强化 MENT 学习的变分预测指导

匿名作者
双盲评审中的论文

摘要

如何做出智能决策是机器学习和人工智能中的一个中心问题。尽管最近深度强化学习(RL)在各种决策问题上取得了成功，但一个重要但未得到充分探索的方面是如何利用 oracle 观察(在线决策过程中不可见的信息，但在离线培训过程中可以获得)来促进学习。例如，人类专家将在一场扑克游戏后查看重播，在其中他们可以检查对手的手牌，以改善他们在玩游戏时从可见信息对对手手牌的估计。在这项工作中，我们基于贝叶斯理论研究这类问题，并使用变分法推导出一个在 RL 中利用 oracle 观察的目标。我们的主要贡献是为 DRL 提出了一个称为变分潜在预言引导(VLOG)的通用学习框架。VLOG 具有更好的特性，如其稳健和有前途的性能，以及与任何基于值的 DRL 算法相结合的多功能性。我们使用从视频游戏到具有挑战性的基于瓷砖的游戏麻将的决策任务，通过经验证明了在线和离线 RL 域中 VLOG 的有效性。此外，我们发布了麻将环境和一个离线 RL 数据集作为基准，以促进未来对 oracle guiding 的研究(https://github . com/py 麻将/py 麻将)。

1 介绍

深度强化学习(DRL)近年来发展迅速(Sutton &Barto, 2018; Mnih et al., 2015; Vinyals et al., 2019).然而，RL 中有一个常见且重要但未得到充分开发的方面:想象一下，在玩了一场扑克游戏后，一个人类玩家可能会查看重播来检查对手的牌，并分析这些信息以改进他/她的下一次游戏策略(或策略)。我们将对手的手这样的信息称为 oracle observation，定义为在在线任务执行期间对代理不可见但在离线训练期间可用的信息。相比之下，任务执行期间可用的信息称为执行者观察。这种情况被称为 oracle 指南(Liet al., 2020; Fang et al., 2021) (见第二节。3 用于正式定义)。甲骨文导读在现实生活中很常见。比如考试的时候(甲骨文观察是类似问题的答案，只有在备考的时候才有)；以及训练一个机器人在月球上执行一些任务(在训练机器人的时候，我们可以给它提供领地的信息，这些信息在执行的时候是没有的)。oracle 观察的类型可以是多种多样的，包括事后诸葛亮的信息(Harutyunyan et al., 2019; Guez et al., 2020)，人类反馈(Knox & Stone, 2009;Loftin et al., 2016; MacGlashan et al., 2017)，通过后处理重新校准数据，并在部分观察设置中隐藏状态(Li et al., 2020).

虽然人类在学习做决策时会自然地执行 oracle 指导，但在 RL 中仍然具有挑战性。困难包括:(1)如何保证使用 oracle 观察的学习改进仅使用执行者观察的主要决策模型，以及(2)如果引入利用 oracle 观察的辅助损失，如何在主要损失和辅助损失之间进行权衡。虽然最近的研究试图在 RL 中模拟 oracle guiding(Guez et al., 2020; Li et al., 2020;Fang et al., 2021)，它们都没有解决这些困难(更多细节请参考相关工作部分)。特别是，所有这些建议的方法都是启发式的:尽管经验结果显示使用 oracle 指导可以提高性能，但理论上并不能保证使用 oracle 观察可以提高执行性能。

在这篇文章中，我们提出了一个全新的基于贝叶斯理论的预言引导思想。以扑克为例，我们知道，如果知道环境的全局真实状态(或简单的状态),学习最佳策略是容易的[1]，包括所有可见或不可见的牌，对手的打法等。(Azar et al., 2017; Jin et al., 2018; 2020). 技能提高的一个关键部分是学会从执行者的观察中估计环境状态的概率分布。人类专家完成这项工作的常用方法是在可以看到预测观察(例如对手的手)的情况下观看比赛回放，然后使用预测估计状态来校正执行者估计状态。我们将此解释为贝叶斯语言:执行者估计的状态作为先验分布，而先知估计的状态作为后验分布。因此，训练目标可以被认为是双重的:学习基于状态的后验估计做出决策，以及学习更接近后验状态的先验状态分布。

我们提出了一个新的基于变分贝叶斯(VB)的学习框架来解决一般的预言引导问题(Kingma & Welling, 2014)，称为变分潜在甲骨文指导(VLOG)。VLOG 拥有几个更好的特性。首先，理论上保证 VLOG 利用 oracle 观察来改进使用执行者观察的决策模型。第二，VLOG 是一个通用的 DRL 框架，可以集成到任何基于值的 RL 算法中，并且与 oracle 观察的类型无关。第三，VLOG 不需要调整额外的超参数。最后，我们从经验上证明了 VLOG 有助于在线和离线 RL 领域中各种决策任务的更好表现。这些任务包括一个简单的迷宫导航，玩视频游戏，以及一个特别具有挑战性的基于瓷砖的游戏麻将，其中人类在学习中大量利用甲骨文的观察(Li et al., 2020).我们还将麻将作为 oracle guiding 的基准测试任务，并发布相应的模拟环境和数据集，以方便未来的研究，从而为社区做出贡献。

2 相关著作

在过去的几年里，对 DRL 和模仿学习的研究兴趣有所增长(Chen et al.,2020) 利用先知或后见之明的信息。对 DRL 来说，Guez et al. (2020); Fang et al.(2021) 将事后观察(执行者对未来步骤的观察)视为培训期间的先知观察。Guez et al. (2020) 使用事后观察来帮助学习当前状态的表述。另一种方法(Fang et al., 2021) 用于股票交易:作者用后见之明的信息训练了一个教师(oracle)策略，并使用网络提取使学生策略的行为更类似于教师策略。这两种方法(Guezet al., 2020; Fang et al., 2021) 是启发式的，集中于利用未来的观察来进行更好的顺序建模，而 VLOG 理论上保证适用于任何类型的 oracle 观察。对于不完全信息博弈的应用(Li et al., 2020)，一个基于 DRL 的人工智能麻将，也介绍了一种方法来利用甲骨文观察(对手的手)更强的性能。它们将 oracle 观察与 executor 观察连接起来作为策略网络的输入，其中 oracle 观察由一个标量变量计时，该变量在训练过程中从 1 退火到 0。但是，中使用的方法 Li et al. (2020) 也是启发式的，只在一个任务中测试过。

变分贝叶斯(VB)是一种成熟的方法，并已在 RL 中得到利用。例如，作为概率推理的控制使用 VB 来连接 RL 的目标函数和概率推理问题的变分下界(Furmston & Barber, 2010; Weber et al.,2015; Levine, 2018). 我们的想法不同，因为它通过最大似然问题构建了一个值回归问题，然后，它应用 VB 来解决它(参见第。4). 此外，VBian 网络模型在 DRL 的应用最近引起了研究者的注意。例如，Ha & Schmidhuber(2018) 提出采用 VAE 来降低图像观察的高维度；Igl et al.(2018); Han et al. (2020); Lee et al. (2020) 提出了变分 RNNs 作为状态转移模型来编码主体的信念状态；Yin et al. (2021) 利用变分序贯生成模型来预测未来观测值，并使用预测误差来推断内在奖励以鼓励探索；和 Okada et al. (2020) 通过在连续控制任务中使用深度贝叶斯规划模型展示性能增益。我们的研究与前面提到的工作不同，它关注于 oracle guiding，并且 VLOG 不涉及学习状态转换模型。

[1] 通常，oracle 观测不一定包含环境状态的所有信息。

3 ORACLE 指导 POMDP

这里我们基于部分可观测马尔可夫决策过程(POMDP)来定义这个问题(Sondik, 1978). 指导 POMDP 的 oracle 与原始 POMDP 的区别在于它有两种类型的观察:执行者和 oracle。执行者观察值 x 对
代理，代理的决策(执行)依赖于此。oracle 观察值 x-在执行过程中是不可访问的，但可以在执行后获得。x 包含在 x ˇ 中，因为前者总是可用的。因此，x-包含的底层环境状态信息并不比 x 少。

oracle 指导 POMDP 的正式定义是一个元组(S，a，P0，t，x，x ˇ，o，o ˇ， γ)，其中 s 和 a 分别是状态和动作空间。P0 指定初始状态分布，使得 P0(s)是状态 s ∈ S 是初始状态的概率。t 指定了状态转换
概率使得 T(S′ ，r|s，a)是到达一个新状态 S′ ∈S 的概率
在状态 s ∈ S 采取行动 a ∈ A 后的即时报酬 R∈R X 表示执行者观察空间，X ˇ 表示甲骨文观察空间。O 指定执行者观察概率，因此 O(x|s)是处于状态 s ∈ S 的执行者观察 x ∈ X 的概率。类似地，O ˇ 指定 oracle 观察概率，因此 O ˇ (X | s)是
状态为 s ∈ S 时的 oracle 观察值 x ˇ ∈x ˇ 。 γ ∈[0，1]是折扣因子。价值函数是
特定状态或状态-动作对的预期回报值。对于一个策略 π，它的 Q 函数定义为 qπ(s，a) := Eπ[∞n=0 γnrt+n|st = s，at = a]，其中 Eπ 表示遵循策略 π 时的期望。状态值函数 vπ 定义为 vπ(s) := Eπ(a s)[qπ(s，a)]。主体的目标是使 E(s)vπ(s)相对于 π 最大化，而价值函数在这个目标上起着重要的作用(Sutton & Barto, 2018).

4 VLOG:变异的潜在神谕指引

让我们引入一个潜在向量 zt，它是一个表示环境状态 st 的概率变量。从贝叶斯的角度来看，我们考虑先验分布 p(zt xt)作为基于执行者观察 xt 的 zt 的代理的估计概率密度函数(PDF)。同时，后验分布 PDF q(ZT | x ´ t)基于 oracle 观测值 x ´ t 建模。

在 RL 中，最基本的要求是基于可用信息，即执行者观察 xt，通过价值函数 approx-imator v(xt):= v(ZT)p(ZT xt)dzt(我们用 v 表示它，但它也可以是 Q 函数，只需用(XT，at)代替 XT)来对回报进行良好的估计。由 vtar 表示的回报目标可以通过任何价值学习算法来估计，例如 TD(0)(Sutton
& Barto, 2018)，彭的 Q(λ)(Kozuno et al., 2021)，等等。(一般情况下，vtar 总是可以由 Bell- man 方程给出。然而，如果可用，也可以使用蒙特卡罗返回作为 vtar)。特别是，我们在决斗架构中采用了双 Q 学习(Wang et al., 2016) 计算 vtar 的有效性和简单性(第。5 和 B.1)。我们希望最大化基于执行者观察 xt(即，对于执行者模型)的回报估计的对数似然目标

$$l := \log P\, v(XT) = vtar \mid XT = \log \int P\, v(ZT) = vtar \mid ZT\, P(ZT \mid XT)\, dzt$$

$$= \log \frac{q(ZT \mid x\check{}\, t)P\, v(z) = vtar \mid z\, P(z \mid x)dz。}{}$$

根据詹森不等式，我们有

$$L \geqslant \int zt \quad \underset{q\text{值对数}}{} \quad \frac{p(v(ZT) = vtar \mid ZT)p(ZT \mid XT)q(ZT \mid x\check{}\, t)}{} \quad \underset{德国国家旅游局}{}$$

$$= \int q(z \mid x\check{}\,)\log P\, v(z) = vtar \mid z\, q(z \mid x\check{}\,)\log \frac{q(ZT \mid x\check{}\, t)}{} \quad \underset{p(zt|}{}$$

$$= eq(z\ \hat{x})\log p\, v(ZT) = vtar \mid ZT\ dkl(q(ZT \mid x\check{}\, t)p(ZT \mid XT)) := LV$$

log。

$$\underbrace{\qquad}_{\text{oracle pred¸i操作错误}} \quad \underbrace{\qquad}_{x} \quad \underbrace{\qquad}_{\text{regulariz¸a术语}} \quad \underbrace{\qquad}_{x}$$

因此，我们可以通过最大化 LVLOG 来最大化我们的初始目标 L，LVLOG 也称为变差下限(Kingma & Welling, 2014)，但是在我们的先知指引计划中。自 p(zt|xt)

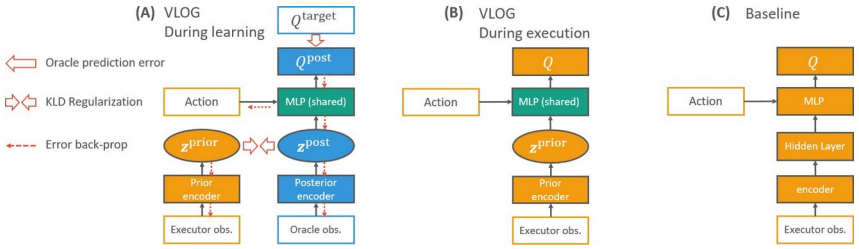

图 1:在(A)学习和(B)执行期间用于深度 Q 学习的 VLOG 的实现。Executor observation xt 和 oracle observation xˆ 使用两个不同的编码器，但共用一个解码器(MLP 用于估计 Q 函数)。(C)不使用 oracle 观察的基线模型。一个长方形
表示确定性层/变量，椭圆表示随机层/变量。

并且 q(zt xˇ t)分别表示从执行者观察和 oracle 观察获得的潜在向量的 PDF，VLOG 中的两个术语的含义现在看起来很清楚:第一个术语，即 oracle 预测误差，有助于改进从后验潜在状态分布(ZT)的值估计
用甲骨文观察计算)；第二项，即正则化项，有助于使 zt 的先验表示更接近于后验表示，作为潜在的神谕引导。我们要强调的是，VLOG 目标是以前的执行者模型 v(xt)的目标下限(使用执行者观察 xt 估计回报)，使用 oracle
这个下限保证了 oracle 观察的使用是积极的
影响了执行者模型的学习，这也是我们最终想要的。现在我们已经推导出
VLOG 的理论目标，下一节将解释它在实践中的实现。

备注 1。可以使用任何形状的近似后验 q，这取决于哪个不同的 VLOG 实例是可能的。此外，可以直接使用 v(xt)来代替 p(zt xt)。这些设计选择允许用户结合任何关于 oracle observation 的先验知识。例如，如果已知 xt 处的状态值的范围是封闭区间[1，u]，则近似后验 q(vt | xˇ t)可以被限制为[1，u]上支持的概率分布族。

4.1 用神经网络实现

受变分自动编码器(VAE，Kingma & Welling (2014))，我们提出了 VLOG 的神经网络架构(图。1)．执行器观察 xt 和 oracle 观察 XT 由两个不同的编码器网络处理，并计算先验和后验
潜在向量的先验分布。在训练期间，xt 和 XT 都可用，并且
通过以端到端的方式最大化 VLOG 目标来更新所有网络参数(图。1A)．在执行期间，代理为决策计算先验分布 p(z xt)(图。1B) 不使用 oracle observation。zt 由参数化正态分布计算

$$p(zt|xt) = N(\ p,\ exp\ (log\ \sigma p)),\ (p,\ log\ \sigma p) = 先验编码器(xt),$$
$$q(ZT\ |\ x\check{}\ t) = N(q,\ exp\ (log\ \sigma q)),\ (q,\ log\ \sigma q) = 后验编码器(x\check{}\ t)。$$

用于计算等式中的(v(zt) = vtar pt)。1，我们简单地假设它遵循正态分布，并在实践中使用 v(zt)和 vtar 之间的均方误差来估计它。在 VAE，重新参数化技巧用于执行端到端的训练(Kingma & Welling, 2014)．那么解码器的输出(值函数)就可以由 v(zt) = decoder(zt)得到。注意，zt 是在训练期间使用后验编码器获得的，而在执行期间使用前验编码器获得的。

4.2 任务无关的潜在瓶颈控制

为了学习更好的表示，我们借用了 β-VAE 的思想(Higgins et al., 2016) 将系数 β 应用于正则项。因此，我们的损失函数(负下限)为

$$J = -eq(z\ \ x\hat{})\log\ p\ v(ZT) = vtar\ |\ ZT + \beta\, dkl(q(ZT\ |\ x\check{}\ t)p(ZT\ |\ XT))。(2)$$

超参数 β 控制着潜在信息瓶颈的容量(Tishby & Za-slavsky, 2015; Alemi et al., 2017).我们发现 β 的选择对的性能很重要

RL 中的 VLOG(参见附录 B.3). 然而，不希望有额外的超参数。受中所用方法的启发 Burgess et al. (2017) 为了控制 β-VAE 中 KL 散度的大小，我们提出了一种任务无关的方法，通过设置 KL 散度 Dtar 的目标来自动调整 β 。特别地，我们最小化辅助损失函数(β 作为优化参数)

$$j\beta = (Dtar - D_{KL}(q(ZT \mid x \sim t) \| p(ZT \mid XT)))\log(\beta)。 \quad (3)$$

这里的直觉是，当先验和后验散度过大时，通过增加 β 来加强正则化，反之亦然。该方法类似于在软演员-评论家中用于自动调整熵系数的方法(Haarnoja et al., 2019), 但是我们以前是吉隆坡的
发散系数。重要的是，我们发现一个性能良好的值 Dtar = 50，与其他设计选择无关。它在一系列不同的任务和网络中运行良好。5). 因此，我们做到了
不需要调整 β 。我们在附录中提供了关于这种方法的更多讨论 D.

5 实验

VLOG 在实践中表现如何？我们调查了 VLOG 在三种类型的任务中使用在线或离线 RL 的经验表现，从简单到困难。在下面的实验中，我们使用了双 DQN 和决斗网络架构(van Hasselt et al., 2016; Wang et al., 2016) 作为基础 RL 算法，RL 的模型和损失函数在附录中定义。B.1。由于 DRL 易受超参数选择的影响，引入任何新的超参数都可能掩盖 oracle 指导的效果。与其他 DQN 变体相比，双 DQN 和决斗架构更适合基本算法，因为它们不需要额外的超参数(Hesselet al., 2018), 例如优先体验重放(Schaul et al., 2016), 噪声网络(Fortunatoet al., 2018),绝对 DQN(Bellemare et al., 2017),和分布式 RL(Kapturowski et al.,2018). 重要的是，我们尽可能对所有方法和环境使用相同的超参数设置(参见附录 B.2).

5.1 迷惑

我们首先展示了 VLOG 如何通过利用学习中的 oracle 观察来帮助形成潜在的表征。试验台是一个迷宫导航任务[2] (图 2A) 有 10 ^ 10 个格子。执行者观察是(x, y)位置，其中 x，y 是在迷宫的每个网格内随机采样的连续值(因此，在两个相邻但被墙隔开的网格中的观察可能非常接近)。在每一步，代理选择一个动作(向上、向下、向右或向左),如果没有被墙阻挡，则移动到另一个网格。

我们在训练时为 VLOG 代理提供了 oracle observation (xc，yc，dg)，其中 xc，yc 是当前网格中心的坐标，dg 是从当前网格到目标的(最短)路径距离。直观地说，虽然原始观察值是(x，y)，但 dg 在迷宫导航任务中更重要。我们实证研究了在学习 VLOG 时使用的甲骨文观察在多大程度上有助于形成 z w r t DG 的潜在表征，而不是位置。编码器和解码器都是具有宽度 256 和 ReLU 激活的 2 层多层感知器(MLP)。VLOG 的潜在向量 zt 的大小是 128，因为我们计算了和 σ (附录 C).

实验表明，基线代理努力达到目标(图。2B)，而 VLOG 代理在学习后稳定地解决了任务。为了检查使用 VLOG 如何影响学习的潜在表示，我们用主成分分析 (PCA，Pearson F.R.S. (1901)).在图 1 中 2C, 我们将到目标 dg 的路径距离映射到颜色，并绘制 VLOG 的 z 的前 2 个分数(使用执行者观察计算)和基线的相应隐藏状态[3] (图 1 中的“隐藏层”。1C). 与基线相比，VLOG 的潜在状态显示了相对平滑和可解释的目标距离表征。然后我们在图 2 中画出迷宫中不同位置的潜在表征。2D.VLOG 的潜在状态更清楚地代表 dg，与结果一致(图。2C).

具体来说，我们研究了一个矩形区域(在图 1 中用矩形表示)。2D) 其中左侧的 2 个网格和右侧的 2 个网格被一堵墙隔开。我们在潜伏中找到了相应的区域

2 https://github . com/MattChanTK/gym-mazehttps://githu

3 由于基线并不总能成功达到目标，我们仅使用成功的试验。

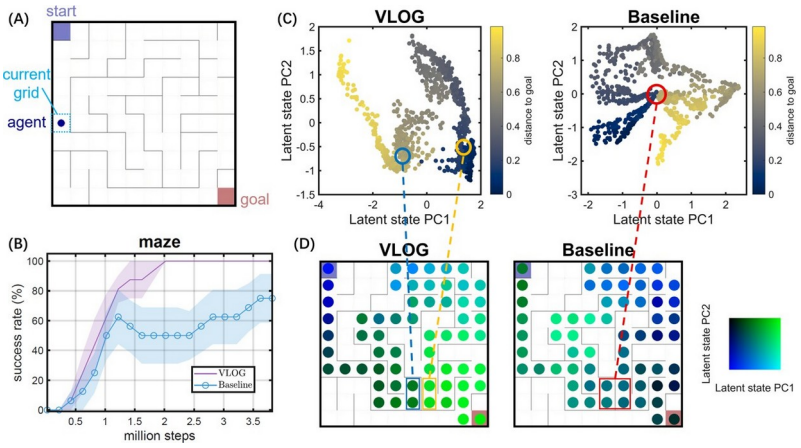

图 2:在一个简单的迷宫中通过甲骨文引导学习潜在的表征。(A)任务说明。(B)使用 VLOG 和不使用 VLOG(基线)的成功率，各使用 8 个随机种子。
(VLOG 中 z 的 PCA 和基线模型中对应的隐藏状态(z 在执行过程中收集，不使用 oracle 观察)。颜色表示到目标的标准化路径距离。用 oracle 观察训练的 VLOG 的潜在状态显示了更有序和可解释的表示，即 w.r.t .到目标的路径距离。我们圈出了对应于迷宫中(x，y)位置接近但距离目标不同的一些地方的潜在区域。(D)学习的潜在表征在迷宫中的位置，其中颜色的绿色和蓝色分量分别对应于潜在状态的前 2 个 PCs 的分数，如色图所示(相似的颜色表示相似的表征)。

PC 空间，并在图中圈出它们。2C. 虽然这 4 个网格在(x，y)上很接近(执行者观察)，但是它们到目标的距离(oracle 观察)是非常明显的。通过利用使用 VLOG 的 oracle guiding，代理可以清楚地区分潜在空间中的左侧 2 个网格和右侧 2 个网格，如图所示。2C，d，左(注意，这里 VLOG 的潜在状态 z 是仅使用执行者观察来计算的)。相比之下，这些网格的潜在表示对于基线模型是重叠的，基线模型没有利用甲骨文观察(图。2C，d，右)。总之，我们用一个玩具示例演示了 VLOG 有效地帮助潜在空间与对任务有用的 oracle state 耦合。接下来的部分将转移到更复杂任务的实验，并讨论 VLOG 如何提高实际性能。

5.2 嘈杂的米娜达

为了评估 VLOG 如何扩展更高维的状态空间，我们在一组 MinAtar 视频游戏上测试了它。米纳塔(Young & Tian，2019) 是一个人工智能代理的测试平台，它实现了 5 个具有离散动作的小型 Atari 2600 游戏(seaquest、breakout、space invaders、freeway 和 asterix)。MinAtar 的灵感来自街机学习环境(Bellemare et al.，2013)但是为了效率简化了环境。观察值为 10×10 像素，多个通道指示不同的对象。

在现实世界中，观察结果通常包含一些噪声。因此，很自然地认为有噪声的观察是部分可观察的执行者观察，而原始的、无噪声的观察是先知观察。假设在每一帧，每个像素可能以 1/8 的独立概率随机“断裂”(图。3A). 在破损像素处的原始观察被擦除，并且在所有通道中被不同的值代替。我们考虑这样的噪声 MinAtar 环境，其中噪声像素作为执行者观察，原始像素作为先知观察。

网络结构与 Maze 的网络结构相同，但编码器被 CNN 取代(附录 C). 我们使用 VLOG 以及基线、oracle 和替代 oracle 指导方法在 MinAtar 的所有 5 个环境中运行了实验(参见附录 A 详细信息)。基线模型总是使用执行者的观察作为网络输入(图。1C). oracle 与 baseline 相同，只是它总是接收 oracle 观察(即作弊，我们运行 Oracle 的实验作为参考)。VLOG ——不，oracle 是 VLOG 的一个删减，我们使用执行者观察

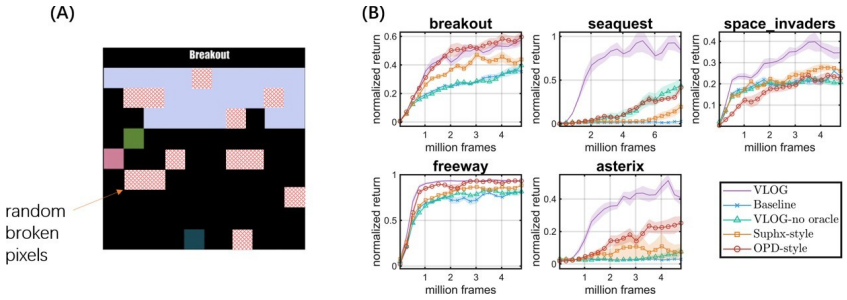

图 3：(A)带有随机破碎像素的 MinAtar 环境的图示。(B)显示 VLOG 的归一化平均回报(除以训练的 oracle 模型的平均回报)和各自使用 8 个随机种子的替代方法的学习曲线。

作为 VLOG 中后部编码器的输入(不使用 oracle 观察)。Suphx 风格的甲骨文引导是麻将 AI Suphx 中使用的甲骨文引导方法(Li et al., 2020)，其中执行者观察和丢失的 oracle 观察(具有丢失概率 pdropout)被连接作为网络的输入。随着训练的进行，pdropout 逐渐从 0 增加到 1，因此训练的网络不需要 oracle 观察作为输入(附录 A).OPD 式的 oracle 指导是在 oracle 策略提炼(OPD)中使用的 oracle 指导方法(Fang et al., 2021)。OPD 式 oracle guiding 首先使用 oracle 观察作为输入来训练教师模型，然后使用辅助损失来训练执行者模型，该辅助损失是执行者和教师模型对价值函数的估计之间的误差(附录 A)..

结果显示，正如预期的那样，oracle 的表现通常最好。为了更清楚地进行比较，我们使用 oracle 模型作为参考，对非 oracle 模型的性能进行了标准化(图。3B).在所有的 oracle 指导方法(VLOG、OPD 式和 Suphx 式)中，VLOG 始终表现最好。值得注意的是，VLOG 和 VLOG-no oracle 在 seaquest 中的表现出奇的好。这可以解释为 Seaquest 是一个具有局部最优的任务[4]，而 VLOG 的隐藏状态中的随机性有助于在潜在空间中的探索以逃离局部最优(类似的想法在 Fortunato et al.(2018)，但是它们的噪声被添加到网络权重中)。除了在 Seaquest 中，VLOG-no oracle 没有显示出与基线显著不同的性能，这表明在该任务集中 VLOG 的性能增益主要来自利用 oracle 观察来形成潜在分布；并且使用变分贝叶斯模型，至少在没有有用的预言信息时不会损害性能。

5.3 麻将离线学习

麻将是一种受欢迎的基于瓷砖的游戏，在全球有数亿玩家(这里我们考虑日本的变体)。该游戏类似于许多其他卡牌游戏(但使用瓷砖代替卡片)，其中多个(通常是四个)玩家交替抽取和丢弃瓷砖(总共 136 个瓷砖)以满足获胜条件。这是一个非常具有挑战性的游戏，其特征在于(1)执行者观察中的不完全信息(玩家看不到对手的私人牌和剩余的要抽的牌)，(2)像在许多纸牌游戏中一样的随机状态转换，以及(3)极高的游戏复杂性(即，不同的合法游戏状态的数量)。麻将的复杂性在于
远大于 10166(附录 E.1).作为参考，围棋的复杂度是 10172(Silver et al.,2016) 无限注扑克的复杂度是 10162(Johanson, 2013).

在麻将中，很难根据执行者的观察做出最优决策，因为结果严重依赖于不可见信息，而不可见状态空间的复杂度平均高达 1048(Li et al.,2020).为了应对这一挑战，Li et al.(2020)介绍
suphx 风格的 oracle 指导并展示了性能提升。因此，我们认为麻将是一种
有前途的 oracle 引导方法测试平台。因为麻将中可能的状态数

4 在 Seaquest 中，代理人驾驶潜水艇潜入海中向敌人射击并营救潜水员以获得分数。然而，潜艇的氧气有限。为了生存，它必须在耗尽氧气之前浮出水面补充氧气，这样它暂时不能得分。一个典型的局部最佳方案是将最后剩余的氧气用于潜水，而不是浮出水面。

非常大。以在线 RL 的方式探索随机动作是昂贵的，并且没有先前的工作可以用纯在线 RL 训练一个强大的麻将 AI。此外，我们还想检查 VLOG 在离线 RL 设置中的有效性。由于这些原因，我们转移到离线 RL(Levine et al., 2020) 对于麻将任务使用专家示范。

我们将来自在线麻将游戏平台天厚(https://tenhou.net/mjlog.html)的约 2300 万步人类专家的棋谱处理成一个离线 RL 数据集(数据是利用麻将中的对称性增加的，见附录 E). 此外，我们创建了一个麻将模拟器作为测试环境。虽然有复杂的方法来编码麻将的状态和动作空间(Li et al., 2020)，我们试图用合理数量的近似值进行简化，因为我们的目标不是创建一个强大的麻将 AI，而是使用麻将作为一个平台来研究甲骨文指导问题。在我们的例子中，动作空间由 47 个独立的动作组成，涵盖了麻将中的所有决策。一个执行者观察是一个编码公共信息和当前玩家私手的矩阵；神谕观察将执行者的观察与对手私人手中的信息联系起来。(参见附录 E). 我们使用一维 CNN 作为编码器，就像麻将中通常使用的那样(Li et al., 2020)，并且 zt 和解码器网络宽度的大小分别增加到 512 和 1024(附录 C).

注意，尽管麻将是 4 人游戏，但是使用离线 RL 数据来训练代理并不涉及多代理 RL(Zhang et al., 2021) 因为离线数据集是固定的:对手有固定的策略，因此可以被视为环境的一部分。我们的实验集中于单代理 RL，以避免考虑多代理 RL 所带来的复杂性。我们研究了两种离线 RL 设置。第一种是保守的 Q-learning (CQL)(Kumar et al., 2020).我们的 CQL 设置通过增加辅助 CQL 损耗而不同于前面章节中的在线 RL 设置(Kumar et al., 2020) Q-学习损失函数(附录 B.1.2).

另一个是行为克隆。尽管 VLOG 是为基于值的 RL 设计的，但我们可以通过让网络预测动作(并且解码器不需要动作输入)来直接将 VLOG 与 BC 合并。通过最小化输出和目标动作(示范)之间的交叉熵来进行学习，如同在分类问题中一样。请注意，我们没有在业务连续性设置中测试 OPD 式的 oracle 指导，因为它等于基线，因为我们可以直接使用演示操作作为 oracle 策略进行提炼。

被评估模型 对比基线模型	持续查询语言 平均值。平均支付匹配胜率 (%)		公元前 。支付匹配胜率 (%)	
甲骨文(作弊)	**224 13**	53.6 0.4	15 12	50.5 0.4
Suphx 风格	59 13	51.2 0.4	37 - 12	50.9 0.4
OPD 风格	-9 13	50.0 0.4	41 12	-
视频博客	**233 13**	**55.7 0.4**	**67 12**	51.4 0.4
VLOG-无 oracle	61 13	52.0 0.4		**52.2 0.4**

表 1:训练后每个模型的性能与麻将上训练的基线模型的性能，每个模型使用 4 个随机种子。每场比赛由四个独立的玩家(代理人)进行一系列的 8 场比赛。如果一个玩家在完成 8 场游戏后，在所有玩家中得分最高，则该玩家赢得这场游戏。每个玩家只为自己而玩，只考虑最大化自己的收益和排名。在 4 个玩家中，两个是测试代理，另外两个是基线模型(代理之间没有交流或团队合作)。基线模型用相同的离线 RL 过程训练，但是没有 oracle 指导(图。1C)). 为了减少方差，我们将两个相同模型的玩家的结果相加，并将其用作相应方法的性能。平均回报是每场比赛点数变化的平均值。每种类型的比赛重复 12，500 次(100，000 场)。数据是平均值±标准差..

因为麻将是一种零和游戏，四人游戏，我们在两种情况下测试了训练模型的性能:使用(训练的)基线模型(表 1) 还有互相玩(互相打架，表 2).在第一个场景中，四个代理在游戏桌上玩相同的游戏，其中两个是测试代理，另外两个是基线模型。尽管每个代理都为自己而战，并且玩家之间没有交流，但我们简单地将两个被测试代理的收益相加，并考虑如果其中一个排名靠前，他们就赢得了一场比赛(因此如果与基线一样强，比赛胜率将为 50%)，用于统计(表 1).

互相争斗(CQL)	平均值。支付匹配胜率(%)	游戏胜率(%)买入率(%) 18.45
甲骨文(作弊)Suphx 式 OPD 式 VLOG	168 11 28.0 0.4 -75 12 23.2 0.4 -209 12 20.1 0.4 116 12 28.7 0.4	4.48 18.32 8.84 14.39 8.06 **19.36 8.28**

互相争斗(公元前)	平均值。支付匹配胜率(%)	游戏胜率(%)成交率(%)
甲骨文(作弊) Suphx 风格的 VLOG VLOG-无 oracle	-27 11 23.9 0.4 -10 11 24.6 0.4 26 11 25.6 0.4 11 11 25.9 0.4	20.69 7.96 20.51 8.03 **20.82 8.00** 20.76 8.15

表 2:经过训练的模特在同一张桌子上互相玩耍。发牌意味着玩家放弃一张牌，另一个玩家通过捡起这张牌组成一手获胜牌来赢得游戏。发牌是对玩家的惩罚，因此最好避免。每种类型的比赛重复 12，500 次(即 100，000 场比赛)。数据是平均值±标准差..

对于 CQL，结果(表 1 向左和 2 上图)显示，VLOG 大大优于基线和替代方法(因为麻将是一种高度随机的游戏，55.7%的匹配胜率表明存在很大的技能差距)。有趣的是，VLOG 甚至可以与 oracle 相媲美。这可以解释为 VLOG 也受益于它的贝叶斯属性，这与 VLOG 一致——没有 oracle 显示出比基准模型显著的性能增益(表 1 左)。尽管如此，oracle 模型学会了减少发牌(即一名玩家丢弃一张牌，另一名玩家通过捡起这张牌组成一手好牌来赢得游戏)，因为它可以清楚地看到对手的私人牌，显示出比其他非作弊模型低得多的发牌率(表 2 上)。

在 BC 设置中，代理没有学习价值函数，而是试图预测人类专家的行动。因此，训练程序不涉及推理玩出来和甲骨文观察之间的关系，而只是模仿人类的行为。从结果中可以看出，oracle 在业务连续性方面并没有显著超过基准(表 1 右和 2 更低)。然而，由于随机建模，VLOG 和 VLOG-no oracle 仍然显示出性能提升。

6 摘要

我们提出了 VLOG——一种利用 oracle 观测值的变分贝叶斯学习框架，以促进 DRL，尤其是在部分可观测的环境中。VLOG 可用于任何 RL 问题，其中有 oracle 观察可以帮助执行者做出决策。

我们首先引入一个潜在向量 z 来表示环境状态。z 的先验和后验分布分别使用 executor 和 oracle 观察来建模。然后，我们推导了一个变分下界(方程。2) 通过最大化，我们可以使用 oracle 观察来优化 executor 模型。我们为 DRL 开发了相应的方法，该方法可以与大多数需要估计价值函数的 RL 算法结合。

如果甲骨文观测包含更多的信息来检索真实的环境状态(或者，它是真实的环境状态)，VLOG 在潜在空间中的甲骨文引导有助于在神经网络中形成更接近真实的潜在表示。我们使用迷宫任务展示了 VLOG 的这一优势。然后，我们扩展 VLOG 来解决基于图像的视频游戏，并将其与其他 oracle 引导的方法进行比较。尽管所有 oracle 引导的方法都显示出性能优于基线模型，但 VLOG 始终表现最佳。最后，我们使用一个具有挑战性的基于瓷砖的游戏麻将转移到离线 RL 域，在该游戏中，执行者玩隐藏信息和随机状态转换，并且观察到 VLOG 获得了最佳的整体性能。

我们还对 VLOG(VLOG-无甲骨文)进行了消融研究，其中后验模型未接受甲骨文观察，但接受了执行者 1 的观察。VLOG——没有 oracle 证明可以从随机性中获益的任务的性能提升；否则，其表现类似于确定性基线。这澄清了 VLOG 的良好性能来源于两个方面:oracle 引导和随机建模。最后，我们发布了麻将离线强化学习数据集和相应的强化学习环境，以方便今后对甲骨文引导的研究。

再现性声明

我们在补充文件中包含了 VLOG 和替代模型的源代码。我们使用的环境和数据集是在线公开的。

道德声明

我们声明没有利益冲突。我们尝试对有颜色识别障碍的人使用友好的颜色(图。2C, d)和不同模型的性能曲线的可区分标记(图。2B, 图 3 和 Fig6).我们的麻将数据集是使用来自 Tenhou.net 的可下载的公开游戏回放数据并经过后期处理而生成的。数据集不包含关于球员的私人信息。由于 VLOG 是一个利用 oracle 信息的通用框架，我们无法预见 VLOG 会被直接用于恶意目的。然而，任何新的 RL 算法可能赋予代理增加的自主性，并最终导致完全自主的代理，其可被用于恶意目的，例如完全自主的士兵。

参考

亚历山大·阿莱米、伊恩·费希尔、约书亚·v·狄龙和凯文·墨菲。深度变化的信息瓶颈。在 2017 年国际学习表征会议上。

Mohammad Gheshlaghi Azar、Ian Osband 和 Re mi Munos。强化学习的极大极小后悔界限。在 2017 年国际机器学习会议上。

马克·G·贝勒马尔、亚瓦尔·纳达夫、乔尔·维内斯和迈克尔·鲍林。街机学习环境:总代理的评估平台。人工智能研究杂志，47:253 - 279，2013。

马克·g·贝勒马尔、威尔·达布尼和 Re mi Munos。强化学习的分布观点。在 2017 年国际机器学习会议上。

克里斯托弗·布格斯、伊琳娜·希金斯、阿尔卡·帕尔、罗克·马泰、尼克·沃特斯、纪尧姆·德雅丁斯和亚历山大·勒施纳。理解 β-VAE 中的解纠缠。2017.

陈典韦，布雷迪周，弗拉德伦科尔敦，和菲利普克拉亨布 hl。靠作弊学习。机器人学习会议，第 66-75 页。PMLR，2020 年。

方、、、刘、董舟、张伟南、、俞勇和。通过 Oracle Policy Distillation 执行订单的通用交易。在 2021 年 AAAI 人工智能大会上。

Meire Fortunato、Mohammad Gheshlaghi Azar、Bilal Piot、Jacob Menick、Matteo Hessel、Ian Os- band、Alex Graves、Volodymyr Mnih、Remi Munos、戴密斯·哈萨比斯、Olivier Pietquin、Charles Blundell 和 Shane Legg。探索嘈杂的网络。在 2018 年国际学习代表大会上。

托马斯·弗姆斯顿和大卫·巴伯。强化学习的变分方法。《第十三届人工智能与统计国际会议论文集》，第 241-248 页。2010 年 JMLR 研讨会和会议记录。

亚瑟·格斯、法比奥·维奥拉、西奥法纳·韦伯、拉尔斯·布辛、史蒂文·卡普托罗夫斯基、多伊娜·普雷科普、大卫·西尔弗和尼古拉斯·赫斯。价值驱动的后见之明建模。神经信息处理系统进展，2020 年。

大卫·哈和朱尔根·施密德胡伯。循环世界模型促进政策演变。神经信息处理系统进展，2018。

Tuomas Haarnoja、Aurick Zhou、Pieter Abbeel 和 Sergey Levine。软行动者-批评者:带随机行动者的非政策最大熵深度强化学习。在机器学习国际会议上，第 1856-1865 页，2018 年。

Tuomas Haarnoja、Aurick Zhou、Kristian Hartikainen、George Tucker、Sehoon Ha、、Vikash Kumar、Henry Zhu、Abhishek Gupta、Pieter Abbeel 和 Sergey Levine。软演员-评论家算法及其应用。arXiv，abs/1812.05905，2019。

韩，多谷健治和谷俊秀。求解部分可观测控制任务的变分递归模型。2020 年学习表征国际会议。

Anna Harutyunyan、Will Dabney、Thomas Mesnard、Mohammad Gheshlaghi Azar、Bilal Piot、Nico- las Heess、Hado van Hasselt、Gregory Wayne、Satinder Singh、Doina Precup 和 Remi Munos。事后诸葛亮。神经信息处理系统进展，2019 年。

Matteo Hessel、Joseph Modayil、Hado van Hasselt、Tom Schaul、Georg Ostrovski、Will Dabney、Dan Horgan、Bilal Piot、Mohammad Azar 和 David Silver。彩虹:结合深度强化学习的改进。2018 年 AAAI 人工智能大会。

Irina Higgins、Loic Matthey、Arka Pal、Christopher Burgess、Xavier Glorot、Matthew Botvinick、Shakir Mohamed 和 Alexander Lerchner。β-VAE:用约束变分框架学习基本视觉概念。在 2016 年国际学习代表大会上。

马克西米利安·伊戈尔、路易莎·津特格拉夫、图安·安勒、弗兰克·伍德和西蒙·怀特森。POMDPs 的深度变分强化学习。在 2018 年国际机器学习会议上。

、袁泽·艾伦-朱、塞巴斯蒂安·布贝克和迈克尔一世·乔丹。q-learning 是可证明有效的吗？神经信息处理系统进展，2018 年。

、杨卓然、王和乔丹。线性函数逼近的可证明有效强化学习。2020 年学习理论会议。

迈克尔·约翰逊。测量大型无限注扑克游戏的规模。arXiv 预印本 arXiv:1302.7008，2013。

史蒂文·卡普托罗夫斯基、格奥尔格·奥斯特洛夫斯基、威尔·达布尼、约翰·全和雷米·穆诺斯。分布式强化学习中的循环经验重放。在 2018 年国际学习代表大会上。

迪德里克·p·金玛和马克斯·韦林。自动编码变分贝叶斯。2014 年国际学习代表会议。

W.布拉德利·诺克斯和皮特·斯通。通过人类强化交互塑造代理:驯服者框架。2009 年知识获取国际会议。

Tadashi Kozuno，郝云·唐，马克·罗兰，雷米·穆诺斯，史蒂文·卡普托罗夫斯基，威尔·达布尼，迈克尔·瓦尔科和大卫·阿贝尔。再论彭的 Q(λ)对现代强化学习的启示。2021 年国际机器学习会议。

Aviral Kumar、Aurick Zhou、George Tucker 和 Sergey Levine。离线强化学习的保守 Q 学习。神经信息处理系统进展，2020 年。

李东旻、阿努沙·纳加班迪、彼得·阿贝耳和谢尔盖·莱文。随机潜在行动者-批评家:具有潜在变量模型的深度强化学习。神经信息处理系统进展，33，2020。

谢尔盖·莱文。作为概率推理的强化学习与控制:教程与综述，2018。

谢尔盖·莱文、阿维拉尔·库马尔、乔治·塔克和贾斯汀·傅。离线强化学习:辅导、回顾和对开放问题的展望。arXiv 预印本 arXiv:2005.01643，2020

、小山田宗越、叶琦玮、、、、杨、李钊、、和。arXiv，abs/2003.13590，2020。

罗伯特·洛夫廷、彭蓓、詹姆斯·麦克格拉森、迈克尔·利特曼、马修·泰勒、杰夫·黄和戴维·罗伯茨。通过人类提供的离散反馈的学习行为:模拟隐式反馈策略以加速学习。自主代理和多代理系统，30(1):30–59，2016。

James MacGlashan，Mark K. Ho，Robert Loftin，Bei Peng，，David L. Roberts，Matthew E. Taylor 和 Michael L Littman。从政策相关的人类反馈中进行交互式学习。在 2017 年国际机器学习会议上。

沃洛季米尔·姆尼赫、科雷·卡武克库奥卢、戴维·西尔弗、安德烈·鲁苏、乔尔·维内斯、马克·贝尔-马雷、亚历克斯·格雷夫斯、马丁·里德米勒、安德烈亚斯·菲杰兰德、格奥尔格·奥斯特罗夫斯基、斯蒂格·彼得森、查尔斯·贝蒂、阿米尔·萨迪克、约安尼斯·安东诺格鲁、海伦·金、达尔山·库马兰、金奎大·维尔-斯特拉、沙恩·莱格和戴密斯·哈萨比斯。通过深度强化学习实现人类水平的控制。自然，518(7540):529，2015。

冈田正史、小坂纪夫和谷口忠弘。Bayesian 人的星球:通过结合贝叶斯推理重新考虑和改进深度规划网络。2020 年智能机器人与系统国际会议。

卡尔·皮尔逊·LIII。在最接近空间点系统的直线和平面上。伦敦、爱丁堡和都柏林哲学杂志和科学杂志，2(11):559–572，1901 年。

汤姆·绍尔、约翰·全、约安尼斯·安东诺格鲁和大卫·西尔弗。优先体验回放。在…里 *2016 年国际学习表征会议*。

David Silver，Aja Huang，Chris J，Arthur Guez，Laurent Sifre，George Van Den Driessche，Julian Schrittwieser，Ioannis Antonoglou，Veda Panneershelvam，Marc Lanctot，等人用深度神经网络和树搜索掌握围棋。自然，529(7587):484，2016。

爱德华·j·桑迪克。无限时域上部分可观测马尔可夫过程的最优控制:贴现成本。奥佩。第 26(2)号决议，第 282-304 页，1978 年。

理查德·萨顿和安德鲁·巴尔托。强化学习:导论。麻省理工学院出版社，2018 年第二版。

纳夫塔利·蒂什比和诺加·扎斯拉夫斯基。深度学习和信息瓶颈原理。在…里 *IEEE 信息理论研讨会。IEEE，2015*。

哈多·范·哈瑟尔特，亚瑟·盖兹和大卫·西尔弗。双 Q 学习的深度强化学习。2016 年 AAAI 人工智能大会。

奥里奥尔·维尼亚尔斯、伊戈尔·巴布什金、沃伊切赫·沙皇内基、马天如·马蒂厄、安德鲁·杜兹克、钟俊英、戴维·崔世安、理查德·鲍威尔、蒂莫·埃瓦尔德斯、佩特科·乔尔杰夫、吴俊赫、丹·霍根、曼努埃尔·克洛斯、伊沃·达尼埃尔卡、黄阿贾、洛朗·西弗尔、特雷弗·蔡、约翰·阿加皮乌、马克斯·贾德尔伯格、亚历山大·萨莎·维兹涅维茨、雷米·勒布隆、托比亚斯·波伦、瓦伦丁·达利巴尔、戴维·布登星际争霸 2 中使用多智能体强化学习的特级大师级别。自然，575(7782):350–354，2019。

王子瑜、汤姆·绍尔、马特奥·赫塞尔、哈多·范·哈瑟尔特、马克·兰托特和南多·弗雷塔斯。用于深度强化学习的 dueling 网络体系结构。在 2016 年国际机器学习会议上。

西奥法娜·韦伯、尼古拉斯·赫斯、阿里·伊斯拉米、约翰·舒尔曼、大卫·温盖特和大卫·西尔弗。强化变分推理。2015 年神经信息处理系统(NIPS)研讨会进展。

尹海燕、陈建达、Sinno Jialin Pan 和 Sebastian Tschiatschek。部分可观测强化学习的序列生成探索模型。《AAAI 人工智能会议论文集》，第 35 卷，第 10700–10708 页，2021 年。

肯尼·扬和田甜。MinAtar:一个受 Atari 启发的试验台，用于彻底和可重复的强化学习实验。arXiv 预印本 arXiv:1903.03176，2019。

张凯庆，杨卓然，和塔梅尔.巴斯.阿尔。多主体强化学习:理论和算法的选择性综述。强化学习与控制手册，第 321–384 页，2021。

A 替代 ORACLE 指导方法的实施

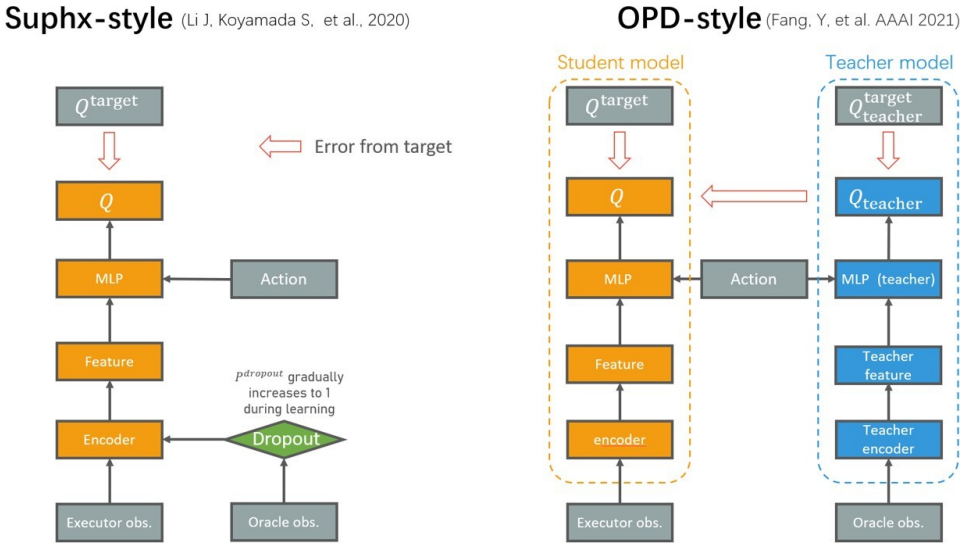

图 4:我们的 Suphx 风格和 OPD 风格的 oracle guiding 的实现图。对于 OPD 风格的 oracle guiding，教师模型被训练，然后在训练学生模型(执行者模型)之前被固定。

在 Suphx 的原著中(Li et al., 2020) 和 OPD (Fang et al., 2021)，作者们正引导我们走向不同的 RL 目标。为了比较，我们使用相似的网络结构在我们的设置中重新实现了 Suphx 风格和 OPD 风格的 oracle guiding(见图。4)尽可能使用相同的超参数。然而，苏霍克式和 OPD 式的甲骨文指导的关键方法仍然与他们的原著相同。

如图 2 所示 4，左，Suphx 风格的模型接收输入

$$xSuphx = concatenate(xt, \delta t \odot x\tilde{}t),$$

其中，δ t 是丢失矩阵，其元素是伯努利变量，P (δ t(i, j) = 1)
1 表示逐元素乘法。在训练期间，pdropout 在前 2/3 个训练时段中从 0 线性增加到 1，并且在剩余的 1/3 个时段中保持为 0。训练的其他部分与基线模型相同。

对于 OPD 式的甲骨文指导，我们直接使用经过训练的甲骨文模型作为教师模型(图。4，右)。损失函数(在 Q 学习中)是

$$JOPD = MSE\ Qtar,\ Q(xt,\ at) + MSE(Qteacher(x\tilde{}t),\ Q(xt,\ at)),$$

其中 Qtar 是通过引导学生模型获得的 Q 学习目标(第一项(MSE (Qtar, Q(xt, at))与基线模型的损失函数相同)。而第二项是带系数的蒸馏损失(注意训练时老师模型是固定的)。我们使用网格搜索为每项任务选择了最佳表现(见表 3).

B RL 算法和超参数

B. 1 RL 算法

B.1.1 决斗双 DQN 迷宫和 MINATAR 任务

正如我们在第二节中讨论的。5，我们使用双 DQN 和决斗网络架构(van Hasseltet al., 2016; Wang et al., 2016) 作为基本的 RL 算法，因为它工作得相对较好(Hesselet al., 2018) 而不引入额外的超参数。

DQN 的决斗建筑(Wang et al., 2016) 定义如下(见附录 C 对于隐藏层大小):

```
决斗网络(nn。模块) :
    定义(s e l f，i n p u t s i z e，动作号，h i d d e n l a y e r s):
        超级(决斗 QNetwork，s e_l f)。i n i t_( ) s
        e l f . i n p u t s i z e = i n_p u t_s i
        z e
        s e l f a c t_I o n n u m = a_c t I o n n u m
        s e l f . h i d d e n l a y e r s = h i d d e n l a y e r s
        网络模块= nn。模块列表()l a s t l a y e r s
        i z e = i n p u t s i z e
        在健康的生活中:
            网络模块。追加(nn。网络模块。追加(nn。ReLU())
            l a s t l a y e_r s i z e = l a y e r s i z e
        s e l f v a l u e_l a y e r = nn。L i n e a r ( l a s t l a y e r s i z e，1)
        s e l f a d v_a n t a g e l a y e r = nn。L i n e a r_( l a s t l a y_e r s
        i z e，a c t i o n n u m ) s e l f . main network = nn。网络模块
    def f o r w a r d ( s e l f，x):
        主网络(x)
        v = s e l f . v a l u e l a y e r ( h)。r e p e a t I n t e r l e a v e(s e l f .
        action num，dim = 1)Q0 = s e l f . a d v a n t a g e l a y e r(h)
        a = Q0 t o r c h . mean(Q0，dim = 1，keepdim=True)。r e p e a t i n t e r l e a
            v e ( s e l f_.动作编号，dim = 1)
        q = v + a
        返回 q
```

双重深度 Q 学习(van Hasselt et al., 2016) 用于计算图中的 Qtarget。1 a(可以使用任何其他算法来计算 Qtarget，而无需更改其他部分)。特别是，就像在 Wanget al. (2016)，我们有

$$Qtarget = rt + \gamma Q(zt，\ arg\ max\ Q(zt+1)，\ a'；\ \theta)；\theta —),$$
$$a'$$

其中 rt 是步骤 t 的奖励，γ 是折扣因子(表 3)，θ 表示用于计算 Q 函数的 Q 网络(MLP 解码器)的参数(附录 C)。注意，z 是由后验解码器以甲骨文观察值 x 作为输入给出的，因为甲骨文预测误差项 eq(z x^)[log p(v(ZT)= vtar | ZT]]在等式中。1 是对后验分布 q(z | x ˇ)的期望。

像在 RL 中通常做的那样(Mnih et al., 2015; Wang et al., 2016; van Hasselt et al., 2016)，我们使用了

与原始 Q 网络结构相同的目标 Q 网络，其参数表示为 θ (表 3).每 1000 步，目标 Q 网络从原始 Q 网络复制参数(表 3)。然后 VLOG 损失函数的第一项(等式。2) 就是简单的给定通过 Qtarget 和 Q 网络(MLP 解码器)的输出之间的均方误差。

B.1.2 决斗双 DQN 与保守的 Q 学习麻将

在麻将中，当我们转移到离线 RL 域(秒。5.3)，直接使用非策略 RL 算法通常会导致非常不满意的性能(Levine et al., 2020)。

因此，我们补充 VLOG 的损失函数(等式。2) 具有辅助的保守 Q 学习(CQL)损失(Kumar et al., 2020)，这是一种最先进的离线 RL 算法:

$$\alpha\ Ex\hat{}，a\sim \quad 日志 \quad \exp(Q(z(x\check{})，a'，\theta))-[Q(z(x\check{})，a，\theta)]，$$
$$'\ a'\in A$$

其中 D 是我们用于麻将的离线数据集，α = 1。所使用的组合损失函数对于麻将(CQL)是 J βJ + JCQL。

B.1.3 麻将中的行为克隆

我们还用 VLOG(秒)测试了行为克隆。5.3) 为了麻将。我们通过让解码器网络预测动作(并且解码器不需要动作输入)，用 BC 实现 VLOG。通过最小化输出和目标动作(示范)之间的交叉熵来进行学习，如同在分类问题中一样。

B.2 超参数选择

我们总结了表中的超参数 3.

说明	标志	价值	评论
		普通	
贴现因素	μ	1(麻将)0.995(其他)	
学习率批量大小		0.0001 1024(麻将)128(其他)	对于在线 RL 的所有
ϵ-greedy 目标网络更新间隔系数	ϵ	0.1(学习)0(评估)1000 四	损失函数 对于所有损失函数
每个梯度步长优化器的环境步长	τ	圣经》和《古兰经》传统中）亚当（人类第一人的名字	
KL 偏差目标值	间奏二	视频博客 50	在所有实验中固定
β 的初始值	酚 乏	0.00001	
		OPD 风格	
保单蒸馏损失系数		0.01(麻将)10(其他)	通过网格搜索

表 3:本文中使用的超参数。对于 OPD 风格，我们从[0.001，0.01，0.1，1，10]中网格搜索其唯一的附加超参数。

B.3 β 的敏感性分析

正如我们在第二节提到的。4.2, 系数 β 对于 VLOG 的学习非常重要。如果我们在整个训练过程中固定 β 的值，过大或过小的 β 都会导致较差的性能。相应的结果显示在图。6.

C 网络结构

为了简单起见，我们尝试对同一模型中的所有全连接层使用相同的隐藏层大小，其中迷宫和 MinAtar 的隐藏层大小为 256，麻将的隐藏层大小为 1024。

C.1 编码器

由于我们针对不同的任务，我们为每种类型的环境使用不同的编码器网络。除了输入特征/通道的大小不同之外，前编码器和后编码器具有相同的结构。

对于迷宫任务，编码器是具有 ReLU 激活的 2 层 MLP。输出大小也等于隐藏层大小。

对于 MinAtar 任务，我们使用二维 CNN 编码器，定义如下:

```
作为 nn 导入到 r c h . nn
c n n m o d u l e l i s t = nn。模块列表( )
c n n m o d u l e l i s t 追加(n n。Conv2d ( n c h a n n e l
s, 16, 3, 1, 0 ) ) c n n m o d u l e l i s t . append ( nn .ReLU())
c n n m o d u l e l i s t 追加(n n。Conv2d ( 1
6, 32, 3, 1, 0 ) ) c n n m o d u l e l i s t . append
( nn .ReLU())
c n n m o d u l e l i s t 追加(n n。Conv2d ( 3 2, 128, 4, 2, 0 )
) c n n m o d u l e l i s t . append ( nn .ReLU())
c n n m o d u l e l i s t 追加(n n。Conv2d ( 1 2
8, 256, 2, 1, 0 ) ) c n n m o d u l e l i s t . append
( nn .ReLU())
c n n m o d u l e l i s t 追加(n n。F l a t t e n())
n n m i n a t a r = nn。S e q u e n t i a l (* c n n m o d u l e l i s t)
```

其中 n 通道是执行程序或 oracle 观察的通道数。输出大小等于隐藏层大小。

对于麻将来说，因为观察的二次元(麻将牌 ID)有局部的上下文关系(Li et al.,2020),我们使用一维 CNN(沿图块 ID 维度的卷积)作为编码器，定义如下:

```
c n n m o d u l e l i s t = nn。模块列表( )
c n n m o d u l e l i s t追加(n n。Conv1d ( n c h a n n e l
s, 64, 3, 1, 1 ) ) c n n m o d u l e l i s t . append ( nn .Conv1d ( 6
4, 64, 3, 1, 1))
c n n m o d u l e l i s t追加(n n。ReLU())
c n n m o d u l e l i s t追加(n n。Conv1d ( 6 4, 6 4, 3, 1, 1
) ) c n n m o d u l e l i s t追加(n n。ReLU())
c n n m o d u l e l i s t追加(n n。Conv1d ( 6
4, 32, 3, 1, 1 ) ) c n n m o d u l e l i s t追加(n n。ReLU())
c n n m o d u l e l i s t追加(n n。F l a t t e n())
cnn 麻将= nn。S e q u e n t i a l (* c n n m o d u
l e l i s t)
```

输出大小为 1088，接近隐藏层大小。

C.2 潜在层

对于 VLOG 和 VLOG(非 oracle)，z 层的大小是隐藏层大小的一半，因为我们需要估计 z 的平均值和方差。对于所有其他模型，隐藏层是一个完全连接的层，具有大小隐藏层大小和 ReLU 激活。

C.3 解码器

所有型号的解码器都是 2 层 MLP，大小为隐藏层大小。除了麻将上的 BC，解码器的输入是潜在层和动作的输出，我们使用了决斗 Q 网络结构(Wang et al., 2016) 以输出标量 Q 值。

对于麻将上的 BC，解码器的输入是潜在层的输出。解码器的输出是动作的 logit，动作可以使用 softmax 获得。

D 任务无关的潜在瓶颈控制

正如我们在第二节中讨论的。4.2, VLOG 损失函数中正则项的系数 β (等式 2) 由等式自适应调整。3,提供了 Dtar，这是另一个超参数。虽然我们的实验证明了这种方法的有效性，但下面将讨论这种设计选择背后的更多想法。

原则上，用一个超参数替换另一个超参数并不总是使训练更容易。然而，在实践中(尤其是深度 RL)，性能将对一些超参数高度敏感(例如，原始软演员评论家算法中的熵系数 α (Haarnoja et al.,2018) 需要在每个机器人任务中进行调整。这是因为任务之间的奖励幅度不同。).因此，用另一个不需要微调的超参数替换一个敏感的超参数将是有益的。例如，在 soft-actor critic 的后续论文中，作者通过引入另一个超参数，即熵目标，使熵系数 α 是自适应的(Haarnoja et al., 2019). 他们根据经验发现，将熵目标设置为代理自由度的负值是很好的，这样可以避免调整 α 。

我们用 Dtar 代替 β 的想法也是出于类似的原因。一个明显的问题是
“oracle 预测误差”项的大小(等式 2) 依赖于任务的回报大小。因此 β 也应该被调整以匹配任务奖励的大小。然而，Dtar 仅与先验 z 和后验 zt 的大小相关，这在任务之间没有太大的差异(通常在 1 的数量级)。实际上，我们发现目标 KLD 的单个值 Dtar = 50 对于包括迷宫、MinAtar 和麻将在内的所有任务都很有效。

调整 β 的另一种方法是采用线性或指数调度程序。例如，在 Burgess et al. (2017), 作者对目标 KL-散度使用了线性调度器，得到了良好的结果

结果。然而，使用调度器引入了更多的超参数(至少两个:初始 β 和最终 β)，这违背了我们降低超参数影响的意愿。

E 麻将

E.1 麻将游戏复杂度的估计

我们考虑 4 人日本麻将[5]。尽管规则有微小的变化，但以下估计适用于一般情况。

为了更容易计算，我们做了两个主要的简化(即，我们估计游戏复杂度的下限):(1)从丢弃中融合[6]不被考虑，并且(2)不考虑除了区块之外的信息，例如上下文游戏的结果、玩家的分数。

共有 34 种瓷砖，每种都有 4 个副本(因此共有 136 种瓷砖)。我们进一步将我们的估计限制到最后一轮(即，最后一张牌被画出)。在 136 张牌中，有 53 张牌在某人手里(4 名玩家手里分别有 14、13、13、13 张牌)。一个人手中牌的排列不会产生不同，而如果不在手中，排列就很重要。…的数目
因此，136 个瓦片的可区分配置可计算为 $\frac{136!}{(13!)^3\,14!\,(4!)^{34}} \sim 10^{145}$

同时，对于每个被丢弃的牌，重要的是知道它是否在被抽取后立即被丢弃。对于 70 个丢弃的瓦片，可能性的数目简单地是 $2^{70} \sim 10^{21}$。

因此，麻将游戏复杂度的下界估计为

$$10^{145} \times 10^{21} \sim 10^{166}.$$

如果考虑到被简化的信息，状态空间可能会大得多。比如我们来考虑一下每个玩家目前的积分。最常见的规则是，每个玩家以 25000 分开始，以 100 分为最小单位(共 1000 个单位)，如果有人得到负分，游戏终止。因此，可能性的数量可以转换为“有多少种方法向 4 个孩子分发 1000 颗糖果”的答案，即 $\binom{1000+1}{(4-1)!}(41)^{4} \sim 10^{8}$.

E.2 观察空间和动作空间编码的细节

在我们的麻将环境中，动作空间由 47 个离散的动作组成(表 5)．因为不是所有的动作在某一步都可用，所以我们也根据规则提供了每一步所有 47 个动作中的有效动作集合，可以在玩和学的时候使用。

甲骨文观察的形状为 $(34, 34, 111)$(执行者观察是甲骨文观察的一部分，形状为 $(34, 34, 93)$)(图。5)．第一维对应于 111 个特征(通道)。观察的第二维度(大小为 34)对应 34 张麻将牌(顺序为字 1-9，点 1-9，竹 1-9，东、南、西、北、白、绿、红)。对于编码器，我们使用沿第二维卷积的 1-D CNN。假设当前玩家为 0 号玩家，其他玩家逆时针编号为 1，2，3。观察中任何元素的值都是 1 或 0，如表中所述 4.

E.3 数据扩充

在(日本)麻将中，有 3 套花色(字、点、竹)牌，4 个风牌和 3 个龙牌。这三套衣服是对称的，因此可以互换[7]。四风牌和三龙牌也是如此。根据这种对称性，我们通过分别随机交换 3 个套装、4 个风牌和 3 个龙牌来扩充离线 RL 数据集。

5 https://en . Wikipedia . org/wiki/日本麻将 https://en.wikipedia.o

6A 梅尔德是一个特定的模式，三个或四个瓷砖。如果满足特定条件，玩家可以从其他人那里捡起被丢弃的牌，通过向公众显示 meld 来形成 meld。

7 获胜手牌图案“全绿”有一个例外。因为这是一个极其罕见的情况，我们简单地忽略它。

粗略的意思	特征索引	含义(t 是相应的瓷砖)
玩家 0 的手牌	0 一 2 3 四 5	如果玩家 0 手里有≥1 t 的牌 如果玩家 0 手里有≥2 t 如果玩家 0 手里有≥3 t 如果玩家 0 手里有 4 t 如果 0 号玩家在这场游戏中弃牌 t 如果 0 号玩家手里有红朵拉 t
玩家 0 的叫牌(从丢弃的牌中融合)	6 七 8 9 10 11	如果玩家 0 有≥1 t 的跟注 如果玩家 0 有≥2 t 的跟注 如果玩家 0 有≥3 t 的跟注 如果玩家 0 有 4 t 的跟注 如果玩家 0 在叫牌中有≥1 t，并且 t 来自其他玩家丢弃的牌，如果玩家 0 在叫牌中有红朵拉 t
玩家 1 的叫牌 玩家 2 的叫牌玩家 3 的叫牌	12 至 17 岁 18 至 23 岁 24 至 29 岁	和 5 到 11 一样，但对 1 号玩家来说 和 5 到 11 一样，但对 2 号玩家来说 与 5 到 11 相同，但对 3 号玩家而言
玩家 0 丢弃的牌	30 31 32 33 34 35 36 37 38 39	如果 0 号玩家弃牌≥1 t 如果玩家 0 弃牌≥2 t 如果 玩家 0 弃牌≥3 t 如果玩家 0 在跟注中有 4 t 如果玩家 0 的第一张两张牌 t 是 Tegiri(如果适用) 如果玩家 0 的第二张两张牌 t 是 Tegiri(如果适 用)如果玩家 0 的第三张两张牌 t 是 Tegiri(如果 适用)如果玩家 0 的第四张两张牌 t 是 Tegiri(如 果适用)如果玩家 0 弃掉了 Red Dora t 如果 0 号玩家弃了 t 来宣布 Riichi
玩家 1 丢弃的牌 玩家 2 的弃牌玩家 3 的弃牌	40 至 49 岁 50 至 59 岁 60 至 69 岁	和 30-39 一样，但是对 1 号玩家来说 与 30-39 相同，但对玩家 2 来说与 30-39 相同，但对玩 家 3 来说
其他公共信息	70 71 72 73 74 75 76 77 78 79	如果 t 是 Dora 指示器(≥ 1 次重复) 如果 t 是 Dora 指示符(≥ 2 次重 复)如果 t 是 Dora 指示符(≥ 3 次重复)如果 t 是 Dora 指示符(4 次重复) 如果 t 是 Dora (≥ 1 次 重复)如果 t 是 Dora (≥ 2 次重复)如果 t 是 Dora (≥ 3 次重复)如 果 t 是 Dora (4 次重复) 如果 t 是桌子的风， 如果 t 是自己的风
最新动作的平铺	80	如果 t 是对应于最新动作的区块
可用操作的信息	81 82 83 84 85 86 87 88 89 90 91 92	如果 at t 在 0 号玩家手里 如果 at t 可以是 Chi，则在 meld 中最小， 如果 at t 可以是 Chi，则在 meld 中居中， 如果 at t 可以是 Pong，则在 meld 中最大 如果 at t 可以是 安-坎如果 at t 可 以是坎 如果 at t 可以成为 Ka-Kan 如果通过丢弃 t 可以得到 Riichi 如果 t 是其他人最近丢弃的牌，则启用 RonAgari 如果 t 是最近绘制的牌，则启用 Tsumo 如果 t 是九州并且在 0 号玩家手里
玩家 1 的手牌(仅适用于甲骨文) 玩家 2 的手牌(仅适用于甲骨文)玩家 3 的手牌(仅适用于甲骨文)	93 至 98 99 到 104 104 到 110	与 0 到 5 相同，但适用于玩家 1 与 0 到 5 相同，但适用于玩家 2 与 0 到 5 相同，但适用于玩家 3

表 4:对我们的麻将环境的 oracle 观察编码的 111 个特性(前 93 个特性对执行者可用)的解释。玩家 0 是当前正在做决策的玩家，玩家 1，2，3 是逆时针方向的对手。Tegiri(“弃牌”)表示抽牌后不会立即弃牌。

行动指数	说明
0	丢弃字符 1
一	丢弃字符 2
2	丢弃字符 3
3	丢弃字符 4
四	丢弃字符 5(优先级较高的非红朵拉)
5	丢弃字符 6
6	丢弃字符 7
七	丢弃字符 8
8	丢弃字符 9
9	丢弃点 1
10	丢弃点 2
11	丢弃点 3
12	丢弃点 4
13	丢弃点 5(优先级较高的非红朵拉)
14	丢弃点 6
15	丢弃点 7
16	丢弃点 8
17	丢弃点 9
18	丢弃竹子 1
19	弃竹 2
20	弃竹 3
21	丢弃竹子 4
22	弃竹 5(优先级较高的非红朵拉)
23	丢弃竹子 6
24	弃竹 7
25	丢弃竹子 8
26	弃竹 9
27	抛弃东风
28	丢弃南风
29	抛弃西风
30	抛弃北风
31	丢弃白龙瓷砖(哈克)
32	丢弃青龙瓷砖(Hatsu)
33	弃红龙瓦(楚)
34	Chi(拾取的瓷砖是 meld 中最小的)
35	池(捡起的牌在中间)
36	Chi(拾取的瓷砖是最大的)
37	（同 PassiveOpticalNetwork）无源光网络
38	安坎
39	赣江
40	卡坎
41	理一
42	留宿（remain overnight 的缩写）
43	津摩
44	重启九州九州海的游戏
45	不要去理一(当理一可能的时候)
46	不作反应(当驰、彭、菅、荣等。是可能的)

表 5:我们麻将环境的动作编码。

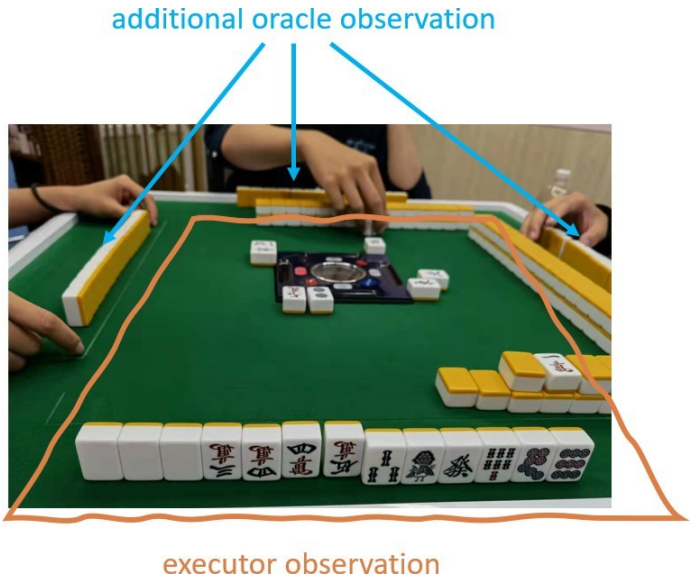

图 5:麻将游戏的图片。执行者观察包括公开可见的信息和玩家的私人手牌。甲骨文观察包括执行者观察和对手私人牌的附加信息。

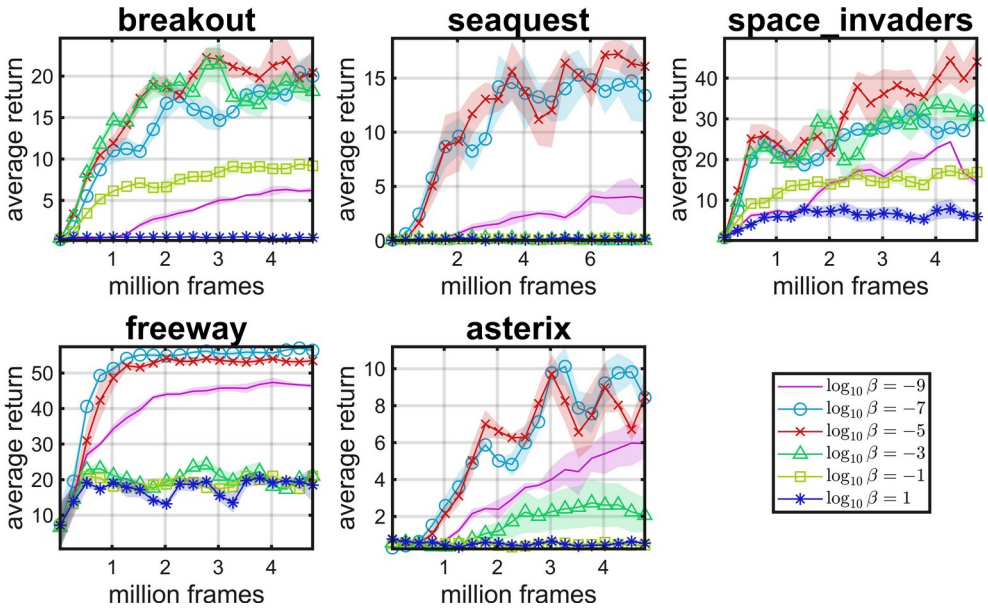

图 6:噪声 MinAtar 环境下 VLOG 的灵敏度分析(相对于 β if β 的选择)是固定的。