Reinforcement Learning Notes

To Be Strong, To Be Gentle

Jiayu Song

2025 年 1 月 23 日

Motivation

Note my understanding and problems

2025 年 1 月 23 日

目录

-1.1 动态规划算法

-1.1.1 概要

基于动态规划的强化学习算法主要有两种：

策略迭代 policy iteration

策略评估 (policy evaluation)，使用贝尔曼期望方程得到一个策略的状态价值函数

策略提升 (policy improvement)，直接使用贝尔曼最优方程来进行动态规划，得到最终的最优状态价值

价值迭代 value iteration

-1.1.2 策略迭代算法

Policy Evaluation

贝尔曼期望方程

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

动态规划 + 贝尔曼期望方程

$$V^{k+1}(s)=\sum_{a\in A}\pi(a|s)(r(s,a)+\gamma\sum_{s'\in S}p(s'|s,a)V^{k}(s'))$$

Policy Improvement

Q: 如果我们有 $Q^{\pi}(s,a)>V^{\pi}(s)$，则说明在状态 s 下采取动作 a 会比原来的策略 $\pi(a|s)$ 得到更高的期望回报

假设存在一个确定性策略 π' 在任意一个状态 s 下，都满足

$$Q^{\pi}(s,\pi'(s))\geq V^{\pi}(s)$$

于是在任意状态 s 下，都有

$$V^{\pi'}(s)\geq V^{\pi}(s)$$

$$\pi'(s)=\arg\max_{a}Q^{\pi}(s,a)=\arg\max_{a}\{r(s,a)+\gamma\sum_{s'}P(s'|s,a)V^{\pi}(s')\}$$

-1.1.3 价值迭代算法

$$V^*(s) = \max_{a \in A} r(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V^*(s')$$

$$V^{k+1}(s) = \max_{a \in A} r(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V^k(s')$$

-1.2 时序差分算法

-1.2.1 概要

无模型的强化学习 model-free Reinforcement Learning

Sarsa 和 Q-Learning

在线策略学习

离线策略学习：更好地利用历史数据，并具有更小的样本复杂度

-1.2.2 时序差分方法

蒙特卡洛方法对价值函数的增量更新方式：

$$V(s_t) \leftarrow V(s_t) + \alpha[G_t - V(s_t)]$$

时序差分算法用当前获得的奖励加上下一个状态的价值估计来作为在当前状态会获得的回报

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

其中 $R_t + \gamma V(s_{t+1}) - V(s_t)$ 被称为时序差分误差 (Temporal Difference Error)

$$\begin{aligned} V_\pi(s) &= \mathbb{E}_\pi[G_t|S_t = s] \\ &= \mathbb{E}_\pi[\sum_{k=0}^{\infty} \gamma^k R_{t+k}|S_t = s] \\ &= \mathbb{E}_\pi[R_t + \gamma \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}|S_t = s] \\ &= \mathbb{E}_\pi[R_t + \gamma V_\pi(S_{t+1})|S_t = s] \end{aligned}$$

蒙特卡洛方法将上式第一行作为更新的目标，时序差分算法将上式最后一行作为更新的目标

-1.2.3 Sarsa 算法

-1.2.4 Q-Learning 算法

-1.3 马尔可夫决策过程

马尔可夫决策过程 (Markov Decision Process, MDP)，在 robotics, control, operations research, economics, game theory, artificial intelligence, reinforcement learning 等领域有着广泛的应用。MDP 是一个五元组 $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$，其中 $\mathcal{S}$ 是状态空间，$\mathcal{A}$ 是动作空间，$\mathcal{P}$ 是状态转移概率，$\mathcal{R}$ 是奖励函数，γ 是折扣因子。

-1.3.1 概念

- Random Variable

  X, Y $X \perp Y$

- Stochastic Process

  $S_t, S_{t+1}, S_{t+2}, \cdots = \{S_t\}_{t=1}^{\infty}$

- 马尔可夫链 Markov Chain: 具有 Markov Property 的随机过程

  $P(S_{t+1}|S_t, S_{t-1}, \cdots, S_1) = P(S_{t+1}|S_t)$

- State Space Model:(HMM, Kalman Filter, Particle Filter)

  Markov Chain + Observation

- Markov Reward Process

  Markov Chain + Reward

-1.3.2 集合

$\mathcal{S}$: state set $\rightarrow s_t$

$\mathcal{A}$: action set $\forall s \in \mathcal{S}, s \rightarrow \mathcal{A}_t$

$\mathcal{R}$: reward set $\rightarrow R_t, R_{t+1}$

$\mathcal{P}$: $\mathcal{P}(s', r|s, a) \triangleq P(S_{t+1} = s'|S_t = s, A_t = a)$

状态转移函数:

$$\mathcal{P}(s', r|s, a) = \sum_{r \in R} \mathcal{P}(s', r|s, a)$$

Policy: π 表示

$$\pi \begin{cases} a \triangleq \pi(s) & \text{deterministic policy} \\ P\{A_t = a|S_t = s\} & \text{random policy} \end{cases}$$

-1.3.3 回报函数

Reward Function

$$G_t \triangleq R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{i=0}^{\infty} \gamma^i R_{t+i+1} \quad \gamma \in [0,1]$$

-1.3.4 价值函数

Value Function

$$\begin{cases} v_\pi(s) \triangleq \mathbb{E}_\pi[G_t | S_t = s] \\ q_\pi(s,a) \triangleq \mathbb{E}_\pi[G_t | S_t = s, A_t = a] \end{cases}$$

$$V_\pi(s) = \sum_{a \in \mathcal{A}(s)} \pi(a|s) q_\pi(s,a)$$

$$q_\pi(s,a) = \sum_{s',r} \mathcal{P}(s', r|s,a)[r + \gamma V_\pi(s')]$$

Bellman Expectation Equation:

$$V_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) \sum_{s' \in \mathcal{S}} \mathcal{P}(s', r|s,a)[r + \gamma V_\pi(s')]$$

$$q_\pi(s,a) = \sum_{s',r} \mathcal{P}(s', r|s,a)[r + \gamma \sum_{a'} \pi(a'|s') q_\pi(s', a')]$$

$$\begin{cases} V_*(s) \triangleq \max\limits_\pi V_\pi(s) \\ q_*(s,a) \triangleq \max\limits_\pi q_\pi(s,a) \end{cases}$$

记 $\pi_* = \arg\max\limits_\pi V_\pi(s) = \arg\max\limits_\pi q_\pi(s,a)$

$V_*(s) = \max\limits_\pi V_\pi(s) = V_{\pi_*}(s)$, $q_*(s,a) = \max\limits_\pi q_\pi(s,a) = q_{\pi_*}(s,a)$

$V_{\pi_*}(s) = \max\limits_a a_{\pi_*}(s,a)$

假定 $V_{\pi_*} < \max q_{\pi_*}(s,a)$，则可以构造 π_{new}

$$\pi_{new} = \begin{cases} \pi_{new}(s) = \arg\max\limits_a q_{\pi_*}(s,a) & \forall s \in S \\ \pi_{new}(\bar{s}) = \pi_*(\bar{s}) || \pi_{new}(a|\bar{s}) = \pi_*(a|\bar{s}) \end{cases}$$

$$\therefore V_{\pi_{new}}(s) = \max_a q_{\pi_*}(s,a) > V_{\pi_*}(s)$$

$\therefore \pi_{new}$ is better than π_*，这与 π_* 最优矛盾

$\therefore V_{\pi_*}(s) = \max_a q_{\pi_*}(s,a)$

$$V_*(s) = \max_a q_*(s,a)$$

$$q_*(s,a) = \sum_{s',r} \mathcal{P}(s',r|s,a)[r + \gamma V_*(s')]$$

Bellman Optimality Equation:

$$V_*(s) = \max_a \sum_{s',r} \mathcal{P}(s',r|s,a)[r + \gamma V_*(s')]$$

$$q_*(s,a) = \sum_{s',r} \mathcal{P}(s',r|s,a)[r + \gamma \max_{a'} q_*(s',a')]$$

-1.4 Paper Reading

-1.4.1 SCI 写作积累

引出自己的概念

“However, we believe that it is a distinct concept that is worth clarifying.”

“To disentangle the parameter count from the filter size,”

表示递进

“More narrowly,”

“Empirically”

同义词替换

解决“solve —> decouple = disentangle”

大量的“a lot of —> prohibitive amounts of; a great number of”

-1.4.2 Mamba: Linear-Time Sequence Modeling with Selective State Spaces

原文状态方程和离散化状态方程：

$$\begin{cases} h^{'}(t) = Ah(t) + Bx(t) \\ y(t) = Ch(t) \end{cases} \tag{1}$$

$$\begin{cases} h_t = \overline{A}h_{t-1} + \overline{B}x_t \\ y_t = Ch_t \end{cases} \tag{2}$$

$$\begin{cases} \overline{K} = (C\overline{B}, C\overline{AB}, ..., C\overline{A}^k\overline{B}) \\ y = x * \overline{K} \end{cases} \tag{3}$$

SSM 离散化过程推导

公式1到公式2是怎样实现的?

构造状态方程

构造新的函数 $\alpha(t)h(t)$，并对新函数求导

$$\frac{d[\alpha(t)h(t)]}{dt} = \frac{d\alpha(t)}{t}h(t) + \alpha(t)h^{'}(t) \tag{4}$$

将公式1代入公式4得到：

$$\begin{aligned}\frac{d[\alpha(t)h(t)]}{dt} &= \frac{d\alpha(t)}{dt}h(t) + \alpha(t)[Ah(t) + Bx(t)] \\ &= [A\alpha(t) + \frac{d\alpha(t)}{dt}]h(t) + B\alpha(t)x(t)\end{aligned} \tag{5}$$

为消除 $h(t)$，则有：

$$\begin{aligned}A\alpha(t) + \frac{d\alpha(t)}{dt} &= 0 \\ \frac{d\alpha(t)}{dt} &= -A\alpha(t) \\ \alpha(t) &= e^{-At}\underbrace{+C}_{overlook}\end{aligned} \tag{6}$$

将公式6代入公式5得到：

$$\frac{d[e^{-At}h(t)]}{dt} = Be^{-At}x(t) \tag{7}$$

两边求积分得到：

$$\begin{aligned}e^{-At}h(t) &= h(0) + \int_0^t Be^{-A\tau}x(\tau)\,d\tau \\ h(t) &= h(0)e^{At} + \int_0^t e^{A(t-\tau)}Bx(\tau)\,d\tau\end{aligned} \tag{8}$$

离散化后对 $\{t_k, t_{k+1}\}$，时间间隔为 $T = t_{k+1} - t_k$，做差分：

$$h(t_{k+1}) = h(t_k)e^{AT} + \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)}Bx(\tau)\,d\tau \tag{9}$$

令 $\eta = t_{k+1} - \tau$ 得到（"-" 改变上下界，和换元有两次变号）：

$$\begin{aligned}h(t_{k+1}) &= h(t_k)e^{AT} + \underbrace{\int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)}}_{-}\underbrace{Bx(\tau)\,d\tau}_{-} \\ &= h(t_k)e^{AT} + \int_0^T e^{A\eta}Bx(\eta)\,d\eta\end{aligned} \tag{10}$$

对 $x(t)$ 应用零阶保持器，在 $\{t_k, t_{k+1}\}$ 区间数值不发生变化，可作为常数提出，于是则有：

$$\begin{aligned}h(t_{k+1}) &= h(t_k)e^{AT} + \int_0^T e^{A\eta}\,d\eta Bx(t_k) \\ &= h(t_k)e^{AT} + [A^{-1}e^{AT} - A^{-1}]Bx(t_k) \\ &= e^{AT}h(t_k) + A^{-1}(e^{AT} - I)Bx(t_k)\end{aligned} \tag{11}$$

时间差 $T = \Delta$(常数)，上式为：

$$h(t_{k+1}) = e^{A\Delta}h(t_k) + A^{-1}(e^{A\Delta} - I)Bx(t_k) \tag{12}$$

所以 Mamba 原文公式2中 $\overline{A}$ 和 $\overline{B}$ 对应的值为：

$$\begin{cases} \overline{A} & = e^{\Delta A} \\ \overline{B} & = A^{-1}(e^{\Delta A} - I)B \end{cases} \tag{13}$$

这一结果与原文中的结果相对应，但是让我想不明白的是：为什么要在 A^{-1} 中加入 Δ^{-1} 并在后面乘 Δ，虽然这样变化整式的结果是没有变化的。

-1.4.3 Social Influence as Intrinsic Motivation for Multi-Agent Deep RL

条件互信息计算公式 Conditional Mutual Information：

$$I(X;Y|Z) = \sum_{x\in X}\sum_{y\in Y} p(x,y|Z)\log(\frac{p(x,y|Z)}{p(x|Z)p(y|Z)})$$

KL Divergence 计算公式：

$$D_{KL}(P||Q) = \sum_{x\in X} P(x)\log\frac{P(x)}{Q(x)}$$

社会影响奖励函数

Agent k 对 Agent j 产生影响会得到相应的影响奖励：

$$\begin{aligned} c_t^k &= \sum_{j=0,j\neq k}^{N} [D_{KL}[p(a_t^j|a_t^k, s_t^j)|| \underbrace{\sum_{\tilde{a}_t^k} p(a_t^j|\tilde{a}_t^k, s_t^j)p(\tilde{a}_t^k|s_t^j)}_{sample\ counterfactual\ actions}]] \\ &= \sum_{j=0,j\neq k}^{N} [D_{KL}[p(a_t^j|a_t^k, s_t^j)|| \underbrace{p(a_t^j|s_t^j)}_{margin\ policy}]] \end{aligned} \tag{14}$$

Basic Soical Influence

结论：最大化动作间的互信息会促进多智体之间的协调

Conclusion: training agents to maximize the MI between their actions results in more coordinated behavior.

Proof of Influence as Mutual Information:

Agent k 对 Agent j 的因果影响：

$$D_{KL}[p(a_t^j|a_t^k, z_t)||p(a_t^j|z_t)] \tag{15}$$

影响奖励（Agent k 和 Agent j）：

$$I(A^j; A^k|z) = \sum_{a^k,a^j} p(a^j, a^k|z) \log \frac{p(a^j, a^k|z)}{p(a^j|z)p(a^k|z)} \tag{16}$$

$$= \sum_{a^k} p(a^k|z) \sum_{a^j} p(a^j|a^k, z) \log \frac{p(a^j|a^k, z)}{p(a^j|z)} \tag{17}$$

$$= \sum_{a^k} p(a^k|z) D_{KL}[p(a^j|a^k, z)||p(a^j|z)] \tag{18}$$

上式表示，在给定条件 z 下，agent k 和 agent j 之间的互信息可以通过计算 a^k 上的期望 KL 散度来表示。

$$\begin{aligned} pmi(a^k; a^j|Z = z) &= \log \frac{p(a^j|a^k, z)}{p(a^j|z)} \\ &= \log \frac{p(a^j, a^k \mid z)}{p(a^j \mid z)p(a^k \mid z)} \end{aligned}$$

$$\begin{aligned} I(A^j; A^k|z) &= \sum_{a^k,a^j} p(a^j, a^k|z) \log \frac{p(a^j, a^k|z)}{p(a^j|z)p(a^k|z)} \\ &= \sum_{a^k,a^j} p(a^j, a^k|z) pmi(a^k; a^j|Z = z) \end{aligned}$$

所以，PMI 在 $p(a^j, a^k \mid z)$ 上的期望是 MI。对应原文的“The expectation of the PMI over $p(a^j, a^k \mid z)$ is the MI. ”

存在的问题：随着智能体数量的增加，策略梯度更新的方差也会增加，会阻碍收敛

Problem: the variance of policy gradient updates increases as the number of agents in the environment grows (Lowe et al., 2017). This issue can hinder convergence to equilibrium for large-scale MARL tasks.

Assumptions

1. 集中训练以计算 c_t^k
2. 受到影响奖励的 agent 只能影响未受到影响奖励的 agent

Experiment I: Basic Influence

influencer: 只有在追逐苹果的时候，才会穿越地图，其他时候呆着不动；

influencee: 在等待生成苹果的时候，随机移动和探索地图；

Influential Communication

Experiment II: Influential Communication

1. speaker consistency $\in [0, 1]$，计算 $p(a^k \mid m^k)$ 和 $p(m^k \mid a^k)$ 的熵，以确定发言者在采取特定动作时发出特定符号的频率。而且期望这一度量值越高越好，例如，当发言者总是在执行相同动作时发出相同的符号。

2. instantaneous coordination(IC) 用来度量互信息

   信号/动作 $IC = I(m_t^k; a_{t+1}^j)$ 计算影响者的信号和被影响者的动作的 MI

   动作/动作 $IC = I(a_t^k; a_{t+1}^j)$ 计算影响者的动作和被影响者的下一个动作的 MI

Conclusion: Because the listener agent is not compelled to listen to any given speaker listeners selectively listen to a speaker only when it is beneficial, and influence cannot occur all the time. Only when the listener decides to change its action based on the speaker's message does influence occur, and in these moments we observe high $I(m_t^k; a_{t+1}^j)$. It appears the influencers have learned a strategy of communicating meaningful information about their own actions, and gaining influence when this becomes relevant enough for the listener to act on it.

Modeling Other Agents

结构：2*FC+LSTM

每个智能体可以“想象”其反事实动作在每个时间步可能采取的动作，并使用内部 MOA 来预测这些动作对其他智能体的影响。然后它可以给自己奖励那些它估计最有影响力的动作。

当 k 能看见 j 时，$p(a_{t+1}^j \mid a_t^k, s_t^k)$ 会变得准确。⇔ “This constraint could have the side effect of encouraging agents to stay in closer proximity.”

Experiment III: Modeling Other Agents

agents with influence achieve higher collective reward than pre-SOTA.

Related Work

- 手工设计的奖励和情感内在奖励：早期的工作如 Sequeira 等人和 Yu 等人研究了手工设计的奖励和情感内在奖励，但这些方法在需要长期策略的复杂环境中效果有限。
- 查看其他智能体的奖励：Peysakhovich 和 Lerer 以及 Hughes 等人研究了智能体如何通过查看和优化其他智能体的奖励来提高集体绩效，但这些假设在实际应用中可能不现实。
- 新兴通信协议：Foerster 等人和 Cao 等人探讨了训练智能体学习通信协议的方法，但发现自私的智能体难以使用无基础的"cheap talk" 通信。
- 机器理论和反事实推理：Rabinowitz 等人的机器理论和 Barton 等人的因果影响度量提供了新的视角，但依然依赖集中控制。