2020・10・01　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(中編)

1116 17 9036

山口真哉

お詫び

風邪であんまり進んでません. ><

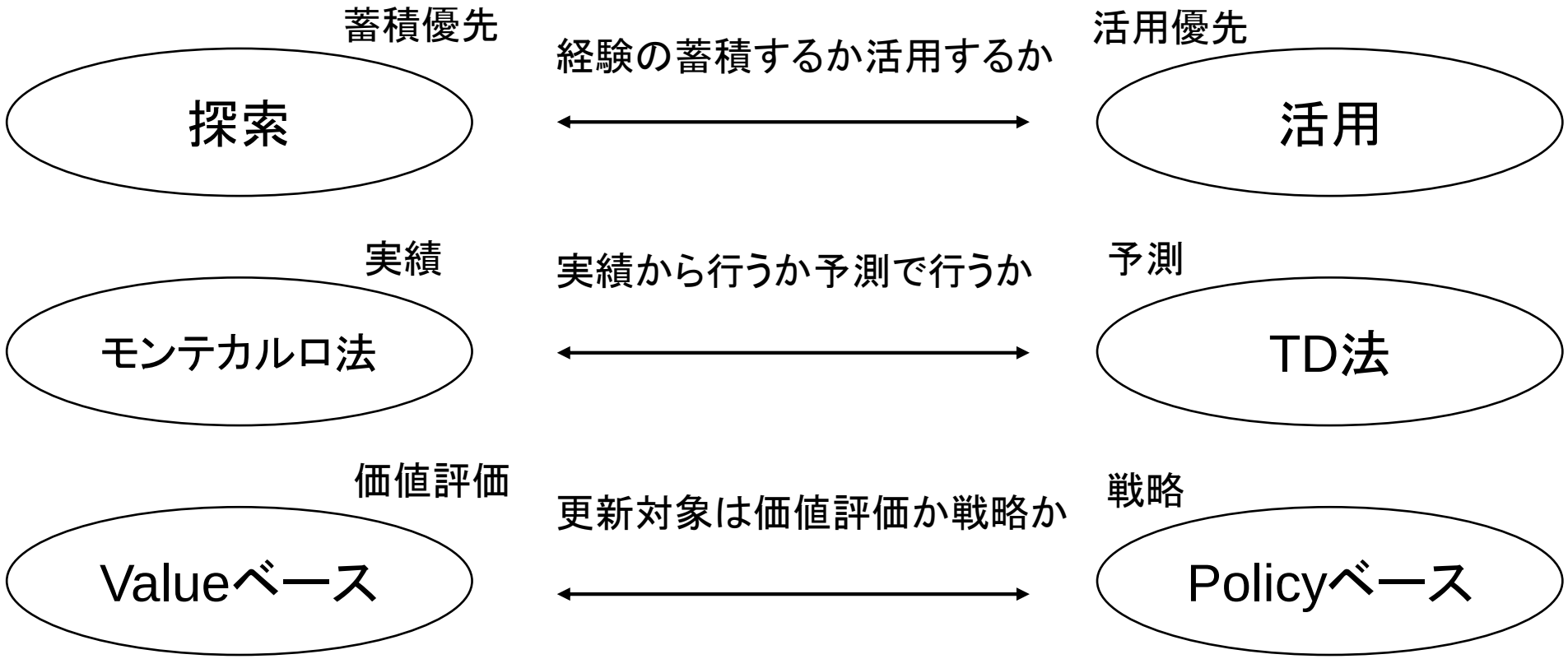
TD法とモンテカルロ法をやります.

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 (先週 ε – Greedy を使ってコイントスをした)
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

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

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)

モンテカルロ法（教科書誤植注意）

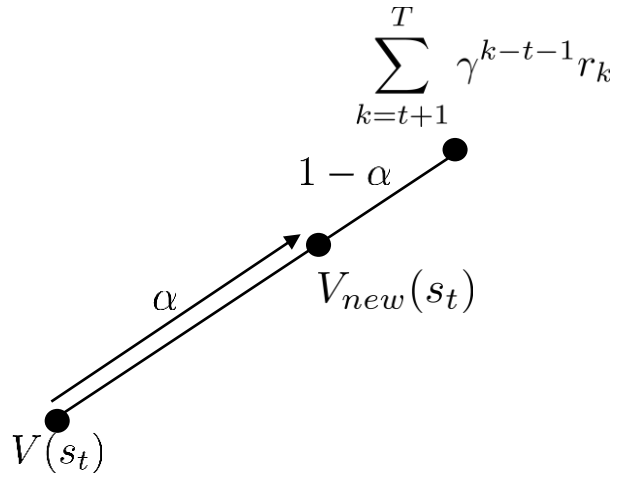
- エピソード終了した後に，獲得できた報酬の総和で行動をもとに修正を行うのはとてもシンプルな方法．
- ただこの場合エピソードが終了するまで修正はできなくなる．
- つまり最適とは言えない行動と途中で分かっていても，エピソード終了まで続けないといけないことを意味する．

更新アルゴリズム（モンテカルロ法)

$$V(s_t) \leftarrow V(s_t) + \alpha \left(\left(\sum_{k=t+1}^{T} \gamma^{k-t-1} r_k \right) - V(s_t) \right)$$

α は学習率と呼ばれる．

予測で更新しないので $\gamma = 1$ の場合がよくある．



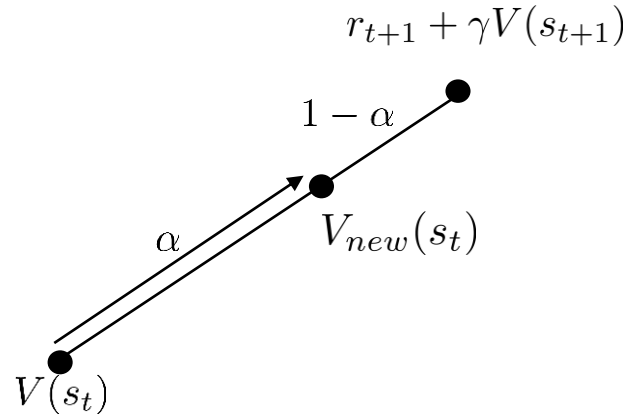
TD法

- エピソード終了を待たず, 予測で修正を行う.
- 修正前の行動を延々と続けることは避けられる. (更新が速い)
- 修正は現時点の見積もりから行われるため正確性に欠ける.
- ベルマン方程式の解に近づくようにアルゴリズムを設計する.

更新アルゴリズム（TD(0)法)

$$V(s_t) \leftarrow V(s_t) + \alpha(r_{t+1} + \gamma V(s_{t+1}) - V(s_t))$$

$V(s_{t+1})$ を終了まで展開するとモンテカルロ法と一致する.



TD法

- 他にも以下のような更新をするアルゴリズムが考えられる.

$$V(s_t) \leftarrow V(s_t) + \alpha(r_{t+1} + \gamma r_{t+2} + \gamma^2 V(s_{t+2}) - V(s_t))$$

- これは1-stepのTD法を改変したもので2期のMulti-step Learningと呼ばれる.
- 同様に3期, 4期のMulti-step Learningも考えらえる.
- k 期のMulti-step Learningに対して, $k \to \infty$ とするとモンテカルロ法と一致する.

TD(λ)法

- 固定のstepだけでなく複数のstepを組み合わせる手法もある.
- 各ステップ先の k に関しての価値 $G_t^{(k)}$ を考える.

$$
\begin{aligned}
G_t^{(1)} &= r_{t+1} + \gamma V(s_{t+1}) \\
G_t^{(2)} &= r_{t+1} + \gamma r_{t+2} + \gamma^2 V(s_{t+2}) \\
&\vdots \\
G_t^{(T-t)} &= r_{t+1} + \gamma r_{t+2} + \cdots + \gamma^{T-t} r_T
\end{aligned}
$$

TD(λ)法

- これらに関して重みがかかるように加重平均を考える.
- 具体的には以下のような重みを考える.

$$G_t^{(1)} : G_t^{(2)} : \ldots : G_t^{(T-t)} = 1 : \lambda : \ldots : \lambda^{T-t-1} \quad (0 <= \lambda <= 1)$$

- $\sum_{n=0}^{T-t-1} \lambda^n = 1$ となるように定数倍を注意してあげると,

$$G_t^{\lambda} = (1-\lambda) \sum_{n=1}^{T-t-1} \lambda^{n-1} G_t^n + \lambda^{T-t-1} G_t^{(T-t)}$$

が導出される.

TD(λ)法

$$G_t^\lambda = (1-\lambda)\sum_{n=1}^{T-t-1}\lambda^{n-1}G_t^n + \lambda^{T-t-1}G_t^{(T-t)}$$

- λを大きくしていくにつれて長い経験を重視する.
- 特に$\lambda = 1$の場合モンテカルロ法と一致する.
- 逆にλを小さくしていくにつれて短い経験を重視する.
- 特に$\lambda = 0$の場合TD(0)法と一致する.
- λを調整することによってどれだけ長いstepを重視するか調整することができる.

まとめ

- 最後まで見てあげて更新するモンテカルロ法と1期先を見て更新するTD法を見た.
- 更新方法は徐々に近づけいく単純なものだった.
- モンテカルロ法とTD法の折衷案としてTD(λ)法を見た.

何か質問はありますか？