第十一章：基于函数逼近的离轨策略方法

Chapter 11 : Off-policy Methods with Approximation

演讲：王嘉宁

2020/05/15

华东师范大学 · 数据科学与工程学院 · X101实验室

个人网站：www.wjn1996.cn
CSDN博客：https://blog.csdn.net/qq_36426650

第十一章：基于函数逼近的离轨策略方法

1、相关回顾

2、离轨策略的稳定性挑战

3、梯度TD方法

4、总结

1、相关回顾

重要度采样

函数逼近法

均方价值误差

收益类型

相关回顾

基于重要度采样的离轨策略

目标策略 π —— 用来学习并最终成为最优策略

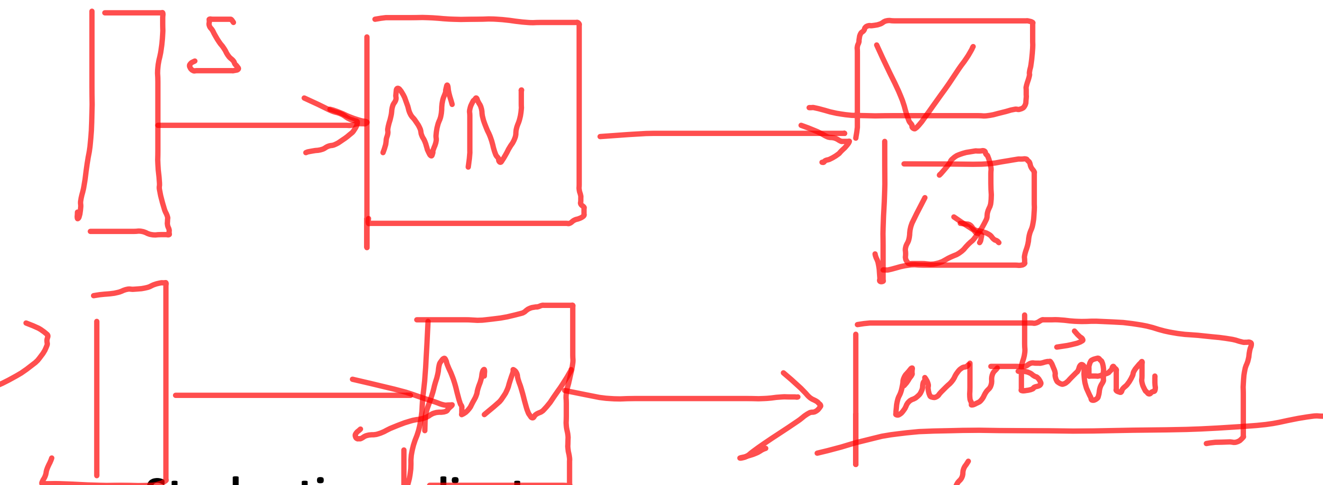
行动策略 b —— 增加试探性，产生行动样本

$$\rho_{t:T-1} = \prod_{k=t}^{T-1} \frac{\pi(A_k \mid S_k)}{b(A_k \mid S_k)}$$

$$\rho_{t:T} = \frac{\pi(A_k \mid S_k)}{b(A_k \mid S_k)}$$

相关回顾

拟合值



函数逼近法

学习一组权重向量实现价值函数的拟合

Input a state vector	$\mathbf{X}(s)=(x_1(s),x_2(s),\ldots,x_d(s))^{\mathrm{T}}$
Neual network function	$\mathbf{W}=(w_1,w_2,\ldots,w_d)^{\mathrm{T}}$
Output a value	$v_\pi(s,\mathbf{W})$

linear

$v = wx$

Stochastic-gradient

$$\mathbf{w}_{t+1} \doteq \mathbf{w}_t - \frac{1}{2}\alpha\nabla\left[v_\pi(S_t)-\hat{v}(S_t,\mathbf{w}_t)\right]^2$$
$$= \mathbf{w}_t + \alpha\left[v_\pi(S_t)-\hat{v}(S_t,\mathbf{w}_t)\right]\nabla\hat{v}(S_t,\mathbf{w}_t),$$

removal

Semi-gradient

$$\mathbf{w}_{t+1}=\mathbf{w}_t+\alpha[R+\gamma\hat{v}(S',\mathbf{w})-\hat{v}(S,\mathbf{w})]\nabla\hat{v}(S,\mathbf{w})$$

自举

相关回顾

学习目标

确定学习的目标——**均方价值误差**

real pre

$$\overline{\mathrm{VE}}(\mathbf{w}) \doteq \sum_{s\in\mathcal{S}} \mu(s)\Big[v_\pi(s) - \hat{v}(s,\mathbf{w})\Big]^2.$$

$$\eta(s) = h(s) + \sum_{\bar{s}} \eta(\bar{s}) \sum_a \pi(a|\bar{s})p(s|\bar{s},a), \quad \text{for all } s \in \mathcal{S}.$$

$$\mu(s) = \frac{\eta(s)}{\sum_{s'} \eta(s')}, \quad \text{for all } s \in \mathcal{S}.$$

相关回顾

收益类型

强化学习的三个设定——**分幕式设定，折扣设定，平均收益设定**

1、折扣收益

$$
\begin{aligned}
G_t &\doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 R_{t+4} + \cdots \\
&= R_{t+1} + \gamma\left(R_{t+2} + \gamma R_{t+3} + \gamma^2 R_{t+4} + \cdots\right) \\
&= R_{t+1} + \gamma G_{t+1}
\end{aligned}
$$

2、差分收益

$$G_t \doteq \underline{R_{t+1} - r(\pi)} + \underline{R_{t+2} - r(\pi)} + \underline{R_{t+3} - r(\pi)} + \cdots .$$

2、离轨策略的稳定性挑战

两种挑战

致命三要素

线性逼近下的探究

贝尔曼误差的不可学习性

离轨策略的稳定性挑战

离轨策略的两种挑战

更新目标的变换 → 使用重要度采样技术应对更新目标 → 收敛性

更新的分布 → 再次使用重要度采样扭曲更新的分布；寻找不依赖于任何特殊分布 → 收敛性

离轨策略的稳定性挑战

半梯度方法

半梯度离轨策略TD(0)

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \alpha \rho_t \delta_t \nabla \hat{v}(S_t, \mathbf{w}_t)$$

$$\delta_t = R_{t+1} + \gamma \hat{v}(S_{t+1}, \mathbf{w}_t) - \hat{v}(S_t, \mathbf{w}_t)$$

$$\delta_t = R_{t+1} - \overline{R}_t + \hat{v}(S_{t+1}, \mathbf{w}_t) - \hat{v}(S_t, \mathbf{w}_t)$$

n步SARSA

$$\mathbf{w}_{t+n} = \mathbf{w}_{t+n-1} + \alpha \rho_{t+1} \dots \rho_{t+n-1} [G_{t:t+n} - \hat{q}(S_t, A_t, \mathbf{w}_{t+n-1})] \nabla \hat{q}(S_t, A_t, \mathbf{w}_{t+n-1})$$

$$G_{t:t+n} = R_{t+1} + \dots + \gamma^{n-1} R_{t+n} + \gamma^n \hat{q}(S_{t+n}, A_{t+n}, \mathbf{w}_{t+n-1})$$

$$G_{t:t+n} = R_{t+1} - \overline{R}_t + \dots + R_{t+n} - \overline{R}_{t+n-1} + \hat{q}(S_{t+n}, A_{t+n}, \mathbf{w}_{t+n-1})$$

t

n

t+n

离轨策略的稳定性挑战

方

半梯度方法

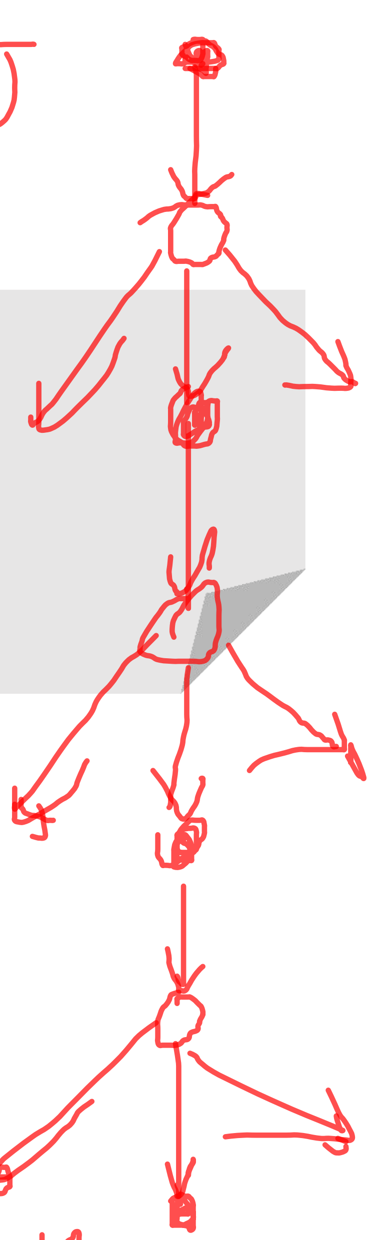
n步树回溯

$$\mathbf{w}_{t+n} = \mathbf{w}_{t+n-1} + \alpha[G_{t:t+n} - \hat{q}(S_t, A_t, \mathbf{w}_{t+n-1})]\nabla\hat{q}(S_t, A_t, \mathbf{w}_{t+n-1})$$

$$G_{t:t+n} = R_{t+1} + \gamma \sum_{a \neq A_{t+1}} \pi(a \mid S_{t+1}) Q_{t+n-1}(S_{t+1}, a) + \gamma\pi(A_{t+1} \mid S_{t+1}) G_{t+1:t+n}$$

Expert

【思考】在基于函数逼近的方法之下，是否能够保证稳定？是否会发散？



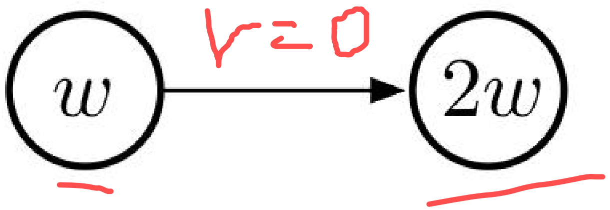
t+n

离轨策略的稳定性挑战

离轨策略发散的案例1

$v = wx$

$w = \{w\}, \quad x = \{x\}$



TD误差 $\quad \delta_t = R_{t+1} + \gamma \hat{v}(S_{t+1}, \mathbf{w}_t) - \hat{v}(S_t, \mathbf{w}_t) = 0 + \gamma 2w_t - w_t = (2\gamma - 1)w_t,$

梯度更新 $\quad w_{t+1} = w_t + \alpha \rho_t \delta_t \nabla \hat{v}(S_t, w_t) = w_t + \alpha \cdot 1 \cdot (2\gamma - 1)w_t \cdot 1 = (1 + \alpha(2\gamma - 1))w_t.$

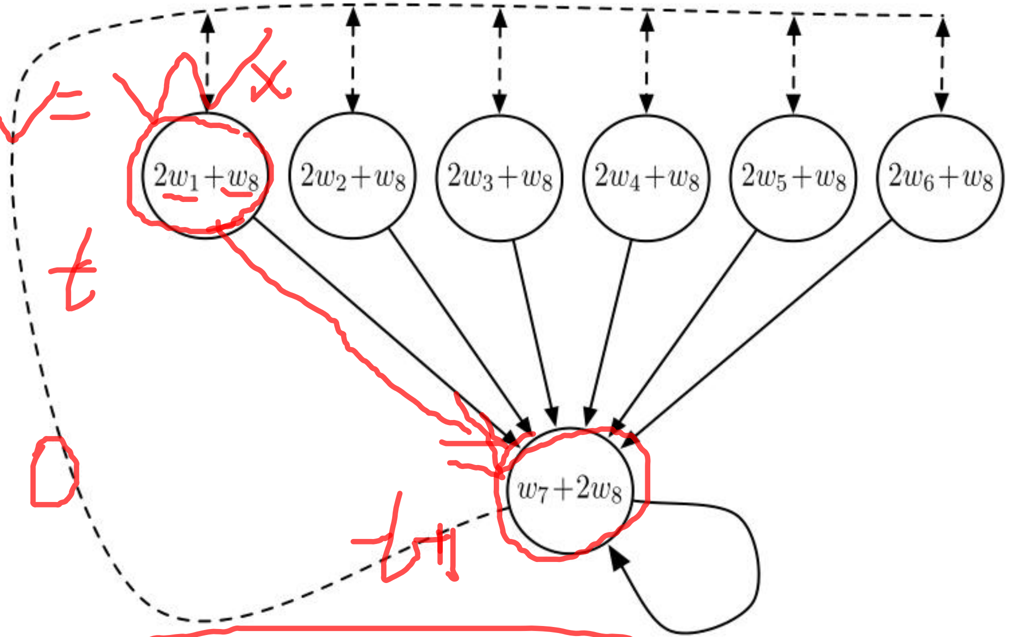
一个转移的重复发生，但没有在其他转移上更新，使得离轨策略容易发散

离轨策略的稳定性挑战

$W = \{w_1, w_2, \ldots, w_8\}$ $X = \{\cdots\}$

离轨策略发散的案例2（贝尔德反例）

$v = W^T X$

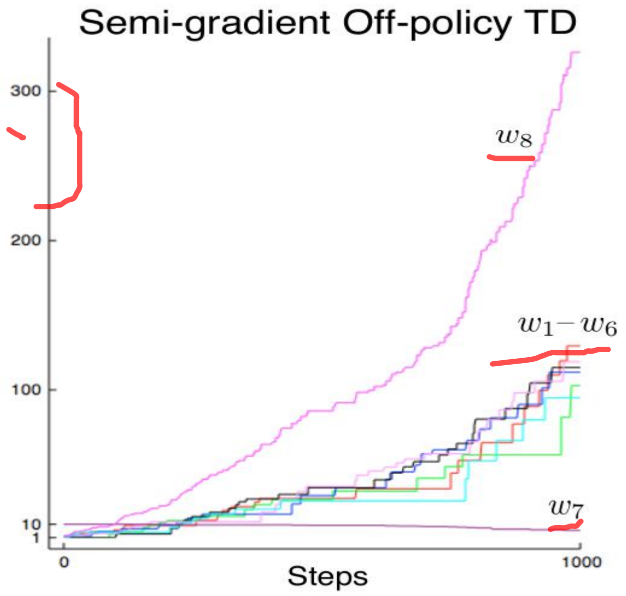


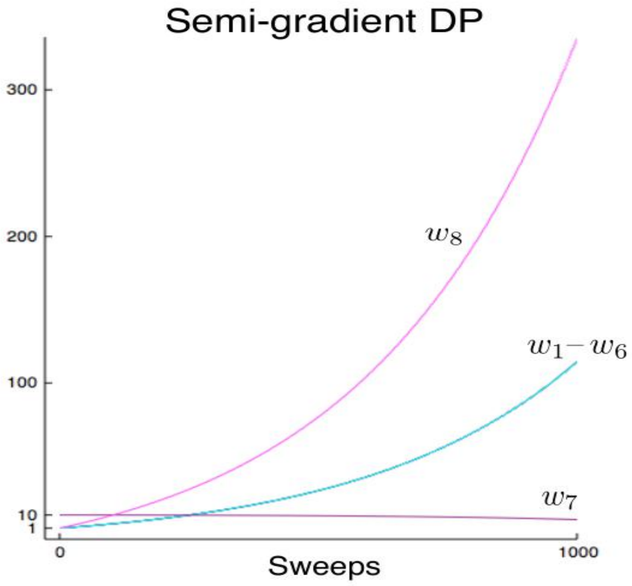
$\pi(\text{solid}|\cdot) = 1$

$\mu(\text{dashed}|\cdot) = 6/7$

$\mu(\text{solid}|\cdot) = 1/7$

$\gamma = 0.99$





注：行动策略以6/7和1/7概率选择虚线或实线；目标策略只选择实线

行动策略会选择目标策略永不执行的动作，导致梯度被容忍

离轨策略的稳定性挑战

致命三要素

函数逼近：使用权重拟合所有状态（状态-动作）和价值的函数，降低内存，提高泛化能力 半梯度

自举法：使用当前的目标估计值进行更新以得到目标估计值（动态规划，时序差分，n步自举等）

离轨策略：根据行动策略形成具有试探性的样本分布来学习目标策略的分布 Q SARSA.

【思考】为什么三者结合起来发散的风险最大？如果要舍弃一个你会选择谁？

离轨策略的稳定性挑战

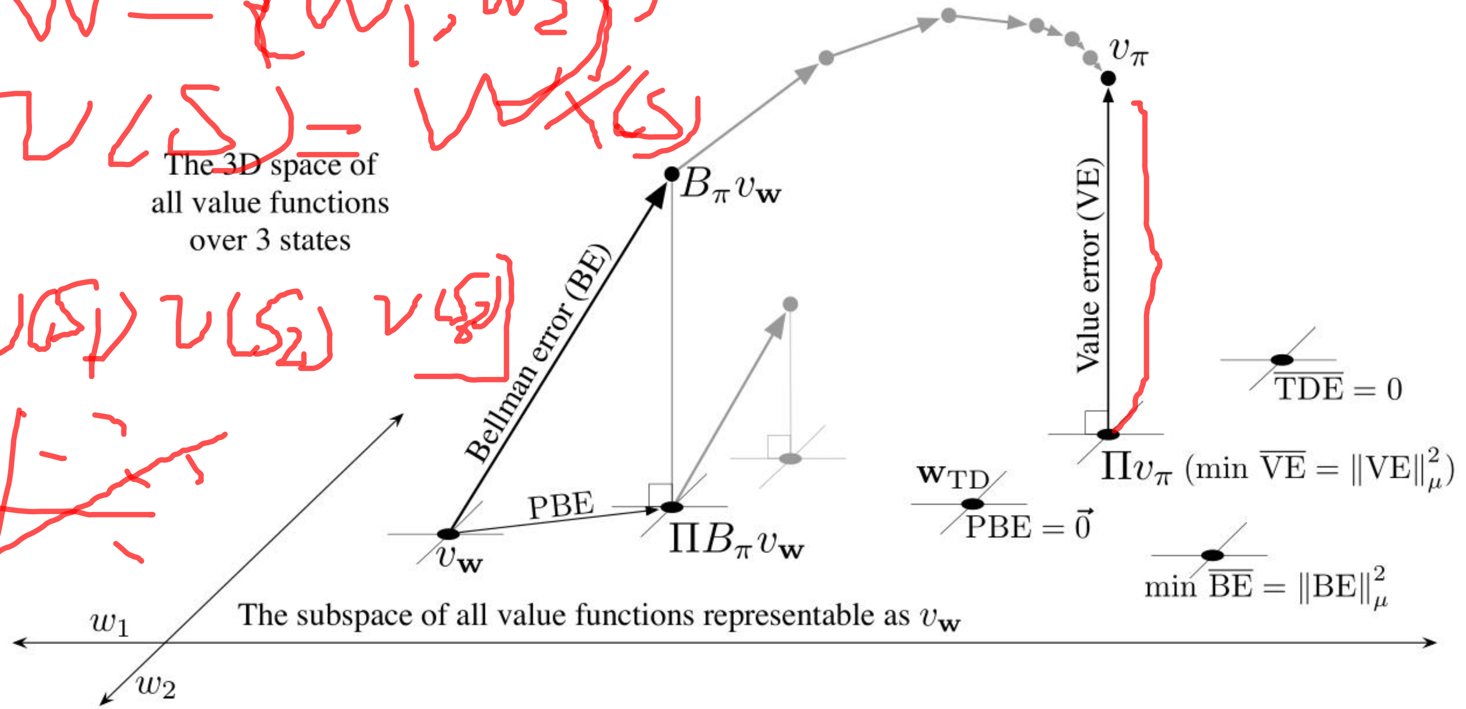
$S=\{s_1, s_2, s_3\}$ $X(s)=(x_1(s), x_2(s))^T$

线性逼近下的探究

$W=(w_1, w_2)^T$

$v(s)=W X(s)$

$[v(s_1)\ v(s_2)\ v(s_3)]$



(1) 假设有3个状态 $S=\{s_1, s_2, s_3\}$

(2) 权重向量为二维 $\mathbf{W}=(w_1, w_2)^T$

(3) 真实价值空间则为三维空间，近似价值空间则为一个平面

- 价值函数之间的距离，通过**投影**来寻找近似解

$$\text{VE} = v_\pi - v_{\mathbf{w}}$$

$$\overline{\text{VE}}(\mathbf{w}) = \sum_{s\in S} \mu(s)[v_\pi(s) - \hat{v}(s,\mathbf{w})]^2$$

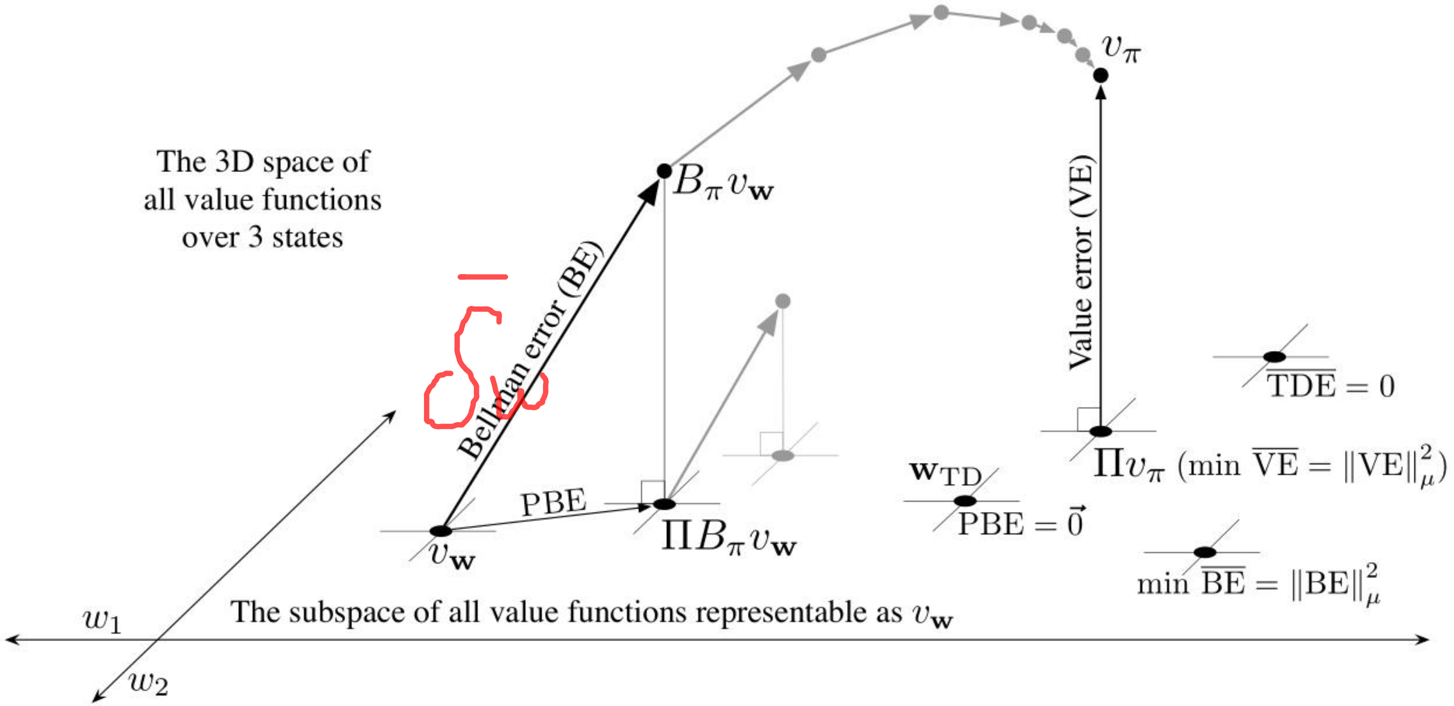
$$\Pi v = v_{\mathbf{w}}, \mathbf{w} = \arg\min_{w\in R^d} \|\overline{\text{VE}}\|_\mu^2$$

投影算子

$$\Pi = \mathbf{X}(\mathbf{X}^\top\mathbf{D}\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{D}$$

离轨策略的稳定性挑战

线性逼近下的探究



(1) 假设有3个状态 $S = \{s_1, s_2, s_3\}$

(2) 权重向量为二维 $\mathbf{W} = (w_1, w_2)^T$

(3) 真实价值空间则为三维空间，近似价值空间则为一个平面

✲ 也可以用自举的方法衡量差异，例如**贝尔曼误差**

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a)\left[r + \gamma v_\pi(s')\right]$$

$$\bar{\delta}_{\mathbf{w}}(s) \doteq \left(\sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a)\left[r + \gamma v_{\mathbf{w}}(s')\right]\right) - v_{\mathbf{w}}(s)$$

$$= \mathbb{E}\left[R_{t+1} + \gamma v_{\mathbf{w}}(S_{t+1}) - v_{\mathbf{w}}(S_t) \mid S_t = s, A_t \sim \pi\right]$$

贝尔曼算子

$$(B_\pi v)(s) \doteq \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a)\left[r + \gamma v(s')\right]$$

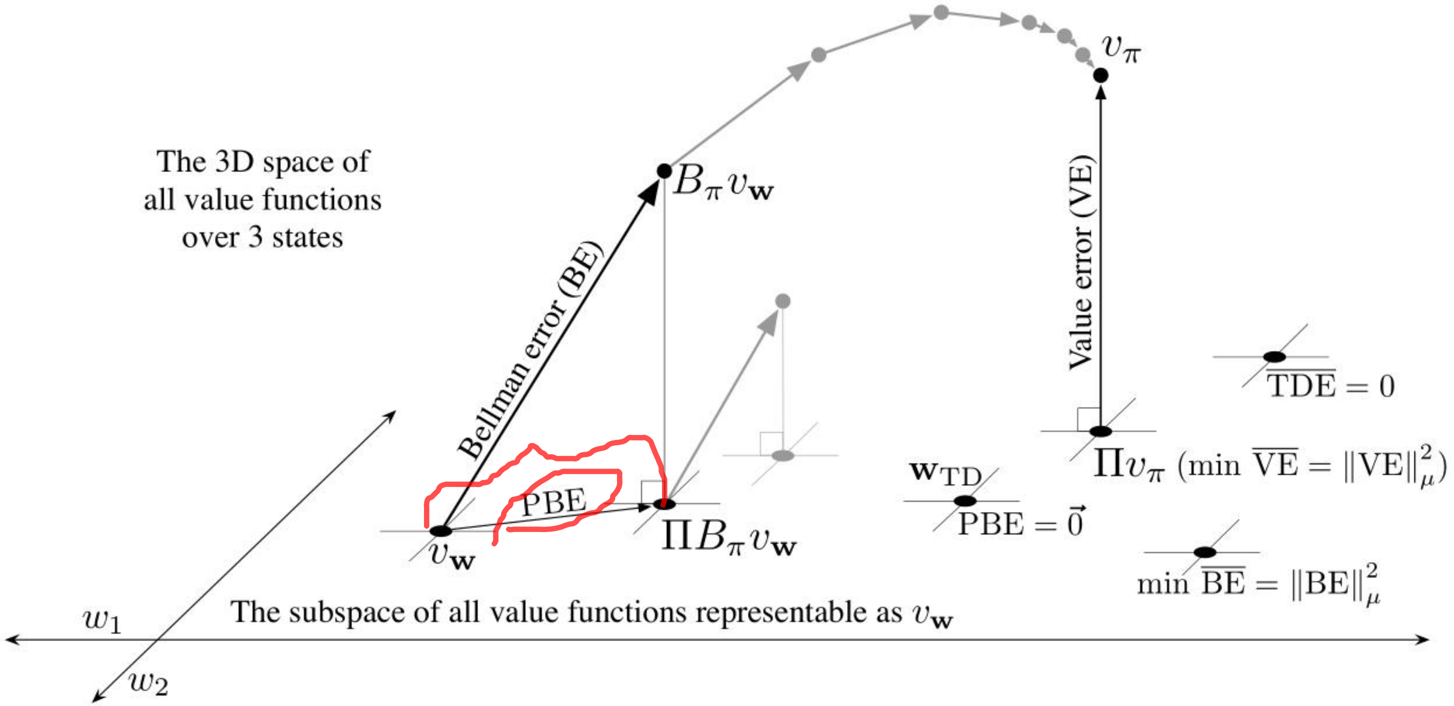
贝尔曼误差向量 $\bar{\delta}_{\mathbf{w}} = B_\pi v_{\mathbf{w}} - v_{\mathbf{w}}$

均方贝尔曼误差 $\overline{\text{BE}}(\mathbf{w}) = \|\bar{\delta}_{\mathbf{w}}\|_\mu^2$

当收敛时形成不动点，即： $v_\pi = B_\pi v_\pi$

离轨策略的稳定性挑战

线性逼近下的探究



(1) 假设有3个状态 $S = \{s_1, s_2, s_3\}$

(2) 权重向量为二维 $\mathbf{W} = (w_1, w_2)^T$

(3) 真实价值空间则为三维空间，近似价值空间则为一个平面

- 也可以结合贝尔曼误差与投影算子，形成**投影贝尔曼误差**

$$(B_\pi v)(s) \doteq \sum_a \pi(a|s) \sum_{s',r} p(s',r\,|\,s,a)\,[r + \gamma v(s')]$$

投影贝尔曼误差向量 $\mathrm{PBE} = \Pi \bar{\delta}_{\mathbf{w}}$

均方投影贝尔曼误差 $\overline{\mathrm{PBE}}(\mathbf{w}) = \left\|\Pi \bar{\delta}_{\mathbf{w}}\right\|_\mu^2$

选用不同的误差测度，会影响最终值函数的近似特性，从而影响策略

which?

离轨策略的稳定性挑战

朴素残差梯度算法——最小化TD误差

$$\delta_t = R_{t+1} + \gamma \hat{v}(S_{t+1},\mathbf{w}_t) - \hat{v}(S_t,\mathbf{w}_t)$$

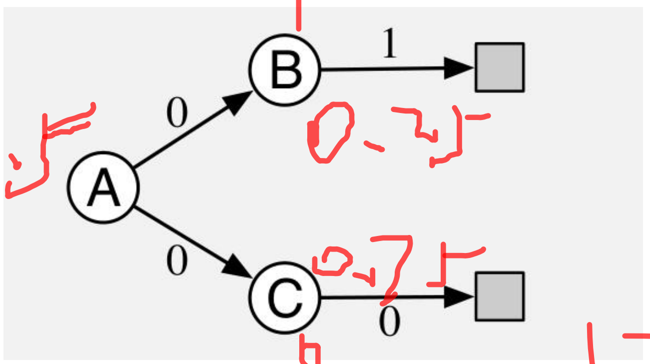
$$\begin{aligned}
\overline{\mathrm{TDE}}(\mathbf{w}) &= \sum_{s\in\mathcal{S}} \mu(s)\mathbb{E}\left[\delta_t^2 \mid S_t = s, A_t \sim \pi\right] \\
&= \sum_{s\in\mathcal{S}} \mu(s)\mathbb{E}\left[\rho_t\delta_t^2 \mid S_t = s, A_t \sim b\right] \\
&= \mathbb{E}_b\left[\rho_t\delta_t^2\right].
\end{aligned}$$

$$\begin{aligned}
\mathbf{w}_{t+1} &= \mathbf{w}_t - \frac{1}{2}\alpha\nabla(\rho_t\delta_t^2) \\
&= \mathbf{w}_t - \alpha\rho_t\delta_t\nabla\delta_t \\
&= \boxed{\mathbf{w}_t + \alpha\rho_t\delta_t\big(\nabla\hat{v}(S_t,\mathbf{w}_t)} - \underline{\gamma\nabla\hat{v}(S_{t+1},\mathbf{w}_t)\big)}
\end{aligned}$$

半梯度

【思考】朴素残差算法是以真实值与预测值之差为驱动学习，使用随机梯度法寻找最优值。如果将TD误差作为函数逼近的学习目标，会怎么样？

使用TD误差可能陷入局部最优，而不是全局最优！



TDE = 1/16

真实值：v(A)=0.5，v(B)=1，v(C)=0

收敛值：v(A)=0.5，v(B)=0.75，v(C)=0.25

结论：真实的价值并不会对应于最小的TDE

离轨策略的稳定性挑战

残差梯度算法——最小化贝尔曼误差

$$\delta_t = R_{t+1} + \gamma \hat{v}(S_{t+1},\mathbf{w}_t) - \hat{v}(S_t,\mathbf{w}_t)$$

$$\begin{aligned}
\mathbf{w}_{t+1} &= \mathbf{w}_t - \frac{1}{2}\alpha\nabla(\mathbb{E}_{\pi}[\delta_t]^2) \\
&= \mathbf{w}_t - \frac{1}{2}\alpha\nabla(\mathbb{E}_{b}[\rho_t\delta_t]^2) \\
&= \mathbf{w}_t - \alpha\mathbb{E}_b[\rho_t\delta_t]\,\nabla\mathbb{E}_b[\rho_t\delta_t] \\
&= \mathbf{w}_t - \alpha\mathbb{E}_b\big[\rho_t(R_{t+1} + \gamma\hat{v}(S_{t+1},\mathbf{w}) - \hat{v}(S_t,\mathbf{w}))\big]\,\mathbb{E}_b[\rho_t\nabla\delta_t] \\
&= \mathbf{w}_t + \alpha\Big[\mathbb{E}_b\big[\rho_t(R_{t+1} + \gamma\hat{v}(S_{t+1},\mathbf{w}))\big] - \hat{v}(S_t,\mathbf{w})\Big]\Big[\nabla\hat{v}(S_t,\mathbf{w}) - \gamma\mathbb{E}_b\big[\rho_t\nabla\hat{v}(S_{t+1},\mathbf{w})\big]\Big]
\end{aligned}$$

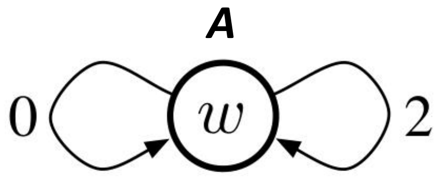
【思考】残差梯度算法的收敛性是否完美?

（1）速度慢;
（2）可能收敛到错误的值
（3）贝尔曼误差是不可学习的 Why?

注：贝尔曼误差是对应TD误差的期望

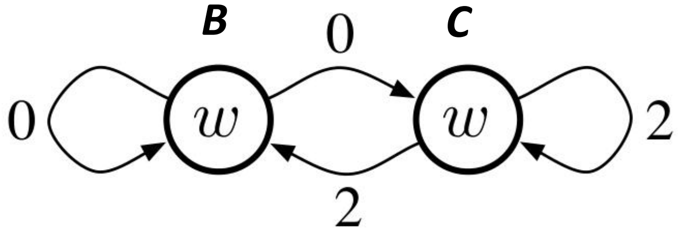
离轨策略的稳定性挑战

价值误差的不可学习性



MRP：0 2 2 0 0 2 0 2 0 2

真实值v(A)=1
收敛值v(A)=0
VE=0



MRP：0 0 2 0 2 0 2 2 2 0

真实值v(B)=0，v(C)=2
收敛值v(B)=1，v(C)=1
VE=1

【思考】什么是不可学习性?

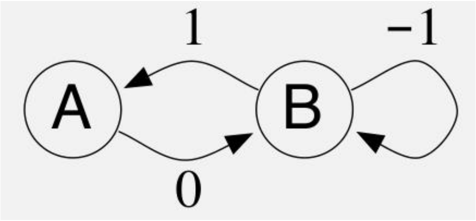
价值误差目标不能从可观测的数据（MRP流）中学习。不同的马尔可夫收益过程，贝尔曼误差不同，但却服从相同的分布，所以说**价值误差并不是数据分布的唯一确定函数。**

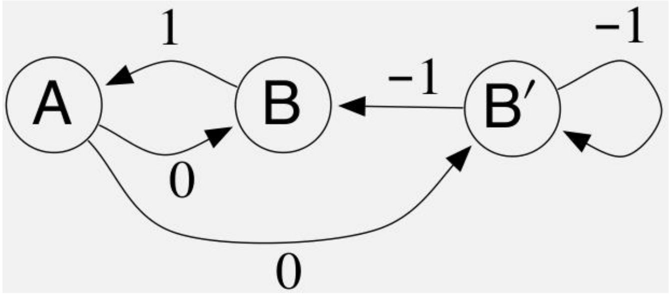
辛运的是，它们都有相同的最优参数W*，因此可以使用下面的**均方回报误差**来表示

$$
\begin{aligned}
\overline{\mathrm{RE}}(\mathbf{w}) &= \mathbb{E}\left[\left(G_t - \hat{v}\left(S_t, \mathbf{w}\right)\right)^2\right] \\
&= \mathbb{E}\left[\left(G_t - v_\pi(S_t) + v_\pi(S_t) - \hat{v}\left(S_t, \mathbf{w}\right)\right)^2\right] \\
&= \overline{\mathrm{VE}}(\mathbf{w}) + \mathbb{E}\left[\left(G_t - v_\pi\left(S_t\right)\right)^2\right]
\end{aligned}
$$

离轨策略的稳定性挑战

贝尔曼误差的不可学习性



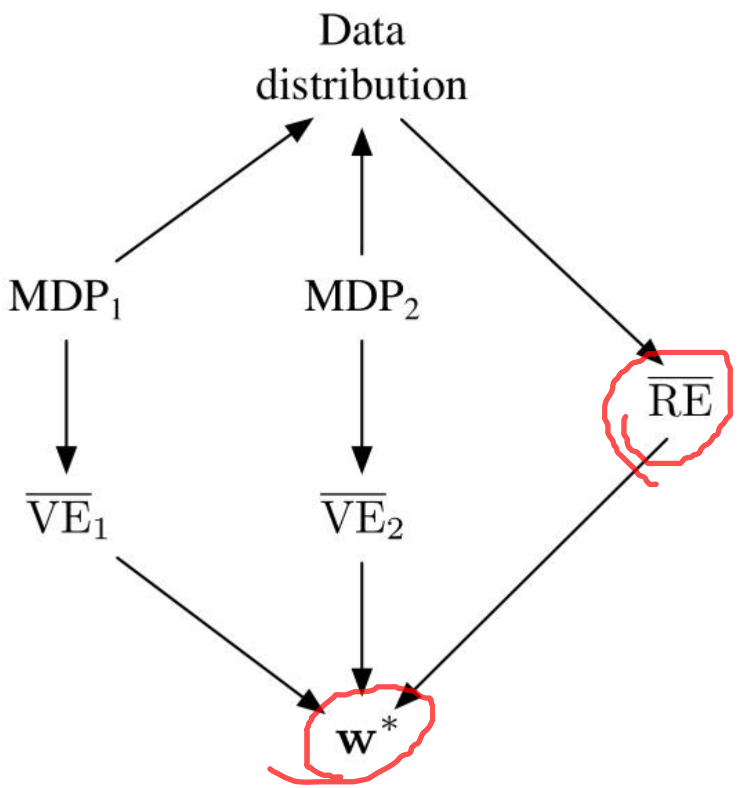


【结论】两个不同的MRP产生相同的数据分布，但有两个不同的均方贝尔曼误差0和2/3。另外最小化的权重向量对于两个MRP也是不同的，因此贝尔曼误差不可仅从数据中学习

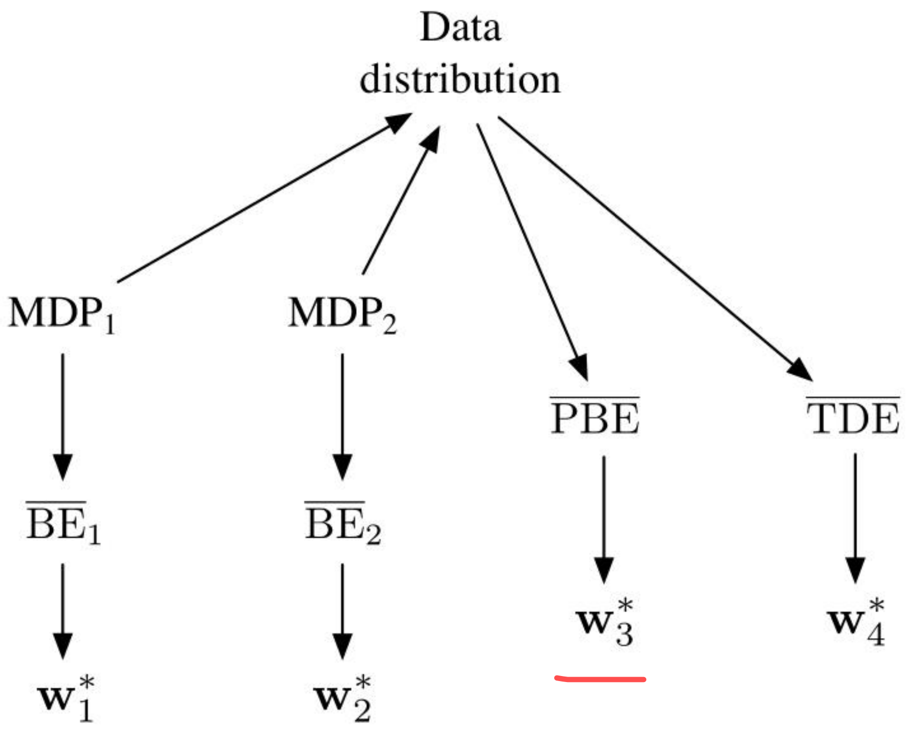
额外知识

离轨策略的稳定性挑战

不可学习性总结



不同的MDP产生相同的数据分布，但有不同的VE误差，所以不可学习。但具有相同的最优参数，所以可利用RE目标学习



不同的MDP产生相同的数据分布，但有不同的BE误差，且具有不相同的最优参数，所以BE不可学习。但PBE和TDE可以。

3、梯度TD方法

- 梯度TD方法
- GTD2与TDC算法
- 兴趣与强调

梯度TD方法

梯度TD方法

$$\Pi = X(X^\top D X)^{-1} X^\top D$$

$$D = \begin{pmatrix} \mu(s_1) & & \\ & \mu(s_2) & \\ & & \ddots \end{pmatrix}$$

$$\begin{aligned}
\overline{\mathrm{PBE}}(\mathbf{w}) &= \left\|\Pi\bar{\delta}_{\mathbf{w}}\right\|_{\mu}^{2} \\
&= (\Pi\bar{\delta}_{\mathbf{w}})^\top \mathbf{D}\Pi\bar{\delta}_{\mathbf{w}} \\
&= \bar{\delta}_{\mathbf{w}}^\top \Pi^\top \mathbf{D}\Pi\bar{\delta}_{\mathbf{w}} \\
&= \bar{\delta}_{\mathbf{w}}^\top \mathbf{D}\mathbf{X}\left(\mathbf{X}^\top\mathbf{D}\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{D}\bar{\delta}_{\mathbf{w}}
\end{aligned}$$

(using (11.13) and the identity $\Pi^\top \mathbf{D}\Pi = \mathbf{D}\mathbf{X}\left(\mathbf{X}^\top\mathbf{D}\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{D}$)

$$= \left(\mathbf{X}^\top\mathbf{D}\bar{\delta}_{\mathbf{w}}\right)^\top \boxed{\left(\mathbf{X}^\top\mathbf{D}\mathbf{X}\right)^{-1}\left(\mathbf{X}^\top\mathbf{D}\bar{\delta}_{\mathbf{w}}\right).}$$

$$\nabla\overline{\mathrm{PBE}}(\mathbf{w}) = 2\nabla\left[\mathbf{X}^\top\mathbf{D}\bar{\delta}_{\mathbf{w}}\right]^\top \left(\mathbf{X}^\top\mathbf{D}\mathbf{X}\right)^{-1}\left(\mathbf{X}^\top\mathbf{D}\bar{\delta}_{\mathbf{w}}\right)$$

$$\nabla(X^\top D \bar{\delta}_w)^\top \cdots + (X^\top D \bar{\delta}_w)^\top (X^\top D X)^{-1} \nabla(X^\top D \bar{\delta}_w)$$

梯度TD方法

梯度TD方法

$$\nabla \overline{\mathrm{PBE}}(\mathbf{w}) = 2\nabla \boxed{\left[\mathbf{X}^\top \mathbf{D}\bar{\delta}_{\mathbf{w}}\right]^\top} \boxed{\left(\mathbf{X}^\top \mathbf{D}\mathbf{X}\right)^{-1}} \boxed{\left(\mathbf{X}^\top \mathbf{D}\bar{\delta}_{\mathbf{w}}\right)}$$

$$\boxed{\mathbf{X}^\top \mathbf{D}\bar{\delta}_{\mathbf{w}} = \sum_s \mu(s)\mathbf{x}(s)\bar{\delta}_{\mathbf{w}}(s) = \mathbb{E}[\rho_t \delta_t \mathbf{x}_t]}$$

$$\boxed{\begin{aligned}\nabla \mathbb{E}[\rho_t \delta_t \mathbf{x}_t]^\top &= \mathbb{E}\left[\rho_t \nabla \delta_t^\top \mathbf{x}_t^\top\right] \\ &= \mathbb{E}\left[\rho_t \nabla (R_{t+1} + \gamma \mathbf{w}^\top \mathbf{x}_{t+1} - \mathbf{w}^\top \mathbf{x}_t)^\top \mathbf{x}_t^\top\right] \\ &= \mathbb{E}\left[\rho_t (\gamma \mathbf{x}_{t+1} - \mathbf{x}_t)\mathbf{x}_t^\top\right].\end{aligned}}$$

$$\boxed{\mathbf{X}^\top \mathbf{D}\mathbf{X} = \sum_s \mu(s)\mathbf{x}_s \mathbf{x}_s^\top = \mathbb{E}\left[\mathbf{x}_t \mathbf{x}_t^\top\right]}$$

$$\nabla \overline{\mathrm{PBE}}(\mathbf{w}) = 2\mathbb{E}\left[\rho_t (\gamma \mathbf{x}_{t+1} - \mathbf{x}_t)\mathbf{x}_t^\top\right] \mathbb{E}\left[\mathbf{x}_t \mathbf{x}_t^\top\right]^{-1} \mathbb{E}[\rho_t \delta_t \mathbf{x}_t]$$

梯度TD方法

$Ax = v$

梯度TD方法

$$\nabla \overline{\mathrm{PBE}}(\mathbf{w}) = 2\mathbb{E}\left[\rho_t(\gamma \mathbf{x}_{t+1}-\mathbf{x}_t)\mathbf{x}_t^\top\right]\mathbb{E}\left[\mathbf{x}_t\mathbf{x}_t^\top\right]^{-1}\mathbb{E}[\rho_t\delta_t\mathbf{x}_t]$$

$Tr(dv^\top x) = Tr(x\,dv^\top)$

若 $\mathbf{v} \approx \mathbb{E}\left[\mathbf{x}_t\mathbf{x}_t^\top\right]^{-1}\mathbb{E}[\rho_t\delta_t\mathbf{x}_t]$

LS

可视为最小二乘法，等同于最小化期望平方误差 $(\mathbf{v}^\top\mathbf{x}_t-\rho_t\delta_t)^2$

$$\mathbf{v}_{t+1} = \mathbf{v}_t + \beta\rho_t\left(\delta_t - \mathbf{v}_t^\top \mathbf{x}_t\right)\mathbf{x}_t$$

$$\nabla(v^\top x_t - \rho_t\delta_t)^2 = 2(v^\top x_t - \rho_t\delta_t)\,\nabla(v^\top x_t - \rho_t\delta_t)$$

$$= 2(v^\top x_t - \rho_t\delta_t)\,x_t$$

梯度TD方法

梯度TD方法

GTD2算法

$$
\begin{aligned}
\mathbf{w}_{t+1} &= \mathbf{w}_t - \frac{1}{2}\alpha\nabla\overline{\mathrm{PBE}}(\mathbf{w}_t) \\
&= \mathbf{w}_t - \frac{1}{2}\alpha 2\mathbb{E}\left[\rho_t(\gamma\mathbf{x}_{t+1} - \mathbf{x}_t)\mathbf{x}_t^\top\right]\mathbb{E}\left[\mathbf{x}_t\mathbf{x}_t^\top\right]^{-1}\mathbb{E}[\rho_t\delta_t\mathbf{x}_t] \\
&= \mathbf{w}_t + \alpha\mathbb{E}\left[\rho_t(\mathbf{x}_t - \gamma\mathbf{x}_{t+1})\mathbf{x}_t^\top\right]\mathbb{E}\left[\mathbf{x}_t\mathbf{x}_t^\top\right]^{-1}\mathbb{E}[\rho_t\delta_t\mathbf{x}_t] \\
&= \mathbf{w}_t + \alpha\mathbb{E}\left[\rho_t(\mathbf{x}_t - \gamma\mathbf{x}_{t+1})\mathbf{x}_t^\top\right]\mathbf{v}_t \\
&= \mathbf{w}_t + \alpha\rho_t\left(\mathbf{x}_t - \gamma\mathbf{x}_{t+1}\right)\mathbf{x}_t^\top\mathbf{v}_t.
\end{aligned}
$$

Sampling

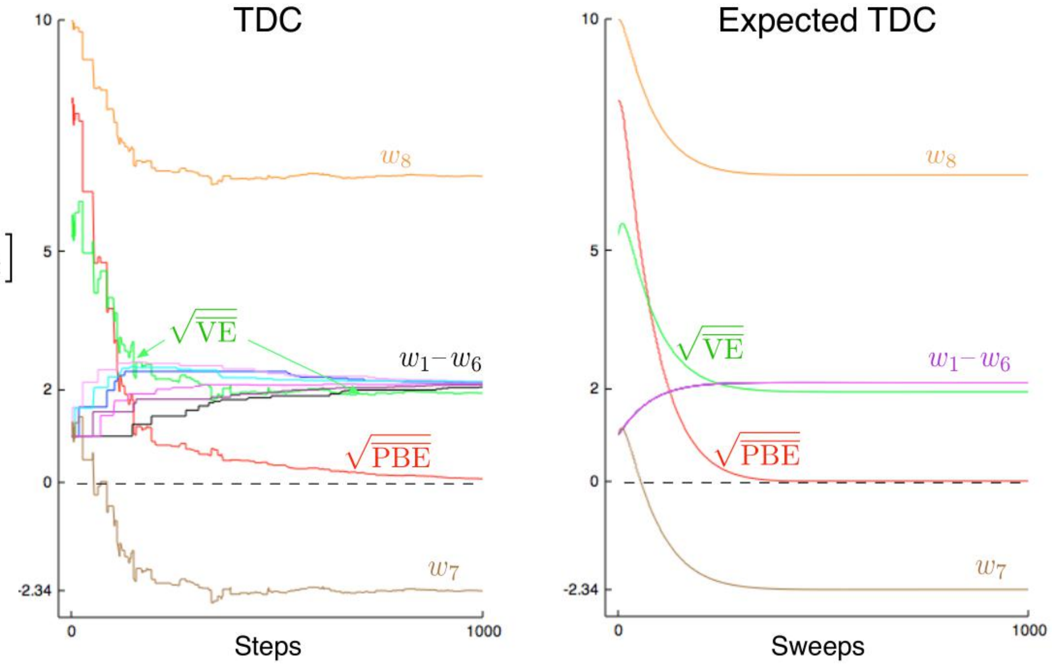
梯度TD方法

梯度TD方法

TDC算法

$$
\begin{aligned}
\mathbf{w}_{t+1} &= \mathbf{w}_t + \alpha \mathbb{E}\left[\rho_t(\mathbf{x}_t - \gamma \mathbf{x}_{t+1})\mathbf{x}_t^\top\right] \mathbb{E}\left[\mathbf{x}_t\mathbf{x}_t^\top\right]^{-1} \mathbb{E}[\rho_t\delta_t\mathbf{x}_t] \\
&= \mathbf{w}_t + \alpha \left(\mathbb{E}\left[\rho_t\mathbf{x}_t\mathbf{x}_t^\top\right] - \gamma\mathbb{E}\left[\rho_t\mathbf{x}_{t+1}\mathbf{x}_t^\top\right]\right) \mathbb{E}\left[\mathbf{x}_t\mathbf{x}_t^\top\right]^{-1} \mathbb{E}[\rho_t\delta_t\mathbf{x}_t] \\
&= \mathbf{w}_t + \alpha \left(\mathbb{E}[\mathbf{x}_t\rho_t\delta_t] - \gamma\mathbb{E}\left[\rho_t\mathbf{x}_{t+1}\mathbf{x}_t^\top\right] \mathbb{E}\left[\mathbf{x}_t\mathbf{x}_t^\top\right]^{-1} \mathbb{E}[\rho_t\delta_t\mathbf{x}_t]\right) \\
&\approx \mathbf{w}_t + \alpha \left(\mathbb{E}[\mathbf{x}_t\rho_t\delta_t] - \gamma\mathbb{E}\left[\rho_t\mathbf{x}_{t+1}\mathbf{x}_t^\top\right] \mathbf{v}_t\right) \\
&\approx \mathbf{w}_t + \alpha\rho_t \left(\delta_t\mathbf{x}_t - \gamma\mathbf{x}_{t+1}\mathbf{x}_t^\top\mathbf{v}_t\right),
\end{aligned}
$$

sampling



【结论】实验表明，以PBE为目标的梯度TD方法可以具有较好的收敛性。
【思考】梯度TD方法优点是可以保证收敛，缺点是什么？

两个学习率 slow w_t v_t

梯度TD方法

强调TD

redistribution

兴趣：非负随机标量，表示在t时刻有多大兴趣要精确估计一个状态的价值，记做 I_t

强调：非负随机标量，决定在t时刻是否强调学习（梯度更新的量），记做 M_t

同轨策略（n步学习）：

$$\mathbf{w}_{t+n} \doteq \mathbf{w}_{t+n-1} + \alpha M_t \left[G_{t:t+n} - \hat{v}(S_t,\mathbf{w}_{t+n-1})\right] \nabla \hat{v}(S_t,\mathbf{w}_{t+n-1}), \qquad 0 \le t < T,$$

$$M_t = I_t + \gamma^n M_{t-n}, \qquad 0 \le t < T,$$

离轨策略（TD）：

$$\delta_t = R_{t+1} + \gamma \hat{v}(S_{t+1},\mathbf{w}_t) - \hat{v}(S_t,\mathbf{w}_t),$$

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \alpha M_t \rho_t \delta_t \nabla \hat{v}(S_t,\mathbf{w}_t),$$

$$M_t = \gamma \rho_{t-1} M_{t-1} + I_t,$$

4、总结

梯度TD方法

GTD2与TDC算法

兴趣与强调

总结

本章要点：

【1】基于函数逼近的离轨策略方法的两大挑战：更新的目标，更新的分布；

【2】致命三要素：函数逼近、自举法、离轨策略。兼顾三者易发散；

【3】不可学习的概念；

【4】梯度TD方法；

离策略学习这个领域比较新，而且很多事情都尚无定论！因此本章作为了解即可。

参考：

【1】https://zhuanlan.zhihu.com/p/69340176

【2】https://blog.csdn.net/qq_25037903/article/details/82713736

【3】《Deep Residual Reinforcement Learning》https://arxiv.org/abs/1905.01072

第十一章：基于函数逼近的离轨策略方法

Chapter 11 : Off-policy Methods with Approximation

演讲：王嘉宁

华东师范大学 · 数据科学与工程学院 · X101实验室

2020/05/15

Thank You For Your Listening