2020･09･24　ゼミ資料

Pythonで学ぶ強化学習

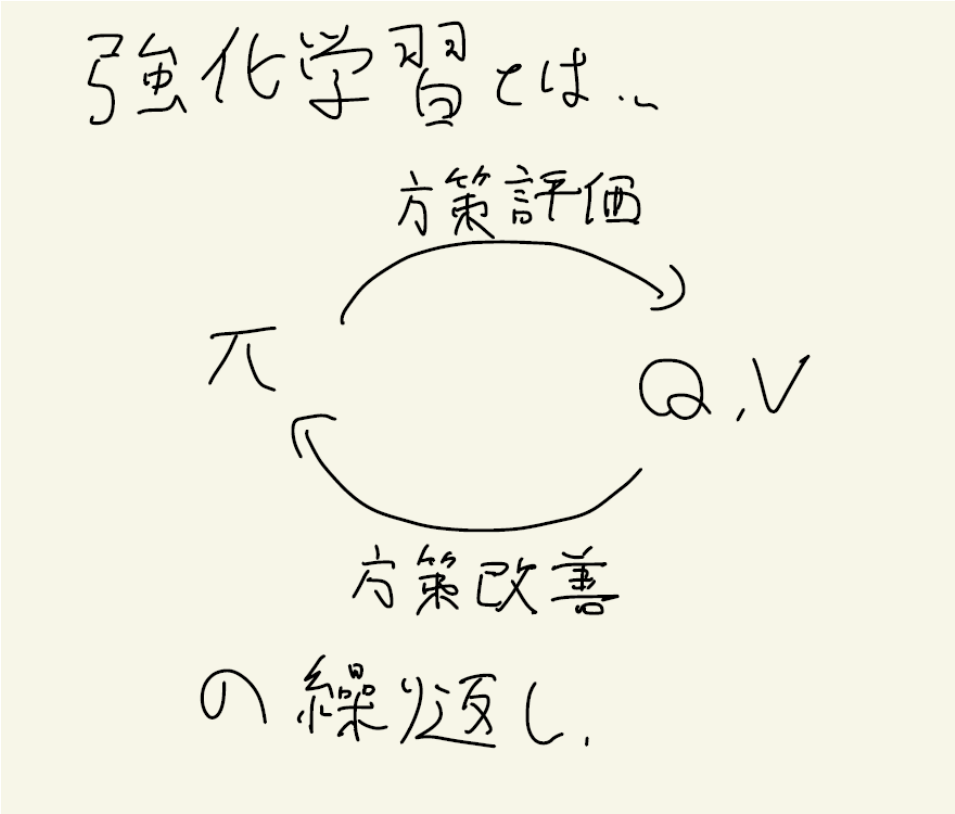
3章 強化学習の解法(2):
経験から計画を立てる(前半)

1116 17 9036

山口真哉

強化学習とは?

行動により報酬が与えられる環境を与えて,
状態で報酬につながる行動が出力されるように学習させる.



やること

- chapter 2 では環境から計画を立てた.
- 今回は経験から計画を立てる.
- 今回は遷移関数と報酬関数がわかっていないことが前提となる.

chap 3

行動した「経験」を活用するにあたって, 検討すべき点が3つある.

1. 経験の蓄積と活用のバランス
2. 計画の修正を実績から行うか予測で行うか
3. 経験を価値評価, 戦略どちらの更新に利用するか

1. 経験の蓄積と活用のバランス

- 遷移確率が不明なため, どのくらいの確率で状態から状態へ遷移するかわからない.

- 前回と同じ状態で同じ行動をしても異なる結果になる可能性がある.
  - 見積もりを正確にするには多くの経験を蓄積する必要がある.

- 一方で見積もりを活用しなければ報酬を得ることができない.

- 石橋を渡る際に叩くほど安全の確信を持てるが(経験の蓄積),
  渡らなければ向こう岸へはいけない(活用).
  - 経験の蓄積か活用かそのバランスをどう取るかが1つ目の問題点

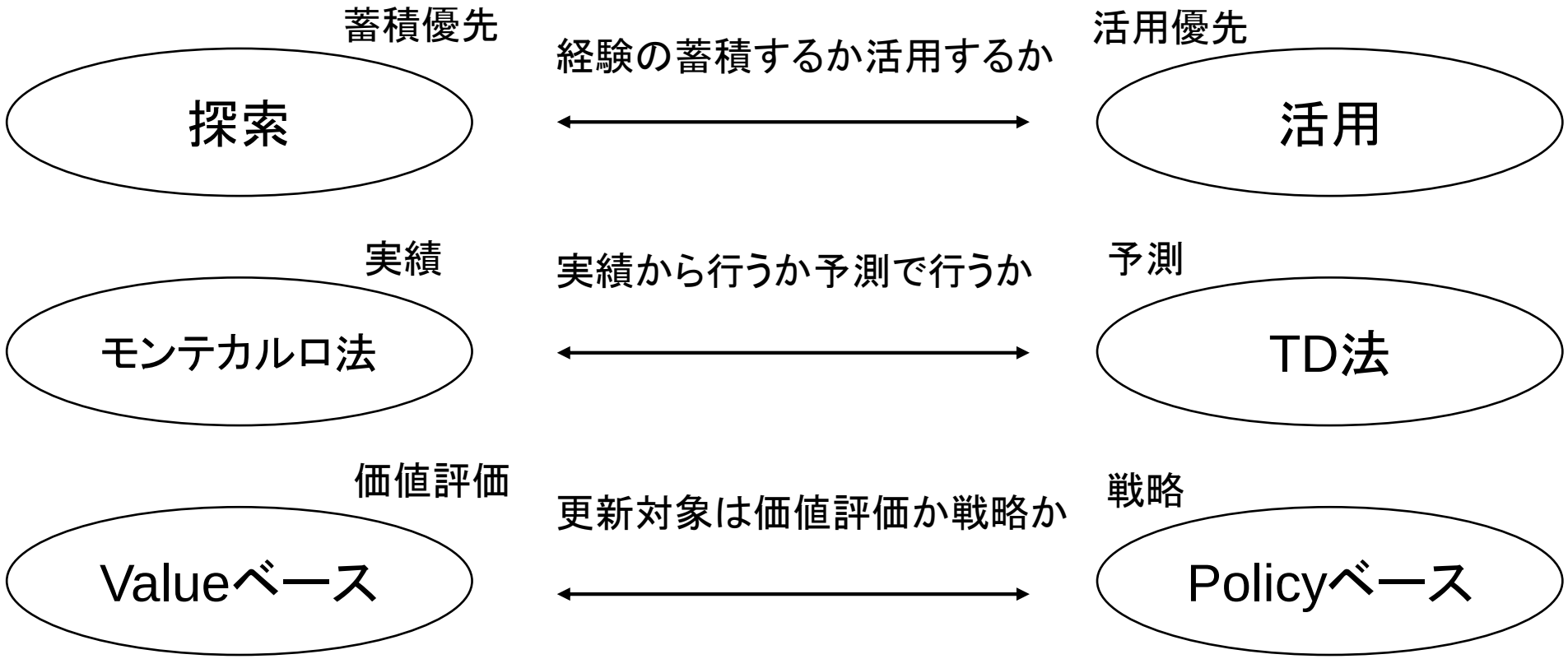
2. 計画の修正を実績から行うか予測で行うか

- 実績とは(即時)報酬の総和のことで報酬の総和が確定するのはエピソード終了時点
  - 計画の修正はエピソード終了時点になる.
- 見積もった報酬の総和, つまり予測で修正する場合は途中でも修正が可能.
- 前者は強化学習が最大化したい実際の報酬の総和に基づいた修正が可能だが
  エピソード終了まで待つ必要がある.
- 後者は素早い修正が可能だが見積もりベースの修正になる.
- 実績か予測かという観点はエピソードの終了が定まる場合のみ成立する.
  - 状態遷移が延々と続く場合には使えないので注意
    (この本ではそのようなケースを扱わない)

3. 経験を価値評価, 戦略どちらの更新に利用するか

- chapeter2で触れたValueベース, Policyベースと同じ観点
- Valueベースでは経験が価値評価の更新で
  Policyベースでは戦略の更新に使われる.
- また両方更新するという二刀流も存在する.

3つをまとめると…


蓄積優先
探索
経験の蓄積するか活用するか
活用優先
活用
実績
モンテカルロ法
実績から行うか予測で行うか
予測
TD法
価値評価
Valueベース
更新対象は価値評価か戦略か
戦略
Policyベース

1. 経験の蓄積と活用のバランス：ε – Greedy

- 環境の情報(遷移確率や報酬関数)が未知の場合，
  自ら行動することで状態の遷移，または得られる報酬を調査していくことになる．

- 調査が目的ならなるべくいろんな状態でいろいろな行動を取るといいが，
  これでは報酬の最大化はできない．
  - ロールプレイングゲームで洞窟の完全なマップを作るのと
    洞窟をいち早く脱出するのとでは目的が異なるのと同じ

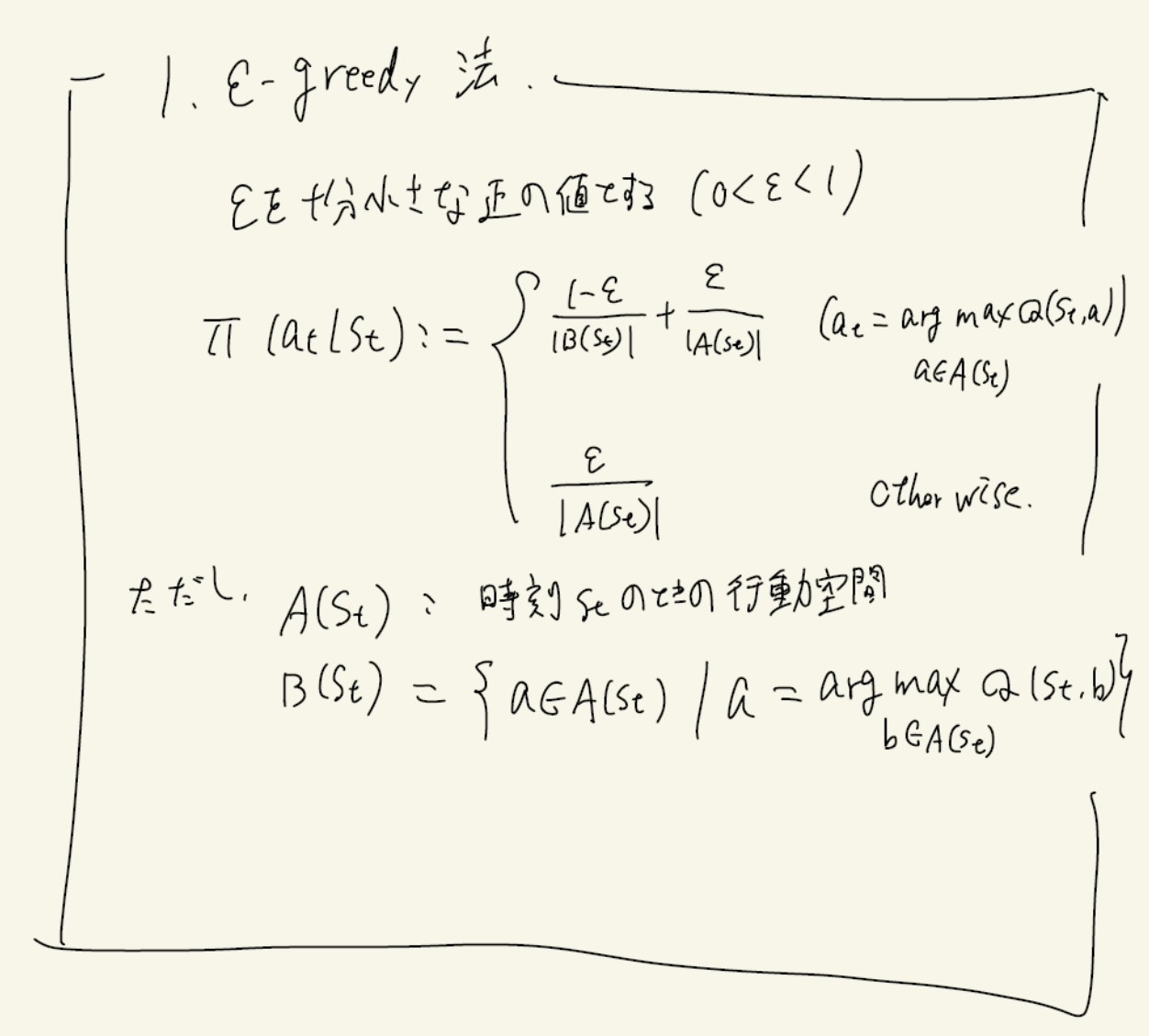
- どれぐらい調査目的の行動をして，どれくらい報酬目的の行動をすべきか
  これを「探索と活用のトレードオフ」と呼ぶ．

- 無限回試行できるなら苦労しないが多くの場合に行動回数に制約がある．
  - うまいことバランスを取りたい．

1. 経験の蓄積と活用のバランス：ε – Greedy

- バランスを取る方法としてε – Greedy法がある.
- ε の確率で調査目的の行動(探索)を行い, それ以外は活用目的の行動を行う.
  - $\varepsilon = 0.2$ なら20%で探索を行い, 80%で活用を行う.

1. 経験の蓄積と活用のバランス：ε – Greedy



エピソード数を重ねるにつれ
ε を小さくすることで学習が収束することが示せる.

ε – Greedy : 多腕バンディング問題

何枚かのコインから1枚を選び, 投げたとき表が出れば報酬が得られるゲームを考える.

なお, 各コインの表の出る確率はバラバラである.

そのため報酬を最大化するためには表が出る確率が高いコインをなるべく早く探索し,

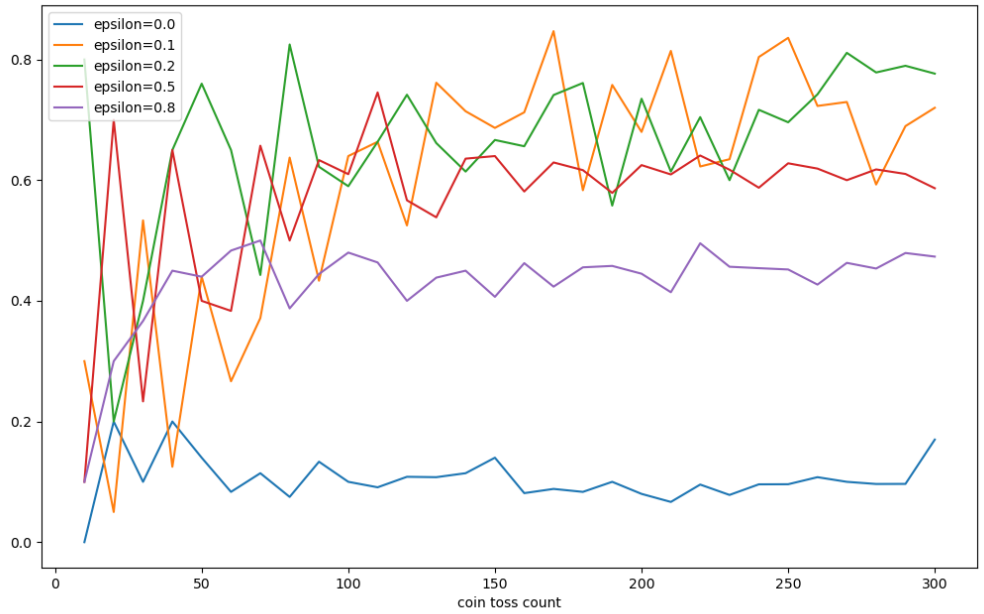
活用によりそのコインをたくさん投げることが重要である.

多腕バンディング問題と呼ばれる.

⇒ 5つのコインを使って実験してみる. 表が出る確率は[0.1, 0.5, 0.1, 0.9, 0.1] で実験する.

ε – Greedy : 多腕バンディング問題

result : X軸（回数) Y軸(平均報酬)



$\varepsilon = 0.1, 0.2$ がよい結果を示した.

注意：$\varepsilon = 0$ の時平均報酬が極端に低いのは初期値によるもので初期値を変えてあげると結果も変わってくる.

何か質問はありますか？