大状态空间中的离线强化学习理论

1 离线强化学习与时域诅咒

1.1 离线强化学习介绍

强化学习（Reinforcement Learning, RL）虽然名字中有“学习”一词，但实际上，它的历史更多是依赖基于采样的方法来完成各种计算任务. Richard Sutton 曾说过：

> “[RL] 领域的许多研究其实并不涉及真正的学习，而更多是在已知环境模型下进行规划.”

这在一些基于经验的（深度）强化学习研究中尤为常见，算法通过与模拟环境的交互，采样数据轨迹，以寻找近似最优的策略. 我们的目标非常明确：在给定的计算资源下找到一个好的策略，这里的计算资源不仅包括算法本身的计算成本，还包括从模拟器中采样数据的成本.

尽管这种方法在处理复杂的仿真任务时取得了显著的成果，但它也逐渐显现出不足之处，特别是在一些潜在的现实应用中. 例如，适应性临床试验、推荐系统与客户关系管理、在线教育等领域. 这些场景的共同特点是：人类患者、用户或学生是“环境”的一部分，而针对人类心理或生物学特征，往往很难构建出准确的模拟器. 因此，现实世界就成为了我们唯一可以依赖的环境，无论是实际的患者、真实的用户，还是在教育系统中的学生.

这就带来了另一个问题. 大多数基于模拟的强化学习算法都是*在线*（online）的：算法在与环境交互、收集数据的过程中，会不断尝试新的决策并观察结果. 然而，在学习的初期，由于算法对环境认知不足，这种探索可能导致不良甚至危险的结果. 在模拟环境中，错误决策的代价并不严重，但在现实环境中，尤其当人类成为环境的一部分时，这种探索可能带来严重的后果.

针对这一挑战，**离线强化学习**（Offline RL）应运而生. 它依靠预先收集的数据进行学习，无需在训练过程中与环境实时交互. 对于现实世界的应用，这些数据通常来自系统正常运行时的日志，且不需要改变原有的决策流程. 这种方式不仅更安全，也更适合那些无法频繁试错的高风险场景.

1.2 回顾：监督学习的理论保障

离线强化学习提倡一种类似于*监督学习*（Supervised Learning）的数据驱动范式. 在本节中，我们将简要回顾监督学习的基本理论框架，并通过重要性采样方法来建立其在离线强化学习中的对应关系.

考虑如下的监督学习的分类问题（其中的符号仅仅在此处讨论监督学习和强化学习的比较中用到，后面的部分并不涉及）：我们从某个分布中*独立同分布*（i.i.d）采样数据集 $\{(X_i, Y_i)\}_{i=1}^n$，其中 $X_i \in \mathcal{X}$ 为输入的*特征*（feature），例如图像，$Y_i \in \{-1, 1\}$ 为二元的*标签*（label），例如图像中是否有猫. 我们的目标是要让模型学得一个映射 $h : \mathcal{X} \to \{-1, 1\}$，称作*假设*（hypothesis）；然

后，我们要从未见数据 X 中尽可能准确地预测其对应的标签 Y. 我们定义风险（risk）为 $\mathcal{R}(h) := \mathbb{E}[\mathbb{I}[Y \neq h(X)]]$，来评判模型的表现.

监督学习理论分析的关键在于，给定一个假设 h，其风险 $\mathcal{R}(h)$ 可以通过数据有效地做估计，其经验估计值为 $\widehat{\mathcal{R}}(h) := \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}[Y_i \neq h(X_i)]$. 这是因为 $\widehat{\mathcal{R}}(h)$ 可以看成是 i.i.d 随机变量 $\{\mathbb{I}[Y_i \neq h(X_i)]\}_{i=1}^{n}$ 的平均值，大数定律（Law of Large Numbers）告诉我们这在统计上是无偏的（unbiased），且 Hoeffding 不等式告诉我们，其和真实期望之间的差距有如下界限，且该界限以至少 $1-\delta$ 的概率成立：

$$\left|\widehat{\mathcal{R}}(h) - \mathcal{R}(h)\right| \leq \sqrt{\frac{1}{2n}\log\frac{2}{\delta}}. \tag{1}$$

公式 (1) 是监督学习中一切理论的基础，我们通过经验风险最小化（Empirical Risk Minimization, ERM）的方式对模型进行训练，也即选取：

$$h^* = \arg\max_{h\in\mathcal{H}} \widehat{\mathcal{R}}(h),$$

其中 $\mathcal{H}$ 是所有可能的假设集合，称作假设空间（hypothesis space）. 而且，对于有限假设空间 $\mathcal{H}$ 而言，我们在公式 (1) 的基础上可以通过 union bound 给出一个泛化误差界，误差界限依赖于 $\log|\mathcal{H}|$，也即假设空间的大小.

在实际应用中，我们可能尝试不同的训练算法，并依赖从保留验证数据集（holdout dataset）中估计的 $\widehat{\mathcal{R}}(h)$ 来进行模型的评估和选择. 这样的流程称作 “ERM-then-evaluate”.

1.3 通过重要性采样的异策略评估

为了在离线强化学习中也建立类似 “ERM-then-evaluate” 的流程，核心问题在于：如何估计候选策略的性能，并建立类似于公式 (1) 的保证. 在强化学习中，学习算法输出的是决策策略（policy），而这些策略通常与最初用于收集数据的策略不同. 也就是说，我们需要评估一个策略的表现，而数据却是由另一个策略收集的. 这种场景被称为**异策略评估**（Off-Policy Evaluation, OPE）问题.

为了介绍 OPE 方法，我们考虑无限时间折扣的（infinite-horizon discounted）**马尔可夫决策过程**（Markov Decision Process, MDP），这是一个标准的强化学习设定. 一个 MDP $(\mathcal{S}, \mathcal{A}, P, R, \gamma, d_0)$ 由以下元素确定：

- 状态空间（state space）$\mathcal{S}$，表示环境中所有可能的状态集合；
- 动作空间（action space）$\mathcal{A}$，表示所有智能体（agent）在每个状态下可以选择的动作集合；
- 转移函数（transition dynamics）$P : \mathcal{S}\times\mathcal{A} \to \Delta(\mathcal{S})$，其中 $\Delta(\mathcal{S})$ 表示状态空间上的概率分布，$P(s' \mid s, a)$ 表示从状态 s 出发，采取动作 a，转移到状态 s' 的概率；
- 奖励函数（reward function）$R : \mathcal{S}\times\mathcal{A} \to [0, R_{\max}]$，$R(s,a)$ 表示从状态 s 出发，采取动作 a 得到的奖励（也称收益），这里我们假设奖励是确定性且非负的；
- 折扣因子（discount factor）$\gamma \in [0,1)$，表示未来奖励的衰减程度，取值范围为 $[0,1)$；
- 初始状态分布（initial distribution）d_0，表示系统开始时各个状态的概率分布.

方便起见，我们假设状态空间 $\mathcal{S}$ 和动作空间 $\mathcal{A}$ 均是离散且有限的；不过，许多结论也可以推广到大状态空间的情况，甚至在适当的测度理论下，也可以扩展到连续状态空间.

一个平稳且随机的（stationary and stochastic）策略 $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ 定义了在每个状态下选择动作的概率分布，$\pi(a \mid s)$ 表示在状态 s 上会采取动作 a 的概率. 一个策略会生成一个轨迹（trajectory），定义为：

$$\tau := (s_0, a_0, r_0, s_1, a_1, r_1, \ldots, s_{H-1}, a_{H-1}, r_{H-1}, \ldots).$$

这样一个轨迹 τ 通过如下的过程生成：刚开始由初始分布 d_0 得到一个初始状态 $s_0 \sim d_0$；在第 t 轮时，我们观测到状态 s_t，并通过策略 π 采取一个动作 $a_t \sim \pi(\cdot \mid s_t)$，环境给予我们对应的奖励 $r_t = R(s_t, a_t)$，进而进行状态转移 $s_{t+1} \sim P(\cdot \mid s_t, a_t)$；如此循环往复.

由此可见，该过程本质上基于采样（sampling）而进行的，而这一采样过程理论上可以无限进行（$t \to \infty$）. 为了方便起见，我们假设在最多 H 步之后，过程会进入一个自循环状态（self-loop state），此时奖励为 0. 这样的状态被称为吸收状态（absorbing state），它标志着系统的终止，例如：患者完成了一个多阶段的治疗程序，用户完成了一个多回合的聊天对话，系统结束了一个多轮的推荐流程，等等. H 也被称作时域（horizon）.

我们对吸收状态的假设，主要是为了后面的重要性采样考虑. 如果并不存在这样的吸收状态，我们可以在一个有效时域（effective horizon）的位置来截断无限长的轨迹，取成 $H = \mathcal{O}(1/(1-\gamma))$. 这是由于折扣因子的存在，使得 H 步之后的奖励对估计误差的影响微不足道了.

在给定 MDP 模型的情况下，我们还需要定义：

- **估计量** $J(\pi)$. 给定一个待估计的策略 π，其通常称作目标策略或评估策略（target/evaluation policy）. 我们通过期望折扣回报（expected discounted return）来评估策略 π，定义为：

  $$J(\pi) := \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r_t \right],$$

  其中 $\mathbb{E}_\pi[\cdot]$ 表示在策略 π 下轨迹分布的期望，我们同样使用 $\Pr_\pi[\cdot]$ 则表示对应的概率. 这是强化学习的学习目标，衡量了一个策略在期望意义下能够获得的总奖励. 当我们考虑过程在 H 步终止时，求和 $\sum_{t=0}^{\infty}$ 可以替换为 $\sum_{t=0}^{H-1}$，因为在 $t = H$ 之后，所有奖励都为 0.

- **数据集** $\mathcal{D}$. 在 OPE 中，一条数据轨迹又称为一个回合（episode），其通过另一种策略 π_D 收集的，其通常被称作行为策略或日志策略（behavior/logging policy）. 具体地，数据集 $\mathcal{D}$ 可以表示为：

  $$\{\tau^{(i)} := (s_0^{(i)}, a_0^{(i)}, r_0^{(i)}, \ldots, s_{H-1}^{(i)}, a_{H-1}^{(i)}, r_{H-1}^{(i)})\}_{i=1}^n,$$

  其中所有动作 $a_t^{(i)}$ 都遵循策略 π_D，也即 $a_t^{(i)} \sim \pi_D(\cdot \mid s_t^{(i)})$.

简而言之，OPE 的关键是：在已知数据是由行为策略 π_D 生成的前提下，准确评估目标策略 π 的期望回报 $J(\pi)$.

一个重要的估计方法是**重要性采样**（Importance Sampling, IS），也称作重要性加权或逆倾向评分（Inverse Propensity Score, IPS），可以利用单条轨迹 τ 对 $J(\pi)$ 进行无偏估计：

$$\mathrm{IS}(\tau) = \left(\prod_{t=0}^{H-1} \frac{\pi(a_t \mid s_t)}{\pi_D(a_t \mid s_t)} \right) \left(\sum_{t=0}^{H-1} \gamma^t r_t \right). \tag{2}$$

我们需要保证：对于所有状态-动作对 (s, a)，如果有 $\pi_D(a \mid s) > 0$，就有 $\pi(a \mid s) > 0$（也即我们确保 $\pi/\pi_D < \infty$）. 这样便有

$$\mathbb{E}_{\pi_D}[\mathrm{IS}(\tau)] = \mathbb{E}_\pi \left[\sum_{t=0}^{H-1} \gamma^t r_t \right],$$

也即 IS 估计器是无偏的. 这是因为重要性权重可以表示成

$$\prod_{t=0}^{H-1} \frac{\pi(a_t \mid s_t)}{\pi_D(a_t \mid s_t)} = \prod_{t=0}^{H-1} \frac{\pi(a_t \mid s_t) P(s_{t+1} \mid s_t, a_t)}{\pi_D(a_t \mid s_t) P(s_{t+1} \mid s_t, a_t)} = \frac{\Pr_\pi[\tau]}{\Pr_{\pi_D}[\tau]},$$

这样就在期望意义下将轨迹分布从 π_D 策略中采样转换成从 π 策略中采样.

注意到我们的条件式 $\pi/\pi_D < \infty$，这被称作*覆盖条件*（coverage condition）. 在动作空间较小的场景下，当策略 π_D 被适当随机化，且对所有动作都分配了非零概率时，这一条件通常可以满足.

公式 (2) 中的 IS 是最基本的形式，但也可以通过多种方式进行改进. 例如，可以使用依赖于数据的归一化因子，形成*加权* IS（weighted IS），或者结合*控制变量*（control variates），得到*双稳健估计器*（doubly robust estimator）. 尽管这些改进方法在实际中表现更好，但它们本质上仍然与公式 (2) 中的 IS 共享类似的样本复杂度.

给定 n 条轨迹 $\{\tau^{(i)}\}_{i=1}^n$，对 $J(\pi)$ 的经验估计可以写作：

$$\hat{J}_{\mathrm{IS}}(\pi) = \frac{1}{n} \sum_{i=1}^{n} \mathrm{IS}(\tau^{(i)}).$$

这可以看成是 n 个 i.i.d 随机变量 $\mathrm{IS}(\tau^{(i)})$ 的平均值，因此可以为其提供类似于监督学习中公式 (1) 的理论保障. 值得注意的是，Hoeffding 不等式依赖于随机变量的取值范围，在公式 (1) 中，每个随机变量均在 $[0,1]$ 之间，而在 IS 中，$\sum_{t=0}^{H-1} \gamma^t r_t$ 的取值范围为 $[0, V_{\max}]$，其中 $V_{\max} := R_{\max}/(1-\gamma)$. 此外，我们还需要对重要性权重进行界定. 假设存在常数 $C_{\mathcal{A}}$，使得：

$$\max_{s,a} \frac{\pi(a \mid s)}{\pi_D(a \mid s)} \le C_{\mathcal{A}},$$

那么便有 $\prod_{t=0}^{H-1} \frac{\pi(a_t|s_t)}{\pi_D(a_t|s_t)} \le (C_{\mathcal{A}})^H$. 这样一来，我们用 Bernstein 不等式，进一步用到重要性权重的“低方差”特性（具体而言，我们用到了如下性质：*考虑任意非负函数 ρ 满足 $\mathbb{E}[\rho] = 1$，我们有 $\mathbb{E}[\rho^2] \le \|\rho\|_\infty$，也即方差的增长与取值范围呈线性关系*），可以得到如下保障：

$$|\hat{J}_{\mathrm{IS}}(\pi) - J(\pi)| \lesssim V_{\max} \sqrt{\frac{(C_{\mathcal{A}})^H}{n} \log \frac{1}{\delta}}, \tag{3}$$

其中 $\lesssim$ 表示 LHS $= \mathcal{O}$(RHS)，忽略了常数因子.

借助 IS 估计器，我们便可以在离线强化学习中建立与监督学习相似的优化框架. 对于一个策略的集合 Π，若希望优化其中的策略，则可以直接求解：

$$\arg\max_{\pi \in \Pi} \hat{J}_{\mathrm{IS}}(\pi).$$

这一过程能够近似找到集合 Π 中表现最优的策略（通过公式 (3) 给出理论保障）. 学到的策略随后可以在保留验证数据集上再次使用 IS 进行评估，以完成模型选择和测试.

1.4 时域诅咒

基于 IS 的框架与监督学习之间存在严格的对应关系，这种对应带来了许多不错的性质. 然而，这一框架在*多步强化学习*（multi-step RL）中存在一个关键缺陷，即样本复杂度中出现了指数项 $(C_{\mathcal{A}})^H$，即使是在评估单个策略时（见公式 (3)），这一项也不可避免. 如果 $C_{\mathcal{A}}$ 接近于 1，则该指数项可能不会带来太大影响，例如当 $C_{\mathcal{A}} = 1 + \mathcal{O}(1/H)$ 时，$(C_{\mathcal{A}})^H$ 将是一个常数. 然而，这种情况限制了目标策略 π 必须非常接近行为策略 π_D，否则样本复杂度将指数增长.

为了更加清晰地说明这一问题，我们考虑一个简单的 MDP 问题，只有一个状态和两个动作，且状态始终转移回自身. 行为策略 π_D 采用均匀的随机策略来收集数据. 从直观上看，即使是一个适度大小的数据集，也应该包含足够的信息来准确评估任何策略. 然而，当我们使用 IS 评估一个确定性的策略 π 时，估计的方差仍然会呈指数增长.

由于目标策略 π 是确定性策略，例如始终选择动作 a_1，那么 $\pi(a_t \mid s_t) = \mathbb{I}\{a_t = a_1\}$；而行为策略 π_D 是均匀随机的，也即 $\pi_D(a_t \mid s_t) = 0.5$. 那么，当轨迹中每个时间步 t 都选择了动作 a_1 时，其重要性权重为 $\prod_{t=0}^{H-1} \frac{1}{0.5} = 2^H$；但如果某个时间不选择了动作 a_2，则权重为 0，因为目标策略绝不会选择 a_2. 在这种情况下，只有概率为 2^{-H} 的轨迹会被赋予巨大的权重 2^H，而其他轨迹的权重为 0. 这就导致 IS 估计器的方差会呈指数增长，也即 $\mathrm{Var} \sim (2^H)^2 = 4^H$.

这一现象显然与直觉相悖——对于这样一个简单的问题，学习过程本不应该需要与时域 H 呈指数关系的数据量! 这种现象称作**时域诅咒**（The Curse of Horizon），这是现代强化学习遇到的一个重要挑战. 这意味着，在多步强化学习中，随着时域 H 的增长，学习和评估的样本复杂度呈指数增长.

2 值函数估计

让我们回到一个状态和两个动作的简单 MDP 的例子：是什么让它看起来较为容易？答案在于马尔可夫性（Markovianity）. 尽管在该 MDP 中存在指数级数量的可能动作序列，但它们都会经过相同的唯一状态！也正因如此，如果一种方法如果能够充分利用马尔可夫性，那么只要数据中覆盖（cover）了目标策略访问的状态和动作，该方法就可以获得稳健的理论保证.

与之相比，重要性采样（IS）方法完全忽略了状态的概念，而是将所有问题都视为一个指数级增长的树，在每个时间步上都会不断分叉，导致估计的方差快速膨胀. 这种处理方式虽然使得 IS 可以直接应用于部分可观测（partially observed）的领域，但对于标准的 MDP 问题来说，其效率并不理想.

如何更加有效地利用马尔可夫性？对于应具有有限且较小的状态空间而言，一种直接的方法是分别估计每个状态-动作对的转移概率和奖励，然后在估计得到的 MDP 中计算策略的回报. 在表格型问题中，这种方法被称作确定性等效方法（certainty equivalence, CE）. 在这种方法中，我们首先建立一个环境的近似模型，然后基于该模型计算最优策略的回报.

然而，当状态空间很大时，这种方法的计算代价将变得不可行. 因此，核心问题变成了：我们如何利用马尔可夫性，使得这种估计方法能够扩展到大状态空间?

为了解决这个问题，我们首先定义强化学习中的一个核心概念：值函数（value function）. 给定策略 $\pi : \mathcal{S} \to \Delta(\mathcal{A})$，我们可以定义 Q-值函数为：

$$Q^\pi(s,a) := \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s, a_0 = a \right],$$

也即：当轨迹从状态-动作对 (s,a) 开始，并且从 $t=1$ 及之后按照策略 π 进行决策时，期望获得的折扣回报. 一旦 Q^π 已知，我们就可以计算目标函数 $J(\pi)$ 为

$$J(\pi) = \mathbb{E}_{s\sim d_0}[Q^\pi(s,\pi)],$$

其中我们简写 $f(s,\pi)$ 为 $\mathbb{E}_{a\sim\pi(\cdot|s)} f(s,a)$，表示通过策略 π 采样的动作的期望函数值. 此外，$\mathbb{E}_{d_0}[\cdot]$ 可以从一组 i.i.d 的状态样本中估计，例如我们可以从初始状态分布 d_0 采样一组数据 $\{s_0^{(i)}\}_{i=1}^n$ 对 d_0 进行估计. 为了简化讨论，我们假设 d_0 是已知的.

我们马上可以看到，学习 Q-值函数是克服“时域诅咒”的一种有效方法. 为此，我们需要回答两个关键问题：

1. 如何从数据中估计 Q^π（见第 2.1 和 2.2 节）.
2. 对于估计的 $J(\pi)$，我们可以得到什么样的理论保障（见第 2.3 节）.

2.1 FQE 和 Bellman 完备性

对值函数 Q^π 的估计通常是基于这样一个事实：它是（针对特定策略的）*Bellman* 算子（Bellman operator）唯一的不动点，也即：

$$Q^\pi = \mathcal{T}^\pi Q^\pi \tag{4}$$

其中（针对特定策略的）Bellman 算子 $\mathcal{T}^\pi : \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \to \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$ 定义为：对任意的 $f \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$，

$$(\mathcal{T}^\pi f)(s,a) := \mathbb{E}_{r=R(s,a),s'\sim P(\cdot|s,a)}\left[r + \gamma f(s',\pi)\right] \tag{5}$$

也即基于策略 π 进行了一次 Q-值函数的转移. 因此，Bellman 算子将当前的 Q-值函数映射为一步期望回报加上未来折扣回报，刻画了强化学习中策略或最优行为的动态递推关系. 其重要的一个性质是 γ-收缩性（γ-contraction），也即：

$$\forall f, f' \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}, \|\mathcal{T}^\pi f - \mathcal{T}^\pi f'\|_\infty \le \gamma \|f - f'\|_\infty.$$

用于求解 Q^π 的计算算法通常基于*动态规划*（Dynamic Programming, DP）. 例如，**值迭代**（Value Iteration, VI）算法反复将 Bellman 算子 $\mathcal{T}^\pi$ 应用于任意初始函数 f_0，而其 γ-收缩性质恰恰保证了 $(\mathcal{T}^\pi)^n f_0 \to Q^\pi$，且在 $\|\cdot\|_\infty$ 范数下具有几何收敛的保障，也即：

$$\|(\mathcal{T}^\pi)^n f_0 - Q^\pi\|_\infty \le \gamma^n \|f_0 - Q^\pi\|_\infty. \tag{6}$$

这一性质推动了许多估计值函数的主流算法的发展，它们试图通过数据来逼近 Bellman 算子.

注意到，公式 (5) 中的 $\mathcal{T}^\pi f$ 实际上具有条件期望的形式，因此可以被写作一个回归问题：

$$\mathcal{T}^\pi f \in \operatorname*{arg\,min}_{f'\in\mathbb{R}^{\mathcal{S}\times\mathcal{A}}} \mathcal{L}(f'; f, \pi),$$

其中损失函数为均方误差损失，也即：

$$\mathcal{L}(f'; f, \pi) := \mathbb{E}_D\left[\left(f'(s,a) - r - \gamma f(s',\pi)\right)^2\right].$$

通过这样的表示方式，我们可能可以通过*最小二乘回归*（least square regression）的方式对 Bellman 算子 $\mathcal{T}^\pi$ 进行逼近. 为此，我们首先需要明确其中所使用的数据形式.

这里的数据集 D 表示离线数据中观测到的 (s,a,r,s') 的经验分布. 在本文后续部分中，我们不再依赖轨迹数据，而是使用这些可从轨迹中提取出来的的转移元组进行学习，也即：

$$\tau^{(i)} \to \left(s_0^{(i)}, a_0^{(i)}, r_0^{(i)}, s_1^{(i)}\right), \left(s_1^{(i)}, a_1^{(i)}, r_1^{(i)}, s_2^{(i)}\right), \ldots.$$

在这种情况下，我们在分析时所需要用到的集中不等式需要考虑来自同一条轨迹的多个元组之间的依赖关系. 为了简化分析，这里采用一种标准的简化设定：假设数据集 $\mathcal{D}$ 包含 n 个 i.i.d 的转移元组：

$$(s,a,r,s') \sim D \iff (s,a) \sim d^D, r = R(s,a), s' \sim P(\cdot \mid s,a).$$

于是我们便可以介绍**拟合 Q**（Fitted-Q）方法，在 OPE 问题中也即**拟合 Q 评估**（Fitted-Q Evaluation, FQE）方法. 它可以看成是许多经验上流行的方法的原型或理论基础，例如时序差分方法（Temporal Difference, TD）. FQE 假设有一个函数类 $\mathcal{F}$ 用于建模 Q^π，并从一个任意的初始化函数 $f_0 \in \mathcal{F}$ 开始. 之后，它反复求解一系列的最小二乘回归问题：

$$f_k \leftarrow \arg\min_{f' \in \mathcal{F}} \widehat{\mathcal{L}}(f'; f_{k-1}, \pi), \tag{7}$$

其中 $\widehat{\mathcal{L}}$ 是损失函数 $\mathcal{L}$ 在基于数据集 $\mathcal{D}$ 下的经验近似：

$$\widehat{\mathcal{L}}(f'; f, \pi) := \frac{1}{|\mathcal{D}|} \sum_{(s,a,r,s') \in \mathcal{D}} \left(f'(s,a) - r - f(s', \pi)\right)^2.$$

我们在分析 FQE 之前，必须指出一个非常严重的问题：FQE 在看似非常合理的假设下，仍有可能会发散！考虑下面的命题.

命题 1. *即使满足以下所有条件，FQE 仍有可能发散：*

1. *数据集大小 $|\mathcal{D}| = \infty$，且在 (7) 中能找到确切的 f' 使得损失函数最小化；*
2. *$\mathcal{F} \subseteq \mathbb{R}^1$ 是一维的线性函数类，且能够精确表示 Q^π，也即 $Q^\pi \in \mathcal{F}$（这称作可实现性，realizability）.*

这种发散现象出现在一组看似“完美”的假设下：无限数据、精确优化、简单的函数类，以及完美的可实现性. 此外，这一被称为“致命三要素”（deadly triad）的问题并不只是理论构造：在实际中，深度强化学习算法常被观察到具有不稳定性和训练发散的现象.

那么，问题到底出在哪里？问题在于：FQE 实际上是求解一系列回归问题 $(s,a) \mapsto r + \gamma f_{k-1}$，$k = 1, 2, \ldots$，其回归目标（即 TD 目标）依赖于上一次迭代得到的函数 f_{k-1}. 因此，为了让 FQE 尽可能“模仿”值迭代（VI）算法，我们需要这个回归序列中的每一步都是良设的（*well-specified*），也就是说，函数空间 $\mathcal{F}$ 必须包含每一步回归的 Bayes 最优预测函数 $\mathcal{T}^\pi f_{k-1}$ 的近似.

但由于 f_{k-1} 本身依赖于数据的随机性，我们希望将其放宽于 $\mathcal{F}$ 中的任意函数，使得所依赖的假设与数据无关. 这就引出了现代强化学习理论中的一个极其重要的表达能力假设，称作**Bellman 完备性**（Bellman Completeness），这也是本节要强调的重点.

假设 1. （*Bellman* 完备性）*对于任意的 $f \in \mathcal{F}$，我们都有 $\mathcal{T}^\pi f \in \mathcal{F}$.*

这个假设意味着，函数空间 $\mathcal{F}$ 对 Bellman 算子 $\mathcal{T}^\pi$ 是封闭的（closed），因此该假设也被称作 *Bellman* 封闭性（Bellman closure/closedness），参见图 1 的可视化说明. 对于有限大小的函数类 $\mathcal{F}$，Bellman 完备性可以推出“可实现性”，因为 $Q^\pi = (\mathcal{T}^\infty) f \in \mathcal{F}$，因此我们可以把 Bellman 完备性看作是一个更强的假设.

需要指出的是，Bellman 完备性和可实现性的假设有根本的不同. 在监督学习中，当函数类的容量不足时，我们通常会使用更加复杂的函数类，以更好地逼近目标函数（至少不会更糟）；然而，Bellman 完备性是一个非单调（non-monotone）假设：当我们引入更复杂的函数类时，反而可能破坏完备性，而不是增强它！正因如此（以及其他原因），模型选择（model selection）在离线强化学习中变得极具挑战性.

Bellman 完备性常常在结构化 MDP 中成立（相比之下，信息论构造不要求对 MDP 的动态模型有任何限制）. 典型的情形包括：低秩 MDP（low-rank MDP），其对应线性函数类；双模同构抽象（bisimulation abstraction），其对应给定状态抽象下的分段常数函数类. 下面以低秩 MDP 为例，作一个简单分析.

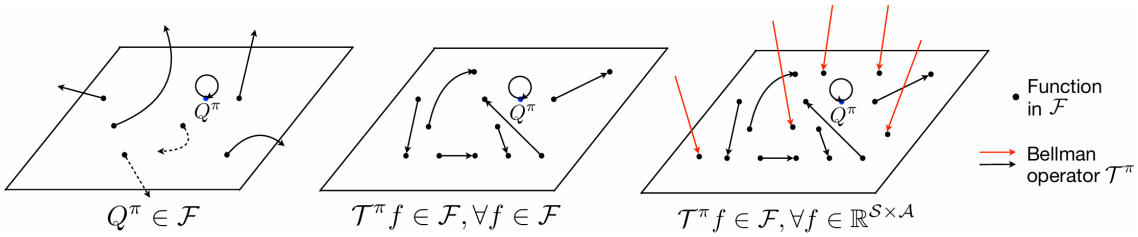

图 1. 对值函数类 $\mathcal{F}$ 上不同表达能力假设的示意图. **左图**：仅假设可实现性，即 $Q^\pi \in \mathcal{F}$，而 Bellman 算子一般会将 $\mathcal{F}$ 中的函数映射到函数类之外；**中图**：Bellman 完备性假设成立，即 $\mathcal{F}$ 对 Bellman 算子 $\mathcal{T}^\pi$ 是封闭的；**右图**：所有函数（包括不属于 $\mathcal{F}$ 的函数）经过 Bellman 映射后，其结果都落在 $\mathcal{F}$ 中（详见第 3.4 节）

[**低秩 MDP**]. 我们称一个 MDP 是秩（rank）为 d 的 MDP，若存在映射 $\phi^* : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ 和 $\psi^* : \mathcal{S} \to \mathbb{R}^d$，以及参数 $\theta_R \in \mathbb{R}^d$，使得

$$P(s' \mid s, a) = \langle \phi^*(s,a), \psi^*(s') \rangle, \quad R(s,a) = \langle \phi^*(s,a), \theta_R \rangle.$$

当 ϕ^* 已知时，其被称作以 ϕ^* 为特征的线性 MDP（linear MDP）. 对于任意函数 $f : \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \to \mathbb{R}$ 和任意策略 π，易知 $\mathcal{T}^\pi f$ 关于 ϕ^* 也是线性的，因此函数类 $\mathcal{F}_{\phi^*} = \{\langle \phi^*, \theta \rangle : \theta \in \mathbb{R}^d\}$ 满足 Bellman 完备性. 若 ϕ^* 不可知，但属于某个特征类 Φ，则 $\bigcup_{\phi\in\Phi} \mathcal{F}_\phi$ 仍然满足 Bellman 完备性.

2.2 Bellman 残差最小化

在我们分析 FQE（Fitted-Q Evaluation）之前，我们讨论一种用于估计值函数的替代算法. 命题 1 中指出的发散现象，在一定程度上可以归因于 FQE 的迭代本质. 相比之下，在监督学习中我们往往只是写出一个损失函数，并对其最小化；而 RL 中恰恰就是缺少一个一致的、“全局”的损失函数. 那么，我们是否可以将值函数的估计形式化为一个损失最小化问题呢?

一个直接的思路是，注意到值函数 Q^π 是 Bellman 方程 $f = \mathcal{T}^\pi f$ 的唯一解，因此我们可以寻找一个函数 f，使得该 Bellman 方程“违背得越少越好”. 具体而言，我们称 $f - \mathcal{T}^\pi f$ 为 *Bellman 误差*（Bellman error/residual），它是状态-动作对 (s,a) 的函数. 为了将其转化成一个标量的目标函数，我们自然地可以考虑该误差在离线数据分布 D 下的期望平方损失：

$$\mathcal{E}(f;\pi) := \mathbb{E}_{d^D}\left[\left(f(s,a) - \mathcal{T}^\pi f(s,a) \right)^2 \right]. \tag{8}$$

我们可以尝试从该数据中估计该损失 $\mathcal{E}(f;\pi)$，并在函数类 $f \in \mathcal{F}$ 上最小化它，以此来近似 Q^π. 这类方法通常被称作 **Bellman 残差最小化**（Bellman Residual Minimization, BRM）.

然而不幸的是，公式 (8) 中的 Bellman 残差 $\mathcal{E}(f;\pi)$ 无法直接被数据准确估计. 问题的根源在于：$\mathcal{T}^\pi f$ 是一个条件期望，而它嵌套在平方项内部：

$$\mathcal{E}(f;\pi) = \mathbb{E}_{(s,a)\sim D}\left[\left(f(s,a) - \mathbb{E}_{r,s'|s,a}[r + \gamma f(s',\pi)] \right)^2 \right],$$

其中 $r, s' \mid s, a$ 是 $r = R(s,a)$，$s' \sim P(\cdot \mid s, a)$ 的简写.

一种朴素的估计方法是我们直接去忽略这个条件期望，也就是把它当成一个常数处理，从而得到 $\widehat{\mathcal{L}}(f; f, \pi)$，即我们在公式 (7) 中定义的经验损失. 然而，这种方法即使在无限数据下也是不正确的！因为

$$\widehat{\mathcal{L}}(f; f, \pi) \overset{n\to\infty}{\longrightarrow} \mathcal{L}(f; f, \pi) \neq \mathcal{E}(f;\pi).$$

更精确地说，这里存在一个*偏差-方差分解*（bias-variance decomposition）：

$$\mathcal{L}(f; f, \pi) = \mathcal{E}(f; \pi) + \mathcal{L}(\mathcal{T}^\pi f; f, \pi).$$

这和监督学习中回归问题的结构类似，在监督学习中，当我们用模型 $h : \mathcal{X} \to \mathbb{R}$ 预测实值标签 Y 时，有如下关系：

$$\mathbb{E}[(Y - h(X))^2] = \underbrace{\mathbb{E}[(\mathbb{E}[Y \mid X] - h(X))^2]}_{\text{超额风险}} + \underbrace{\mathbb{E}[(\mathbb{E}[Y \mid X] - Y)^2]}_{\text{标签噪声}}.$$

我们将误差项分解为了*超额风险*（excess risk）项和不可避免的*标签噪声*（label noise）项. 在监督学习中，我们的目标是最小化超额风险. 这是因为，标签噪声是 $\mathbb{E}[(\mathbb{E}[Y \mid X] - Y)^2]$ 与预测函数 h 无关的，只有数据本身决定，因此，最小化总误差 $\mathbb{E}[(Y - h(X))^2]$ 就等价于最小化超额风险 $\mathbb{E}[(\mathbb{E}[Y \mid X] - h(X))^2]$，即使我们无法估计标签噪声，也不会影响目标函数的最小化过程.

然而，在强化学习中，这里的“噪声”项 $\mathcal{L}(\mathcal{T}^\pi f; f, \pi)$ 依赖于候选函数 f 本身，这就导致了最小化 $\mathcal{L}(f; f, \pi)$ 并不等价于最小化真实的 Bellman 残差 $\mathcal{E}(f; \pi)$. 由于该“噪声”项的期望为 0，一种自然的解决方案为：我们从每个状态-动作对 (s, a) 中采样两个独立的下一状态 s'，从而构造一个无偏的估计量来计算 $\mathcal{E}(f; \pi)$，这被称作**双重采样**（Double Sampling）方法. 事实上，在回归问题中，我们也有类似的做法来消除“噪声”项的影响：对于同一个 X，我们采样两个独立的标签 Y, Y'，然后通过 $\mathbb{E}[(h(X) - Y)(h(X) - Y')]$ 对超额风险进行估计.

然而，双重采样只能在模拟器中实现，在实际的离线环境中并不能够同时采样两个独立的下一状态 s'. 因此，BRM 算法提供了一种解决方法：我们显式地估计 $\mathcal{L}(\mathcal{T}^\pi f; f, \pi)$，然后从 $\mathcal{L}(f; f, \pi)$ 中减去它. 注意到，$\mathcal{L}(\mathcal{T}^\pi f; f, \pi)$ 实际上是一个*贝叶斯误差*（Bayes error），它表示在已知最优预测器 $\mathcal{T}^\pi f$ 的前提下，预测目标 $r + \gamma f(s', \pi)$ 的期望误差. 因此，我们可以将其形式化成

$$\mathcal{L}(\mathcal{T}^\pi f; f, \pi) = \min_{g \in \mathbb{R}^{\mathcal{S} \times \mathcal{A}}} \mathcal{L}(g; f, \pi). \tag{9}$$

当我们用函数类 $\mathcal{G}$ 来建模 g 时，在实际数据集 $\mathcal{D}$ 上，估计 Q^π 的过程就变成了

$$\widehat{f}^\pi = \arg\min_{f \in \mathcal{F}} \widehat{\mathcal{E}}(f; \pi) := \max_{g \in \mathcal{G}} \left(\widehat{\mathcal{L}}(f; f, \pi) - \widehat{\mathcal{L}}(g; f, \pi) \right). \tag{10}$$

易知，只有当 $\min_{g \in \mathcal{G}} \mathcal{L}(g; f, \pi) \approx \mathcal{L}(\mathcal{T}^\pi f; f, \pi)$ 时，$\widehat{\mathcal{E}}(f; \pi)$ 才是 $\mathcal{E}(f; \pi)$ 的良好估计. 这要求我们对所有 $f \in \mathcal{F}$，$\mathcal{T}^\pi f \in \mathcal{G}$. 有趣的是，如果我们直接将 $\mathcal{F}$ 本身作为 $\mathcal{G}$，那么这个条件就等价于 $\forall f \in \mathcal{F}, \mathcal{T}^\pi f \in \mathcal{F}$，这正好是假设 1 中提出的 *Bellman 完备性*！因此，为了方便起见，在本文接下来的分析中，我们将默认使用 $\mathcal{G} \equiv \mathcal{F}$，除非特别说明.

在 Bellman 完备性的假设下，我们可以用一些标准的集中不等式得到如下的理论保障. 对于待评估的策略 $\pi \in \Pi$，以至少 $1 - \delta$ 的概率，对任意 $f \in \mathcal{F}$ 满足：

$$|\widehat{\mathcal{E}}(f; \pi) - \mathcal{E}(f; \pi)| \lesssim \frac{V_{\max}^2}{n} \log \frac{|\mathcal{F}|}{\delta}. \tag{11}$$

如果我们假设 $Q^\pi \in \mathcal{F}$（也即可实现性，这在 $\mathcal{G} = \mathcal{F}$ 的假设下可由完备性直接得到），由 (10) 便有：

$$\mathcal{E}(\widehat{f}^\pi; \pi) \leq_\epsilon \widehat{\mathcal{E}}(\widehat{f}^\pi; \pi) \leq \widehat{\mathcal{E}}(Q^\pi; \pi) \leq_\epsilon \mathcal{E}(Q^\pi; \pi) = 0,$$

其中 $\leq_\epsilon$ 是 $\leq$ 在二者之间相差一个与公式 (11) 右侧数量级一致的微小误差的放松. 因此，我们可以推出如下泛化误差界：

$$\mathcal{E}(\widehat{f}^\pi; \pi) \lesssim \frac{V_{\max}^2}{n} \log \frac{|\mathcal{F}|}{\delta}. \tag{12}$$

该结论适用于有限函数类 $\mathcal{F}$，也可以类似扩展到具有有界 L_∞ *覆盖数*（covering number）的连续函数类.

为了方便后续分析，我们引入*分布加权 p-范数*的记号 $\|\cdot\|_{p,\mu}$. 给定某个定义在 $\mathcal{X}$ 上的分布 μ，函数 f 的 μ-分布加权 p-范数定义为：

$$\|f\|_{p,\mu} := \mathbb{E}_\mu \left[|f|^p\right]^{1/p}.$$

这样一来，我们便可以将公式 (12) 写作：

$$\|\widehat{f}^\pi - \mathcal{T}^\pi \widehat{f}^\pi\|_{2,d^D} \lesssim V_{\max}\sqrt{\frac{\log(|\mathcal{F}|/\delta)}{n}}. \tag{13}$$

我们现在希望将关于 $\widehat{f}^\pi$ 的 Bellman 误差界，转化为对 $f - Q^\pi$ 的误差界，进而得到目标 $J(\pi)$ 的估计误差. 经典 RL 理论中的结果告诉我们 Bellman 误差是如何转化为 Q-值函数的误差的：对所有 $f \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$，有：

$$\|f - Q^\pi\|_\infty \le \frac{\|f - \mathcal{T}^\pi f\|_\infty}{1-\gamma}. \tag{14}$$

然而，正如公式 (6) 中值迭代算法的收敛性一样，这类 L_∞ 的结果难以在大状态空间中得到应用. 这是因为，公式 (14) 要求我们能够控制 L_∞ 范数下的 Bellman 误差，而我们通过公式 (13) 只能控制一个更弱的、基于加权 L_2 范数的 Bellman 误差.

我们将在下一节中进一步处理这个不一致的问题，并分析当我们不再采用经典的 L_∞ 分析方式时，状态（以及动作）的*覆盖*（coverage）这一概念是如何自然的引入的. 在这之前，我们再对本节中的 BRM 方法做一些说明.

- **相关估计方法**. 除了公式 (10) 中的 BRM 外，还存在其他估计 $\mathcal{E}(f;\pi)$ 的方法. 其中一个值得注意的方法是利用 Fenchel 对偶性：

  $$\mathcal{E}(f;\pi) = \min_{g\in\mathbb{R}^{\mathcal{S}\times\mathcal{A}}} \mathbb{E}_D\left[g(s,a)(f(s,a) - r - \gamma f(s',\pi)) - \frac{1}{2}g(s,a)^2\right].$$

  当我们限制 $g \in \mathcal{G}$ 且 $(f - \mathcal{T}^\pi f) \in \mathcal{G}$ 时，该式依然成立. 正如 BRM 需要 $\mathcal{G}$ 能表达 $\mathcal{T}^\pi f$ 一样，这种方法也要求 $\mathcal{G}$ 能表达 Bellman 误差 $f - \mathcal{T}^\pi f$，因此它们的样本复杂度分析是类似的. 事实上，我们可以在这两种假设中相互切换：如果 $\mathcal{T}^\pi f \in \mathcal{G}$，那么 $\mathcal{F} - \mathcal{G} := \{f - g : f \in \mathcal{F}, g \in \mathcal{G}\}$ 可以表达 Bellman 误差；反之亦然.

- **计算问题**. 公式 (10) 中的 BRM 需要求解一个*极小极大*（minimax）优化问题. 当 $\mathcal{F}$ 为线性函数类时，该问题存在闭式解，且其解与 LSTDQ 相一致；但如果 $\mathcal{F}$ 是非线性的函数类（如神经网络），那么公式 (10) 的计算可行性就不太明确了.

- **逼近误差**. 到目前为止，我们都假设表达能力（也即“可实现性”）完全成立，例如 $Q^\pi \in \mathcal{F}$，$\mathcal{T}^\pi f \in \mathcal{G}$. 我们同样也可以考虑“近似”的版本，这会带来*逼近误差*（approximation error），其可能通过某种范数 $\|\cdot\|$ 诱导的 $\min_{f\in\mathcal{F}}\|f - Q^\pi\|$ 来表示，并会进入最终的误差上界. 如何增加这样的逼近误差项，以及使用哪种范数，通常可以从我们下一节要讲的*误差传播分析*（error propagation analysis）中得到. 对于 Bellman 完备性分析，我们前面分析的标准的“和式误差”（additive error）：

  $$\max_{f\in\mathcal{F}}\min_{g\in\mathcal{G}}\|g - \mathcal{T}^\pi f\|_{2,D}.$$

  在引入逼近误差项后，会带来一种“积式逼近”（multiplicative approximation），这种形式也可以纳入 BRM 的分析框架中.

2.3 状态-动作覆盖条件下的理论保障

本节中，我们将会看到估计值函数是如何带来更好地状态-动作覆盖的. 我们会着重于第 2.2 节中的 BRM 算法，因为它的分析更为简洁清晰；然后我们也会简要说明 FQE 的结果.

正如第 2.2 节末指出的，传统的基于 L_∞ 范数的分析对我们来说是不足的，我们需要引入一个更精细的刻画来替代公式 (14)，这需要我们考虑分布的结构，具有“分布感知”(distribution-aware) 的特性. 下面的引理揭示了这一点：

引理 2. （*Bellman* 误差的 *telescoping*）对于任意策略 π 和任意函数 $f \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$，有：

$$J_f(\pi) - J(\pi) = \frac{1}{1-\gamma}\mathbb{E}_{d^\pi}[f - \mathcal{T}^\pi f],$$

其中 $J_f(\pi) := \mathbb{E}_{s\sim d_0}[f(s,\pi)]$. d^π 为状态-动作对基于策略 π 的折扣占用分布（*discounted occupancy*），定义为：

$$d^\pi = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t d_t^\pi,$$

其中 $d_t^\pi(s,a) := \Pr_\pi[s_t = s, a_t = a]$ 表示在策略 π 下，第 t 步时状态为 s 并采取动作 a 的概率.

引理 2 告诉我们，如果将 f 当作 Q^π 来使用，那么 $J_f(\pi)$ 恰是 $J(\pi)$ 在 $f \approx Q^\pi$ 下的估计. 此时，其与真实回报 $J(\pi)$ 之间的误差正好等于 Bellman 误差 $f - \mathcal{T}^\pi$ 在 d^π 分布下的期望.

事实上，我们已经非常接近最终的理论保障了！引理 2 告诉我们需要控制 Bellman 误差在占用分布 d^π 下的形式，而公式 13 告诉我们可以控制 Bellman 误差在 d^D（也即离线数据分布）下的平方形式. 因此，最后的假设就是一个可以将不同分布下的误差进行转换的结果，见下面的引理.

引理 3. （覆盖条件下的误差转换）对于定义在空间 $\mathcal{X}$ 上的任意函数 ξ，令 $\mu, \nu \in \Delta(\mathcal{X})$ 为两个分布，且 $1 \le p < \infty$，则有：

$$\|\xi\|_{p,\nu}^p \le \|\nu/\mu\|_\infty \cdot \|\xi\|_{p,\mu}^p,$$

其中 $\|\nu/\mu\|_\infty := \max_x \nu(x)/\mu(x)$. 我们约定 $0/0 = 0$，且对 $a \neq 0$ 有 $a/0 = \infty$.

引理 3 暗示了一个在 BRM 分析（公式 (10)）中使用的**覆盖条件**（coverage condition）：

假设 2. （覆盖条件）对于占用分布 d^π 和数据分布 d^D，我们假设有 $\|d^\pi/d^D\|_\infty \le C_\pi < \infty$.

基于这个假设，我们就能得到用 BRM 估计 $J(\pi)$ 的第一个误差保障：

$$\begin{aligned}
\left|J_{\hat{f}^\pi}(\pi) - J(\pi)\right| &= \left|\frac{1}{1-\gamma}\mathbb{E}_{d^\pi}\left[\hat{f}^\pi - \mathcal{T}^\pi\hat{f}^\pi\right]\right| && (15)\\
&\le \frac{1}{1-\gamma}\left\|\hat{f}^\pi - \mathcal{T}^\pi\hat{f}^\pi\right\|_{1,d^\pi} && (16)\\
&\le \frac{1}{1-\gamma}\left\|\hat{f}^\pi - \mathcal{T}^\pi\hat{f}^\pi\right\|_{2,d^\pi} && (17)\\
&\le \frac{\sqrt{C_\pi}}{1-\gamma}\left\|\hat{f}^\pi - \mathcal{T}^\pi\hat{f}^\pi\right\|_{2,d^D}\\
&\lesssim \frac{V_{\max}}{1-\gamma}\sqrt{\frac{C_\pi \log(|\mathcal{F}|/\delta)}{n}}. && (18)
\end{aligned}$$

其中，第一行是由于引理 2；第二行是由于 $\|\cdot\|_{1,d^\pi}$ 范数的定义；第三行是由于对分布加权 p-范数而言，若 $p \le p'$，则 $\|f\|_{p,\mu} \le \|f\|_{p',\mu}$；第四行是由于引理 3；第五行是代入了公式 (13).

假设 2 中的覆盖系数 C_π 也被称作集中系数（concentrability coefficient），它用于衡量分布 d^D 相对于占用分布 d^π 的最大密度比. 事实上，从上面的推导可见，C_π 用于控制如下的量：

$$\frac{\|\widehat{f}^\pi - \mathcal{T}^\pi \widehat{f}^\pi\|_{2,d^\pi}^2}{\|\widehat{f}^\pi - \mathcal{T}^\pi \widehat{f}^\pi\|_{2,d^D}^2}.$$

然而，当函数类 $\mathcal{F}$ 具有某些额外的结构时，这个比值可能很松（甚至趋于 ∞）. 特别地，由于 $\widehat{f}^\pi \in \mathcal{F}$，我们可以直接定义一个数据无关的量，对上述比值进行放松，进而替代对 C_π 的估计而得到一个更优的估计. 这个先验量是平方误差版的集中系数：

$$C_\pi^{\mathrm{sq}} := \max_{f \in \mathcal{F}} \frac{\|f - \mathcal{T}^\pi f\|_{2,d^\pi}^2}{\|f - \mathcal{T}^\pi f\|_{2,d^D}^2}. \tag{19}$$

当函数 $\mathcal{F}$ 是以 $\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ 为特征的线性类时，即 $\mathcal{F} \subseteq \{\phi(s,a)^\top \theta\}$，那么 Bellman 完备性蕴含 $\mathcal{T}^\pi f \in \mathcal{F}$，从而 $f - \mathcal{T}^\pi f$ 也属于该线性类. 此时，公式 (19) 有非常直观的上界：

$$C_\pi^{\mathrm{sq}} \le \max_{u \in \mathbb{R}^d} \frac{u^\top \Sigma_\pi u}{u^\top \Sigma_D u} = \lambda_{\max}(\Sigma_\pi^{1/2} \Sigma_D^{-1} \Sigma_\pi^{1/2}), \tag{20}$$

其中 $\Sigma_\pi = \mathbb{E}_{d^\pi}[\phi\phi^\top]$ 和 $\Sigma_D = \mathbb{E}_{d^D}[\phi\phi^\top]$ 分别为对应分布下的协方差矩阵，$\lambda_{\max}(\cdot)$ 表示最大特征值. 易知，这是一个广义瑞利商（generalized Rayleigh quotient）优化问题. 通过求解 $\Sigma_\pi^{1/2} \Sigma_D^{-1} \Sigma_\pi^{1/2}$ 的最大特征值，我们发现：控制 C_π^{sq} 的要求仅是数据分布 d^D 在状态-动作空间内能够命中（hit）占用分布 d^π 所激活的所有特征方向. 换句话说，对所有 $\theta^\top \Sigma_\pi \theta > 0$ 的方向 θ，也有 $\theta^\top \Sigma_D \theta > 0$. 即使在 $\|d^\pi / d^D\|_\infty = \infty$ 的极端情况下，这个覆盖参数也依然是控制的.

事实上，我们还可以进一步收紧覆盖参数，通过将公式 (15) 中的目标转化为对数据分布 d^D 下的 Bellman 误差范数进行控制：

$$C_\pi^{\mathrm{avg}} := \max_{f \in \mathcal{F}} \frac{(\mathbb{E}_{d^\pi}[f - \mathcal{T}^\pi f])^2}{\mathbb{E}_{d^D}[(f - \mathcal{T}^\pi f)^2]}. \tag{21}$$

在和之前相同的线性函数类的假设下，我们进一步有如下结果：

$$C_\pi^{\mathrm{avg}} \le \mathbb{E}_{d^\pi}[\phi]^\top \Sigma_D^{-1} \mathbb{E}_{d^\pi}[\phi]. \tag{22}$$

这个界相比于公式 (20) 而言是一个非常显著的改进，因为我们现在只需要在 $\mathbb{R}^d$ 中覆盖一个方向，也即 d^π 下的特征的均值方向！这表明，我们在公式 (17) 中使用 Cauchy-Schwarz 不等式有时会让结果变得很松.

需要指出的是，并非所有的设定或方法都适用于这种更紧的覆盖（即公式 (21) 中的 C_π^{avg}）. 在某些情况下，我们仍需回退至使用公式 (19) 或类似的形式（见第 3.4 节）. 例如，在 C_π^{avg} 覆盖条件下通过 BRM 学得的 $\widehat{f}^\pi$ 可用于估计回报 $J(\pi)$，并能够提供对 $J(\pi)$ 的估计保证. 然而，如果我们希望获得更强的泛化能力，例如希望在某个分布 ν 下控制 $\mathbb{E}_\nu[(f - Q^\pi)^2]$，则需要确保数据分布 d^D 能够覆盖以 ν 为初始分布所诱导的占用分布 d_ν^π，且应在 C_π^{sq} 的覆盖下成立.

下面我们在 Bellman 完备性和覆盖条件的假设下，对 FQE 方法进行分析. 不同之处在于，这里 f_k 并不是与其自身保证 Bellman 一致性（Bellman-consistent），也即不能保证 f_k 满足 Bellman 方程！在 FQE 中，f_k 是和上一轮迭代的函数 f_{k-1} 相关，这就使得 FQE 的误差传播分析更加复杂. 当然，分析的整体思路和 BRM 的分析还是类似的，我们做一个简单的推导.

对于 FQE 的关键观察是：运行 K 步迭代的 FQE 可以被看作是在一个截断有限时域的（truncated finite-horizon）MDP 上学习一个非平稳策略（non-stationary policy）的值函数. 此时，我

们在策略 π 上的期望回报定义为

$$J_K(\pi) := \mathbb{E}_\pi \left[\sum_{t=0}^{K-1} \gamma^t r_t \right].$$

其可以很好地近似无穷时域下的期望回报 $J(\pi)$，误差仅为一个可控的残差项 γ^K，而 K 可以任意选取得足够大，以减小该残差项. “非平稳性”告诉我们，其值函数是时间相关的（time-dependent）：

$$Q_k^\pi(s,a) := \mathbb{E}_\pi \left[\sum_{t=0}^{k-1} \gamma^t r_t \mid s_0 = s, a_0 = a \right],$$

其表示前 k 步执行策略 π 的期望回报（从状态-动作对 (s,a) 开始）. 类似地，对应的回报可以通过该值函数来刻画：$J_{Q_K^\pi}(\pi) = J_K(\pi)$.

注意到，这个非平稳的值函数也满足 Bellman 方程：$Q_k^\pi = \mathcal{T}^\pi Q_{k-1}^\pi$，且 $Q_0^\pi \equiv 0$. FQE 的输出 $f_K, \dots, f_1$ 近似于 $Q_K^\pi, \dots, Q_1^\pi$. 这样一来，若 $\|f_k - \mathcal{T}^\pi f_{k-1}\|_{2,d^\mathcal{D}}$ 很小，则集合 $\{f_K, \dots, f_1\}$ 在这种有限时域问题中作为时间相关的函数是 Bellman 一致的. 基于这种理解，我们可以写出引理 2 在有限时域问题下的版本：

引理 4. *对于任何非平稳策略 $\pi_{K:1}$ 和函数 $f_{K:1}$，有：*

$$J_{f_K}(\pi_K) - J_K(\pi_{K:1}) = \sum_{t=0}^{K-1} \gamma^t \mathbb{E}_{d_t^{\pi_{K:1}}} \left[f_{K-t} - \mathcal{T}^{\pi_{K-t-1}} f_{K-t-1} \right].$$

该引理适用于一个 K 步的非平稳策略 $\pi_{K:1}$，该策略在第 t 步根据 $\pi_{K-t}(\cdot \mid s)$ 采取动作. 目前我们只需要考虑 $\pi_K = \cdots = \pi_1 = \pi$ 的情况即可，更一般的情况将在后面使用. 其余的分析过程与公式 (16)∼(18) 类似，只是现在我们需要对每一个 d_t^π 分别进行覆盖，而不是将 d^π 整体覆盖.

2.4 策略优化

到目前为止，我们主要关注的都是 OPE 问题，这在保留验证和测试中至关重要. 然而，在训练过程中，我们需要执行策略优化（policy optimization），也即找到最优的策略. 一个直接的方式是：

$$\arg\max_{\pi \in \Pi} J_{\widehat{f}^\pi}(\pi), \tag{23}$$

其中 $J_{\widehat{f}^\pi}(\pi)$ 的定义见引理 2，$\widehat{f}^\pi$ 是基于公式 (10) 或其他方法估计得到的. 然后，我们可以将公式 (18) 中的预测误差上界，在 $\pi \in \Pi$ 上进行一致（uniform）控制，从而直接转化为以下关于策略优化的保障.

定理 5. *令 $\mathcal{G} = \mathcal{F}$，策略 $\widehat{\pi}$ 为通过公式 (23) 进行策略优化得到的. 假设对于所有的策略 $\pi \in \Pi$，假设 1 和假设 2 均成立，那么以至少 $1-\delta$ 的概率，对任何 $\pi_{\mathrm{cp}} \in \Pi$，有：*

$$J(\pi_{\mathrm{cp}}) - J(\widehat{\pi}) \lesssim \frac{V_{max}}{1-\gamma} \sqrt{\frac{\max_{\pi \in \Pi} C_\pi}{n} \cdot \log \frac{|\mathcal{F}| \cdot |\Pi|}{\delta}}.$$

这里，π_{cp} 是我们可能希望与 $\widehat{\pi}$ 比较的任何策略，其可以设为 $\arg\max_{\pi \in \Pi} J(\pi)$. 我们以这种形式给出，是为了方便与第 3.1 节的改进算法保障进行比较. 额外的 $|\Pi|$ 来自对所有 $\pi \in \Pi$ 上 $J(\pi)$ 精确估计误差的 union bound，并可以推广到连续的策略类，只需要采用适当的覆盖数进行刻画. 另外，界中的 C_π 也可以被更紧的覆盖定义来替代，例如公式 (21) 中的 C_π^{avg}.

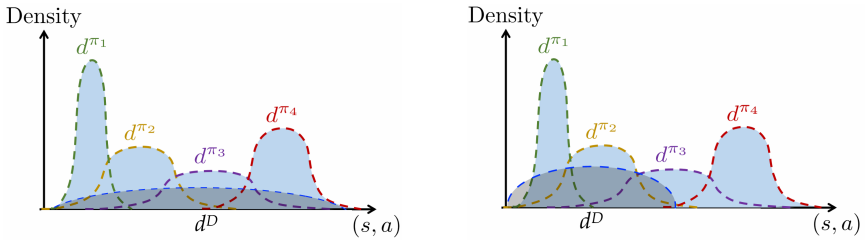

图 2. 不同覆盖假设的示意图. **左图**：全策略覆盖；**右图**：仅覆盖了策略 π_1 和 π_2（分析详见第 3.1 节）.

在定理 5 中，有 $\max_{\pi\in\Pi} C_\pi$ 一项，要求数据分布 d^D 对所有策略 $\pi \in \Pi$ 提供足够的覆盖，这被称作**全策略覆盖**（all-policy coverage）. 这事实上对 d^D 施加了很大的负担，要求它具有很强的探索性（exploratory）.

存在一些情形，使得 $\max_{\pi\in\Pi} C_\pi$ 即使在最优的 d^D 下也会非常大（和 $|\Pi|$ 线性相关，或和 H 指数相关），例如，当 MDP 是一个确定性的完全树（complete tree）时，对任意数据分布 d^D 均有 $\max_{\pi\in\Pi} C^\pi \geq |\Pi| = |\mathcal{A}|^H$；也存在一些情形，其中 MDP 的某些结构特性仍然可能使得 $\max_{\pi\in\Pi}$ 在大状态空间中得到控制. 一个例子是低秩 MDP：对于任意的策略类 Π，总存在某个数据分布 d^D 使得 $\max_{\pi\in\Pi} C_\pi \leq |\mathcal{A}|d$, 其中 d 为特征的维度.

公式 (23) 为我们提供了一个理论上的算法. 如果直接对其进行实现，需要遍历整个（可能很大的）策略类 Π，这通常是计算上不可行的. 因此，我们需要使用一些计算上更可行的算法，在类似或者稍强的假设下给出对应的保障. 它们通常基于动态规划（dynamic programming）方法，例如值迭代（value iteration）和策略迭代（policy iteration）.

我们之前都是运用（针对特定策略的）Bellman 方程 $Q^\pi = \mathcal{T}^\pi Q^\pi$ 来逼近对应的值函数 Q^π. 事实上，我们还有 Bellman 最优方程（Bellman optimality equation），$Q^* = \mathcal{T}Q^*$. 这里，$\mathcal{T}$ 同样也称作 Bellman 算子，定义为：对任意的 $f \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$，

$$(\mathcal{T}f)(s,a) := \mathbb{E}_{r=R(s,a),s'\sim P(\cdot|s,a)}[r + \gamma \max_{a'\in\mathcal{A}} Q^*(s',a')].$$

这个 Bellman 算子 $\mathcal{T}$ 的不动点 Q^* 被称作最优值函数，通过贪心（greedy）方法可以得到一个最优策略：

$$\pi_{Q^*}(s) := \arg\max_{a\in\mathcal{A}} Q^*(s,a).$$

可以证明，该策略在对应的 MDP 中对所有初始状态同时实现最优回报，也记作 π^*.

因此，我们可以通过与 $\mathcal{T}^\pi$ 相同的方式逼近 $\mathcal{T}$ 的不动点，例如：公式 (7) 中迭代迭代的方式，或公式 (10) 中求解一个极小极大优化问题的方式. 如果我们采用迭代的方式进行求解，对应的算法称作**拟合 Q 迭代**（Fitted-Q Iteration, FQI），其对应的迭代公式为：

$$f_k \leftarrow \arg\min_{f\in\mathcal{F}} \sum_{(s,a,r,s')\in\mathcal{D}} \left(f(s,a) - r - \gamma \max_{a'\in\mathcal{A}} f_{k-1}(s',a')\right)^2.$$

我们通过迭代（或求解优化问题），最终可以得到对最优值函数 Q^* 的估计 $\widehat{f} \in \mathcal{F}$，进而输出对应的贪心策略 $\pi_{\widehat{f}}$ 为 $\pi_{\widehat{f}}(s) := \arg\max_{a\in\mathcal{A}} \widehat{f}(s,a)$. 通过这样的方式，算法不需要再维护一个独立的策略类，而是通过从函数类 $\mathcal{F}$ 中贪心地导出它来实现.

尽管 FQI 在计算上更加高效，我们仍将对其基于 BRM 直接优化的极小极大的变体进行分析，下面的引理分析了这里的误差传播.

引理 6. 对于任意 $f \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$，策略 $\pi, \pi' : \mathcal{S} \to \Delta(\mathcal{A})$，有：

$$J(\pi') - J(\pi) = \frac{1}{1-\gamma}\left(\mathbb{E}_{s\sim d^{\pi'}}[f(s,\pi') - f(s,\pi)] + \mathbb{E}_{d^{\pi'}}[\mathcal{T}^{\pi} f - f] + \mathbb{E}_{d^{\pi}}[f - \mathcal{T}^{\pi} f]\right).$$

当我们选择 $\pi' = \pi^*$，$\pi = \pi_f$ 时，由于 π_f 是对 f 的贪心策略，右侧第一项 $\mathbb{E}_{s\sim d^{\pi^*}}[f(s,\pi^*) - f(s,\pi_f)] \le 0$；且因为 $\mathcal{T}^{\pi_f} f = \mathcal{T} f$，我们得到：

$$J(\pi^*) - J(\pi_f) \le \frac{1}{1-\gamma}\left(\mathbb{E}_{d^{\pi^*}}[\mathcal{T} f - f] + \mathbb{E}_{d^{\pi_f}}[f - \mathcal{T} f]\right).$$

这个结果对应 FQE 分析中的引理 2，但针对的是学习最优值函数 Q^*. 其告诉我们，当以下两点成立时，$J(\pi^*) - J(\pi_f)$ 会很小：

1. $\mathbb{E}_{d^D}[(\widehat{f} - \mathcal{T}\widehat{f})^2]$. 这一点我们之前已经分析过，如公式 (13).
2. 在某种覆盖条件下，数据分布 d^D 能同时覆盖两个策略的占用分布 d^{π^*} 和 d^{π_f}.

让我们着重分析第二点. d^D 能覆盖 d^{π^*} 是非常合理且不可避免的：若我们想要学习最优策略，自然需要数据分布携带该策略相关的信息. 然而，d^D 要能覆盖学习得到的策略 π_f 的分布，这是我们无法事先控制的. 因此，为了建立一个不依赖于数据随机性的假设，我们可以把要求放宽为对所有 π_f 可能输出的策略都成立，并且用 $\max_{f\in\mathcal{F}} C_{\pi_f}$ 来度量覆盖的质量. 这正是“全策略覆盖”的假设，即定理 5 中的 $\max_{\pi\in\Pi} C_\pi$ 一项.

除了值迭代算法，另一种用于 MDP 的基本规划算法是策略迭代（policy iteration），其通过在前一个策略的 Q-值函数下生成贪心策略的方式得到下一个策略，进而进行迭代：$\pi_{k+1} \leftarrow \pi_{Q^{\pi_k}}$，也即 $\pi_{k+1}(s) = \arg\max_{a\in\mathcal{A}} Q^{\pi_k}(s,a)$. 类似地，**拟合策略迭代**（Fitted Policy Iteration, FPI）也是这样一种算法，其中 Q^{π_k} 是通过前面介绍的函数类 $\mathcal{F}$ 中的算法进行逼近的，我们将在第 3.3 节中看到 FPI 一个变体的分析.

3 悲观策略优化

到现在为止，我们看到的所有策略优化的理论保障，都需要全策略覆盖，并依赖于 $\max_{\pi\in\Pi} C_\pi$ 一项，其中 C_π 可以替换为改进后的版本. 这就要求数据集（也即数据分布）具有足够的探索性（exploratory），但在实际中往往并不成立，甚至有时会对底层的 MDP 结构加以限制（见第 2.4 节）. 因此，一个自然的问题（也是离线强化学习研究的核心问题之一），是：我们是否可以设计出适用于任意离线数据集的算法，并给出对应的理论保障？

在这一个部分，我们将展示，通过引入**悲观算法**（pessimistic algorithms），可以将“全策略覆盖”的假设放宽为**单策略覆盖**（single-policy coverage），且仍然能得到较好的表现（见图 2 的右图）.

3.1 面对不确定性的悲观性

考虑一个简单的**多臂赌博机问题**（Multi-Armed Bandit, MAB），它可以被看作是只有一个状态的 MDP（因此转移总是回到自身），每个动作（又称“摇臂”，arm）会带来一个随机奖励. 对于每个 $a \in \mathcal{A}$，我们用 $R(a)$ 表示其真实的期望奖励. 这里的候选策略（candidate policy）对应于选择不同的摇臂进行确定性操作. 在这个问题中，前面所有的优化算法都简化为如下的简单流程：

1. 估计每个摇臂的期望奖励 $\widehat{R}(a)$；

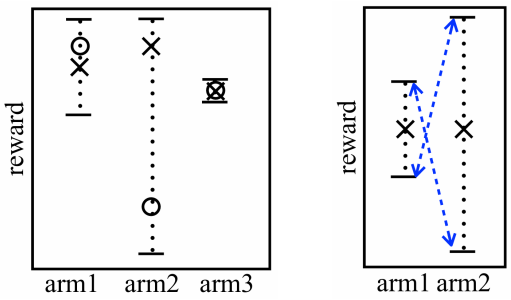

图 3. 多臂赌博机（MAB）问题中的不确定性. 其中“O”代表真实期望奖励，“X”代表点估计. **左图**：贪心（greedy）的方式会选择第 2 个摇臂，损失更大；而悲观（pessimistic）的方式会选择第 3 个摇臂，损失更小（尽管其是次优的，但可以被最优摇臂的不确定性控制）. **右图**：回报最大化（return optimization）的方式选择了第 1 个摇臂，遗憾最小化（regret minimization）的方式选择了第 2 个摇臂；两个摇臂具有相同的遗憾（即双向箭头的高度），但在这两者之间随机选择会使得总遗憾减半.

2. 输出估计值最高的摇臂.

“全策略覆盖”要求每个摇臂在离线数据中都被充分采样，以便所有摇臂的奖励估计都是准确的. 如果这个条件不满足，我们可能会由于随机波动而选择一个奖励较低但采样很少的摇臂，而错过另一个虽然样本多但奖励更高的摇臂（见图 3）.

解决这个问题的关键在于**不确定性量化**(uncertainty quantification). 我们不仅仅要看点估计(point estimation),还应该考虑置信区间(confidence interval, CI). 在 MAB 问题中,我们可以通过集中不等式构造每个摇臂 a 的置信区间 $[R^-(a), R^+(a)]$，并保证真实期望 $R(a) \in [R^-(a), R^+(a)]$ 是以高概率成立的.

为了能够更可靠地找到一个奖励更高的摇臂，我们选择下置信界（Lower Confidence Bound, LCB）$R^-(a)$ 最大的摇臂. 这是因为，基于点估计的算法，其理论保障中需要考虑所有摇臂中最坏的估计误差；而 LCB 算法，我们只需要考虑最优摇臂的不确定性！定义我们通过 LCB 算法选择的摇臂为 $\widehat{a} = \arg\max_{a\in\mathcal{A}} R^-(a)$，对应的最优摇臂为 $a^* = \arg\max_{a\in\mathcal{A}} R(a)$，二者期望奖励的差值可以做如下控制：

$$R(a^*) - R(\widehat{a}) \le R^-(a^*) - R^-(\widehat{a}) + R(a^*) - R^-(a^*) \le R(a^*) - R^-(a^*). \tag{24}$$

这告诉我们，当我们优化下置信界（LCB）时，理论保障只取决于我们对最优策略低估了多少. 这体现了一个重要的原则：**面对不确定性的悲观性原则**（Pessimism in Face of Uncertainty, PFU）. 与之对应的是在线强化学习中的乐观性原则. 乐观性鼓励探索（exploration），即跳出当前的数据分布；而悲观性鼓励利用（exploitation），即在当前数据分布中做出保守决策. “探索-利用”的权衡（exploration-exploitation tradeoff）也是强化学习中的基本命题之一.

下一个问题是：如何将该原则应用到(多步的)MDP. 一个自然的想法是考虑对期望回报 $J(\pi)$ 的不确定性量化：如果我们知道 $J(\pi)$ 以高概率落在由数据计算出的置信区间 $[J^-(\pi), J^+(\pi)]$，那么我们可以选择最大化该下置信界（LCB）的策略：$\arg\max_{\pi\in\Pi} J^-(\pi)$.

一个这样的区间可以直接我们在第 2 节中分析的 BRM 的理论保障给出：

$$J^{\pm}_{\mathrm{cov}}(\pi) := J_{\widehat{f}^\pi}(\pi) \pm \mathrm{eb}(C_\pi) = J_{\widehat{f}^\pi}(\pi) \pm \mathcal{O}\left(\frac{V_{\max}}{1-\gamma}\sqrt{\frac{C_\pi}{n}\log\frac{|\mathcal{F}|\cdot|\Pi|}{\delta}}\right),$$

其中 $\mathrm{eb}(C_\pi)$ 为误差界（error bound），其通过公式 (18) 中对 Π 进行 union bound 后得到. 注意，只要我们有 Bellman 完备性（假设 1），即 $\mathcal{T}^\pi f \in \mathcal{F}$，$\forall f \in \mathcal{F}, \pi \in \Pi$，这就是一个有效的置信界.

根据与公式 (24) 相同的逻辑，最大化 $J^-_{\text{cov}}(\pi)$ 的策略 $\widehat{\pi}$ 可以在所需的单策略覆盖条件下，与任意 $\pi_{\text{cp}} \in \Pi$（例如最优策略 $\arg\max_{\pi\in\Pi} J(\pi)$）直接比较：

$$J(\pi_{\text{cp}}) - J(\widehat{\pi}) \le J^-_{\text{cov}}(\pi) + \text{eb}(C_{\pi_{\text{cp}}}) - J^-_{\text{cov}}(\widehat{\pi}) \le \text{eb}(C_{\pi_{\text{cp}}}),$$

第二个不等号是因为 $\widehat{\pi} = \arg\max_{\pi\in\Pi} J^-_{\text{cov}}(\pi)$. 我们将其和定理 5 比较，其误差界将 $\max_{\pi\in\Pi} C_\pi$ 这样一个“全策略覆盖”的条件替换为 $C_{\pi_{\text{cp}}}$ 这样一个“单策略覆盖”的条件!

这样做看似成功消除了对“全策略覆盖”的依赖，事实上这种方法在理论上也是不可行的! 误差界 $\text{eb}(C_\pi)$ 依赖于 C_π. 在多臂赌博机问题中，C_π 是易知的（它是离线数据中选择某个摇臂的概率的倒数）；然而，在 MDP 问题中，C_π 是难以得到的. 因为它涉及到折扣占用分布 d^π，而 d^π 又依赖于 MDP 的动态结构（见引理 2），这在免模型（model-free）学习的场景下通常是不可知的. 因此，下面的几节我们将展示，如何在不知道 C_π 的情况下，仍然获得相同的理论保障.

3.2 通过版本空间的悲观算法

回顾公式 (10)，我们通过在 $f \in \mathcal{F}$ 上最小化如下损失来获得 $\widehat{f}^\pi$：

$$\widehat{\mathcal{E}}(f;\pi) := \max_{g\in\mathcal{F}} \widehat{\mathcal{L}}(f;f,\pi) - \widehat{\mathcal{L}}(g;f,\pi).$$

由于 $\widehat{\mathcal{E}}(f;\pi) \approx \mathcal{E}(f;\pi) = \|f - \mathcal{T}f\|^2_{2,D}$，因此当 $f = Q^\pi$ 时，$\widehat{\mathcal{E}}(f;\pi)$ 很小. 于是，我们可以得到，以至少 $1-\delta$ 的概率，对于所有 $\pi \in \Pi$，都有：

$$Q^\pi \in \mathcal{F}^\pi_{\epsilon_0} := \{f \in \mathcal{F} : \widehat{\mathcal{E}}(f;\pi) \le \epsilon_0\}, \tag{25}$$

其中 $\epsilon_0 = \mathcal{O}\left(\frac{V^2_{\max}}{n} \log \frac{|\mathcal{F}|\cdot|\Pi|}{\delta}\right)$ 是公式 (11) 的 RHS（再加上对 Π 的 union bound），确保 Q^π 不会被排除在外. 我们称 $\mathcal{F}^\pi_{\epsilon_0}$ 为 Q^π 的版本空间（version space）. 这使得我们可以定义 $J(\pi)$ 的一个下置信界：

$$J^-_{\text{VS}}(\pi) := \min_{f\in\mathcal{F}^\pi_{\epsilon_0}} J_f(\pi) \le J_{Q^\pi}(\pi) = J(\pi). \tag{26}$$

类似于在第 3.1 节对 $\arg\max_{\pi\in\Pi} J^-_{\text{cov}}(\pi)$ 的分析，我们需要控制 $J(\pi_{\text{cp}})$ 和 $J^-_{\text{VS}}(\pi_{\text{cp}})$ 之间的差值. 我们结合 $\mathcal{F}^\pi_{\epsilon_0}$ 的定义与公式 (11)，我们可以直接得到

$$\mathcal{E}(f;\pi) \le 2\epsilon_0, \quad \forall f \in \mathcal{F}^\pi_{\epsilon_0}.$$

这也适用于 $f^\pi_{\min} := \arg\min_{f\in\mathcal{F}^\pi_{\epsilon_0}} J_f(\pi)$，因为 $f^\pi_{\min} \in \mathcal{F}^\pi_{\epsilon_0}$. 使用与公式 (15)∼(18) 相同的 telescoping 和误差传播分析，我们可以得到：

$$J(\pi_{\text{cp}}) - J^-_{\text{VS}}(\pi_{\text{cp}}) \le \frac{\sqrt{C_\pi}}{1-\gamma}\sqrt{2\epsilon_0}.$$

将 ϵ_0 的表达式代入，我们即可得到下面定理中的理论保障.

定理 7. *固定任意 $\pi_{\text{cp}} \in \Pi$. 假设：1) 对于 $\pi = \pi_{\text{cp}}$，Bellman 完备性（假设 1）和覆盖条件（假设 2）成立；2) 对于所有 $\pi \in \Pi$，都有 $Q^\pi \in \mathcal{F}$. 那么，策略 $\widehat{\pi} = \arg\max_{\pi\in\Pi} J^-_{\text{VS}}(\pi)$ 满足：以至少 $1-\delta$ 的概率，*

$$J(\pi_{\text{cp}}) - J(\pi) \lesssim \frac{V_{max}}{1-\gamma}\sqrt{\frac{C_{\pi_{\text{cp}}}}{n}\log\frac{|\mathcal{F}|\cdot|\Pi|}{\delta}}.$$

类似地，我们也可以将覆盖条件 $C_{\pi_{\mathrm{cp}}}$ 改进为更加精细的版本，例如 $C^{\mathrm{avg}}_{\pi_{\mathrm{cp}}}$ 等. 与定理 5 相比，我们在这也将“全策略覆盖” $\max_{\pi\in\Pi} C_\pi$ 的要求放宽为“单策略覆盖” $C_{\pi_{\mathrm{cp}}}$. 事实上，即使最优策略未被覆盖，我们依然可以选择任何由数据良好覆盖的对比策略 π_{cp}，从而保证这个界是有意义的（non-vacuous）.

定理 7 也放宽了对表达能力的假设：它仅要求 π_{cp} 满足 Bellman 完备性，以保证 $J^-_{\mathrm{VS}}(\pi_{\mathrm{cp}})$ 是一个紧的下界，这是误差上界中我们需要付出的部分；另一方面，对于其他策略 π，我们只需要 $J_{\mathrm{VS}}(\pi)$ 是*有效的*（valid），也即 $J^-_{\mathrm{VS}}(\pi) \le J(\pi)$ 即可. 对于任意 $\pi\in\Pi$，只要 $Q^\pi\in\mathcal{F}$，我们就有 $\mathcal{T}^\pi Q^\pi = Q^\pi \in \mathcal{F}$，这意味着即使函数类 $\mathcal{F}$ 对 π 不满足 Bellman 完备性，我们仍然有 $\mathcal{E}(Q^\pi;\pi)\approx 0$. 因此，$Q^\pi$ 永远不会被排除在版本空间 $\mathcal{F}^\pi_{\epsilon_0}$ 之外，从而保证了 $J^-_{\mathrm{VS}}(\pi)$ 的有效性.

3.3 一个 Oracle-高效算法：PSPI

前面的算法尽管在覆盖条件取得了显著提升，但仍然仅仅是理论上成立的（与公式 (23) 的方式），因此在这里我们引入一个更加计算友好的版本. 在研究函数逼近的强化学习中，计算效率通常以 **oracle-效率**的形式来表述，即假设如果我们拥有某些优化子程序的黑盒 oracle. 这样一来，只需满足以下两个条件，则计算上是 *oracle-高效的*（oracle-efficient）：

1. 这些子程序在实践中可以被合理近似；
2. Oracle 本身在特定情况下（如表格型问题、线性函数类）可以被高效实现，无需进一步假设.

典型例子如 FQE（Fitted-Q Evaluation）和 FQI（Fitted-Q Iteration）算法，它们的 oracle 假设非常直接，即在函数组 $\mathcal{F}$ 上进行的*最小二乘回归*（least square regression）. 然而，对于版本空间的悲观性，我们需要一个 oracle 来为任意给定的策略 π 计算最坏情况下的 Q-值函数：

$$f^\pi_{\min} := \arg\min_{f\in\mathcal{F}^\pi_{\epsilon_0}} J_f(\pi). \tag{27}$$

我们知道 $f\in\mathcal{F}^\pi_{\epsilon_0}$ 意味着 $\widehat{\mathcal{E}}(f;\pi)\le\epsilon_0$. 因此，这是一个带约束的优化问题，其可以写成如下等价的*拉格朗日形式*（Lagrangian）：

$$\min_{f\in\mathcal{F}} J_f(\pi) + \lambda\widehat{\mathcal{E}}(f;\pi).$$

可以证明的是，当函数组 $\mathcal{F}$ 是基于 d 维特征映射 $\phi(s,a)$ 的线性函数时，上述优化问题 (27) 可以被完全高效地求解. 这是因为，$\lambda\widehat{\mathcal{E}}(f;\pi)$ 之于特征映射 ϕ 是一个二次函数，且 Hessian 为半正定矩阵. 由于 $J_f(\pi)$ 对 f（从而对 ϕ）也是线性的，整个问题事实上构成了一个*凸二次规划*（convex quadratic programming）问题，因此是计算上高效的.

在给出这样的（计算高效的）oracle 后，我们有如下这个在单策略覆盖下的 oracle-高效算法，称作**悲观软策略迭代**（Pessimistic Soft Policy Iteration, PSPI）. 其流程如下：

1. 初始化策略 π_1，使其在动作空间 $\mathcal{A}$ 上均匀随机
2. **For** $k=1,2,\dots,K$:
   a) 使用 oracle 计算 $f_k := f^{\pi_k}_{\min}$
   b) 策略更新：$\pi_{k+1}(a\mid s)\propto\pi_k(a\mid s)\exp\left(\eta f_k(s,a)\right)$
3. 输出策略 $\pi_1,\dots,\pi_K$ 的均匀混合，记为 $\mathrm{Unif}[\pi_{1:K}]$

上述算法中所指的“均匀混合”(uniform mixture)指的是轨迹级别(trajectory-level)的混合. 也就是说，在实际执行策略时，每次生成一条新轨迹之前，会随机均匀采样一个 $i \in \{1, \dots, K\}$，然后用策略 π_i 来生成整条轨迹. 注意：只有在生成新的轨迹时，才会重新采样 i.

通常，这种轨迹级别的混合会导致整体的生成策略不满足马尔可夫性（Markovian），也即依赖于历史，即使 $\pi_1, \dots, \pi_K$ 这些基础策略满足马尔可夫性. 这是因为进行从轨迹到轨迹的切换，我们需要记得上一次用的是哪个 π_i，这便是隐含的历史信息. 因此，这种学习可以被看做是一种不规范学习（improper learning），也即学习过程中允许输出超出原有策略类别的对象. 这种放松在线强化学习中引入无遗憾算法（no-regret algorithm）时很常见. 通过这种均匀混合，一个直接的结论便是：

$$J(\mathrm{Unif}[\pi_{1:K}]) = \frac{1}{K} \sum_{k=1}^{K} J(\pi_k),$$

即均匀混合策略的期望回报是各个策略期望回报的简单平均.

PSPI 算法中关键的两步，除了 a) 步调用公式 (27) 中的 oracle 来计算 $f^{\pi}_{\min}$，我们还需要执行 b) 步策略更新. 此时我们不再单独定义一个策略类，而是像第 2.4 节中的 FQI、FPI 那样，从值函数类 $\mathcal{F}$ 诱导出一个新的隐式策略类. 不同之处在于：FQI、FPI 是基于对 $f \in \mathcal{F}$ 的贪心策略，也即 argmax；这里我们是基于对 $f \in \mathcal{F}$ 的 softmax 操作，即用 softmax 形式组合值函数 $f \in \mathcal{F}$. 形成的隐式策略类如下：

$$\Pi = \left\{ \pi(\cdot \mid s) \propto \exp\left(\eta \sum_{i=1}^{k} f_i(s, \cdot) \right) : 1 \le k \le K, f_{1:k} \in \mathcal{F} \right\}.$$

如果直接朴素地存储这一策略，需要存储 $|\mathcal{S}| \times |\mathcal{A}|$ 个数，也即每个状态-动作对概率的表格型表示，这在大状态空间中是不现实的. 但实际上，在计算 $f_{k+1} = f^{\pi_{k+1}}_{\min}$ 时，我们只需要在数据集 $\mathcal{D}$ 上评估 π_{k+1} 即可！因此，懒惰评估（lazy evaluation）是足够的，即仅在需要计算某个特定状态 s 的策略输出时，再按需计算 $\pi(\cdot \mid s)$. 注意到，以下两种更新方式是等价的：

$$\pi_{k+1}(\cdot \mid s) \propto \exp\left(\eta \sum_{i=1}^{k} f_i(s, \cdot) \right) \quad \Longleftrightarrow \quad \pi_{k+1}(\cdot \mid s) \propto \pi_k(\cdot \mid s) \exp\left(\eta f_k(s, \cdot) \right).$$

这样一来，只要我们存储了之前每个值函数 $f_i \in \mathcal{F}$ 的模型参数，我们就可以（在需要时）高效计算任意动作 a 在状态 s 下的策略输出 $\pi(a \mid s)$，因为归一化这一步骤只会带来与动作空间大小 $|\mathcal{A}|$ 成线性关系的计算复杂度.

我们下面对 PSPI 算法作简单的理论分析，这可以对 softmax 这一策略更新的步骤做更好地解释. 令策略 π' 为任意比较策略 π_{cp}，策略 π 为算法在 K 轮后输出的策略，则有

$$J(\pi_{\mathrm{cp}}) - J(\mathrm{Unif}[\pi_{1:K}]) = \frac{1}{K} \sum_{k=1}^{K} (J(\pi_{\mathrm{cp}}) - J(\pi_k)). \tag{28}$$

使用引理 6 对每一个 $J(\pi_{\mathrm{cp}}) - J(\pi_k)$ 进行误差分解：

$$J(\pi_{\mathrm{cp}}) - J(\pi_k) = \frac{1}{1-\gamma} \left(\mathbb{E}_{s \sim d^{\pi_{\mathrm{cp}}}} [f_k(s, \pi_{\mathrm{cp}}) - f_k(s, \pi_k)] + \mathbb{E}_{d^{\pi_{\mathrm{cp}}}} [\mathcal{T}^{\pi_k} f_k - f_k] + \mathbb{E}_{d^{\pi_k}} [f_k - \mathcal{T}^{\pi_k} f_k] \right).$$

我们需要分别对这三项进行控制. 对于第三项，我们可以“反着”使用引理 2，有：

$$\mathbb{E}_{d^{\pi_k}} [\mathcal{T}^{\pi_k} f_k - f_k] = J(\pi_k) - J_{f_k}(\pi_k) \le 0.$$

这是因为 $f_k = f^{\pi_k}_{\min}$，且 $J_{f_k}(\pi_k) = J^{-}_{\mathrm{VS}}(\pi_k)$ 是 $J(\pi_k)$ 的一个悲观估计（见公式 (26)）.

对于第二项，我们应该注意到，在 Bellman 完备性的前提下，只要是属于版本空间 $\mathcal{F}_{\epsilon_0}^{\pi_k}$ 的函数（包括 $f_k = f_{\min}^{\pi_k}$），它们都满足 $\mathbb{E}_D[(f - \mathcal{T}^{\pi_k} f)^2]$ 是有界的. 这样一来，我们便可以通过“单策略覆盖”这一条件，用 $C_{\pi_{\mathrm{cp}}}^{\mathrm{avg}}$ 来将 $\mathbb{E}_D[(f - \mathcal{T}^{\pi_k} f)^2]$ 一项转化为对 $\mathbb{E}_{d^{\pi_{\mathrm{cp}}}}[(f - \mathcal{T}^{\pi_k} f)^2]$ 的控制.

因此，悲观地选择 $f_k = f_{\min}^{\pi_k}$ 的设计，自动处理了上述误差分解中的第二项和第三项. 现在我们只需要处理第一项，即：

$$\mathbb{E}_{s \sim d^{\pi_{\mathrm{sp}}}} \left[f_k(s, \pi_{\mathrm{cp}}) - f_k(s, \pi_k) \right]. \tag{29}$$

到目前为止，我们尚未使用任何关于 π_k 的性质. 因此，设计 π_k 的唯一目标（即策略更新规则的目标）是：让公式 (29) 尽可能负！理想情况下，我们希望选择 π_k 是对值函数 f_k 的贪心策略，但是这违背了算法的因果顺序，因为 $f_k = f_{\min}^{\pi_k}$ 是以 π_k 作为输入计算出来的！

我们注意到，公式 (29) 是在比较策略的占用分布 $d^{\pi_{\mathrm{cp}}}$ 下取的期望，和每一轮的策略 π_k 无关. 因此，我们在公式 (28) 中直接把所有 K 轮的这类项聚合起来一起分析：

$$\mathbb{E}_{s \sim d^{\pi_{\mathrm{cp}}}} \left[\frac{1}{K} \sum_{k=1}^{K} \left(f_k(s, \pi_{\mathrm{cp}}) - f_k(s, \pi_k) \right) \right].$$

与其在每一轮单独想办法构造一个策略 π_k，使得它最大化 $f_k(s, \cdot)$（这与之前提到的算法的因果顺序矛盾），我们现在可以把任务变得更简单：我们需要设计整个 $\pi_1, \ldots, \pi_K$ 的序列，来最大化 $\sum_k f_k(s, \cdot)$，并将其与某个固定的基准策略 π_{cp} 进行比较！

对于固定的状态 s，这样的设计问题符合一个经典的“无遗憾学习”（no-regret learning）设定. 我们引入**专家问题**（Expert Problem），也即在单纯形上的在线线性优化. 给定一个离散空间 $\mathcal{X}$，我们有如下的一个 K 轮的在线交互流程：在第 k 轮，智能体（agent）选择一个分布 $p_k \in \Delta(\mathcal{X})$，环境选择一个有界的函数 $f_k \in \mathbb{R}^{\mathcal{X}}$（可能是对抗的（adversariial）代价函数）. 我们的目标是要通过提出的分布最小化遗憾（regret），定义为：

$$\mathrm{Regret}(K) = \sum_{k=1}^{K} \left(\mathbb{E}_p[f_k] - \mathbb{E}_{p_k}[f_k] \right),$$

其中 $p \in \Delta(\mathcal{X})$ 为某种静态的基准分布.

接下来，我们把专家问题这样一个框架应用到我们的问题上：离散空间 $\mathcal{X}$ 对应于动作空间 $\mathcal{A}$，分布 p_k 对应于策略 $\pi_k(\cdot \mid s)$，代价函数 $f_k \in \mathbb{R}^{\mathcal{X}}$ 对应于值函数估计 $f_k(s, \cdot) \in \mathbb{R}^{\mathcal{A}}$，基准分布 $p \in \Delta(\mathcal{X})$ 对应于比较策略 $\pi_{\mathrm{cp}}(\cdot \mid s) \in \Delta(\mathcal{A})$.

因此，一些经典的算法，例如镜像下降（Mirror Descent），可以为最终输出提供一个遗憾界（regret bound），便可以直接应用在这里的情形. 实际上，PSPI 算法的 b) 策略更新步骤，就是在每一个状态 s 上单独运行一轮镜像下降！事实上，这是一种乘性权重更新（Multiplicative Weight Update, MWU），也是一种自然策略梯度（Natural Policy Gradient, NPG）方法.

只要选择合适的学习率参数 η，镜像下降的理论可以保证：

$$\frac{1}{K} \sum_{k=1}^{K} \left(f_k(s, \pi_{\mathrm{cp}} - f_k(s, \pi_k)) \right) \lesssim \frac{V_{\max}}{1 - \gamma} \sqrt{\frac{\log |\mathcal{A}|}{K}}.$$

这表明，通过增加迭代轮数 K，我们可以有效控制公式 (29)，即误差分解中的第一项. 这样一来，在 Bellman 完备性和“单策略覆盖”的假设下，我们为误差分解的公式 (28) 中的三项都给出了对应的上界，进而为 PSPI 算法给出了理论保障.

3.4 逐点悲观算法：PEVI

上一节中的 PSPI 算法要求使用一个非平凡的优化 oracle 来计算 $f^{\pi}_{\min} := \arg\min_{f \in \mathcal{F}^{\pi}_{\epsilon_0}} J_f(\pi)$. 尽管在线性 MDP 下，该 oracle 是高效的（可以转化成一个二次规划问题），但我们并不知道在一般情况下该 oracle 是否容易计算. 因此，一个值得探讨的问题是：我们是否可以在更易于计算的 oracle 下，也能实现“单策略覆盖”.

一个这样的例子是**悲观值迭代** (PEssimistic Value Iteration, PEVI) 算法. 它需要一个回归 oracle，对任意 $v \in \mathbb{R}^{\mathcal{S}}$，该 oracle 都可以拟合映射 $(s,a) \mapsto r + \gamma v(s')$，且具备逐点不确定性量化（pointwise uncertainty quantification）. 设 f^* 为 Bayes 最优函数，$\widehat{f} \in \mathcal{F}$ 为我们学得的函数，“逐点不确定性量化”通过一个非负的“bonus 项” $b \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}_{\geq 0}$ 使得以高概率满足

$$f^*(s,a) \in \left[\widehat{f}(s,a) - b(s,a), \widehat{f}(s,a) + b(s,a)\right], \forall (s,a) \in \mathcal{S}\times\mathcal{A}.$$

因此，PEVI 算法便是在具备上述 oracle 的前提下，通过每轮的“bonus 项” $b(s,a)$ 来修改 FQI 算法，从而实现悲观性. 具体地，其流程如下：

1. 初始化悲观值函数为 $f^-_0 \equiv 0$
2. **For** $k = 1, 2, \ldots, K$:

   a) 通过 oracle，使用数据集 $\mathcal{D}$ 拟合回归模型：$(s,a) \mapsto r + \gamma \max_{a'} f^-_{k-1}(s',a')$，得到 $\widehat{f}_k \in \mathcal{F}$ 和对应的“bonus 项” b_k.

   b) 构造悲观值函数估计：$f^-_k(s,a) := \widehat{f}_k(s,a) - b_k(s,a)$.
3. 输出一个非平稳（non-stationary）策略序列 $\pi_{K:1} := \{\pi_K, \ldots, \pi_1\}$，其中 $\pi_k := \pi_{f^-_k}$ 为每一轮输出的策略，也即对悲观值函数 f^-_k 的贪心策略.

我们下面先对 PEVI 算法做简单的分析. 分析的第一步是要证明是要证明悲观性成立，也即 $f^-_k(s,a) \leq Q^{\pi_{k-1:1}}_k(s,a)$. 我们假设“所有的不确定性量化都是有效的”这一高概率事件成立.

悲观性可以通过归纳法来分析，基于 Bellman 算子的单调性（monotonicity）：对于任意函数 $f, f' \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$，$f(s,a) \leq f'(s,a)$，和任意策略 π，都有 $\mathcal{T}^{\pi} f \leq \mathcal{T}^{\pi} f'$. 基础情况是，$f^-_0 \equiv Q^{(\cdot)}_0 \equiv 0$ 显然成立. 基于归纳假设 $f^-_{k-1} \leq Q^{\pi_{k-2:1}}_{k-1}$，便有：

$$f^-_k \leq \mathcal{T} f^-_{k-1} = \mathcal{T}^{\pi_{k-1}} f^-_{k-1} \leq \mathcal{T}^{\pi_{k-1}} Q^{\pi_{k-2:1}}_{k-1} = Q^{\pi_{k-1:1}}_k.$$

上述第一步是因为不确定性量化的有效性，以及 $f^*_k = \mathcal{T} f^-_{k-1}$ 是第 k 步回归所逼近的 Bayes 最优目标；第二步是因为 π_{k-1} 对 f^-_{k-1} 是贪心策略，因此 Bellman 算子 $\mathcal{T}^{\pi_{k-1}}$ 和 Bellman 最优算子 $\mathcal{T}$ 对于 f^-_{k-1} 有相同的效果；第三步是因为归纳假设和 Bellman 算子的单调性；第四步是基于有限时域版本的 Bellman 方程.

于是，我们便可以使用有限时域版本的引理 6 进行误差传播分析：对于任意比较策略 π_{cp}，

$$\begin{aligned} J_K(\pi_{\mathrm{cp}}) - J_K(\pi_{K:1}) = \sum_{t=0}^{K-1} \gamma^t \Big(& \mathbb{E}_{s\sim d^{\pi_{\mathrm{cp}}}_t}[f^-_{K-t}(s,\pi_{\mathrm{cp}}) - f^-_{K-t}(s,\pi_t)] \\ & + \mathbb{E}_{d^{\pi_{\mathrm{cp}}}_t}[\mathcal{T}^{\pi_t} f^-_{K-t-1} - f^-_{K-t}] + \mathbb{E}_{d^{\pi_{K:1}}_t}[f_{K-t} - \mathcal{T}^{\pi_t} f_{K-t-1}] \Big). \end{aligned}$$

由于策略 $\pi_{K:1}$ 是对 $f^-_{K:1}$ 的贪心策略，因此对任意的 $s \in \mathcal{S}$，均有 $f^-_{K-t}(s,\pi_{\mathrm{cp}}) \leq f^-_{K-t}(s,\pi_t)$，第一项 ≤ 0；由悲观性可知第三项 ≤ 0；因此只剩下第二项：

$$J_K(\pi_{\mathrm{cp}}) - J_K(\pi_{K:1}) \leq \sum_{t=0}^{K-1} \gamma^t \mathbb{E}_{d^{\pi_{\mathrm{cp}}}_t}\left[\mathcal{T} f^-_{K-t-1} - f^-_{K-t}\right],$$

这里我们将 Bellman 算子替换成了 Bellman 最优算子，因为它们对 f_k^- 的作用是相同的（π_k 是对 f_k^- 的贪心策略）. 因此，我们可以再次利用不确定性量化来界定期望中的 Bellman 误差：

$$\mathcal{T} f_{K-t-1}^- = f_{K-t}^* \le \widehat{f}_{K-t} + b \le f_{K-t}^- + 2b.$$

于是我们得到如下的理论保障：

$$J_K(\pi_{\mathrm{cp}}) - J_K(\pi_{K:1}) \le 2 \sum_{t=0}^{K-1} \gamma^t \mathbb{E}_{(s,a)\sim d_t^{\pi_{\mathrm{cp}}}}[b(s,a)]. \tag{30}$$

注意到，公式 (30) 中的理论保障依赖于 $b(s,a)$，即 “bonus 项” 的紧致性. 下面我们将考虑一些具体的设定，在这些设定中，$b(s,a)$ 可以显式计算，且公式 (30) 的 RHS 可以被和之前章节中类似的表达式进一步控制.

首先，和 FQI/FQE 一样，我们需要一些假设，以确保我们所进行的每一次回归问题都是良设的. 注意在这里，基于 Bellman 最优算子的 Bellman 完备性（也即 $\mathcal{T} f \in \mathcal{F}, \forall f \in \mathcal{F}$）是不够的！因为我们并不是对 $\mathcal{F}$ 中的函数作用 Bellman 最优算子，而是对 $f_k^- = \widehat{f}_k - b$ 这个悲观值函数——由于减去了 “bonus 项” b，所以 f_k^- 可能落在 $\mathcal{F}$ 之外. 因此我们假设一个更强的条件，即 $\mathcal{T} f \in \mathcal{F}$，$\forall f \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$，以避免这个问题（如图 1 所示）.

然后，我们需要对每个预测值提供一个逐点置信区间（pointwise confidence interval）. 一个典型的设定是线性设定：当函数类 $\mathcal{F}$ 是由特征映射 ϕ 所诱导的线性函数类，即 $\mathcal{F}_\phi = \{\langle \phi, \theta\rangle : \theta \in \mathbb{R}^d\}$ 时，我们可以用岭回归（ridge regression）来计算点估计 $\widehat{f}_k$，同时使用一个二次型的 “bonus 项” 来构造不确定性量化：

$$b(s,a) = \frac{\beta}{\sqrt{n}} \sqrt{\phi(s,a)^\top \left(\Sigma_{\mathcal{D}}^{\mathrm{ridge}}\right)^{-1} \phi(s,a)}, \tag{31}$$

其中 β 为可调整的参数，通常取决于特征维度 d 和时域 H，而 $\Sigma_{\mathcal{D}}^{\mathrm{ridge}}$ 为正则化后的协方差矩阵：

$$\Sigma_{\mathcal{D}}^{\mathrm{ridge}} = \frac{1}{n}\left(\sum_{(s,a)\in\mathcal{D}} \phi(s,a)\phi(s,a)^\top + I\right).$$

我们之前定义的线性 MDP 就满足上面所需的所有条件. 事实上，它们是等价的：

命题 8. *如果 $\mathcal{F} - \mathcal{F}_\phi$ 是由特征映射 ϕ 所诱导的线性函数类，且满足 $\mathcal{T} f \in \mathcal{F}, \forall f \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$，那么该 MDP 一定是一个以 ϕ 为特征的线性 MDP，即 $f^* = \phi^\top \theta^*$.*

将公式 (31) 代入公式 (30)，我们得到：

$$J_K(\pi_{\mathrm{cp}}) - J_K(\pi_{K:1}) \le \frac{2\beta}{(1-\gamma)\sqrt{n}} \left((1-\gamma)\sum_{t=0}^{K-1} \gamma^t \mathbb{E}_{(s,a)\sim d_t^{\pi_{\mathrm{cp}}}}\left[\sqrt{\phi(s,a)^\top \left(\Sigma_{\mathcal{D}}^{\mathrm{ridge}}\right)^{-1} \phi(s,a)}\right]\right).$$

括号中的表达式在 PEVI 中起到了 “覆盖性”（coverage）的作用. 尽管有一些微小差异，但其平方大致为：

$$\left(\mathbb{E}_{(s,a)\sim d^{\pi_{\mathrm{cp}}}}\left[\sqrt{\phi(s,a)^\top \Sigma_D^{-1} \phi(s,a)}\right]\right)^2, \tag{32}$$

其中 $d^{\pi_{\mathrm{cp}}}$ 为策略 π_{cp} 的占用分布，而 $\Sigma_D = \mathbb{E}_{(s,a)\sim d^D}[\phi(s,a)\phi(s,a)^\top]$ 为数据的协方差矩阵. 注意到，公式 (32) 非常接近于 $\sigma_{\max}(\Sigma_\pi^{1/2}\Sigma_D^{-1}\Sigma_\pi^{1/2})$，也就是第 2.3 节公式 (20) 中定义的平方误差版的覆盖参数 $C_\pi^{\mathrm{sq}}(\mathcal{F}_{\phi^*})$ 的上界！

事实上，如果我们将公式 (32) 中的平方根拿到外面，那么表达式就变为 $\mathrm{tr}\left(\Sigma_\pi^{1/2}\Sigma_D^{-1}\Sigma_\pi^{1/2}\right)$，这可以成为最大奇异值 $\sigma_{\max}$ 的上界（但在最坏情况下，可能相差一个维度因子 d）. 另一方面，如果我们将平方根保留在期望内部，会使得公式 (32) 严格小于 $\mathrm{tr}\left(\Sigma_\pi^{1/2}\Sigma_D^{-1}\Sigma_\pi^{1/2}\right)$. 因此，公式 (32) 和公式 (20) 的关系并不是严格清晰的.

3.5 深度强化学习中的悲观性

我们在第 3.3 节看到的 PSPI 算法和在第 3.4 节看到的 PEVI 算法，在线性设定下都是可以高效计算的. 一个自然的问题是：它们是否可以在更实际的函数逼近机制中进行实现，例如深度强化学习中使用的深度神经网络. 我们在这一部分将简要回顾这些方法在理论和实践之间的差距.

- **PSPI**. PSPI 的主要问题在于，其核心的 oracle（公式 (27) 中定义的）本质上是一个极小极大优化，这在实践中很难实现. 实际上，大多数试图最小化 Bellman 误差的算法都需要通过极小极大优化来解决双重采样问题（见第 2.2 节）. 虽然已有一些实证研究常使用深度神经网络来实现这类方法（且在某些问题上也取得了成功），但这种极小极大优化过程非常难以调参、其优化性质与经典的动态规划算法非常不同.

  在实践中，PSPI 的实现往往仍会采用 DP/TD 式的值函数更新方式. 但这会破坏其理论保障！例如，在有限时域问题中，DP 是*自底向上的*（bottom-up），也即其从后向前拟合值函数——这会冻结后面阶段的值函数，导致前面阶段不能通过悲观策略向后传播不确定性量化，从而阻碍初始时间步的悲观策略对后续估计产生影响.

  PSPI 的另一个实践障碍在于其*自然策略梯度*（Natural Policy Gradient, NPG）式的策略更新. 其需要保留所有历史的值函数迭代，在实践中可能造成很大的存储负担. 此外，NPG 更新所诱导的策略类不再是一个独立的策略类，它需要依赖值函数的组合，也就限制了其在大动作空间的可扩展性.

- **PEVI**. 与 PSPI 不同，PEVI 被设计成一个动态规划算法，因此我们可以非常方便地将其中的回归 oracle 设置为使用神经网络的回归. 不过，问题在于“bonus 项”的设计，它需要实现逐点的不确定性量化，而这在超出线性回归的情况下非常难以获得.

  因此，一个常用的启发式方法是：用神经网络进行回归，然后取网络的最后一层作为特征 ϕ，再利用它来计算公式 (31) 中二次型的“bonus 项”. 但这样做破坏了原有的理论保障！

References

[1] Jiang, N. and Xie, T. (2025). Offline Reinforcement Learning in Large State Spaces: Algorithms and Guarantees. Invited submission under review at *Statistical Science (STS)*.

[2] Antos, A., Szepesvári, C. and Munos, R. (2008). Learning near-optimal policies with Bellman-residual minimization based fitted policy iteration and a single sample path. In *Machine Learning* **71**, 89–129.

[3] Ernst, D., Geurts, P. and Wehenkel, L. (2005). Tree-based batch mode reinforcement learning. In *Journal of Machine Learning Research* **6**, 503–556.

[4] Jiang, N., Krishnamurthy, A., Agarwal, A., Langford, J. and Schapire, R. E. (2016). Contextual decision processes with low Bellman rank are PAC-learnable. In *Proceedings of the 34th International Conference on Machine Learning* **70**, 1704–1713.

[5] Jin, C., Liu, Q. and Miryoosefi, S. (2021). Bellman eluder dimension: New rich classes of RL problems, and sample-efficient algorithms. In *Advances in Neural Information Processing Systems* **34**, 13406–13418.

[6] Jin, C., Yang, Z., Wang, Z. and Jordan, M. I. (2020). Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, 2137–2143.

[7] Jin, Y., Yang, Z. and Wang, Z. (2020). Is Pessimism Provably Efficient for Offline RL? In *Proceedings of the 38th International Conference on Machine Learning*, **139**, 5084–5096.

[8] Precup, D., Sutton, R. S. and Singh, S. P. (2000). Eligibility Traces for Off-Policy Policy Evaluation. In *Proceedings of the Seventeenth International Conference on Machine Learning*, 759–766.

[9] Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. MIT Press.

[10] Tsitsiklis, J. N. and Van Roy, B. (1996). Feature-based methods for large scale dynamic programming. In *Machine Learning* **22**, 59–94.

[11] Xie, T., Cheng, C.-A., Jiang, N., Mineiro, P. and Agarwal, A. (2021). Bellman-consistent Pessimism for Offline Reinforcement Learning. In *Advances in Neural Information Processing Systems*, **34**, 6683–6694.

[12] Xie, T., Foster, D. J., Bai, Y., Jiang, N. and Kakade, S. M. (2023). The Role of Coverage in Online Reinforcement Learning. In *The Eleventh International Conference on Learning Representations*.