbb{1}(s_{k,t} = s, a_{k,t} = a) r_{t,k}$$

  - Compute V^π using MLE MDP [3] (e.g. see method from lecture 2)

MLE：Most Likelihood Estimation

授業はここまで

[3] Requires initializing for all (s, a) pairs

Alternative: Certainty Equivalence V^π MLE MDP Model Estimates

- Model-based option for policy evaluation without true models
- After each (s, a, r, s') tuple
  - Recompute maximum likelihood MDP model for (s, a)

$$\hat{P}(s'|s,a) = \frac{1}{N(s,a)} \sum_{k=1}^{K} \sum_{t=1}^{L_k - 1} 1(s_{k,t} = s, a_{k,t} = a, s_{k,t+1} = s')$$

$$\hat{r}(s,a) = \frac{1}{N(s,a)} \sum_{k=1}^{K} \sum_{t=1}^{L_k - 1} 1(s_{k,t} = s, a_{k,t} = a) r_{t,k}$$

  - Compute V^π using MLE MDP [4] (e.g. see method from lecture 2)
- Cost: Updating MLE model and MDP planning at each update ($O(|S|^3)$ for analytic matrix solution, $O(|S|^2|A|)$ for iterative methods)
- Very data efficient and very computationally expensive
- Consistent
- Can also easily be used for off-policy evaluation

s_1	s_2	s_3	s_4	s_5	s_6	s_7
$R(s_1) = +1$ *Okay Field Site*	$R(s_2) = 0$	$R(s_3) = 0$	$R(s_4) = 0$ 	$R(s_5) = 0$	$R(s_6) = 0$	$R(s_7) = +10$ *Fantastic Field Site*

- Mars rover: R = [1 0 0 0 0 0 +10] for any action
- $\pi(s) = a_1\ \forall s$, $\gamma = 1$. any action from s_1 and s_7 terminates episode
- Trajectory = (s_3, a_1, 0, s_2, a_1, 0, s_2, a_1, 0, s_1, a_1, 1, terminal)
- First visit MC estimate of V of each state? [1 1 1 0 0 0 0]
- Every visit MC estimate of V of s_2? 1
- TD estimate of all states (init at 0) with $\alpha = 1$ is [1 0 0 0 0 0 0]
- What is the certainty equivalent estimate?
  $\hat{r} = [1\ 0\ 0\ 0\ 0\ 0\ 0]$,
  $\hat{p}(terminate|s_1, a_1) = \hat{p}(s_1|s_2, a_1) = \hat{p}(s_2|s_3, a_1) = 1$,
  V = [1 1 1 0 0 0 0]

これらの推定値は正しい？

Some Important Properties to Evaluate Policy Evaluation Algorithms

方策評価アルゴリズムで大事なこと

マルコフ性の頑健性

- Robustness to Markov assumption
- Bias/variance characteristics

bias / variance の characteristics（って何？）

- Data efficiency

データの効率性

- Computational efficiency

計算効率性

Summary: Policy Evaluation

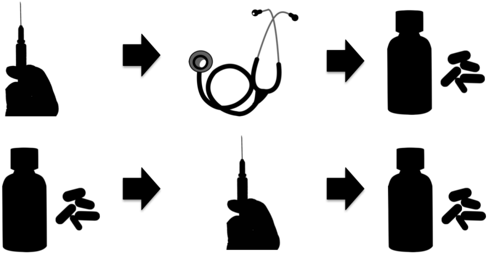
方策評価についてのまとめ

- Dynamic Programming
- Monte Carlo policy evaluation
  - Policy evaluation when we don't have a model of how the world works
    - Given on policy samples
    - Given off policy samples
- Temporal Difference (TD)
- Metrics to evaluate and compare algorithms

This Lecture: Policy Evaluation

- Estimating the expected return of a particular policy if don't have access to true MDP models
- Dynamic programming
- Monte Carlo policy evaluation
    - Policy evaluation when don't have a model of how the world work
        - Given on-policy samples
        - **Given off-policy samples** 方策なしのサンプルが与えられた場合
- Temporal Difference (TD)
- Metrics to evaluate and compare algorithms

MC Off Policy Evaluation



action を起こすのにコストがかかる場合がある

- Sometimes trying actions out is costly or high stakes
- Would like to use old data about policy decisions and their outcomes to estimate the potential value of an alternate policy

過去のデータを、方策を決定するために使えないだろうか

Monte Carlo (MC) Off Policy Evaluation

目的：得られた方策pi_2から、方策pi_1，価値関数V^pi_1 を推定する

- Aim: estimate value of policy π_1, $V^{\pi_1}(s)$, given episodes generated under behavior policy π_2
  - $s_1, a_1, r_1, s_2, a_2, r_2, \ldots$ where the actions are sampled from π_2
- $G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \cdots$ in MDP M under policy π
- $V^{\pi}(s) = \mathbb{E}_{\pi}[G_t | s_t = s]$

異なる方策からのデータがある（ここでいうbehavior policy pi_2 のみ？）

- Have data from a different policy, behavior policy π_2
- If π_2 is stochastic, can often use it to estimate the value of an alternate policy (formal conditions to follow)

pi_2が確率的である場合、代わりの方策の価値を推定するために使える（？）

- Again, no requirement that have a model nor that state is Markov

モデルもマルコフ性もいらない

Monte Carlo (MC) Off Policy Evaluation: Distribution Mismatch

方策によってepisodeもreturnも異なりますよね

- Distribution of episodes & resulting returns differs between policies

Importance Sampling

Goal：確率分布p(x)下でのf(x)の期待値を推定する

- Goal: estimate the expected value of a function $f(x)$ under some probability distribution $p(x)$, $\mathbb{E}_{x \sim p}[f(x)]$

データを分布q(s)からサンプルする

- Have data $x_1, x_2, \ldots, x_n$ sampled from distribution $q(s)$
- Under a few assumptions, we can use samples to obtain an unbiased estimate of $\mathbb{E}_{x \sim q}[f(x)]$

$$\mathbb{E}_{x \sim q}[f(x)] = \int_x q(x) f(x)$$

Importance Sampling (IS) for Policy Evaluation

- Let h_j be episode j (history) of states, actions and rewards

$$h_j = (s_{j,1}, a_{j,1}, r_{j,1}, s_{j,2}, a_{j,2}, r_{j,2}, \ldots, s_{j,L_j(terminal)})$$

Importance Sampling (IS) for Policy Evaluation

- Let h_j be episode j (history) of states, actions and rewards

$$h_j = (s_{j,1}, a_{j,1}, r_{j,1}, s_{j,2}, a_{j,2}, r_{j,2}, \ldots, s_{j,L_j(terminal)})$$

$$\begin{aligned}
p(h_j|\pi, s = s_{j,1}) =& p(a_{j,1}|s_{j,1})p(r_{j,1}|s_{j,1}, a_{j,1})p(s_{j,2}|s_{j,1}, a_{j,1}) \\
& p(a_{j,2}|s_{j,2})p(r_{j,2}|s_{j,2}, a_{j,2})p(s_{j,3}|s_{j,2}, a_{j,2}) \ldots \\
=& \prod_{t=1}^{L_j-1} p(a_{j,t}|s_{j,t})p(r_{j,t}|s_{j,t}, a_{j,t})p(s_{j,t+1}|s_{j,t}, a_{j,t}) \\
=& \prod_{t=1}^{L_j-1} \pi(a_{j,t}|s_{j,t})p(r_{j,t}|s_{j,t}, a_{j,t})p(s_{j,t+1}|s_{j,t}, a_{j,t})
\end{aligned}$$

Importance Sampling (IS) for Policy Evaluation

- Let h_j be episode j (history) of states, actions and rewards, where the actions are sampled from π_2

$$h_j = (s_{j,1}, a_{j,1}, r_{j,1}, s_{j,2}, a_{j,2}, r_{j,2}, \ldots, s_{j,L_j(terminal)})$$

$$V^{\pi_1}(s) \approx \sum_{j=1}^{n} \frac{p(h_j|\pi_1, s)}{p(h_j|\pi_2, s)} G(h_j)$$

? ? ?

Importance Sampling for Policy Evaluation

- Aim: estimate $V^{\pi_1}(s)$ given episodes generated under policy π_2
  - $s_1, a_1, r_1, s_2, a_2, r_2, \ldots$ where the actions are sampled from π_2
- Have access to $G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \cdots$ in MDP M under policy π_2
- Want $V^{\pi_1}(s) = \mathbb{E}_{\pi_1}[G_t | s_t = s]$
- IS = Monte Carlo estimate given off policy data

? ? ?

- Model-free method
- Does not require Markov assumption
- Under some assumptions, unbiased & consistent estimator of V^{π_1}
- Can be used when agent is interacting with environment to estimate value of policies different than agent's control policy
- More later this quarter about batch learning

Today's Plan

- Last Time:
    - Markov reward / decision processes
    - Policy evaluation & control when have true model (of how the world works)
- Today
    - Policy evaluation when don't have a model of how the world works
- **Next Time**:
    - **Control when don't have a model of how the world works**

次回は世界がどう動くかについてのモデルがない場合のコントロールについて