Ch2.

ex. 多臂老虎机

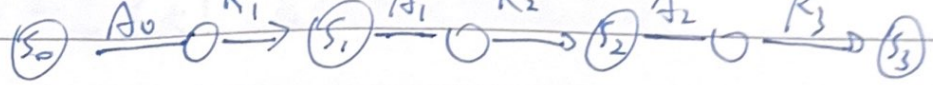
1. 多臂老虎机
2. 平衡exploit - explore

ε-greedy { ε random choice
1-ε chose the best.

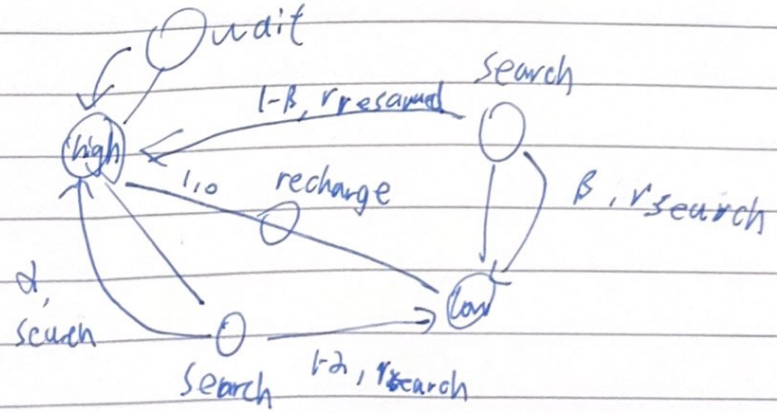
MDP. (Markov decision Process)

① should be Markov chain.

ex



ex



DP.

best policy = max{ all sub state' policy + transfer }

缺点：状态空间指数上增长

优点：状态空间小时相当快（比直接暴力要快很多）

ex. 0-1背包：给出 n个存价物品与k元钱，如何取金使得收益最大！（存价物品价值不同）

强化学习方法：

对比 { DP：基于结构从根节点全局向下
MC：大样本从抽样回报角度

on-policy：ε-greedy.

重要性采样（off-policy）

How to use $E_b(G)$ to estimate $E_\pi(G)$?

$P_\pi(\{(S_i, A_i)\}_{i=t}^{T}) = \prod_{i=t}^{T} \pi(A_i|S_i) p(S_{i+1}|S_i, A_i)$

$\therefore p_{t:T-1} = \frac{\prod_{i=t}^{T} \pi(A_i|S_i)}{\prod_{i=t}^{T} b(A_i|S_i)}$

一般重要性采样：

$V(s) \approx \frac{\sum_{k}^{N} \rho_{T^k(s):T^k(s)} G_{T^k(s)}}{N}$

其中，$T^k(s) = \min\{i : S_i^k = s\}$

加权重要性采样

$$V_n(s) = \frac{\sum_k^n \rho_{t:T-1}^k G_t^k(s)}{\sum_k^n \rho_{t:T-1}^k}$$

(有偏估计，但 $b \to 0$)

常量实现：

$$V_n = \frac{\sum_{k=1}^{n-1} W_k G_k}{\sum_k^{n-1} W_k}$$

$$\to V_{n+1} = V_n + \frac{W_n}{C_n}(G_n - V_n)$$

其中 $C_{n+1} = C_n + W_{n+1}$

Ch3.

TD方法（时间差分法）

对比：

$$\begin{cases} MC: V(S_t) \leftarrow V(S_t) + \alpha[G_t - V(S_t)] \\ TD: V(S_t) \leftarrow V(S_t) + \alpha[R_{t+1} + \gamma V(S_{t+1}) - V(S_t)] \end{cases}$$

相比于DP，TD采用估计值而非真实值去更新模型，
相比于MC，TD采用 one-step 去估计未来型.

上述为TD(0).

Sarsa: on-plicy TD

TD(0) for V(action):

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)]$$

Sarsa: $S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1}$.

Q-learning : off-policy TD

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)]$$

收敛于 q_*. 类似 sarsa.

Expected Sarsa:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma E[Q(S_{t+1}, A_{t+1}) | S_{t+1}]$$

$$- Q(S_t, A_t)] \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma \sum_a \pi(a|S_{t+1}) Q(S_{t+1}, a)$$
$$- Q(S_t, A_t)]$$

Ch4.

DQN (Deep Q-Networks)

Q-learning 的深度学习的版本

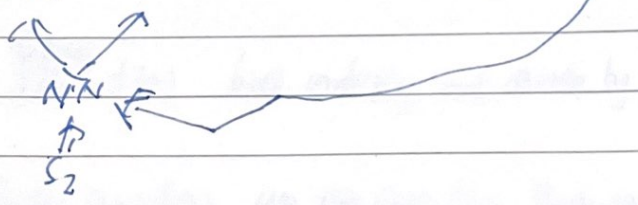
更新函数 Q-learning

Loss: $L(\theta) = E[(Target\ Q - Q(s,a;\theta))^2]$

$Target\ Q = r + \gamma \max_{a'} Q(s', a'; \theta)$

2(Q real − Q estimated)

Q(s') 真实 $R + \gamma * \max[Q(s', a_1), Q(s', a_2)]$



非常重要的一块知识..

Ch5. Ch6.

Policy Gradient 方法

直接更新策略网络的 DQN.

$L(\theta) = E[r_1 + \gamma r_2 + \gamma^2 r_3 + \cdots | \pi(\cdot, \theta)]$.

思路：不仅选一个好的动作，还判断不好.

loss: $L(\theta) = \sum \log \pi(a|s, \theta) f(s, a)$

$\nabla_\theta E_x[f(x)] = \nabla_\theta \sum_x p(x) f(x)$ def.

$= \sum_x \nabla_\theta p(x) f(x)$ Swap sum and gradient

$= \sum_x p(x) \frac{\nabla_\theta p(x)}{p(x)} f(x)$ both multiply and divide by $p(\theta)$

$= \sum_x p(x) \nabla_\theta \log p(x) f(x)$ use the fact that $\nabla_\theta \log(z) = \frac{1}{z} \nabla_\theta z$

$= E_x[f(x) \nabla_\theta \log p(x)]$ def of exp.

off-policy version

- Greedy-GQ 适于前3，比较难收敛.
- GTD
- DDPG.

Ch7.

Entropy：火高的话。

$$H(x) \triangleq E[I(x)] = -\sum_{x \in X} p(x) \log_b p(x)$$

DRL:

$$\pi^* = \arg\max_{\pi} E_{(s_t, a_t) \sim p_\pi}\left[\sum_t R(s_t, a_t)\right]$$

Max-Entropy:

$$\pi^* = \arg\max_{\pi} E_{(s_t, a_t) \sim p_\pi}\left[\sum_t \underbrace{R(s_t, a_t)}_{\text{reward}} + \underbrace{\text{entropy}(\pi(\cdot|s_t))}_{\text{Entropy}}\right]$$

本质思想：不遗漏到任意一个有用的 action，有用的 trajectory.

SAC $\begin{cases} \text{一个策略网络} \\ \text{两个Q网络} \end{cases}$

Ch8.

Ex: Coffe-making robot.

Challenges:
1. Diversity of problems
2. Multi-dimensional and multi-modal goals
3. Scalability
4. Instability
5. Optimality

Inverse RL (逆向强化学习) aka. IRL.

exp
- Soft Q imitation
- SQIL
- DQfD
- GAIL.

Other paradigms:

~~Eaten~~. ex: meta-learning
- Transfer Learning
- ⋮

Ch4. Practical Reinforcement Learning

① RL Project Life Cycle.

try → fail → 卡住 → fail → change Data → success

② RL project defination.

1. Sequential
2. strategic

本章主要一些职业和发展的形而上内容，不再赘述。

Ch 10. Operational RL.

framework

Abstractions

Simplicity

Observability

Scalability

minimal dependencies

Documentation and examples

Included algorithms

Ancillary algorithms

Support

comercial offerings.

Deployment

Log file.

Parameter server

Policy severs

Replay buffers

Actors and critics

Learning

Data augmentation.

Ch 11. Summary.

前言：区别于DL的一种工具，仅此而已。