探索利用困境综述

Minghuan Liu

2019 年 5 月 9 日

1 引言

探索与利用困境 (exploration and exploitation dilemma)]Robbins (1985) 是强化学习中的重要问题。问题来源于强化学习问题本身的特点: 为了获取更高的回报，agent 必须根据以往的经验在不同的状态选择合适的动作，也就是利用 (exploit) 已知的动作来获取更高的分数; 而要发现这些动作，则需要进行必要的探索 (exploration)。此困境即在于二者均需要进行不断的试错。每个算法都必然需要包含这两部分。如果一个算法只包含探索，那么有效的动作将永远都不会得到利用，我们将永远不会得到最优解; 而如果一个算法只包含利用，那么就会成为贪心算法，不会发现更好的全局最优解。因此，在有限的时间内，如何使得算法达到更好的收敛效果，是探索与利用困境的本质所在。

在强化学习算法中，常常使用随机策略，这些随机策略耦合了探索和利用，其中探索部分交给了随机变量。常用的随机策略包括以下几种:

1. $\epsilon - greedy$:

$$\pi(a|s) = \begin{cases} 1 - \epsilon + \frac{\epsilon}{|A(s)|}, & if a = argmax_a Q(s,a) \\ \frac{\epsilon}{|A(s)|}, & if a \neq argmax_a Q(s,a) \end{cases} \tag{1.1}$$

ps: 含义就是以 ϵ 的概率随机抽取动作空间中的动作。

2. Botzman 策略:

$$\pi(a|s) = \frac{exp(kQ(s,a))}{\sum_{a'} exp(kQ(s,a'))} \tag{1.2}$$

3. 高斯策略 (常用于连续系统):

$$\pi(a|s) = \mu_\theta + \epsilon, \epsilon \sim N(0, \sigma^2) \tag{1.3}$$

上面列举的这些策略本质上大都是在贪婪策略或确定性策略上面加上一个随机无向的噪声，通过噪声进行探索的，因此也被称为抖动策略 (dithering strategy)。抖动策略的好处是: (i) 计算容易，不需要复杂的计算公式; (ii) 能保证充分探索。但坏处就在于: (i) 需要大量探索，数据利用率低;(ii) 需要无限长时间。其实不难理解，因为抖动策略在智能体行动的所有时刻对于动作空间的探索都是随机的，不会随着经验的增加而改变。因此，抖动策略并没有很好的平衡探索与利用，在实现复杂任务时往往需要更长的训练时间和海量的数据。

因此，我们需要更好的策略来平衡探索和利用。下面，本文将由浅入深介绍探索与利用在强化学习领域的进展，以及取得更好平衡效果的算法的思想和数学证明。

2 多臂赌博机问题

多臂赌博机 (multi-armed bandit)Robbins (1985) 是一个简化的强化学习环境，它的问题如下: 假设有 n 个选择，每个选择都会以一定的概率分布带来收益，我们要在有限的时间步内获得更多的预期收益，应当在每个时间步执行哪个动作。很明显，我们希望执行预期回报最多的动作，但我们需要先探索哪个动作具有最大的回报，再进行利用。由于在选择某个动作时无法同时解决探索和利用，这二者便常常被称为一对矛盾或困境。探索和利用的选择取决于动作值估计的精度，不确定性以及剩余的时间步。

对于每个动作，我们称它可以获得的平均回报 $Q(a) = \mathbb{E}[r|a]$ 为这个动作的值 (action value)。因此，若我们使用贪心策略，每次选择当前已知的值最高的动作，此即利用; 但若我们不按照贪心策略选择，则为探索。设想当我们选择动作时，贪心策略的动作值是可以大概率肯定的，而某些动作可以获得差不多甚至更高的动作值，但却具有很高的不确定性。因此，若剩余的时间步尚多时，我们最好选择探索非贪心的动作以获取更高的回报。

多臂赌博机和一般的强化学习问题的最大不同有两点: (i) 多臂赌博机的目标策略只需要在单一情况中找到最优动作，这个情况要么是静态的，要么是动态地随时间变化的; 而一般的强化学习要求找到到最优策略则是要找

到一种可以在不同状态下找到最优动作的映射，要进行更复杂的序列决策; (ii) 多臂赌博机对于动作的选择只影响即时的回报，而强化学习问题在每个时刻的动作的选择会影响后续的回报。对于多臂赌博机来说，有很多方法对探索和利用问题做出了很好的平衡，但却是包含了很多先验知识和假设。尽管在真正的强化学习中如何平衡探索和利用是一个艰巨的挑战，但多臂赌博机为我们提供了解决问题的清晰的简明形式。因此，在多臂赌博机中，我们更关注理论上的完整性和收敛性。一个从概率角度全面审视该问题的 overview 可见Berry and Fristedt (1985)。

多臂赌博机包含多种变体，例如：Contextual bandit，Adversarial bandit，Infinite-armed bandit，Non-stationary bandit，Dueling bandit，Collaborative bandit，Combinatorial bandit 等。本文只讨论最简单的多臂赌博机问题。

2.1 基本方法

本小节主要介绍一部分在思路上最简单且最基础的方法。这些方法由于其简单和实用性，在实际应用中经常被单独或者组合使用。其中包含 $\epsilon - greedy$ 方法、$softmax$ 策略 (亦为 Botzman 策略) 以及最优初始值方法。这些基本方法主要是针对动作值来进行动作的选择。

我们可以利用 MonteCarlo 方法对动作值函数进行估计。在每次选择动作之后，记录动作和其得到的回报。因此，对于动作 a 的值函数的估计为:

$$Q(a_t) = \frac{R_1 + R_2 + ... + R_{K_a}}{N_t(a)} \tag{2.1}$$

其中 $K_a = N_t(a)$ 为动作 a 被执行的次数。由大数定理，当实验的次数足够多时即当 $N_t(a) \to \infty$ 时，$Q_t(a)$ 将会收敛至真值 $Q^*(a)$。若我们采用此方法直接计算均值，则我们储存的数据将会随时间步的进行而增大，因此在第 k 步我们可以采取增量方法计算该值:

$$Q_{k+1} = Q_k + \frac{1}{k}[R_k - Q_k] \tag{2.2}$$

这个更新规则的一般形式如下:

$$NewEstimate \leftarrow OldEstimate + StepSize[Target - OldEstimate] \tag{2.3}$$

其中 $[Target - OldEstimate]$ 即代表了估计的误差，指示了更新的方向，而 $StepSize$ 则是步长参数 (TD 学习、MonteCarlo 学习都用到了这个形式的更

新公式)。我们令步长参数 $\alpha_k(a)$ 表示第 k 次动作选择的步长参数，则若其满足如下限制，则可以保证 $Q_t(a)$ 的收敛性。

$$\sum_{k=1}^{\infty} \alpha_k(a) = \infty \quad and \quad \sum_{k=1}^{\infty} \alpha_k^2(a) = \infty \tag{2.4}$$

由此可知，对于样本平均来说，此时 $\alpha_k = \frac{1}{k}$，满足式 Eq. (2.4)。而对于 $\alpha_k = \alpha$ 即为常数的情形，其收敛性无法得到保证，但正适合对于非静止的环境，即各个动作的回报真值分布随时间的变化而变化，此时我们估计的 $Q(a_t)$ 将和近期的回报的关联更大。由于调参的复杂性和缓慢收敛的问题，参数序列的形式通常不会被使用。

为了衡量策略对探索和利用的平衡效果，我们定义 regret 来表示每一步平均的可能的机会损失:

$$l_t = \mathbb{E}[V^* - Q(a_t)] \tag{2.5}$$

其中

$$V^* = Q^*(a^*) = \max_{a \in \mathcal{A}} Q^*(a)$$

表示最优的动作值。定义 total regret 表示总的机会损失:

$$\begin{aligned} L_t &= \mathbb{E}[\sum_{\tau=1}^{t} V^* - Q(a_\tau)] \\ &= \sum_{a \in \mathcal{A}} \mathbb{E}[N_t(a)](V^* - Q(a)) \\ &= \sum_{a \in \mathcal{A}} \mathbb{E}[N_t(a)]\Delta_a \end{aligned} \tag{2.6}$$

其中 $\Delta_a = (V^* - Q(a))$ 表示该动作和最优动作之间价值的 gap，$N_t(a)$ 表示动作 a 被选择的次数。很明显，最大化累计回报的目标实际上等价于最小化 total regret，而且我们希望一个好的算法可以让 gap 大的动作的选择次数更小。对于这样一个指标，一个很明显的问题是我们在实际问题中的 gap 难以得到，因为 V^* 未知而且 $Q^*(a)$ 同样需要估计，但指标仍旧可以对策略的评价具有很好的指导意义。我们在讨论的时候，通常假设 Δ_a 是已知的定值。

若随着实验的进行，total regret 是线性增长，则说明在每一个时间步算法对于各个动作选择的概率并没有改变，也即每一步的 regret 没有变化，没有更好的利用已探索的信息，因此不能算是一个好的算法；一个可以更好解决探索与利用的策略应当具有亚线性的 total regret，也即随着实验的进

行，每个时间步的 regret 是逐渐降低的，算法的选择会逐渐摒弃较少可能具有较大回报的算法。

Lai and Robbins (1985) 证明了，对于所有可能成为一个最优的次优策略的 total regret，需要至少拥有对数增长形式的渐进下界:

$$\lim_{t\to\infty} L_t \leq 8\ln t \sum_{a|\Delta_a>0} \frac{\Delta_a}{KL(R^a||R^{a^*})} \tag{2.7}$$

2.1.1 $\epsilon - greedy$

$\epsilon - greedy$ 仅利用动作值进行动作选择，其公式已在 Eq. (1.1) 给出，其本质仍是抖动策略。作为对贪婪策略的改进，随着游戏轮数的进行，$\epsilon - greedy$ 可以保证所有动作都被无限次采样，从而保证 $Q(a)$ 的收敛性。这意味着选择最优动作的概率可收敛至大于 $1-\epsilon$，即接近确定。然而，这些只是渐近保证，而实际有效性是无法得到保证的。很明显，$\epsilon - greedy$ 每次对于动作的选择的概率都至少为 $\frac{\epsilon}{\mathcal{A}}$，所以 regret l_t 至少为 $\frac{\epsilon}{\mathcal{A}}\sum_{a\in\mathcal{A}}\Delta_a$，因而是具有线性的 total regret。有趣的是，尽管在理论上，$\epsilon - greedy$ 不是最优方法，但Kuleshov and Precup (2014) 用大量实验数据说明了，实际中 $\epsilon - greedy$ 往往可以比一些复杂方法取得更好的结果。

2.1.2 $softmax$ 策略 (Botzman 策略)

由于 $\epsilon - greedy$ 在探索时对所有动作的选择都采取相等的可能，即使对于已经确定会获得很差回报的动作同样如此，因此并不能很好的利用已有的信息。我们希望对于具有更高回报可能的动作采取更大的可能进行探索，对剩下的动作利用权重进行排序并选择。一个自然的想法就是采用 $softmax$ 方法 based on Luce (2012)，利用 Botzman 分布或称为 Gibbs 分布，类似于 Eq. (1.2)，在时刻 t 选择动作 a 的概率如下式:

$$\mathbb{P}_t(a) = \frac{exp(Q_t(a)/\tau)}{\sum_{i=1}^{n} exp(Q_t(i)/\tau}} \tag{2.8}$$

其中 τ 是被称为温度的超参数，是一个正数，可以用来操控各个动作被选择的概率。若 τ 较大，则所有动作可近似视为等可能选择; 若 $\tau \to 0$，则成为贪心策略。一个适合的 τ 应当和 $Q(a)$ 在一个数量级。

容易知道 $softmax$ 策略的 regret 为 $\sum_{a\in\mathcal{A}} P_t(a)\Delta_a$ 也为定值，因而也是具有线性的 total regret。其实本质上，$softmax$ 策略和 $\epsilon - greedy$ 都是改进后的贪婪策略。

$softmax$ 策略和 $\epsilon - greedy$ 孰优孰劣依赖于具体任务和人为调参等因素。$softmax$ 策略的主要困难在于参数的设置：通常来说，设置 ϵ 要比设置 τ 更容易，因为后者需要具有知道可能动作值等先验知识，而这些知识通常难以获得。目前关于此方面的研究也因此而停滞，有人认为将动作值通过指数转换为概率这一做法本身就是错的，但此方法可以带来一些启发。

2.1.3 乐观初始化

初始的动作值 $Q_1(a)$ 往往会影响对动作值的估计。若采用样本平均的方式，则只要所有动作都至少被选择一次，初始值造成的偏差都会消失; 但对于步长参数为常数的情形，则偏差的影响一直存在。初始值可以作为一种有效的先验知识存在。

可以利用“乐观”的初始值设置来鼓励探索Sutton and Barto (2018)，$Q_1(a) = r_{max}$。例如，如果所有赌博机的回报都服从方差为 1，均值为 0 的高斯分布，则当我们设置初始值为 5 时，该值是明显偏大的。当某一动作被选择时，获得更小的回报，此时智能体会感到“失望”，进而尝试其他的动作，这会使得在初期，动作空间可以被充分探索。对静态场景来说，这是一个简单有效的可以鼓励探索的方法，但却并不是一个好的通用方法，比如对非静态场景并不适合。实际上，初始时刻只会出现一次，因此并没有必要花费过多精力在上面。

本质上，初始化只是鼓励了智能体在开始尽量多探索，并没有改变动作选择策略，所以使用乐观初始化技巧后并不会改变线性的 total regret。

2.2 复杂方法

上一节所述的基本方法尽管简单，并且都具有线性的 total regret，但在实际应用中往往是有效的，因为我们对于 $Q(a)$ 的估计是随着实验的进行而逐渐准确。因而对于简单的问题，这些方法往往都是具有一定的指导作用的，可以在有限的时间内获得较优解。然而随着估计逐渐准确，这些方法的 total regret 将永远局限在线性增长，因此无法获得更高的预期回报。对于解决探索与利用的平衡问题还有更好更复杂的解决办法。而这些方法可以分为频率论观点的方法，以及贝叶斯方法。频率论方法通常将对未知参数的估计视为其真实值，而贝叶斯方法中则将估计作为从先验分布中导出的后验分布。两类方法都可对问题进行较好的求解，但这两种方法是相关的，且其中的研究方法可以互相利用。

2.2.1 Decaying $\epsilon - greedy$

首先介绍频率论的方法。前面所述的对动作值 $Q(a)$ 进行估计，并利用 regret 对策略的优劣进行衡量，即是频率论的思想。在这些方法里利用样本值对 $Q(a)$ 进行点估计，并利用 $Q(a)$ 决定策略。

对于一般的 $\epsilon - greedy$ 算法，随机采样参数 ϵ 为常值，因此当探索充分的时候，动作选择策略将仍然会拥有和刚开始探索一样的探索功能。这实际上是一种浪费，因为我们希望智能体在探索充分后可以充分利用以取得最大收益，后面继续的探索将不再具有意义。因此一种直观的想法就是可以令 ϵ 随着时间的进行而降低。例如，考虑如下的变化规律:

$$\begin{aligned} & c > 0 \\ & d = \min_{a|\Delta_a > 0} \Delta_a \\ & \epsilon_t = \min\left\{1, \frac{c|\mathcal{A}|}{d^2 t}\right\} \end{aligned}$$

下面考虑该策略的 regret，仅考虑 $\epsilon < 1$ 的情况:

$$l_t = \frac{c}{d^2 t} \sum_{a \in \mathcal{A}} \Delta_a \tag{2.9}$$

则 total regret L_t 将为:

$$\begin{aligned} L_t &\approx \int_t l_t dt \\ &= \int_t \frac{1}{t} \frac{c}{d^2} \sum_{a \in \mathcal{A}} \Delta_a \\ &= \frac{c}{d^2} \sum_{a \in \mathcal{A}} \Delta_a \ln t \end{aligned} \tag{2.10}$$

从以上简单证明不难看出，L_t 将呈对数规律增长 (实际上，由于 t 是离散的，L_t 能视为近似呈对数规律增长)，这样的 $decaying \epsilon_t - greedy$ 算法的 total regret L_t 将为亚线性，随着时间步的增加，L_t 的增加将逐渐减小，Cesa-Bianchi and Fischer (1998) 详细说明了这一点。但这样的 ϵ 的变化规律需要事先知道关于 gap 的信息。我们需要考虑寻找具有亚线性的 regret 且不需要有关回报知识的方法。对于 Botzman 策略，Cesa-Bianchi and Fischer (1998) 也证明了对于采用衰减的参数 τ 可以给出对数增长的上限，这里不再展开介绍。

需要注意的是，有关 $\epsilon - greedy$ 的方法仍有一些扩展的研究工作，例如基于学习进度进行参数自调整Tokic (2010); Tokic and Palm (2011)，以及与环境相关 (contextual) 的方法Bouneffouf et al. (2012) 等。他们都在 $\epsilon - greedy$ 基础上进行了相关改进，本文不再展开介绍。

2.2.2 Upper Confidence Bound(UCB) 方法

另一种思路是用区间估计来估计动作值的不确定程度，根本思路是对不确定程度表示“乐观”，利用动作值的不确定程度来刺激探索的力度，因为不确定的动作将更有可能取得更高的动作值。若某一动作会以较高的不确定程度取得较低的回报，则该动作肯定不会被选择; 但若某动作的不确定程度较低，意味着该动作有可能取得较高回报，此时应该对该动作多执行探索。因此，区间估计的方法被提出。

区间估计会给出相应动作值的置信区间，例如，以 95% 的置信度该动作获得的回报在 9 到 11 之间，而不是确定地给出该动作值的单点估计。此时下一步的策略选择应该是选择动作值的置信区间拥有较高上限的动作，鼓励了智能体对不确定性较高的动作的探索。但是区间估计方法在实际应用中受到区间估计的负责统计方法的限制，一些假设也常常不能被满足。

尽管区间估计的方法存在一些问题，但这一想法是合理有效的。因此，这一思想延伸出了上置信界即 UCB 方法族Auer et al. (2002)。UCB 方法对每个动作值会估计一个上置信限度 $\hat{U}_t(a)$，使得 $Q(a) \leq \hat{Q}_t(a) + \hat{U}_t(a)$ 具有相当大的概率。而我们希望不确定程度随着动作尝试次数的增加而降低，即 $\hat{U}_t(a) \propto \frac{1}{N_t(a)}$。UCB 的策略选择具有最大上置信界的动作，即:

$$a_t = \arg\max_{a \in \mathcal{A}} \hat{Q}_t(a) + \hat{U}_t(a) \tag{2.11}$$

此问题转换为如何求得上置信限度 $\hat{U}_t(a)$。利用 Hoeffding 不等式:

Hoeffding 不等式定理: 令 X_1,X_2,...,X_t 是在 [0,1] 上的服从独立同分布的随机变量，再令 $\overline{X_t} = \frac{1}{\tau}\sum_{\tau=1}^{t} X_\tau$ 为样本均值, 则:

$$\mathbb{P}(\mathbb{E}[X] > \overline{X_t} + u) \leq e^{-2tu^2} \tag{2.12}$$

利用该定理，我们可以得到:

$$\mathbb{P}(Q(a) > \hat{Q}_t(a) + \hat{U}_t(a)) \leq e^{-2N_t(a)U_t^2(a)} \tag{2.13}$$

令 $e^{-2N_t(a)U_t^2(a)} = p$，即 $Q(a)$ 取值大于上置信界的概率不超过 p，则可解出:

$$U_t(a) = \sqrt{\frac{-\log p}{2N_t(a)}} \tag{2.14}$$

我们希望随着时间步的增加，动作的不确定程度可以降低，即 p 逐渐降低，可令 $p = t^{-4}$，则可以得到:

$$U_t(a) = \sqrt{\frac{2\log t}{N_t(a)}} \tag{2.15}$$

由此我们即可得到 UCB1 算法:

$$a_t = \arg\max_{a \in \mathbb{A}} Q(a) + \frac{2\log t}{N_t(a)} \tag{2.16}$$

UCB1 算法被证明具有对数增长形式的 total regret。Auer et al. (2002) 证明了在任意时间步，UCB1 的算法都具有对数增长形式的上界 (而非渐进上界)，即:

$$L_t \leq \left[8\ln t \sum_{a:Q(a)<Q(a^*)} \left(\frac{1}{\Delta_a}\right)\right] + \left(1 + \frac{\pi^2}{3}\right)\left(\sum_{a \in \mathcal{A}} \Delta_a\right) \tag{2.17}$$

UCB 系列算法利用严谨的数学证明给预期的 regret 提供了强有力的保证，i.e.，提供了明确的渐进上界 $O(\log n)$。由于在原理上的严谨证明和其简单优雅的实现形式，UCB 算法被视为可以较好解决 (solve) 多臂赌博机问题的解法。

2.2.3 Pursuit 算法

以上算法均基于每个动作的估计动作值来选择动作，而也可以利用学习自动机的方法来维持一个显式的策略，例如 pursuit 算法Thathachar (1984)。其选择每个动作的概率会分别根据实验结果进行更新。最简单的 pursuit 算法按照均匀分布为策略赋予初始值，即 $p_i(0) = 1/k$，然后在每个时间步 t 按照如下规则更新:

$$p_i(t+1) = \begin{cases} p_i(t) + \beta(1 - p_i(t)), & if i = argmax_j \hat{\mu}_j(t) \\ p_i(t) + \beta(0 - p_i(t)), & otherwise \end{cases} \tag{2.18}$$

其中 $\beta \in (0,1)$ 称为学习率。Thathachar (1984) 在学习自动机中利用 PAC 的相关理论给出 pursuit 算法的收敛性的证明，即该算法最终会收敛至最优动作。由于证明较为复杂，故本文不再详细讨论。

实际上，学习自动机可以作为强化学习的特例，而多臂赌博机也可以作为学习自动机的特例，此部分不是本文重点，故不再展开。但需要注意的是 pursuit 算法和强化学习问题中的 actor-critic 算法其实是相关的，actor-critic 中对策略的更新参考了 pursuit 算法的更新方式。

2.2.4 POKER 策略

此外，Vermorel and Mohri (2005) 提出了 POKER(Price of Knowledge and Estimated Reward) 策略，其基本思想是为动作可能 获得的知识的价格 (price)，来衡量动作的不确定程度。这种被称为"信息价值 (Infomation Value)"的概念已得到了深入研究。在多臂赌博机文献中，它有时被称为"探索奖励 (exploration bonuses)"Dearden (2000); Meuleau and Bourgine (1999)，目的在于以相同量纲量化与回报不确定性。该方法也考虑了动作值的分布，希望可以从已经观察到的情况估计未观察到的属性。此外，剩余轮数 (称为 horizon) 也得到了充分的考虑。一个 POKER 策略的价值衡量为:

$$p_{a_t} = Q_{a_t} + \mathbb{P}[q_a \geq \hat{V}_t^* + \sigma_{\mu_t}]\sigma_{\mu_t} H \tag{2.19}$$

其中 Qa_t 是对动作值的当前估计，而 $\hat{V}_t^*$ 则表示对当前最优动作值的估计，$\sigma_{\mu_t} = \mathbb{E}\left[V^* \hat{V}_t^*\right]$ 表示预期回报增益 (expected reward improvement)，V^* 表示最优动作值。H 是剩余轮数，称为视野 (horizon)。在实际中 σ_{μ_t} 将近似求解。因此，整个第二项将表示该动作可能带来的信息或知识增益 $\mathbb{P}[q_a \geq \hat{V}_t^* + \sigma_{\mu_t}]$ 表示信息增益大于预期增益的概率。

Vermorel and Mohri (2005) 证明了 POKER 是一种 zero-regret 的策略，即随着剩余轮数的增加，regret 将逐渐趋于 0。这样利用知识获取 (Knowledge Acquirement) 来引导探索的想法很受欢迎，在很多强化学习问题中也利用类似的方式进行探索。

2.2.5 概率匹配

以上方法均没有给定动作的回报分布。而若每组动作的回报分布 $\mathbb{P}[R]$ 是已知的，我们可以利用该先验知识进行利用，并计算选择任何动作后的后验奖励概率即 $\mathbb{P}[R|h_t]$ 来进行探索，h_t 为历史动作-奖励序列。Scott (2010)

从 Beyesian Modeling 的角度对多臂赌博机问题进行了研究，并提出了概率匹配 (Probability Matching) 的方法。这样的思路启发了很多可以归类为贝叶斯方法的研究 (Beyesian Bandit)。但是，贝叶斯方法的缺陷在于需要更准确的先验知识来引导正确的结果。在贝叶斯方法中，可以利用 Beyesian regret 来衡量 bayesian optimality Kaufmann et al. (2012)。

概率匹配是贝叶斯方法中比较重要的方法之一。其思想即是根据该动作可能成为最优动作的概率来选择动作，即 $\pi(a|h_t) = \mathbb{P}[Q(a) > Q(a'), \forall a' \neq a|h_t]$。在实际使用中，概率匹配常常利用 Thompson Sampling Thompson (1933); Russo et al. (2018) 实现。

$$\begin{aligned}\pi(a|h_t) &= \mathbb{P}[Q(a) > Q(a'), \forall a' \neq a|h_t] \\ &= \mathbb{E}_{\mathcal{R}|h_t}\left[\mathbb{I}(a = \arg\max_{a \in A} Q(a))\right] \\ &= \int \left[\mathbb{I}(\mathbb{E}(r|a^*, x, \theta) = \max_{a \in A} \mathbb{E}(r|a, x, \theta)\right] P(\theta|h_t) d\theta\end{aligned} \tag{2.20}$$

在实验的每一轮迭代中，我们可以得到后验 $P(\theta|h_t)$，进而通过 sampling 的方式得到一个 θ^*，然后我们选择动作 $a^* = \arg\max_{a \in \mathcal{A}} \mathbb{E}[r|\theta^*, a, x]$。因此，Thompson Sampling 的思路是智能体在每一轮中都可以根据当前知识来随机采样参数，然后根据参数采取行动。然而很多时候，从后验分布进行维护和采样是计算繁琐的，因此使用时常常使用近似采样技巧 Russo et al. (2018)。

Thompson Sampling 和 UCB 算法都可以从理论上以最优策略解决多臂赌博机问题，因此，可以将 UCB 中的 regret 转化为 Thompson sampling 的 Beyesian regretRusso and Van Roy (2014) 以及统一 regret 分析方法 Russo and Van Roy (2013)。

2.2.6 Beyesian UCB

Kaufmann et al. (2012) 提出 Beyesian UCB 方法，将 UCB 的思想应用到贝叶斯方法中，而且在某些情况下 (Bernoulli rewards) 证明可以达到 Eq. (2.7) 中要求的下界。只考虑动作之间相互独立的情况，一种通用的 Beyesian UCB 方法的形式为:

$$a_t = \arg\max_{a \in \mathcal{A}} \mathcal{Q}\left(1 - \frac{1}{t(\log n)^c}, \lambda_a^{t-1}\right) \tag{2.21}$$

其中 $\mathcal{Q}(t,\rho)$ 是分布 ρ 的 quantile function，满足 $\mathbb{P}_\rho(X \leq \mathcal{Q}(t,\rho)) = t$(即此时该分布的上 α 分位点 $\alpha = t$)，$\lambda_a^t(1 \leq j \leq K)$ 为动作值 $Q(a)$ 的均值 μ_a 的后验分布，可由动作 a 的边际后验分布 π_a^t 得到。需要注意的是，在贝叶斯策略中，$\Pi^t = \prod_{a\in\mathcal{A}} \pi_a^t$ 表示 t 轮游戏后参数 θ 的后验分布，而 Π^0 则表示初始的先验分布。每次选择动作 a 后，便会更新 a 的后验分布。而动作 a 的 Beyesian update 可由下式给出:

$$\pi_a^t(\theta_j) \propto v_{\theta_a}(X_t)\pi_a^{t-1}(\theta_a) \tag{2.22}$$

其中，$v_{\theta_a}(X_t)$ 表示在时间步 t 出现的结果为 $X_t = Y_{a,t} \in 0,1$(即动作 a 是否被选取) 的边际分布。

具体举例来说明这一策略。假设回报服从均值为 μ_a，方差为 σ^2 的高斯分布，令先验为 $\pi_a^0 = \frac{1}{\sigma^2}$，则 Beyesian UCB 策略为:

$$a_t = \arg\max_{a\in\mathcal{A}} \frac{S_a(t)}{N_a(t)} + \sqrt{\frac{S_a^{(2)}(t)}{N_a(t)}}\mathcal{Q}\left(1 - \frac{1}{t(\log n)^c}, \mathcal{T}(N_a(t)-1)\right) \tag{2.23}$$

其中，$T(k)$ 表示自由度为 k 的 Student-t 分布。可以看到，Beyesian UCB 在形式上和 UCB 算法是一致的。Beyesian UCB 可以直观理解为利用置信区间对不确定性进行衡量，而置信区间随着时间步 t 的增大而不断缩小。

2.2.7 Gittin Indices

Gittins (1979) 这份著名工作中提出利用 Gittin Indices 来求解某一类特定的 bandit problem 的最优策略的方法，是一类基于贝叶斯模型的方法。Gittin Indices 又称为 Bayes-adaptive RL。随着实验进行，回报分布不断演进成为不同的信息状态，将问题转化为 Bayes-adaptive MDP 问题。给定先验，每个可能的事件链的回报和分布就可以计算。但是，该方法需要巨大的计算成本，因为结果树的增长非常迅速，需要使用更有效的方法来近似计算。

但由于理论和计算的复杂程度使得该方法并没有广泛的适用性，也阻碍了该理论的进一步发展，但仍有工作Brezzi and Lai (2002) 提出了 Gittin Indices 的近似方法，减小了计算的复杂度。除此之外，Gittin Indices 也存在三个理论上的问题Scott (2010)，分别是: 不完全学习 (incomplete learning)，Gittin 策略最终将收敛至为唯一动作，但仍存在一定的次优概率; Gittin

Indices 要求每个动作都以独立的参数定义，对于联合定义的动作则无法达到最优解。因此，Gittin Indices 仅适合具有确定性最优策略的问题，而不适合具有混合策略的问题; 若定义的折扣方式 (discounting scheme) 不是几何形式 (geometric) 的则 Gittin Indices 无法收敛。

2.3 小结

目前在多臂赌博机问题中关于探索与利用的研究已经很多，本文不可能一一讲解。但本文从不同角度，给出了大量经典方法及其原理，具有较好的通用性和实用性。事实上，多臂赌博机不仅仅作为强化学习的简化形式，有很多现实问题已经以多臂赌博机建立了模型，并利用相关方法取得了较好的结果，例如在线广告的竞价预测Ikonomovska et al. (2015) 以及资源分配等。

3 一般强化学习问题

多臂赌博机作为简化版的强化学习环境，可以为探索与利用问题的研究提供更简单和清晰的形式。有一些在多臂赌博机中得到研究的算法也可以应用到一般的强化学习问题上来。例如 $\epsilon - greedy$，基于 UCB 的 UCT 算法等。强化学习问题是一个马尔科夫链 (MDP)，较单一状态下多次进行动作选择的多臂赌博机问题相比，强化学习要学习的是一个最优映射，可以从不同状态中选择最优的动作，并使得这样的状态-动作序列可以获得更高的回报。

在强化学习框架下，问题将更复杂。相较于多臂赌博机，强化学习问题对于最优解和状态动作值 $Q(s,a)$(类似于多臂赌博机中的 $Q(a)$) 的估计将更困难。而在很多强化学习问题中，更多时候面对的问题甚至不在于哪个算法可以更快达到收敛，而是是否可以收敛的问题。有时候是因为任务过于复杂，导致状态空间和动作空间过于庞大，而在这种情况下，仅靠增强策略的探索能力仍旧无法收敛；此外，在某些场景中环境中的回报是稀疏的，甚至有时没有外部回报，在这种情况下，无法仅靠回报或者 $V(S)$ 及 $Q(s,a)$ 来指导探索和利用。

在复杂的环境中，如果仍然采用抖动策略进行随机探索，效果将会很差。因为抖动策略依赖的实际上是随机行为。如果偶然导致奖励，则这些对应的行为就会被强化，并且智能体未来会更倾向于采取这些有利的行为。当

奖励足够密集或者状态空间和动作空间不太大时，随机行动可以带来合理概率的奖励，因而比较奏效。但是，复杂的环境往往需要很长的特定动作的序列才能获取奖励，这样的序列随机发生的可能性非常低，仅靠随机策略难以发现较优解。此外，在一般的强化学习问题中往往无法找到最优解，很多时候只能找到次优解，而更多时候都会陷入局部最优。在这样的情况下，我们需要策略拥有不依赖外界的更好的探索能力。

在强化学习的问题下，对动作的探索变成了对动作和状态的探索，或者是对轨迹的探索。在多臂赌博机中，探索的根本思路在于以更高的概率选取在当前更有可能获得更高回报的动作：一种方案是直接对策略，也即对选择动作的概率进行调整，另一种方式则像 UCT 一样，将当前对动作价值的估计加上额外项，并采用贪心策略。我们同样可以将这一的思想借鉴过来，鼓励智能体探索更有可能获得更高回报的轨迹。由于轨迹不仅包含了动作 a，也包含了状态 s，因此在探索时，要考虑两方面的探索。二者可以分别考虑，也可以同时考虑。通常情况下，由于状态空间的维度往往远大于动作空间，因此状态数量也往往要大于动作数量，因此探索问题的重心通常放在对状态 s 的探索中。

由于强化学习问题中的动作空间有时会涉及连续空间，因此此时无法使用简单的贪心策略。但对于离散动作空间来说，$\epsilon - greedy$，$softmax$ 策略都是可以应用的简单探索方式。但不管针对什么样的动作空间，我们都可以通过调整对状态值 V 或状态-动作值 Q 来引导策略探索可能更优的轨迹，并且往往是直接在回报 r 中增添额外项。这样的方式在强化学习问题中成为了一个研究分支，称为内在奖励（Intrinsic Reward）或者探索奖励（Exploration Bonus），而内在奖励的方法则可以通过各种方式获得。此外，还有一些其他的方法不属于此类，但比较零散，难以归类，因此本文将这些方法放在一个小节中一一讨论。

3.1 基于内在奖励的方法

我们首先要知道什么是内在奖励。我们知道在强化学习的问题定义下，存在 r 作为环境给智能体的反馈或回报，而智能体学习的目标正是最大化这样的累计回报，因此这样的回报是促使智能体学习的重要信号。这样的从环境中获得的奖励信号通常被称为是外在奖励，而内在奖励则是由智能体自身给出。本质上，基于内在奖励的方法可以看做是一种区间估计的方式，给出当前动作可能成为最优动作的奖励，因此早期很多工作都深受此想法

的影响。

基于内在奖励的方法的研究由来已久，并且已经有详细的综述文献给予了总结Oudeyer and Kaplan (2009); Schmidhuber (2010)。基于内在奖励的方法的核心在于如何计算或者生成内在奖励，例如希望其与状态的不确定性/惊喜程度/新鲜程度相关，这通常要借助一些手段或者技术来实现。因此本节将主要从内在奖励的不同组成来对这些方法进行分类简述。

3.1.1 基于计数的方法

基于计数的方法思路较为简单，即通过维护访问过的状态的数量，并利用计数值计算内在奖励。

最早，Wiering and Schmidhuber (1998) 提出利用计数来提供额外的奖励，称作“Directed Exploration”，这样的奖励函数定义了“有趣（interesting）”的状态。其定义了两种奖励函数，一种称作 recency-based，定义为 $R = \frac{-t}{K_T}$，其中，t 是当前的时间步，K_T 代表常数，则该奖励将会鼓励智能体选择最近没有被选择的动作；另一种奖励函数称作 frequency-based，定义为 $R = \frac{-C_{s_t}(a_t)}{K_C}$，其中，$C_{s_t}(a_t)$ 代表在当前状态下执行各动作的次数，K_C 代表常数，因此该奖励会鼓励智能体探索使用最不频繁的动作。该研究基于探索奖励建立了模型，并利用模型计算动作价值的区间估计，进而求解最优策略。此工作即是著名的 MBIE（model based internal estimation）。后来Strehl and Littman (2004, 2008) 重新描述了 MBIE 方法，重新给出了信赖区间的定义，在该定义中，信赖域的公式与 $n(s,a)$，即状态-动作对的访问次数有关，进而在Strehl and Littman (2005) 中证明了 MBIE 是 PAC 的，并给出了 MBIE 估计的回报上界为 $\tilde{R}(s,a) = \hat{R}(s,a) + \sqrt{\frac{\ln 2/\delta_R}{2n(s,a)}}$，其中，$\delta_R$ 是根据 Hoeffding bound 不等式，至少有概率 $1-\delta_R$ 的概率使回报落在指定区间。Strehl and Littman (2008) 中还提出了 MBIE-EB 方法，其对价值函数的估计在回报的基础上加入了探索奖励项即 $R(s,a) + \frac{\beta}{\sqrt{n(s,a)}}$。这些工作和 R_{max}Brafman and Tennenholtz (2002)，E^3Kearns and Singh (2002) 等方法被称为 PAC-MDP 方法。这些方法的主要思路是，如果一个智能体已经观察到某个状态-动作对足够多次，那么可以使用例如 Hoeffding bound 等偏差不等式，确保经验估计可以接近真实环境的动力学模型。但是，如果没有观察到状态-动作对足够多次，则假设它具有非常高的价值，这将鼓励智能体尝试没有观察到足够多次的状态-动作对，直到最终我们有一个适当准确的系统模型，这种技术一般被称为面对不确定性的乐观主义。

和 Gittin Indices 类似，在强化学习问题中同样有人采用贝叶斯方法来对环境建模。这些方法提出了信念状态（belief state）的概念。最优贝叶斯策略选择的动作不仅基于它们将如何影响环境的下一个状态，还基于它们将如何影响下一个信念状态；而且，由于更好地了解 MDP 通常会带来更大的未来回报，贝叶斯策略很自然会在平衡探索和利用上进行权衡。但是这些方法通常是难以求解的。这些方法通常被称为 Bayesian Reinforcement Learning 方法。Kolter and Ng (2009) 结合了贝叶斯方法与 PAC-MDP 方法，对贝叶斯价值函数的估计在回报的基础上加入了探索奖励项即 $R(s,a)+\frac{\beta}{1+n(s,a)}$，并且证明了尽管这样的算法无法求解出最优策略，但可以接近最优策略。

这样基于计数的方法明显只适用于低维离散动作和状态空间，无法解决高维或者连续空间下的强化学习问题。基于此，Tang et al. (2017) 提出了利用哈希函数 $\phi(s_t)$ 缩小状态空间，以计算内在奖励，计算 $R(s,a)+\frac{\beta}{\sqrt{n(\phi(s))}}$。而Bellemare et al. (2016) 则利用概率分布计算的密度函数（density model）推出伪计数（pseudo-count），计算内在奖励，其形式为 $R(x,a)+\frac{\beta}{\sqrt{\hat{n}(x)+0.01}}$，这里的 x 是游戏画面的某个像素点，则 $\hat{n}(x)$ 计算的是某处像素点的伪计数。并通过证明其和信息增益（information gain）的关联证明了基于计数方法中采用的信赖域和传统基于内在奖励方法的关联性，本质是统一的。Ostrovski et al. (2018) 借鉴前面工作的思想，利用 PixelCNN 来计算伪计数，其使用的奖励形式为 $R(x,a)+\frac{1}{\sqrt{\hat{n}_n(x)}}$。

3.1.2 基于信息论的方法

很多基于内在奖励的探索方法从信息论的角度出发，用各种指标衡量新状态或者新动作对信息的贡献程度，或者降低不确定性的程度，鼓励智能体探索包含更多信息的轨迹，很多方法和变分推断相结合。Mohamed and Rezende (2015) 提出使用信息论中的互信息（mutual information）来计算所谓的 empowerment，通过选择可以最大化 empowerment 的动作，智能体希望达到可以到达更大数量的未来状态。作者提出，智能体可以选择只最大化该指标，因为环境中的奖励不一定够好；智能体也可以将 empowerment 作为 reward shaping 的组成部分。empowerment 的计算公式可以写作：

$\epsilon(s)=\max_w \mathcal{I}^w(a,s'\mid s)=\max_w \mathbb{E}_{p(s'|a,s)w(a|s)}\left[\log\left(\frac{p(a,s'|s)}{w(a|s)p(s'|s)}\right)\right]$。

Still and Precup (2012) 同样使用互信息作为探索奖励，该研究提出，可以将动作视为状态的一种表示。因此可以将状态映射到动作视为一种有损压缩，若一组状态共享相同的最优动作，则该动作可以被视为这组该状态“聚

类”的压缩表示，因此在具有相同回报的策略中，要找到最紧凑的策略，即压缩了最多状态信息的动作，因此要最小化动作与状态的互信息，目标是在最大化收益的同时，最小化 $I_q^\pi(A_t, X_t) = \sum_{x\in\mathcal{X}} \sum_{a\in\mathcal{A}} \pi(a \mid x) p^\pi(x) \log \left[\frac{\pi(a|x)}{p^\pi(x)}\right]$。

Houthooft et al. (2016) 利用智能体对动力学模型的信心（belief）的信息增益（information gain）来进行探索，同样是基于好奇心驱动的思想，鼓励智能体选择会带来更惊奇（surprise）的状态的动作，其计算的是在已知历史轨迹 ξ_t 后加入动作 a_t 和状态 s_{t+1} 后的信息增益作为内在奖励，即 $R(s_t, a_t) + \eta D_{KL}[p(\theta \mid \xi_t, a_t, s_{t+1} \parallel p(\theta \mid \xi_t)]$。

3.1.3 基于预测误差的方法

基于预测误差的方法通常会建立环境的一步预测模型，来对未来状态进行估计，并以实际情况与估计的差别程度作为对智能体的探索奖励，由此鼓励智能体探索更少见的状态。Pathak et al. (2017) 使得好奇心驱动的探索再次成为研究热点，该研究提出使用深度网络建立环境的一步预测模型，对环境状态的特征表示进行预测，并利用其与真实的环境状态特征表示的差值范数作为惊喜的奖励，其对内在奖励的定义为：$\frac{\eta}{2} \parallel \hat{\phi}(s_{t+1}) - \phi(s_{t+1}) \parallel_2^2$，其中 $\hat{\phi}(s_{t+1}) = f(\phi(s_t), a; \theta_F)$。

目前大部分探索的 Benchmark 环境是以 MonteZuma 为代表的一系列稀疏回报的游戏环境，而在这些环境中，目前公认的 Sota 是Burda et al. (2019) 提出的 RND 方法。该方法同样是由好奇心驱动，但其通过最小化一个由状态作为输入的可训练的神经网络与另一个随机初始化的固定网络的差值，并由该差值提供内在奖励，其观点在于少见的状态将不会训练的很好，因此会带来很大的差值。该方法的内在奖励可以写为 $\hat{f}(x; \theta) - f(x)$。

3.2 其他方法

这部分算法由于较为零散（也可能是调研程度不够），难以划分成大类，因此按方法整理在此。

3.2.1 UCT

UCT 算法的全称是 Upper Confidence Bounds for Tree 也即 UCB for Tree Kocsis and Szepesvári (2006)，本质上是一种蒙特卡洛树搜索方法（MCTS），与 UCB 算法结合起来。UCT 将 MCTS 的每次结点选

择的问题视为多臂赌博机问题，并利用 UCB 的策略选择结点，即 $a_i = \arg\max v_i + C \times \sqrt{\frac{\ln N}{n_i}}$，其中，$v_i$ 是节点估计的值，n_i 是节点被访问的次数，而 N 则是其父节点已经被访问的总次数，C 是可调整参数。UCT 在原理上有良好的性质，MCTS 估计会在搜索的开始不大可靠，而最终会在给定充分的时间后收敛到更加可靠的估计上，在无限时间下能够达到最优估计。并且，UCT 在实践中也具有良好的效果。有了 UCT 之后，围棋 AI 的战力才大大提升。

3.3 小结

强化学习问题中的探索与利用本质上是针对 MDP 的探索与利用，且研究也由来已久，本文只从个人调研的角度，对这些方法进行了简单的分类，可能分类仍有不准确之处，或者可以从不同的角度进行归纳。在前强化学习时代，即没有引入深度学习方法时，探索与利用问题的重点仍旧在于如何提高算法的采样效率，引导智能体尽早探索可能取得更优解的策略，更多地从信息和不确定性的角度来计算和衡量；但从深度强化学习时代开始，更多的关注点被放在稀疏回报情况下的怎样的探索策略可以更快收敛，以及怎样尽可能收敛到更优解，甚至有人研究不存在环境奖励情况下的探索问题（我认为这样的问题不再适合强化学习来解决），更多的从改变仅有环境奖励的角度出发。探索问题是一个涉及到强化学习本质的问题，也是需要理论和实践相结合的方向，需要有像 UCB 一样理论优美的解决方案，但目前来说，由于状态空间和动作空间的巨大，要做到仍旧比较困难。

References

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems*, pages 1471–1479, 2016.

Donald A Berry and Bert Fristedt. Bandit problems: sequential allocation of

experiments (monographs on statistics and applied probability). *London: Chapman and Hall*, 5:71–87, 1985.

Djallel Bouneffouf, Amel Bouzeghoub, and Alda Lopes Gançarski. A contextual-bandit algorithm for mobile context-aware recommender system. In *International Conference on Neural Information Processing*, pages 324–331. Springer, 2012.

Ronen I Brafman and Moshe Tennenholtz. R-max-a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3(Oct):213–231, 2002.

Monica Brezzi and Tze Leung Lai. Optimal learning and experimentation in bandit problems. *Journal of Economic Dynamics and Control*, 27(1): 87–108, 2002.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *ICLR*, 2019.

Nicolo Cesa-Bianchi and Paul Fischer. Finite-time regret bounds for the multiarmed bandit problem. In *ICML*, pages 100–108. Citeseer, 1998.

Richard W Dearden. *Learning and planning in structured worlds*. PhD thesis, University of British Columbia, 2000.

John C Gittins. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 148–177, 1979.

Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. In *NeurIPS*, pages 1109–1117, 2016.

Elena Ikonomovska, Sina Jafarpour, and Ali Dasdan. Real-time bid prediction using thompson sampling-based expert selection. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1869–1878. ACM, 2015.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On bayesian upper confidence bounds for bandit problems. In *Artificial Intelligence and Statistics*, pages 592–600, 2012.

Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. *Machine learning*, 49(2-3):209–232, 2002.

Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *European conference on machine learning*, pages 282–293. Springer, 2006.

J Zico Kolter and Andrew Y Ng. Near-bayesian exploration in polynomial time. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 513–520. ACM, 2009.

Volodymyr Kuleshov and Doina Precup. Algorithms for multi-armed bandit problems. *arXiv preprint arXiv:1402.6028*, 2014.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

R Duncan Luce. *Individual choice behavior: A theoretical analysis*. Courier Corporation, 2012.

Nicolas Meuleau and Paul Bourgine. Exploration of multi-state environments: Local measures and back-propagation of uncertainty. *Machine Learning*, 35(2):117–154, 1999.

Shakir Mohamed and Danilo Jimenez Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. In *NeurIPS*, pages 2125–2133, 2015.

Georg Ostrovski, Marc G Bellemare, Aaron van den Oord, and Rémi Munos. Count-based exploration with neural density models. *ICML*, 2018.

Pierre-Yves Oudeyer and Frederic Kaplan. What is intrinsic motivation? a typology of computational approaches. *Frontiers in neurorobotics*, 1:6, 2009.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *CVPRW*, pages 16–17, 2017.

Herbert Robbins. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*, pages 169–177. Springer, 1985.

Daniel Russo and Benjamin Van Roy. Eluder dimension and the sample complexity of optimistic exploration. In *Advances in Neural Information Processing Systems*, pages 2256–2264, 2013.

Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.

Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, Zheng Wen, et al. A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96, 2018.

Jürgen Schmidhuber. Formal theory of creativity, fun, and intrinsic motivation (1990–2010). *IEEE Transactions on Autonomous Mental Development*, 2(3):230–247, 2010.

Steven L Scott. A modern bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, 26(6):639–658, 2010.

Susanne Still and Doina Precup. An information-theoretic approach to curiosity-driven reinforcement learning. *THEOR BIOSCI*, 131(3):139–148, 2012.

Alexander L Strehl and Michael L Littman. An empirical evaluation of interval estimation for markov decision processes. In *16th IEEE International Conference on Tools with Artificial Intelligence*, pages 128–135. IEEE, 2004.

Alexander L Strehl and Michael L Littman. A theoretical analysis of model-based interval estimation. In *Proceedings of the 22nd international conference on Machine learning*, pages 856–863. ACM, 2005.

Alexander L Strehl and Michael L Littman. An analysis of model-based interval estimation for markov decision processes. *Journal of Computer and System Sciences*, 74(8):1309–1331, 2008.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. # exploration: A study of count-based exploration for deep reinforcement learning. In *Advances in neural information processing systems*, pages 2753–2762, 2017.

MAL Thathachar. A class of rapidly converging algorithms for learning automata. In *IEEE International Conference on Cybernetics and Society*, pages 602–606, 1984.

William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25 (3/4):285–294, 1933.

Michel Tokic. Adaptive ε-greedy exploration in reinforcement learning based on value differences. In *Annual Conference on Artificial Intelligence*, pages 203–210. Springer, 2010.

Michel Tokic and Günther Palm. Value-difference based exploration: adaptive control between epsilon-greedy and softmax. In *Annual Conference on Artificial Intelligence*, pages 335–346. Springer, 2011.

Joannes Vermorel and Mehryar Mohri. Multi-armed bandit algorithms and empirical evaluation. In *European conference on machine learning*, pages 437–448. Springer, 2005.

Marco Wiering and Jürgen Schmidhuber. Efficient model-based exploration. In *Proceedings of the Sixth International Conference on Simulation of Adaptive Behavior: From Animals to Animats*, volume 6, pages 223–228, 1998.