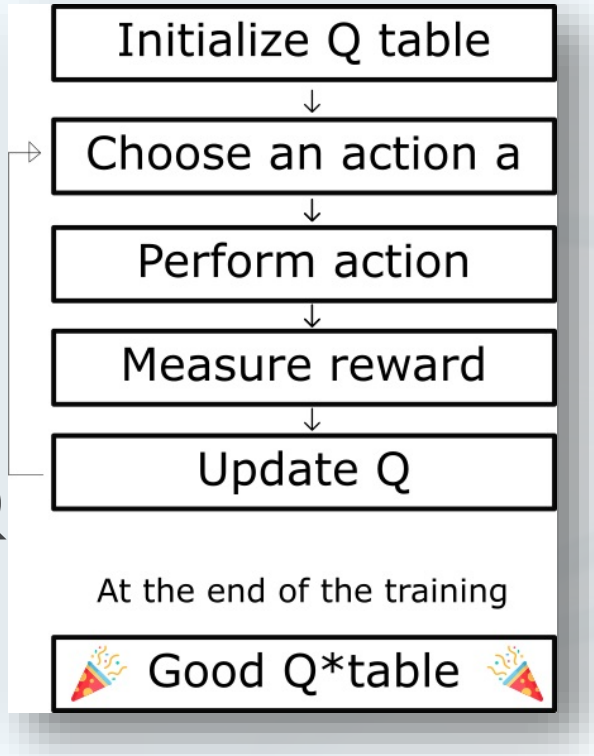
人工智能与信息社会

基于神经网络的智能系统II：熟能生巧-持续更新

陈斌 北京大学 gischen@pku.edu.cn

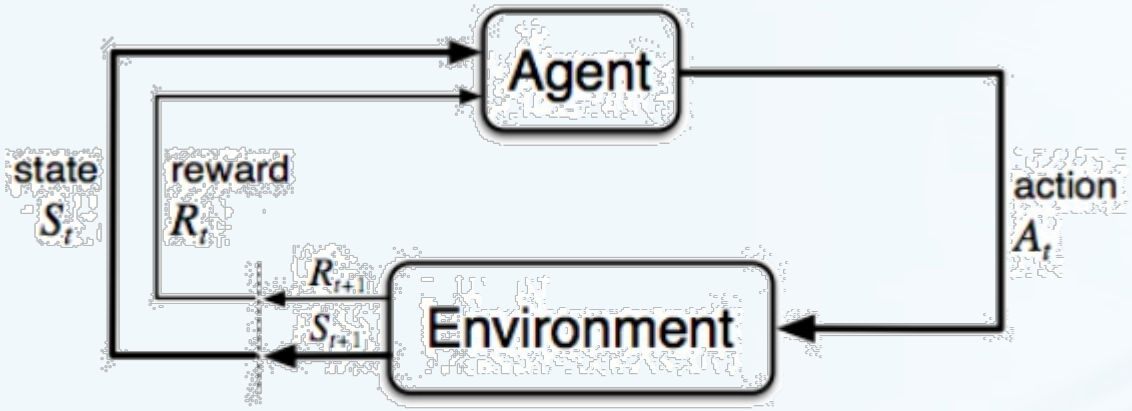
学习流程

- 初始化Q函数
- 不断重复每一局游戏
  - 选择动作
  - 得到回报
  - 更新Q函数
- 最终得到一个好的Q函数



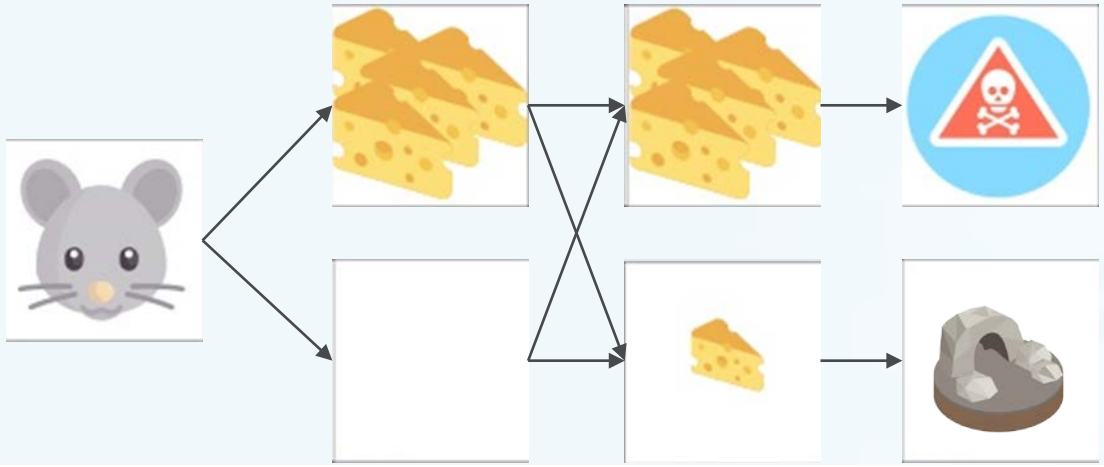
动作-状态序列

- 每一局游戏都是一个动作状态序列
- 下一个状态只和当前的状态+动作有关（马尔可夫性质）



长期回报

- 除了试错式搜索之外，强化学习的另一个重要的特点是回报的滞后性。
- 当前状态下的动作所产生的回报不仅取决于下一个状态，还取决于整个序列之后的每一个状态。



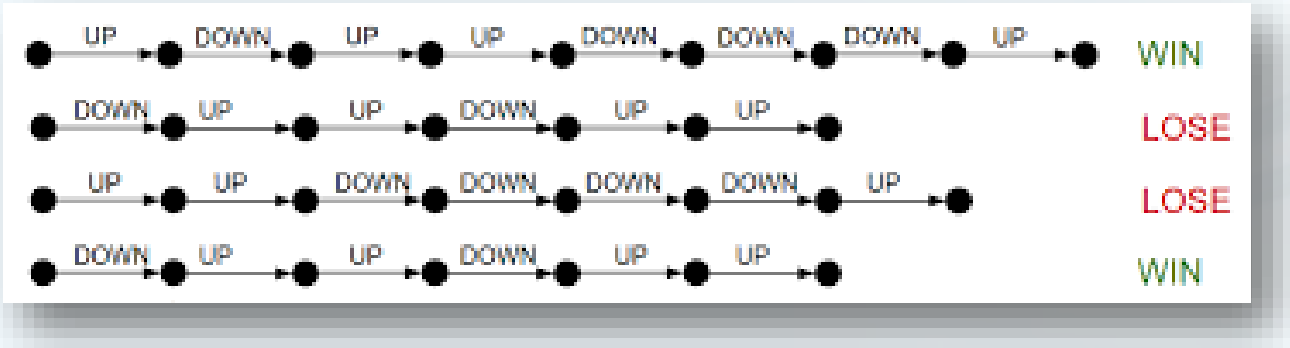
回报率

- 当前的动作对下一状态的影响是最直接的，对后续状态影响没那么直接。
- 某些动作产生的当前回报值比较高，但从长远来看，可能并没有那么高。
- 因此我们用一个回报率来平衡下一状态回报和更远状态回报。

$0.9x$ 状态1 $+0.81x$ 状态2 $+0.729x$ 状态3 $+\ldots\ldots$

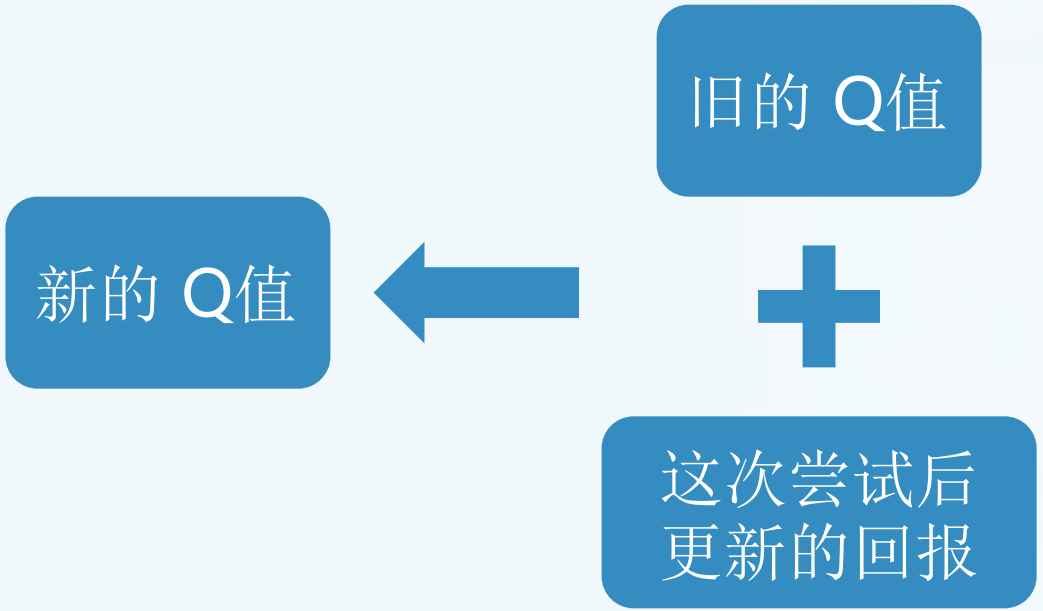
回报函数

- 每一次游戏会产生不同的状态动作序列，即每一次对后续状态的回报计算都不相同。
- 我们用后续状态的期望，即所有之后的序列的回报平均值作为回报函数。
- 回报函数值就是Q值。



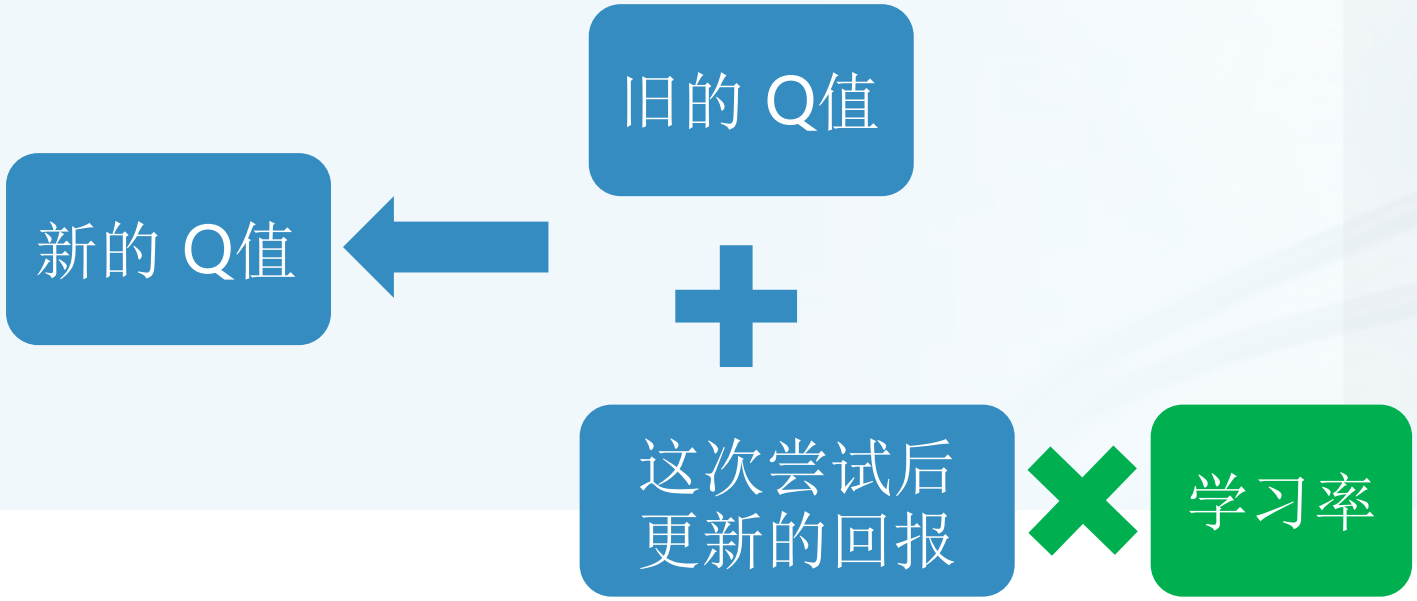
学习过程

- 每完成一局之后，就持续更新Q函数。
- 完成的局数越多，更新的次数就越多，结果也越准确。



学习率

- 既要利用好已经学好的值，也要善于学习新的值。
- 这两者就通过学习率来平衡，一开始学习率可以大一些，最后稳定时学习率可以小一些。



熟能生巧

- 通过上述公式学习，在足够多的尝试之后，AI所学到的状态动作值函数Q就能够达到一个较优的结果。
- 再根据这个Q函数来选择动作，就 “熟能生巧”了！