強化学習分科会第2回

今回の目標

- 強化学習で扱う問題はどのような種類の問題なのかを理解する。
- 具体的な定式化や成り立ってほしい理想的な性質、および現実の問題への適用法を知る。

各種単語の定義

- エージェント（学習器）：意思決定機関
- 環境　　：学習器の選択した行動に応じて変化する
- 状態(s_t)　：環境のなかで、学習器が取得できる情報のこと
- S^+　　：全状態＋終端状態(ex:ゲームオーバー)
- S　　：全状態
- 行動（a_t）：学習器の選択する行動
- A(s_t)　　:状態s_tにおいて選択できる行動全体の集合
- 方策(π)　：状態sにおいて行動aを選択する確率を$\pi_t(s,a)$と表す
- 報酬(r_{t+1}):行動の結果によって得られる。
  学習器は累積報酬の最大化を目指す。

エージェントと環境の境界

- 練習問題3.3

車の運転

脳→手足→アクセル・ハンドル・ブレーキ→タイヤのトルク
→車の速度・位置

どこにエージェントと環境の境界をおくか？

学習で制御したい部分をエージェント（学習器）

学習の評価に用いたい部分を環境

報酬の設定に関する注意

「どう解かせたいか」ではなく、「何をさせたいか」で定める

報酬の最大化＝目標達成

よくあるミス

チェスの報酬で相手のコマをとったら報酬＋

⇒学習器は勝利を犠牲に相手のコマをとりにいく

正しい報酬

勝ったら＋１、負けたら−１、引き分け±０

収益

収益とは累積報酬のことである。学習器は収益の最大化を目標とする。タスクに応じ、次の二つの定義がある

	エピソード的タスク	連続タスク
	オセロゲームなど単位時間に簡単に分割でき、かつ最終時間T(終端状態へたどり着くまでの時間)が有限であるタスク	物理制御のように時間が連続していたり、株の売買の学習のように終端状態がないなどの理由で最終時間T＝∞となるタスク
収益	$R_t = r_{t+1} + r_{t+2}$+……+r_T	$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$

割引率γ:0≦γ＜1

未来にもらう報酬は今もらう報酬より価値が低くなる

収益

エピソードタスク：$R_t = r_{t+1} + r_{t+2}$+……+r_T

連続タスク：$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$

エピソードタスクにおいて

t>Tのときr_t=0　　γ＝1とすれば両者は

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

でまとまる

状態のマルコフ性の要請(理想)

- マルコフ性とは、確率論分野の言葉

　次の状態への遷移確率が、直前の状態でのみに依存すること

この本においては

　次の状態への遷移確率が、直前の状態と直前の自分の行動でのみに依存すること

・非マルコフな状態の例

　将棋の棋譜　直前の相手の手(状態)と自分の手(行動)だけからでは相手が次に指してくる手(状態)の確率分布は求まらない

・マルコフな状態の例

　将棋の盤面　直前の盤面(状態)と自分の手(行動)のみから、相手が指し終わった次の盤面(状態)の確率分布が定まる。

　過去の盤面の様子(過去の状態)には依存しない。

マルコフ性の数式的表現

Pr{A|B}　　:Bの元でAが起こる条件付確率

非マルコフのとき考えるべき遷移確率

$\Pr\{s_{t+1} = s', r_{t+1} = r | s_t, a_t, r_t, \ldots\ldots, r_1, s_0, a_0\}$

　それまでのすべての報酬、行動、状態から次の報酬や状態が定まる

マルコフの時考えるべき遷移確率

$\Pr\{s_{t+1} = s', r_{t+1} = r | s_t, a_t\}$

　直前の状態と行動のみから次の報酬や状態が決まる。

実際の問題への適用

- 実際の問題はほぼ非マルコフであるが、マルコフであると近似して考えればよい。

例）将棋の盤面を状態とする。

　実は相手は10手前に自コマが取られそうになる状況に陥っていた。(危機は回避され、結局コマは取られていないので現在の盤面にはその情報は残っていない。)

　その影響により今は極度にコマ損を恐れた手を返してくるように状態の遷移確率が変化しているかもしれない。(非マルコフ)

でも大体マルコフとして良い。状態をうまく定義することが大事

マルコフ決定過程

マルコフ決定過程(MDP)：マルコフ性を満たす強化学習タスク

有限MDP　　：MDPのうち、状態と行動が有限のもの

遷移確率：

$$\boldsymbol{\mathcal{P}}^{a}_{s\,s'} = \Pr\{s_{t+1} = s' \,|s_t = s, a_t = a\}$$

状態sにおいて行動aを行ったとき、次の状態がs'となる確率

$$\boldsymbol{\mathcal{R}}^{a}_{s\,s'} = \mathrm{E}\{r_{t+1}|s_{t+1} = s', s_t = s, a_t = a\}$$

状態sにおいて行動aを行い、状態がs'になった時にもらえる報酬の期待値。(報酬の分散の情報は失われている)

有限マルコフ決定過程の例

空き缶拾いロボ

状態　S={high, low}　:ロボ内部のバッテリー

行動　A(high)={wait, search}　A(low) ={wait, search, recharge}

wait　:その場で誰かが空き缶を持ってきてくれるのを待機

search　:空き缶を探しに行く

recharge:バッテリーを充電する

有限マルコフ決定過程の例

表 3.1　リサイクルロボットの例における有限 MDP に対する遷移確率と期待報酬.

s	s'	a	$\mathcal{P}^a_{ss'}$	$\mathcal{R}^a_{ss'}$
high	high	search	α	$\mathcal{R}^{\text{search}}$
high	low	search	$1-\alpha$	$\mathcal{R}^{\text{search}}$
low	high	search	$1-\beta$	-3
low	low	search	β	$\mathcal{R}^{\text{search}}$
high	high	wait	1	$\mathcal{R}^{\text{wait}}$
high	low	wait	0	$\mathcal{R}^{\text{wait}}$
low	high	wait	0	$\mathcal{R}^{\text{wait}}$
low	low	wait	1	$\mathcal{R}^{\text{wait}}$
low	high	recharge	1	0
low	low	recharge	0	0

注：現状態 s，次の状態 s'，現状態で取ることが可能な行動 $a \in \mathcal{A}(s)$ の可能な組合せに対して 1 つの行がある.

遷移グラフ

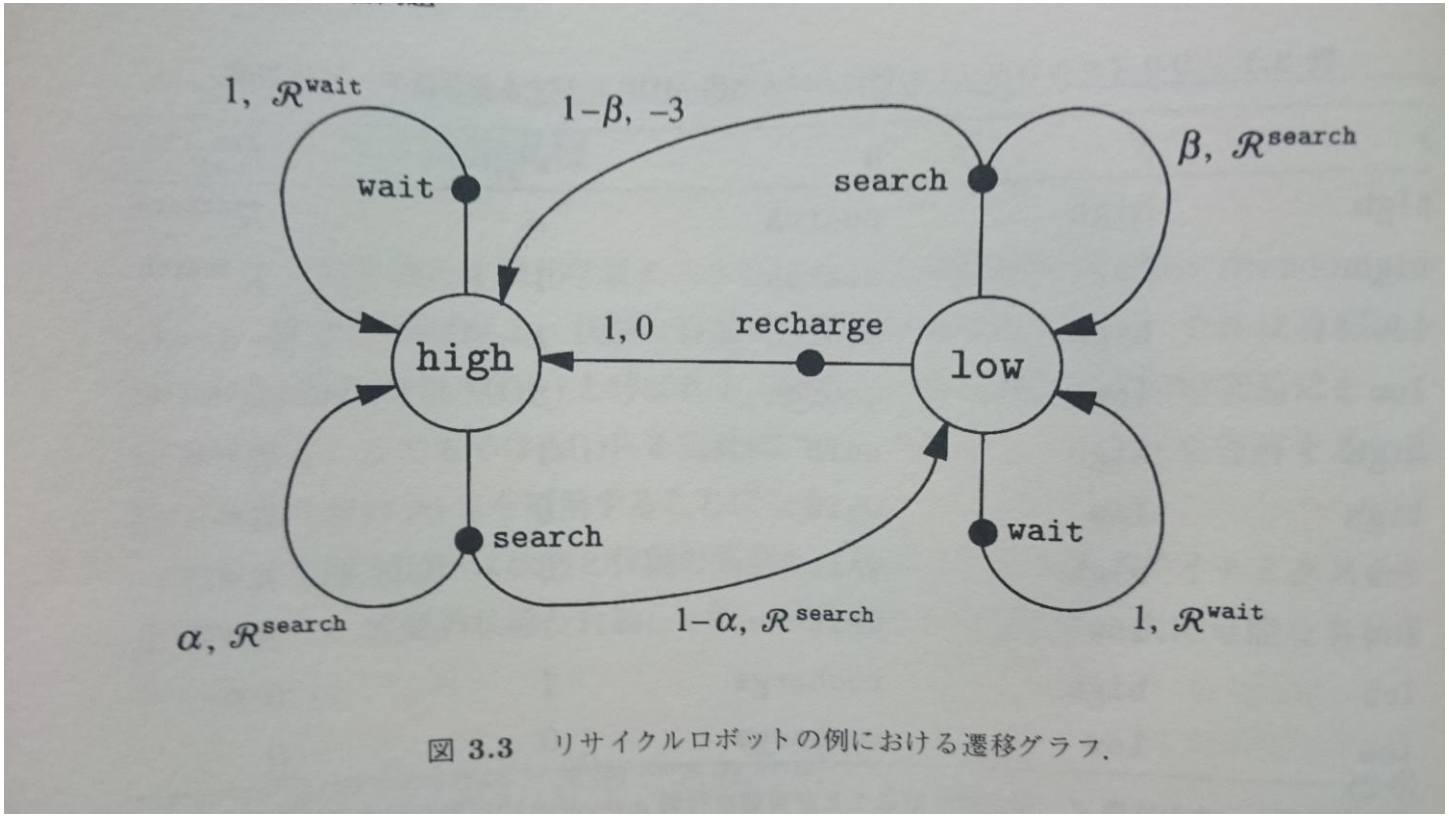

図 3.3　リサイクルロボットの例における遷移グラフ.

high　:状態ノード

search　:行動ノード

価値関数

価値関数：方策πに従って行動したときの収益の期待値

状態価値関数:

$$V^{\pi}(s) = E_{\pi}\{R_t | s_t = s\} = E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| s_t = s\right\}$$

行動価値関数：

$$Q^{\pi}(s, a) = E_{\pi}\{R_t | s_t = s, a_t = a\}$$

$V^{\pi}Q^{\pi}$は実際の試行の収益Rの平均値としてモンテカルロ法で求まる

Bellman方程式

$$V^{\pi}(s) = \sum_{a} \pi(s,a) \sum_{s\prime} \boldsymbol{\mathcal{P}}^{a}_{s\,s\prime} [\boldsymbol{\mathcal{R}}^{a}_{s\,s'} + \gamma V^{\pi}(s')]$$

が任意のsについて成り立つ。

ここからsの数だけ一次式が成り立つのでVπ(s)は一意的に求まる。

最適価値関数

全てのsについて$V^{\pi}(s) \geqq V^{\pi'}(s)$なら、π≧π‘と定める。

有限MDPではすべての方策π’についてπ≧π‘となるような最適方策が少なくとも一つ存在する。

最適方策群π*　　:最適方策となる方策πの集合

最適状態価値関数:V*(s) $= \max_{\pi} V^{\pi}(s)$

最適行動価値関数:Q*(s,a) $= \max_{\pi} Q^{\pi}(s,a)$

最適Bellman方程式

$$V^*(s) = \max_a \sum_{s'} \boldsymbol{\mathcal{P}}^a_{s\,s'}[\boldsymbol{\mathcal{R}}^a_{s\,s'} + \gamma V^*(s')]$$

これを解けばV*は求まる。

V*に対してgreedyな方策をとればそれが最適方策

以上理想論。

Bellman最適方程式が解ききれない理由

Bellman最適方程式が仮定する条件

①環境のダイナミクスを正確に把握していること

②解の計算ができる十分な計算力があること

③マルコフ性

ゲームAIだと特に②がダメ。よって近似解を求める

動的計画法、ヒューリスティック探索はBellman方程式の近似解を求める操作だと考えられる(詳しくは次章以降)