机器学习

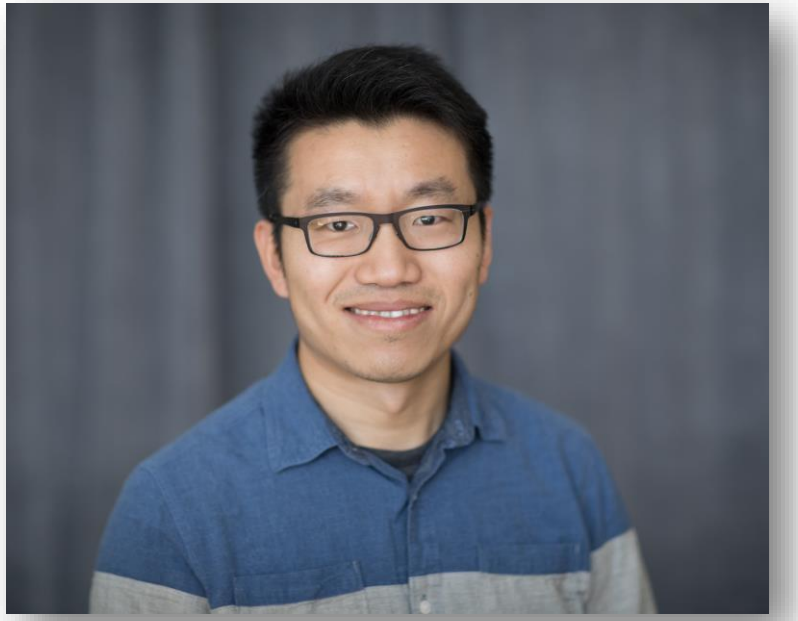
深 度 强 化 学 习

Multi-arm Bandits 算法总览

滴滴内部
学习资料
请勿外传

Tony Qin (秦志伟)

讲师介绍



Tony Qin (秦志伟)

负责大数据&滴滴研究院 强化学习组

致力于强化学习和机器学习算法的通用化，
以及在公司各主要业务方向的实践应用。

多臂老虎机问题

- Multi-arm bandits
  - 多台老虎机（每台一臂）
  - 一次拉一个老虎机
  - 执行动作后立即得知结果（赢钱数量）
  - 每台老虎机的中奖几率不同，且未知

- 目标
  - 最大化一定步数内的总奖励期望
  - 假设每一步拉一个老虎机
  - “只能拉1000次，怎么拉才能获得尽可能多的钱？”

多臂老虎机问题

- 每个老虎机都有自己的奖励概率
- 动作价值：选择一个给定动作能得到的奖励期望
  - $q_*(a) \coloneqq E[R_t|A_t = a]$：第t步时选择a老虎机的期望收益
- 如果知道动作价值
  - 最优策略就是不断拉价值最高的那个老虎机
- 但是，不知道（确切）动作价值
  - 可能有预估：$Q_t(a) \approx q_*(a)$

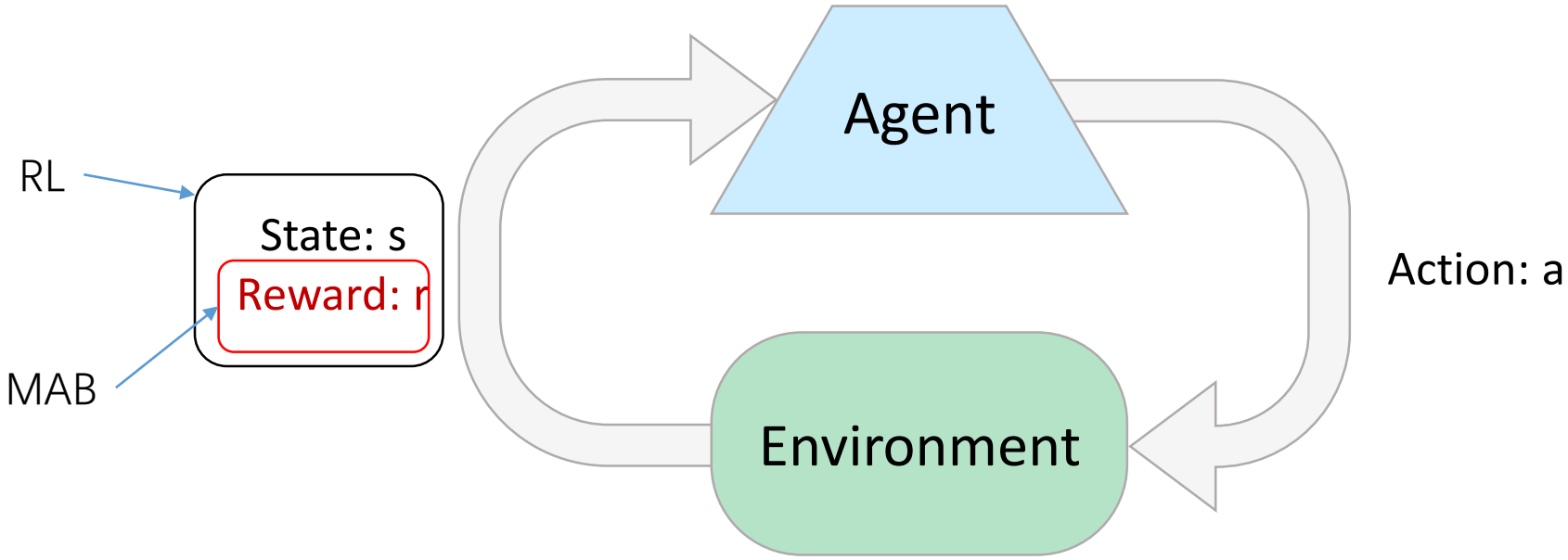
多臂老虎机问题

- 和强化学习问题的不同
- 简化版的MDP问题
  - 动作并不会影响老虎机的状态（奖励期望，q_*）
  - 策略（动作）也不取决于状态
  - 需要学习的策略仅针对单一状态

- 通过观察奖励预估动作价值
- 大思路：逐渐将执行动作的机会集中在（我们觉得）优质的老虎机上

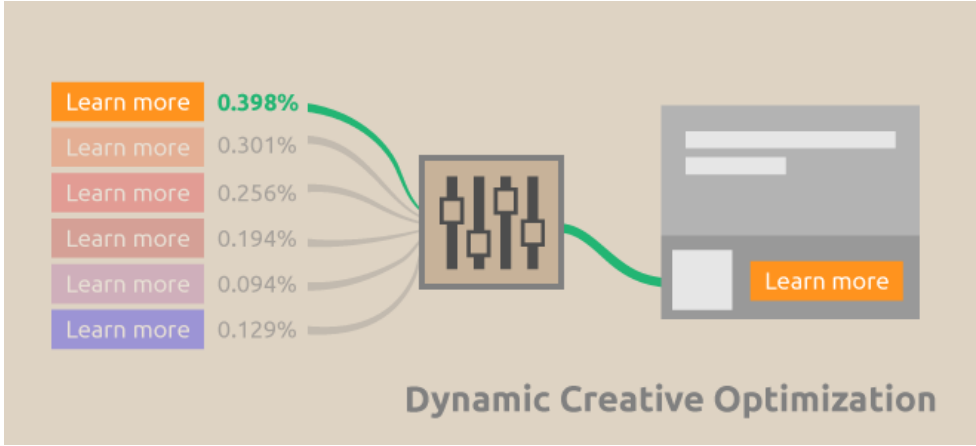
MAB v.s. RL

- 强化学习
  - 需要学习应对多种情况的策略
  - 策略 $\pi(a|s)$ 给出动作或动作分布



MAB的应用

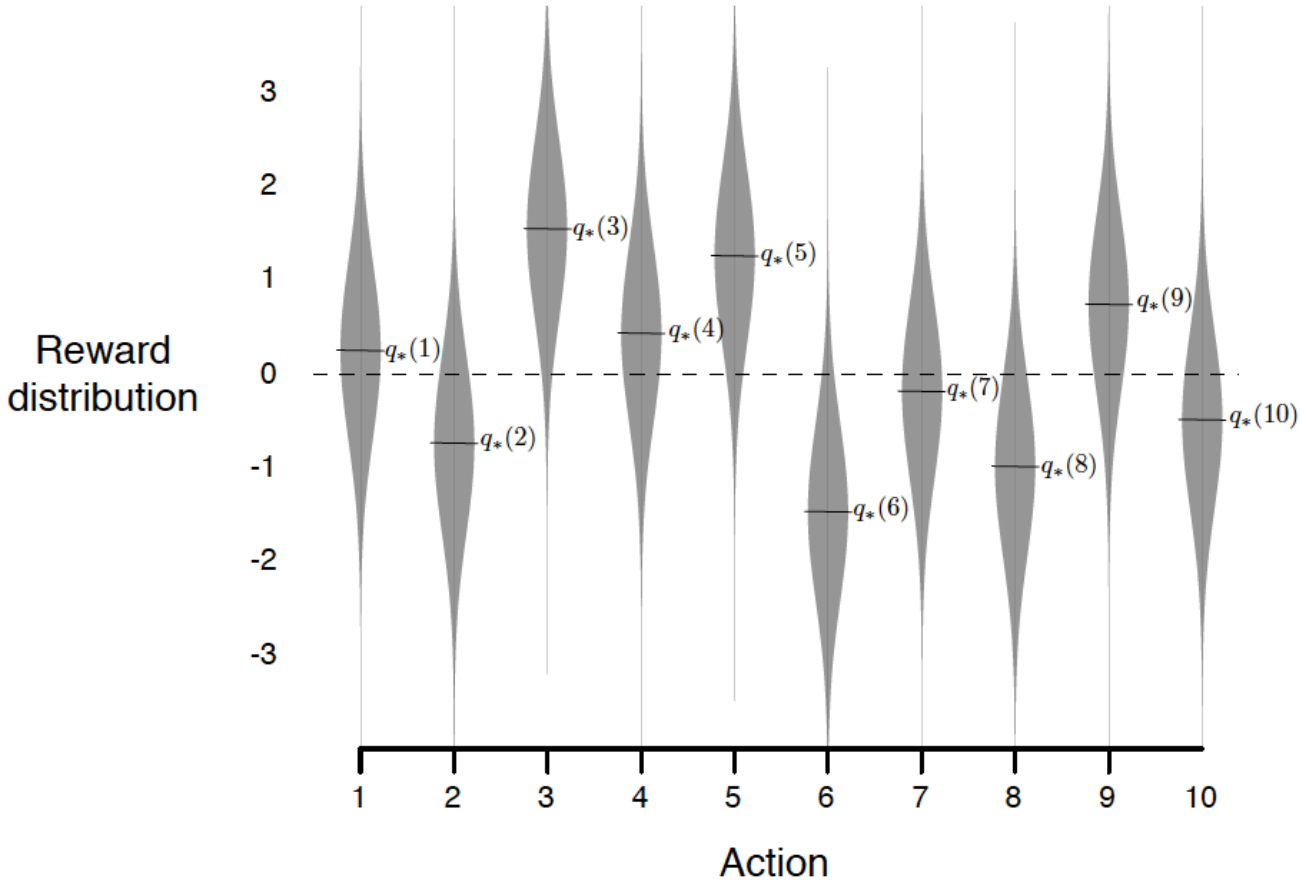
- 广告投放显示
  - 同一广告有不同设计
- 对象人群组成未知
  - K组群，不同人数，每组偏爱特定的设计



- 每次选取一个设计显示给一个随机抽取的顾客看
  - 选取的顾客喜好未知
  - 喜好分布
  - 动作价值：点击率（CTR）

MAB的应用

- 把显示机会合理分配到更好的设计上
- 线上AB实验？
  - 各种选择只测了一次
  - “多臂”时下结论更难
- 动态“AB测试”
  - 差的选项逐步减少显示机会而被自然淘汰



探索，探索，探索

- 真实动作价值未知
  - 只有估测，有一定的信心范围
  - 当前预估最好的动作不一定长期最好
- 动作价值只能通过观察奖励（反馈）预估
  - 要有足够的探索
- 探索程度取决于价值预估值，预估的不确定性，和剩余步数
  - 如果非常确定A比B好，无需探索
  - 执行过程早期，更倾向于探索

计算动作价值

- 样本平均
  - $Q_t(a) := \frac{\sum_{i=1}^{t-1} R_i 1_{A_i=a}}{\sum_{i=1}^{t-1} 1_{A_i=a}}$
  - t 步前的平均奖励
- 假设环境静态(stationary), 大数定理: $Q_t(a) \to q_*(a),\ t \to \infty$
- 增量计算
  - $Q_{t+1} = Q_t + \frac{1}{t}(R_t - Q_t)$
  - 新预估 <-- 旧预估 + 步长*（目标-旧预估）

Non-stationarity

- 老虎机的状态随时间变化
  - 样本平均会很大落后于状态变化
- 用固定步长
  - $Q_{t+1} = Q_t + \alpha(R_t - Q_t) = (1-\alpha)^t Q_1 + \sum_{i=1}^{t} \alpha(1-\alpha)^{t-i} R_i$
  - “忘掉久远的记忆”
- 收敛性
  - $\alpha = \frac{1}{t}$：收敛到定值
  - 固定步长：不收敛，跟踪最近的奖励变化

探索方法：ϵ-greedy

- 每次选预估价值最高的老虎机
  - Greedy, 永远exploit
- 探索
  - ϵ的概率随机执行动作
- 在步数极限
  - 每个老虎机都被拉了无数次
  - $Q_t(a) \rightarrow q_*(a)$

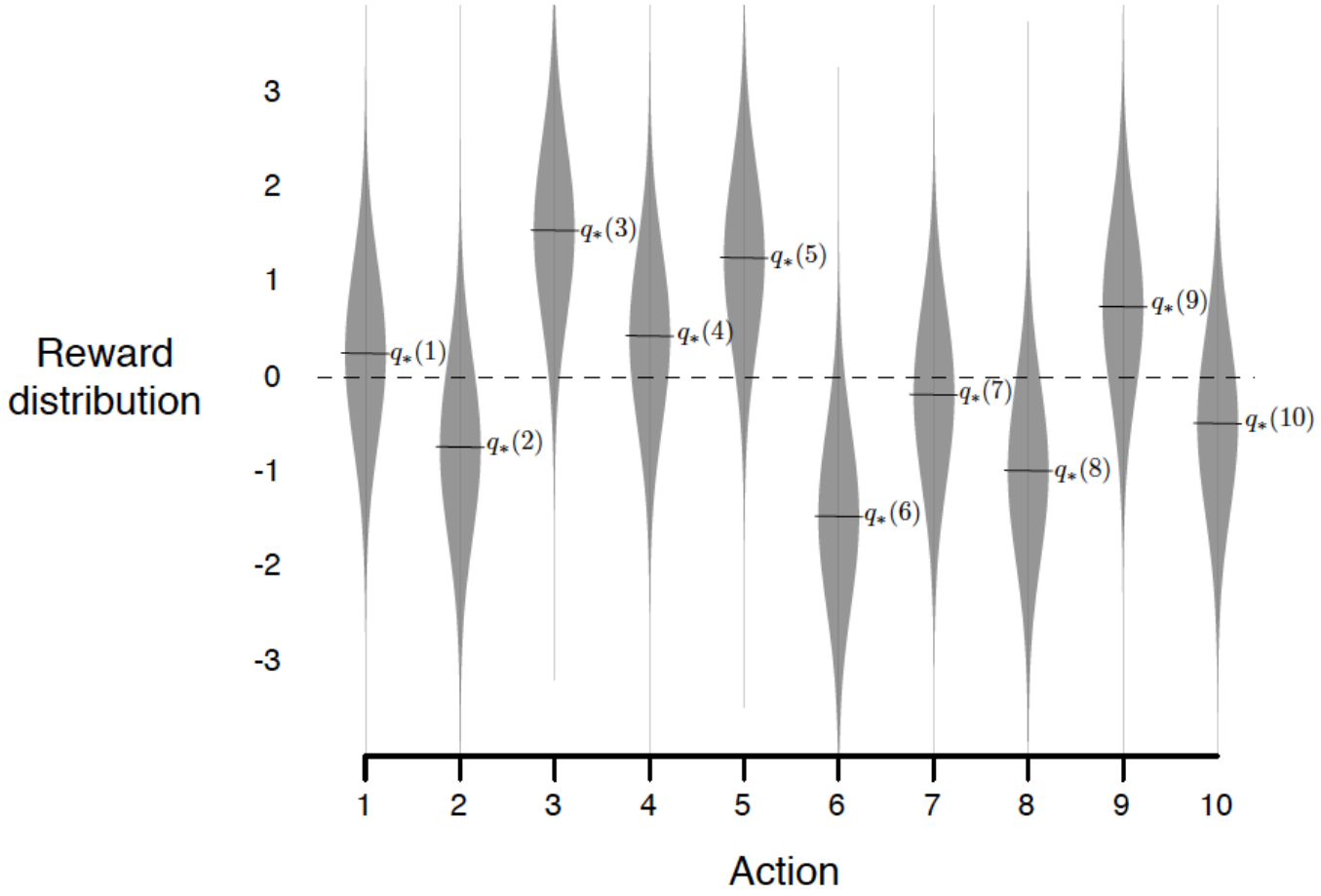
A simple bandit algorithm

Initialize, for $a = 1$ to k:
$Q(a) \leftarrow 0$
$N(a) \leftarrow 0$

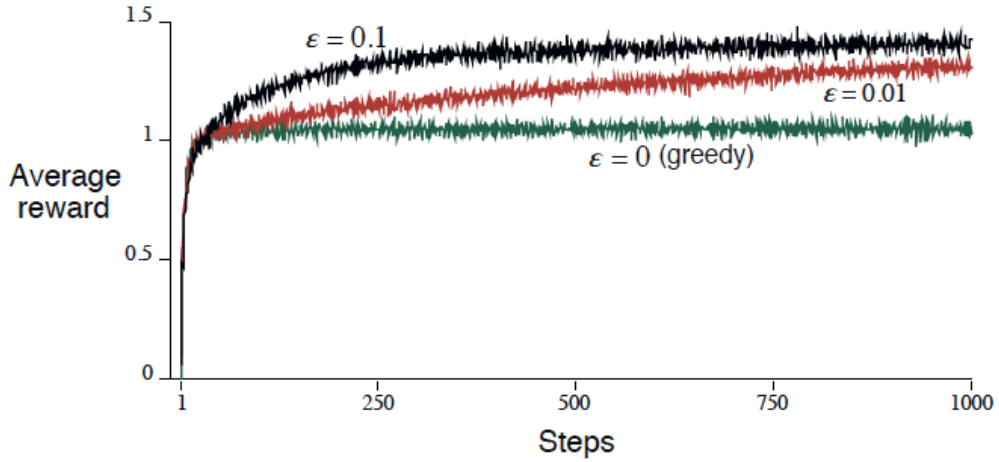
Repeat forever:
$A \leftarrow \begin{cases} \arg\max_a Q(a) & \text{with probability } 1-\varepsilon \quad \text{(breaking ties randomly)} \\ \text{a random action} & \text{with probability } \varepsilon \end{cases}$
$R \leftarrow bandit(A)$
$N(A) \leftarrow N(A) + 1$
$Q(A) \leftarrow Q(A) + \frac{1}{N(A)}[R - Q(A)]$

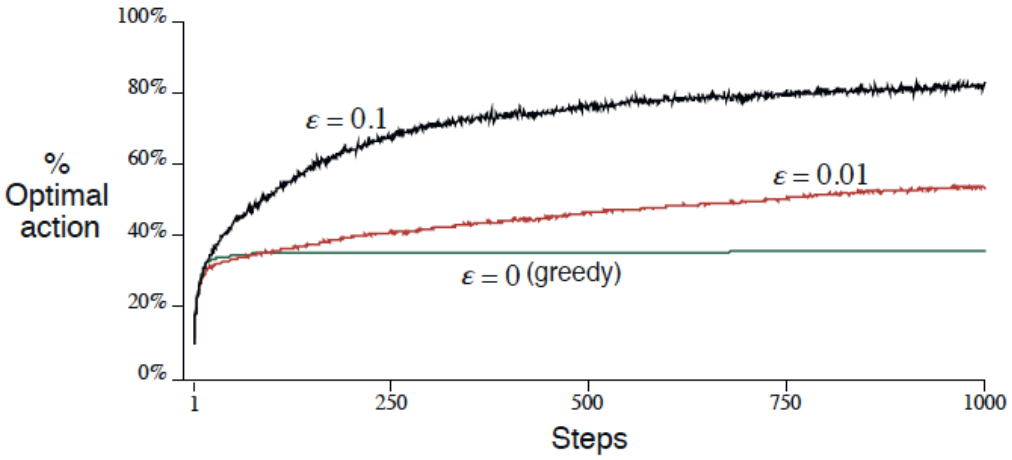
对比实验

- 10臂老虎机
  - 每个臂真值价值$q_*(a)$
  - $q_*(a)$~Gaussian(0,1)
  - 2000个独立10臂问题
- 在时间点t 选择动作A_t
  - $R_t \sim Gaussian(q_*(A_t), 1)$
- 每个10臂问题1000步



ϵ-greedy 不同探索程度





UCB (Upper Confidence Bound)

- ϵ-greedy 做的是无区分性的探索
  - 对于动作价值确定的选项效率不高

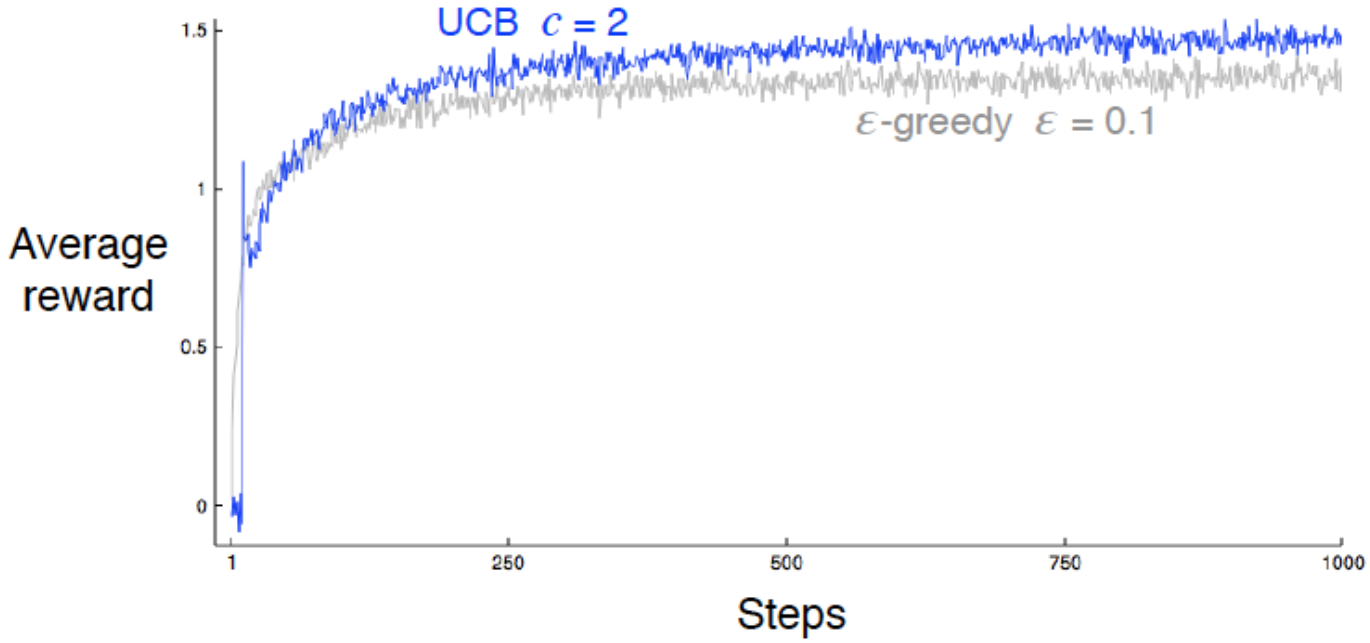
- 根据需要自动控制探索程度
  - Upper Confidence Bound：综合考虑已知价值和不确定性
  - $A_t := argmax_a(Q_t(a) + c\sqrt{\frac{\log t}{N_t(a)}})$ ← 比较各臂价值的上限

预估价值　信心度　不确定性

- A被选，计数增加，不确定性降低
- A没被选，t增加，不确定性上升，但越来越慢

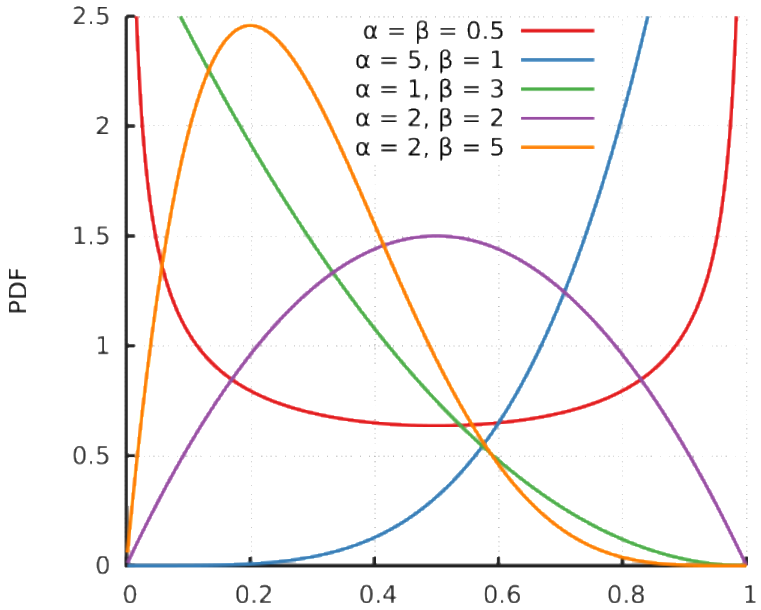
ϵ-greedy v.s. UCB 对比

- 相同的10臂老虎机问题



Thompson Sampling

- 基于贝叶斯统计的方法
  - 和UCB一样，都是有区分性的探索方法
  - UCB：显示的计算不确定性；TS：用抽样
- 设每个老虎机的奖励要么是0，要么是1
  - 估计每一个老虎机的出钱概率是多少



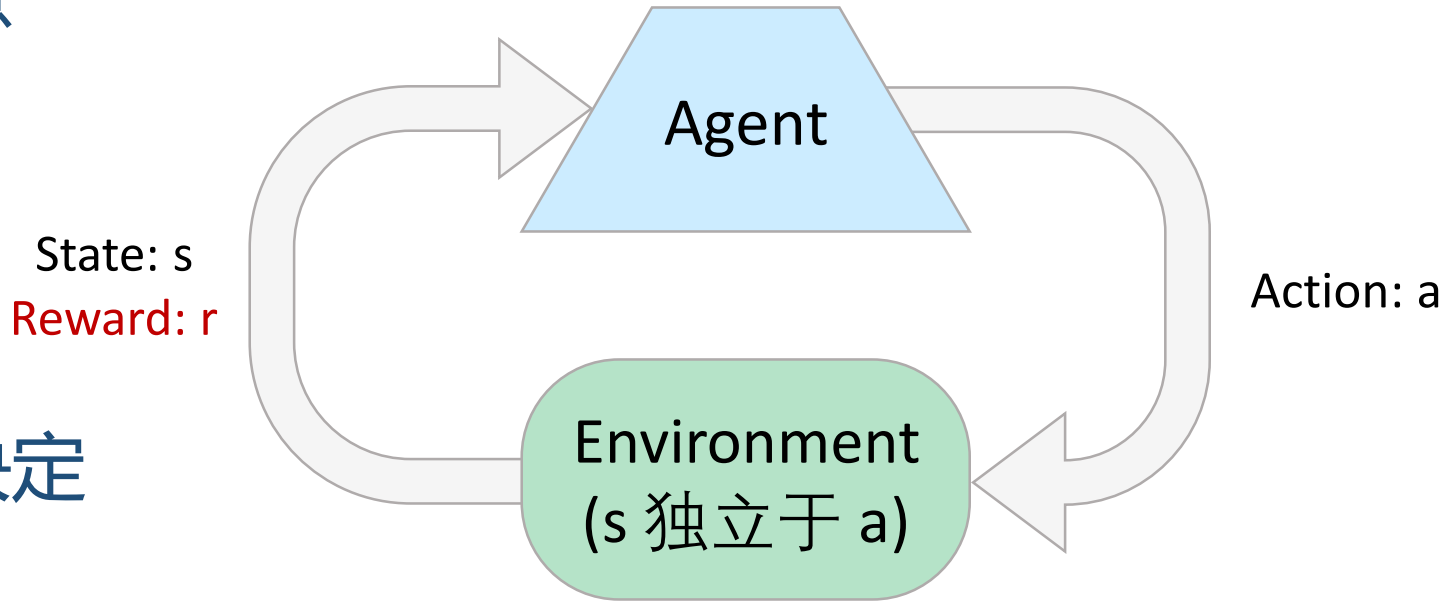
- 怎么估计？
  - 认为每一个老虎机的出钱概率服从Beta分布
  - Beta分布：一枚硬币，抛了a次正面，b次反面，则抛一次为正面的概率为Beta(a, b)

Thompson Sampling

- 对于每个老虎机：
  - 有a次给了奖励，b次没有给奖励
  - 那么认为其出钱概率服从beta分布：$p \sim Beta(a, b)$
  - “概率的概率分布”
- 要做决定的时候：
  - 从每一台老虎机的beta分布中sample一个它的出钱概率
  - 选择出钱概率最大的那个老虎机
  - 观察是否给了奖励：a+1或者b+1，更新它的beta分布

Contextual Bandits

- 能从环境观察到更多信息
  - 状态信息
  - 奖励分布和状态相关
  - 动作任然不会影响状态



- 动作价值不再只由动作决定
  - $Q(s,a)$
  - 状态-动作价值

- 环境是非静态

应用举例

- 老虎机
  - 能观察到额外一盏灯，不同颜色代表老虎机们中奖概率变化了
  - 策略需要根据灯颜色不同而改变

- 广告显示 / 新闻个性化推送
  - 目标：最大化CTR
  - 能得知当前访问用户的信息
  - 男性少年更偏爱苹果产品而不是退休金计划

评测结果

- 学习篮：bandits算法
- 部署篮：根据当时学到的价值预估做greedy策略

algorithm	size = 100%		size = 30%		size = 20%		size = 10%		size = 5%		size = 1%	
	deploy	learn	deploy	learn	deploy	learn	deploy	learn	deploy	learn	deploy	learn
ϵ-greedy	1.596 0%	1.326 0%	1.541 0%	1.326 0%	1.549 0%	1.273 0%	1.465 0%	1.326 0%	1.409 0%	1.292 0%	1.234 0%	1.139 0%
ucb	1.594 0%	1.569 18.3%	1.582 2.7%	1.535 15.8%	1.569 1.3%	1.488 16.9%	1.541 5.2%	1.446 9%	1.541 9.4%	1.465 13.4%	1.354 9.7%	1.22 7.1%
ϵ-greedy (seg)	1.742 9.1%	1.446 9%	1.652 7.2%	1.46 10.1%	1.585 2.3%	1.119 −12%	1.474 0.6%	1.284 −3.1%	1.407 0%	1.281 −0.8%	1.245 0.9%	1.072 −5.8%
ucb (seg)	1.781 11.6%	1.677 26.5%	1.742 13%	1.555 17.3%	1.689 9%	1.446 13.6%	1.636 11.7%	1.529 15.3%	1.532 8.7%	1.32 2.2%	1.398 13.3%	1.25 9.7%
ϵ-greedy (disjoint)	1.769 10.8%	1.309 −1.2%	1.686 9.4%	1.337 0.8%	1.624 4.8%	1.529 20.1%	1.529 4.4%	1.451 9.4%	1.432 1.6%	1.345 4.1%	1.262 2.3%	1.183 3.9%
linucb (disjoint)	1.795 12.5%	1.647 24.2%	1.719 11.6%	1.507 13.7%	1.714 10.7%	1.384 8.7%	1.655 13%	1.387 4.6%	1.574 11.7%	1.245 −3.5%	1.382 12%	1.197 5.1%

Contextual Bandits的定位

- 介于RL和MAB之间
  - RL: 动作改变状态，奖励由状态，动作决定；$R(s,a), Q(s,a)$
  - CB: 动作不改变状态，奖励由状态，动作决定；$R(s,a), Q(s,a)$
  - MAB:动作不改变状态，奖励只由动作决定；$R(a), Q(a)$

- 需要学习策略 $\pi(s)$

LinUCB

- 与Function Approximator的思想相同，用函数近似期望收益
  - 例如线性函数：t步时做动作a的期望收益 $E\left[r_{t,a}\middle|x_{t,a}\right]=x_{t,a}^{T}\theta_{a}^{*}$
  - $x_{t,a}^{T}$：做a动作时的状态特征
  - θ_{a}^{*}：特征权重
- 对于每一个动作，学习一个这样的估计函数
- 当面临新的状态s的时候
  - 先估计每个动作的期望收益$E\left[r_{t,a}\middle|x_{t,a}\right]=x_{t,a}^{T}\theta_{a}^{*}$
  - 再根据UCB算法挑一个动作做（综合考虑探索和贪心）

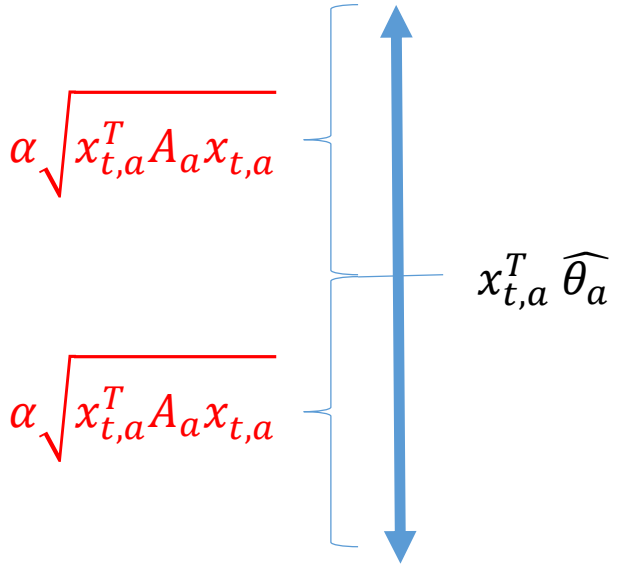
LinUCB

- 训练估价函数就是线性回归的过程
- D_a：做动作a时的状态特征
  - 所有推送了广告a的人的{年龄，性别，国籍……}
- b_a：反馈（标签）
  - 点了a这个广告没有？ 1=点了，0=没有
- Ridge回归即可得到基于数据的解析解θ_a
  - $\min_{\theta} \left|\left|D_a\theta_a - b_a\right|\right|^2 + \left|\left|\theta\right|\right|_2^2$
  - $\widehat{\theta_a} = (D_a^T D_a + I)^{-1} D_a^T b_a$

LinUCB

- 以至少$1-\delta$的概率
  - $\left|x_{t,a}^T\,\widehat{\theta_a}-E\left[r_{t,a}\middle|x_{t,a}\right]\right|\le\alpha\sqrt{x_{t,a}^T(D_a^TD_a+I)^{-1}x_{t,a}}$，$\alpha=1+\sqrt{\frac{\ln\left(\frac{2}{\delta}\right)}{2}}$
  - 误差范围上限
- 动作选取和UCB类似
  - $a_t:=argmax_a(x_{t,a}^T\,\widehat{\theta_a}+\alpha\sqrt{x_{t,a}^TA_ax_{t,a}})$

均值　　标准方差



  - $A_a=(D_a^TD_a+I)^{-1}$

小结

- 多臂老虎机问题
- 探索方法
  - ϵ-greedy
  - UCB
  - Thompson Sampling
- Contextual Bandits
  - LinUCB
- 应用

Q&A

Thank You

滴滴
滴滴一下 美好出行

www.xiaojukeji.com

扫钉钉群二维码，加入我们