arXiv:2303.03572v1 [cs.LG] 7 Mar 2023

学习何时干预业务流程：结合因果推断与强化学习的处方性流程监控

Zahra Dasht Bozorgi[1], Marlon Dumas[2], Marcello LaRosa[1], Artem Polyvyanyy[1],Mahmoud Shoush[2],和 Irene Teinemaa[3]*

[1] 墨尔本大学，澳大利亚帕克维尔，VIC 3010
zahra.dashtbozorgi@student.unimelb.edu.au
{marcello.larosa,artem.polyvyanyy}@unimelb.edu.au[2] 塔尔图大学，爱沙尼亚塔尔图 Narva mnt 18, 51009
{marlon.dumas,mahmoud.shoush}@ut.ee[3]
irene.teinemaa@gmail.com

摘要。 提高流程的成功率，即以正面结果结束的案例所占的百分比，是流程改进中一项常见的目标。在运行时，工作人员通常可以执行某些动作（又称干预措施），以提高案例以正面结果结束的概率。例如，在贷款发放流程中，一种可能的干预措施是提供多个贷款报价，以提高客户接受贷款的概率。每项干预措施都有成本。因此，在制定针对案例的干预策略时，管理者需要考虑干预措施的净收益。此外，干预措施的效果会随时间变化：在案例流程较早阶段进行干预，可能比在较晚阶段更有效。本文提出了一种处方性监控方法，用于自动完成这一决策任务。该方法结合因果推断与强化学习，学习能够最大化净收益的干预策略。该方法利用保形预测技术，通过区分可能以正面或负面结果结束的案例与结果不确定的案例，加快强化学习机制的收敛。在两个真实数据集上的评估表明，所提出的方法优于最先进的基线方法。

关键词： 处方性流程监控 · 因果推断 · 强化学习。

1 引言

处方性流程监控是一类用于推荐动作（以下简称干预措施）的技术；如果执行这些动作，预计可以针对一个或多个流程绩效指标 [9]优化流程。例如，在贷款发放流程中，对于正在犹豫是否接受贷款的客户，可以采取发送一份条件更优的额外贷款报价这一干预措施。该

* 现就职于 DeepMind。

干预措施旨在提高一项称为成功率的绩效指标——以正面结果结束的案例所占的百分比；在此语境下，这意味着最终接受贷款报价。

每项干预都有成本。这项成本通常相当高，因此不可能对每个案例都进行干预。此外，干预效果可能因案例而异。对某个案例效果良好的干预，对其他案例可能无效。实施干预的另一个重要方面是时机。回到贷款发放的例子，在流程后期发送额外报价，效果可能不如在早期发送。

以往关于处方性流程监控的研究提出，利用基于历史执行数据训练的机器学习模型预测每个案例的结果，从而生成干预建议 [6, 21]。具体而言，将在线强化学习与预测模型相结合来推荐干预措施，已显示出良好的结果 [11]。然而，这种既有方法存在两个关键局限。首先，由于它是根据结果预测来提出建议，因此倾向于对可能以负向结果告终的案例进行干预，即使干预不太可能使该案例的结果从负向转为正向 [3]。换言之，这种既有方法没有考虑干预措施的有效性。第二个局限与在线强化学习的使用有关，因为它需要通过试错来学习。这意味着强化学习智能体在最终学会良好执行任务之前会犯错。此外，智能体可能收敛得很慢，例如，强化学习智能体可能需要观察数百个案例，才能收敛到令人满意的干预策略。

本文提出了一种用于学习处方式流程监控干预策略的强化学习方法，以如下方式克服上述局限：

–为考虑干预措施的有效性，该方法将因果效应估计纳入强化学习流程。– 为离线训练强化学习智能体，该方法利用所谓的替代结果扩充现有数据集。扩充后的数据集模拟了真实环境，使强化学习智能体能够在线下根据自身选择获得反馈。– 为加快强化学习智能体的收敛，该方法采用一种称为保形预测的方法——这是一种预测建模方法，可将几乎肯定会以正类结束的案例与不确定的案例区分开来。借助这些信息，强化学习智能体能够避免干预那些无论如何都很可能会取得正面结果的案例。

本文其余部分安排如下。第 2 节回顾相关工作。第 3 节介绍相关概念和符号。第 4 节讨论所提出的方法，第 5 节报告实验评估。最后，第 6 节总结全文并讨论未来工作。

2 相关工作

处方性流程监控： 许多研究探讨了处方式监控问题。此处，我们介绍与本研究密切相关的工作。有关处方式监控方法的完整综述，请参阅 [9]。

Teinemaa等. [21] 提出了一种基于预测的系统，利用经验阈值法在需要干预时触发警报。Fahrenkrog-Petersen等. [6], 随后扩展了这项工作，发现当阈值达到后稍晚些再触发干预，可以以更低的成本改善结果。Metzger等. [11],的研究使用在线强化学习来学习触发干预措施的最佳时机。他们证明，强化学习的表现优于经验阈值法。然而，这些方法并未探讨干预的有效性，或对其有效性作出过于简单的假设。本文旨在解决这些局限。

在另一类研究中，研究者使用处方方法来推荐下一个最佳活动。de Leoni 等人. [10] 为帮助客户求职的流程工作人员规定了下一项任务。Weinzierl 等人 [23] 则推荐预计最有可能带来正面结果的下一项活动。最近，Padella 等人 [14]为业务流程的处方分析提出了一个可解释性框架。这些研究与我们的工作不同之处在于，下一个最佳动作并非一种特殊干预，不会仅在案例处于负向状态时才实施；这些动作属于正常流程执行的一部分。

因果过程挖掘：因果过程挖掘是流程挖掘中一个新兴的子领域，关注发现和估计流程中的因果效应。Koorn 等人 [8] 发现了工作人员对攻击性情境的应对与其有效性之间的因果关系。他们在 [7] 中考虑了可能存在的混杂变量。一些方法使用结构方程模型来发现根本原因，并回答有关非期望结果的反事实问题 [15, 16]。这些方法关注因果效应的发现，而我们使用因果效应估计来改善结果。在我们之前的一项工作中，我们提出了一种基于因果效应的基于规则的推荐方法，以改善流程结果 [2]。在另一项工作中，我们使用因果效应来减少运行时的流程时长 [3]。虽然我们后续的工作侧重于寻找最适合干预的案例，但在本研究中，我们关注何时对案例进行干预。

3 背景

本节介绍本文其余部分所用的流程挖掘相关定义，并介绍本研究方案所依托的因果推断、强化学习和保形预测概念。

3.1 流程挖掘

定义 1（事件、轨迹、事件日志）。事件是一个元组（a，c，t，（d_1，v_1），.....，（d_m，v_m））， $m \in \mathbb{N}_0$，其中， a是活动名称， c是案例标识符， t是时间戳，而（d_1，v_1）.....，（d_m，v_m）是属性值对。轨迹是由具有相同案例标识符且时间戳按升序排列的事件组成的有限序列 $\sigma = \langle e_1, \ldots, e_n \rangle$、 $n \in \mathbb{N}$。事件日志或日志是轨迹的多重集。

定义 2（k-前缀）。轨迹 $\langle e_1, \ldots, e_n \rangle$、 $n \in \mathbb{N}_0$的*k*-前缀是一个序列 $\langle e_1, \ldots, e_k \rangle$、 $0 \leq k \leq n$。

定义 3（序列编码器）。序列编码器 f: $S \to X_1 \times \cdots \times X_p$是一个函数，它接收一条（部分）轨迹 σ，并将其转换为 p 维向量空间 $X_1 \times \cdots \times X_p$ 中的特征向量X，其中 $X_i \subseteq \mathbb{R}$， $1 \leq i \leq p$。

3.2 因果推断

我们使用*Neyman-Rubin* 潜在结果框架 [18] 进行因果推断。干预，或干预，是指在流程执行过程中可以采取的一种动作，以优化案例的结果。本文考虑二元处理设置，其中干预由二元变量 $T \in \{0, 1\}$表示， $T = 1$ 表示对案例实施干预，$T = 0$ 则表示未实施干预。根据潜在结果框架，每个案例都有两个潜在结果：$Y(1)$表示接受干预时的结果，$Y(0)$表示未接受干预时的结果。干预的有效性会因案例以及案例进程中的不同时间点而异。为了衡量干预的有效性，我们使用条件平均处理效应（CATE）：

定义 4（条件平均处理效应）。设 X 为一组用于描述案例的属性。则该案例的条件平均处理效应（CATE）定义如下：

$$CATE : \theta(x) = \mathbb{E}[Y(1) - Y(0) \mid X = x].$$

为识别因果效应，我们采用潜在结果框架中的四个标准假设：正值性、可忽略性、一致性和无干扰。

假设 1 （正值性）。每个案例都有可能被选中接受干预，即对于每种干预 t 以及表示前缀的每个向量 x， $P(T = t \mid X = x) > 0$。

假设 2 （可忽略性）。在给定处理前混杂因素 X 的条件下，干预分配与潜在结果相互独立： $Y(1)\ Y(0) \perp\!\!\!\perp T \mid X$。

假设 3 （一致性）。选择干预后观察到的结果与潜在结果一致：对于所有 t，若 $T = t$，则 $Y = Y(t)$。

假设 4 （无干扰）。对一个案例的治疗决策不会影响其他案例的治疗决策。

文献中已经提出了许多估计条件平均处理效应的方法。我们在本文中采用的其中一种方法是因果森林。

因果森林 是 Athey等人提出的一种因果估计器 [22]。它由一组因果树组成 [1]。因果树是决策树的一种改进，

使其适用于估计因果效应。在决策树中，划分旨在区分类别；而在因果树中，划分旨在提高预期因果效应。因果树与决策树的一个主要区别是诚实划分。这意味着在训练过程中，数据会被分成两组：一组用于构建树，另一组用于在划分后估计处理效应。因果森林通过子抽样方法汇总许多经过诚实划分的因果树的结果构建而成。

Realcause是一种生成具有两种潜在结果的真实数据的方法。在本文中，我们保留真实的前缀，并且只生成两种潜在结果。图 1 展示了 Realcause 的架构。首先，使用所有数据（即前缀）训练一个神经网络，以获得隐藏表示。然后，将接受干预和未接受干预的样本分开，并

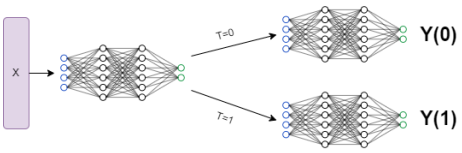


图 1：Realcause 潜在结果生成架构

为每组分别训练两个神经网络。后续神经网络的激活函数用于参数化分布，随后可从中采样，为每个输入样本生成两种潜在结果。

3.3 强化学习

强化学习是机器学习的一个分支，用于训练能够做出一系列决策的模型。通常，智能体被置于一个环境中。它会观察一个状态 s ，并从动作空间的一组动作中执行一个动作 a ，随后观察到作为对 a的响应的奖励 r 和下一个状态 s' 。智能体的目标是最大化累积奖励总和。智能体通常会先随机选择动作，并获得这些动作带来的奖励。通过试错，它会逐渐找出在什么条件下采取哪些动作能获得更多奖励。智能体在学习过程中面临的主要问题之一，是如何在探索与利用之间取得良好的平衡。利用是指智能体运用已获得的知识，执行能够确保获得大量奖励的动作。探索是指智能体尝试新的动作，以寻找其他可能带来奖励的动作。只有在探索与利用之间取得良好的平衡，智能体才能发现所有能带来奖励的动作。

3.4 保形预测

保形预测是一种可为任何预测模型 [17]生成预测集的方法。给定预测模型提供的不确定性分数，保形预测会输出一个类别集合，并在数学上保证该集合包含真实类别。下面我们将介绍如何构造这些预测集，以及它们能够提供哪些保证。

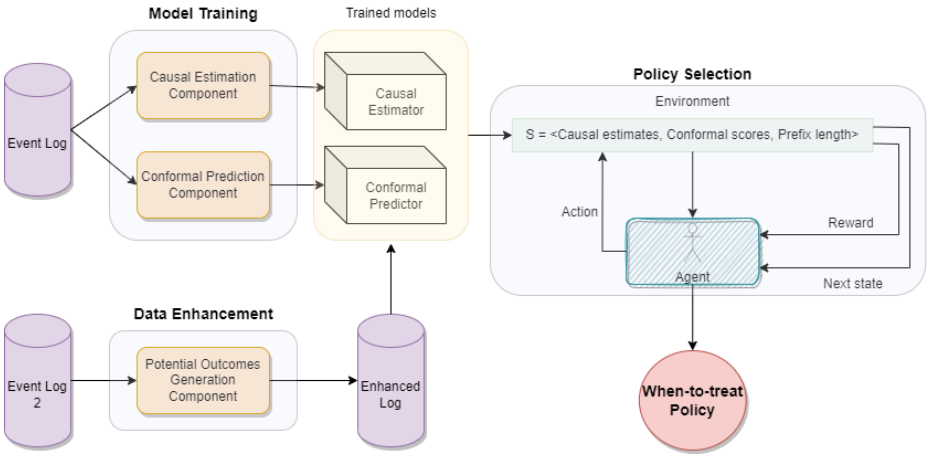


图2：所提方法概述

ˆ 假设我们已经训练了一个预测模型 f，该模型会输出每个预测类别的概率。我们使用一小组大小为 nˆ ˆ的未见校准数据来构造 保序分数 $s_i = 1 - f(x_i)_{Y_i}$，其中 $f(x_i)_{Y_i}$ 表示真实类别的预测概率。如果模型的错误非常严重，则分数 s_i 较高，这意味着 i 与训练数据不一致。接下来，给定 $s_1, \ldots, s_n$，我们定义 ˆ$q = \lceil (n+1)(1-\alpha) \rceil / n$，其中 $\lceil . \rceil$ 是向上取整函数， α 是用户定义的误差容忍阈值。最后，使用一个单独的测试集，我们将保形预测集定义为 $C(X_{test}) = \{y : s(X_{test}) \leq \hat{q}\}$。在 [17], 中已证明，该预测集以概率 $1 - \alpha$保证包含真实类别。

4 方法论

假设智能体 $\mathcal{A}$ 负责对进行中的案例作出是否干预的决策。 $\mathcal{A}$的任务是在每次事件发生后审视每个案例，并决定是否对该案例进行干预。在每个决策点（每次事件发生后）， $\mathcal{A}$ 都需要回答以下两个问题：现在实施干预会让该案例的结果从负向变为正向吗？无论是否进行干预，我对该案例结果有多大把握？为了帮助 $\mathcal{A}$ 回答这些问题，我们使用以往流程执行记录训练两个机器学习模型：一个预测模型，用于估计每种可能结果的概率；一个因果模型，用于估计所选干预的条件平均处理效应。这是该方法的第一阶段。在第二阶段，我们创建一个逼真的环境，让 $\mathcal{A}$ 尝试不同的干预策略并学习其中的最佳策略。为此，我们使用生成模型，为每种前缀长度生成潜在结果，使其在统计上与实际结果无法区分。最后，在第三阶段，我们让$\mathcal{A}$ 通过试错学习最佳干预策略。我们设计了一个奖励函数，指导 $\mathcal{A}$ 判断何时作出正确或错误的决策。图 2 概述了我们的方法。下面将介绍各个阶段。

4.1 模型训练

因果效应估计 为了回答 $\mathcal{A}$的第一个问题，我们训练一个因果估计器。不同于旨在估计 $P(Y|X)$的预测方法，因果估计器估计的是 $CATE$： $P(Y(1) - Y(0)|X)$。通常，在因果估计文献中，案例分为四类：(a) 可说服者，(b) 不可打扰者，(c) 无望者，(d) 确定者（见图 3）。通过因果估计，我们可以将案例划分为这些类别。 $CATE > 0$ 描述可说服者， $CATE < 0$ 描述不可打扰者， $CATE = 0$ 描述确定者和无望者。由于因果估计器难以评估，CATE 的点估计可能不足以作为制定干预决策的可靠依据。因此，我们训练模型计算 CATE 的置信区间。模型接收一个前缀 σ_k，其 $k \leq n$ 为 n，其中 n 是案例的总长度，并返回$\theta_{u,\ k}$ 和$\theta_{l,\ k}$，分别作为前缀长度为 k时因果效应估计值的上界和下界。我们使用因果森林 [22]来计算这些置信区间。因果森林和正交随机森林（ORF） [13] 是两种能够生成有效置信区间的因果估计器。我们选择因果森林，是因为我们发现，在干预和结果均为二元变量时，它的训练速度更快，而性能与 ORF 相近。但总体而言，只要所选因果估计器能够生成置信区间，我们的框架就不受其影响。

保形预测 拥有条件平均处理效应的置信区间后，智能体 $\mathcal{A}$便能够逐渐收敛到良好的策略。然而，仍可能存在因果估计器无法确定的前缀。例如，当 $\theta_{u,\ k} > 0$ 且 $\theta_{l,\ k}$ <0 时。因此，我们假设，如果加入一种额外的严格不确定性概念，例如保形预测，就能帮助智能体更快收敛。回顾一下，借助保形预测，我们可以构造出在数学上有保证包含真实类别的预测集。对于二元结果预测，这些集合将是以下几种之一：(a) $\{\}$、(b) $\{0, 1\}$、(c) $\{0\}$和(d) $\{1\}$。在(a)中

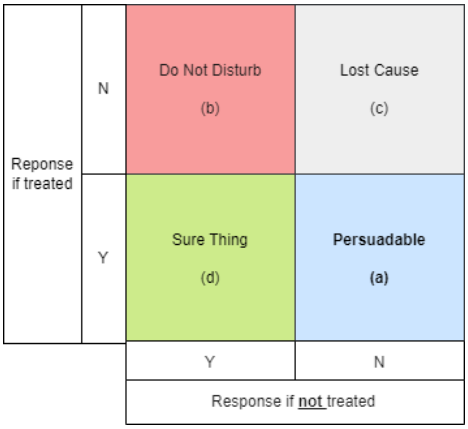


图3：根据案例对干预的响应进行分组

而在（b）中，模型无法确定结果。但在（c）和（d）中，我们可以严格保证该案例最终分别会产生负向和正向结果。这分别对应无望者和确定者。保形预测根据前缀中的信息，告诉我们对结果有把握。因此，向智能体 $\mathcal{A}$展示这些预测集，应该能帮助它缩小需要干预的前缀的搜索空间，因为

它会确定不对保形预测识别出的无望者和确定者进行干预。

要使用保形预测，我们首先需要训练一个预测模型。该模型是从数据中学习得到的函数，它接收一个前缀 σ_k，并返回两个概率： p_k(1)，即前缀长度为 k时结果为正向的概率；以及 p_k(0)，即结果为负向的概率。预测类别定义为概率最高的类别：

$$p(\sigma_k) = Argmax(p_k(1), p_k(0)) \tag{1}$$

我们使用 CatBoost [5] 训练预测模型，因为近期研究表明它表现良好 [14]。然后，我们按照第 3.4 节所述应用保形预测算法，以获得预测集。接着，我们将预测集编码为置信度度量 ρ_k，表示该案例将产生正面结果的置信程度。如果预测集为 (c)，则 $\rho_k = 0$；如果预测集为 (d)，则 $\rho_k = 1$；如果预测集为 (a) 或 (b)，则 $\rho_k = 0.5$。

4.2 数据增强

在在线强化学习中，智能体 $\mathcal{A}$ 必须在不同状态下尝试不同的动作，以学习最优策略。但由于我们希望使用数据来寻找这一策略，因此需要模拟智能体进行学习的环境。该环境需要足够真实，以反映数据中的信息。在这一阶段，我们希望生成替代结果，供 $\mathcal{A}$ 稍后在学习过程中使用，以评估其所掌握的知识并据此做出决策。

在真实世界的数据中，根据 Neyman-Rubin 框架，我们只能观察到潜在结果中的一个。因此，我们无法得知真实的个体层面的因果效应。最近，一种解决方案是使用生成式机器学习来生成与原始数据在统计上无法区分的真实数据。然后，我们可以使用这些生成模型，为描述某个前缀的同一特征

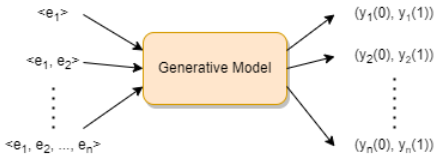


图 4：使用生成式机器学习生成潜在结果

向量获取两个结果：如果 $T = 1$，则为一个结果；如果 $T = 0$，则为另一个结果。其中一种方法称为 Realcause [12]，它使用生成式机器学习模型生成包含真实因果效应的真实数据。在我们之前的工作中 [4]，我们成功应用 Realcause 方法生成了描述案例的编码特征向量。然而，在本文中，我们希望尽可能保持智能体 $\mathcal{A}$的环境真实。因此，我们使用 Realcause 仅为每个案例的每个前缀生成潜在结果（见图 4）。

Realcause 最初旨在处理独立数据点。但在我们的方法中，每个数据点都是一个前缀，而前缀是案例的一部分。因此，为了引导生成模型发现这种依赖关系，我们采取以下步骤：我们

将前缀编号作为特征纳入模型，其中对于每个案例，该特征描述k 每个前缀 σ_k。接下来，我们将案例标识符转换为数值。然后，在模型训练中纳入数值案例 ID 和前缀编号。由于 Realcause 会直接学习数据生成分布，因此它能够发现具有相同数值案例 ID 的前缀编号之间的关系 $1 \dots n$。我们使用两个具有两层隐藏层的多层感知机来对处理组和未处理组建模。我们假设流程结果服从伯努利分布。生成模型在原始事件日志上训练完成后，我们可以在干预和未干预两种条件下对结果模型进行采样，从而生成增强版事件日志，并获得每个前缀的两种潜在结果。我们在策略选择阶段使用增强版日志。

4.3 策略选择

针对 $\mathcal{A}$构建包含两个潜在结果的学习环境后，下一步是让 $\mathcal{A}$ 通过学习干预策略，将其知识映射为动作。策略是一个函数 π ，它将前缀 σ_k 映射为动作 t。我们的目标是找到一个能使净收益函数最大化的策略。为此，我们使用强化学习。具体而言，我们采用 [11]提出的基于策略的强化学习框架。下文将使用强化学习来表述这一学习问题。

如第 3.3 节所述，在强化学习中，智能体置身于一个环境中，并观察状态 s_k ，这些状态描述了每个前缀长度 k下的环境。智能体从动作空间 $\mathcal{T}$中选择一个动作 t ，并观察奖励 r 和下一个状态 s_{k+1}。随后，智能体通过试错学习最佳行为，直到找到选择动作的最优策略。为了将这一学习问题转化为处方式监控，我们对 Metzger等人提出的基于强化学习的方法进行了修改 [11]。与他们类似，我们定义一个二元动作空间 $\{0, 1\}$ ，其中 $t = 1$ 表示实施干预，$t = 0$表示不实施干预。Metzger等人将状态 s 描述为一个元组$s = (\delta_k, \gamma_k, k)$，其中 δ_k 是对正面结果偏差的预测， γ_k是该预测的可靠性评分， k 是前缀长度。我们建议修改 s ，使其包含估计的条件平均处理效应区间和保形预测分数： $s = (\theta_{u,\ k}, \theta_{l,\ k}, \rho_k, k)$。

学习问题的一个重要方面是定义合适的奖励函数。智能体 $\mathcal{A}$ 在强化学习中的目标是最大化累积奖励。由于我们认为最佳策略是能够最大化净收益的策略，因此最直接的奖励就是每个案例结束时获得的收益或损失。因此，我们决定将干预成本和正向案例带来的收益纳入奖励函数。这种设计选择背后的直觉是，干预策略与成本和收益之间的比率直接相关。例如，如果干预成本低且收益高，即使我们不确定其有效性，也可以更频繁地实施干预。但如果干预成本高，那么仔细选择干预的案例和时机就更加重要；只有在智能体确信干预必要且有效时，才实施干预。

奖励函数还需要包含每个决策点上干预有效性的信息。回想一下，在数据增强阶段，我们为每个案例的每种前缀长度生成了 $Y(1)$ 和 $Y(0)$。我们可以为每个前缀计算 $Y(1) - Y(0)$，从而得到每个决策点上的真实处理效应。我们将这一真实效应纳入奖励函数，以引导智能体判断干预的有效性。奖励函数的具体细节见表1。

	真实治疗效果		
智能体的干预	正向	负向	Zero
Yes	收益 - 成本	-成本-收益	负面结果：-成本 正面结果：-成本
No	-收益	Gain	负面结果：0 正面结果：收益

表1：所提出的奖励函数

当智能体实施干预时，如果处理效应为正向，我们会给予$r = Gain - Cost$的奖励，因为我们在支付干预成本的同时，获得了正面结果带来的收益。如果处理效应为零，我们会给予智能体 $r = -Cost$的负向奖励，以此惩罚它，因为它白白付出了干预成本，而干预也没有效果。如果处理效应为负向，我们会给予智能体更低的负向奖励 $r = -Cost - Gain$，以此惩罚它。我们选择这样的奖励，是因为智能体不仅白白付出了干预成本，还在干预有害于结果时实施了干预，从而造成了进一步的损害。

当智能体不实施干预时，如果处理效应为正向，我们会给予它 $r = -Gain$ 的奖励，以此惩罚它，因为它在有必要采取行动时未能行动。如果处理效应为零，我们会考察结果。如果结果为正向，说明智能体正确地决定不实施干预，节省了干预成本，因此 $r = Gain$。如果结果为负向，说明该案例已无可挽回，即使实施干预也不会有任何改变。因此，尽管智能体不实施干预的决定是正确的，我们仍会给予它 $r = 0$ ，因为没有获得任何收益。最后，如果处理效应为负向，我们会给予它较高的正向奖励 $r = Gain$ ，因为在干预有害于结果时选择不干预，带来了正面结果。

该奖励函数能够很好地模拟遵循智能体策略所获得的净收益，但有三种情况例外。第一，如果干预本来会有效，但智能体没有进行干预，实际收益为 0，而我们会通过$-Gain$惩罚智能体。第二，如果智能体进行了负向干预，我们会损失干预的成本，因此净收益为 $-Cost$，但奖励为 $-Cost - Gain$。第三，如果智能体进行了干预，且干预无效，但结果为正向，则净收益为 $Gain - Cost$，但我们会给予 $r = -Cost$。我们加入了这些额外的惩罚，以向智能体表明它做出了错误的决策。根据 [20], ，奖励函数通常需要进行调整，以加快学习和收敛，并避免陷入局部最优。我们发现，这些额外的惩罚对于加快学习过程是必要的。

我们为策略选择阶段使用了一组独立的案例，其时间范围晚于用于模型训练的案例。智能体通过一系列

回合进行学习。每个回合对应一个案例。在每个回合中，事件会按照时间戳的升序呈现给智能体。在每个回合结束时，我们向智能体揭示累积奖励。与 [11], 一样，我们使用近端策略优化（PPO） [19]作为强化学习算法。我们还将策略表示为一个多层感知机。该网络之后可以作为现实场景中的起点。由于它是在潜在结果与其真实对应结果在统计上相似的数据上训练的，因此，与让智能体从头开始相比，它犯的错误会更少。

5 结果

本节介绍实验设置并报告实验结果。我们使用 Python 3.8 开发了我们的方法。我们使用 CatBoost 库构建预测模型，并使用 EconML 实现因果森林。生成模型使用 PyTorch 开发。所有数据均按时间进行拆分，以确保移除训练集中时间范围与测试集时间范围重叠的案例。我们将数据分成两半。第一半用于训练预测模型和因果模型，第二半用于通过强化学习进行策略选择。预测模型、因果模型和生成模型采用相同的预处理和特征工程步骤。我们对类别属性进行独热编码。为编码时间戳信息，我们创建了以下时间特征：“案例开始后的时间”“距上次事件的时间”“距午夜的时间”“月份”“星期”和“小时”。此外，为了捕捉案例之间的时间关系，我们创建了一个名为“距首个案例的时间”的特征，它是一个案例属性，用于表示案例开始时间与首个案例开始时间之间的间隔。为预处理数据，我们对特征进行标准化。我们使用两个数据集进行实验，并将我们的方法与处方性流程监控领域的最先进方法进行比较。

5.1 数据集

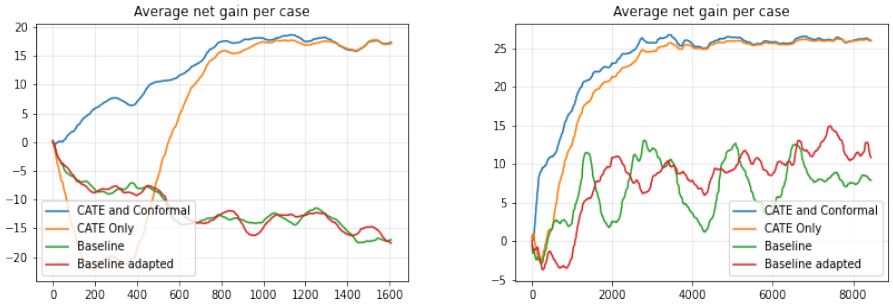
我们在两个公开可用的数据集上进行了实验，即 BPIC12 和 BPIC17。据我们所知，我们选择这两个数据集，是因为它们是唯二在日志中包含干预且该干预能够影响流程结果的公开数据集。这两个日志都包含贷款发放流程的轨迹。如果客户接受贷款报价，我们就认为流程结果为正向。通常，每位客户只会收到一份贷款报价。但我们也观察到，有些案例中，同一位客户会收到不止一份报价。我们发现，这类案例的成功率更高。因此，我们将向同一位客户提供多份贷款报价视为一种可能的干预。尽管这两个日志都涉及同一流程，但二者存在一些差异。BPIC17 日志中的案例数量明显更多，包含的特征也更多。

5.2 性能度量

在本实验中，我们通过衡量以下收益函数来评估我们方法中各个组件的效果：

$$NetGain = Y(t) * gain - t * cost, \tag{2}$$

其中，$t \in \{0, 1\}$ 是智能体建议的干预选项。净收益衡量我们从每个案例中获得的金额。负向净收益表示亏损。我们首先仅向智能体提供 CATE估计值$\theta_{u,\ k}$ 和 $\theta_{l,\ k}$ 来运行实验。在第二个实验中，我们同时向强化学习智能体提供 CATE估计值和保形预测分数 ρ_k 。我们还将本文方法与 [11] 提出的方法进行比较，因为后者是解决何时治疗问题的最先进方法。图 5 展示了两个数据集的结果。



(a) BPIC 2012　　(b) BPIC 2017

图 5：案例的平均净收益。

在上面的图中，我们考虑一种昂贵的干预。具体而言，我们假设正面结果的收益为 $50，干预成本为 $25。我们还试验了廉价干预（例如，$1）。但由于廉价干预不需要严格的策略，我们将相关结果放在补充材料中。图 5 的 x 轴表示智能体决定采取行动的案例，y 轴表示每个案例的净收益。我们可以看到，使用 CATE 上界和下界的智能体，表现优于依据案例结果预测进行决策的智能体。这是因为使用预测的智能体会针对最终结果不佳的案例。但使用 CATE 的智能体掌握了有关干预有效性的信息，只有在确信干预能将负面结果转变为正面结果时才会采取干预。换言之，它们针对的是图 3 中的可说服者。此外，我们观察到，同时使用保形预测分数和 CATE 估计值的智能体收敛更快，这从经验上证明了我们的假设：保形预测能够识别确定者和无望者，从而缩小智能体的搜索空间，帮助其减少错误。如果在真实环境中在线应用强化学习，这一点非常重要

因为如果没有保形预测，我们会发现，最初几百个案例的净收益为负。

我们运行了两个版本的基线。第一个版本（以绿色显示）使用 [11] 中提出的奖励函数。第二个版本（以红色显示）对基线进行了调整，使其使用我们提出的奖励函数（表1）。我们可以看到，在 BPIC17 数据集中，基线表现不稳定。这是因为在某些案例中，智能体掌握的预测信息与干预的有效性相符，因此智能体能够正确决定是否进行干预。但有时，智能体会在需要干预但干预无效时采取干预措施。使用我们提出的奖励函数时，可以看到性能缓慢提升。这是因为奖励函数使智能体逐渐学会更有选择性地决定何时采取干预措施。但对其环境的描述（预测及其可靠性）所包含的信号不足，无法使其更快收敛。在 BPIC12 数据集中，两个版本的基线都产生了负向净收益。这是因为该日志远小于 BPIC17 日志。因此，智能体看不到足够的样本，无法发现能够产生正向净收益的策略。

我们还尝试了直观奖励函数，该函数不会因智能体做出错误决策而施加任何额外惩罚（见第 4.3 节）。然而，我们发现这个奖励函数无法产生良好的结果。这是因为该奖励函数提供的信号不足以让智能体区分正确与错误的决策，导致智能体进行不必要的治疗。调整智能体的探索/利用率可以解决这个问题。不过，由于我们使用的算法（PPO）不允许手动调整这一比率，因此我们将其留待未来工作。该实验的结果以及替代奖励函数的具体说明可在补充材料中找到。

5.3 增强日志上的统计检验

在本节中，我们评估所生成潜在结果的质量。为此，我们进行了几项统计双样本检验。我们检验了生成结果与真实结果是否来自不同分布这一假设，并报告 p 值。任何高于 0.05 的 p 值都表明，该检验无法得出两个样本来自不同分布的结论。我们采用了与 [4] 中相同的检验，并在表 2 中报告结果。可以看出，所有 p 值都高于 0.05 的阈值，这意味着所有检验都无法检测出增强后的结果与真实结果来自不同分布。

	Kolmogorov Smirnov	Epps Singleton	弗里德曼 拉夫斯基	k 近邻 邻居	能量	Wasserstein 1	Wasserstein 2
BPIC12	0.244	0.311	0.109	0.088	0.228	0.323	0.329
BPIC17	1.0	0.996	0.647	0.359	0.92	0.916	0.87

表 2：统计检验的 p 值表

5.4 有效性威胁

实验中使用的事件日志数量较少，这对外部效度构成了威胁。我们生成了两个潜在结果来增强实验中使用的日志，这对内部效度构成了威胁。这些潜在结果的质量取决于输入数据的质量。因此，分析人员在应用我们的方法之前，必须确保拥有足量的高质量数据。使用观察性数据估计条件平均处理效应会对构念效度构成威胁。当事件日志中未包含所有可能的混杂因素时，与纯粹基于相关性的方法相比，因果模型能减少偏差，但无法消除偏差。另一个对构念效度的威胁在于，我们在强化学习阶段将案例视为回合。这意味着，如果日志中存在较长的案例，智能体可能会接触到未来信息。为减轻这一威胁，我们采取了以下措施。首先，我们从环境描述中排除了所有案例信息。智能体只能看到CATE估计值、保形预测分数和前缀编号。然而，由于这些估计值和分数是根据使用案例信息的模型得出的，智能体仍可能间接接触到未来信息。其次，我们按照案例的结束时间戳对案例进行排序，尽可能减少向智能体泄露未来信息。

6 结论

本文介绍了一种学习策略的方法，用于为业务流程中的案例指定干预措施，以最大化这些干预措施所产生的净收益。所提出的方法通过以下方式改进了现有的在线强化学习方法：i）输入因果效应估计值，以避免无效干预并提高净收益；ii）利用保形预测加速强化学习智能体的收敛；iii）采用因果数据集增强方法，模拟一个可供强化学习智能体离线训练的环境。

在所提出的方法中，每个学习回合对应一个案例。这种做法可能会使智能体接触到只有未来才会知道的信息，因为不同案例可能存在时间重叠（数据泄漏）。改进这项工作的一个自然方向是改变回合的定义，以避免此类泄漏。例如，每个回合可以涵盖同一天内发生的、涉及多个案例的所有事件。这里的难点在于找到合适的时间步长，以充分利用现有数据集。未来工作的另一个方向是尝试允许手动控制探索/利用率的强化学习算法，从而为进一步调优提供足够的控制空间。我们还将研究间接调整这一比率的方法，例如在奖励函数中添加系数。未来工作的其他方向包括尝试其他强化学习技术，并通过案例研究验证所提出的方法。

可复现性 我们的工具源代码、所用数据集和实验结果可在 https://github.com/zahradbozorgi/WhenToTrea 查阅 t

致谢 研究经费由澳大利亚研究理事会（资助项目 DP180102839）、欧洲研究委员会（PIX项目）和爱沙尼亚研究委员会（资助项目 PRG1226）提供。

参考文献

1. Athey, S., Imbens, G.：针对异质性因果效应的递归分区。《美国国家科学院院刊》（2016） 2. Bozorgi, Z.D., Teinemaa, I., Dumas, M., Rosa, M.L., Polyvyanyy, A.：流程挖掘与因果机器学习相结合：从事件日志中发现因果规则。载于：第2届 ICPM（2020） 3. Bozorgi, Z.D., Teinemaa, I., Dumas, M., Rosa, M.L., Polyvyanyy, A.：面向成本感知型周期时间缩减的处方性流程监控。载于：第3届 ICPM（2021） 4. Bozorgi, Z.D., Teinemaa, I., Dumas, M., Rosa, M.L., Polyvyanyy, A.：基于因果效应估计的处方性流程监控。审稿中（2022） 5. Dorogush, A.V., Ershov, V., Gulin, A.：CatBoost：支持类别特征的梯度提升。NIPS 的机器学习系统研讨会（2017） 6. Fahrenkrog-Petersen, S.A., Tax, N., Teinemaa, I., Dumas, M., de Leoni, M., Maggi, F.M., Weidlich, M.：现在告警，稍后处理：用于处方性流程监控的基于告警的系统。《知识与信息系统》（2022） 7. Koorn, J.J., Lu, X., Leopold, H., Martin, N., Verboven, S., Reijers, H.A.：挖掘统计关系，以改善医疗保健流程中的决策（2022） 8. Koorn, J.J., Lu, X., Leopold, H., Reijers, H.A.：寻求意义：发现业务流程中的行动-响应-效果模式。载于：BPM会议论文集（2020） 9. Kubrak, K., Milani, F., Nolte, A., Dumas, M.：处方性流程监控：何去何从？《PeerJ计算机科学》8（2022） 10. de Leoni, M., Dees, M., Reulink, L.：基于处方分析的流程感知推荐系统的设计与评估。载于：第2届 ICPM（2020） 11. Metzger, A., Kley, T., Palm, A.：通过在线强化学习触发主动式业务流程调整。载于：BPM会议论文集（2020） 12. Neal, B., Huang, C.W., Raghupathi, S.：Realcause：真实场景因果推断基准测试（2021） 13. Oprescu, M., Syrgkanis, V., Wu, Z.S.：用于因果推断的正交随机森林。载于：第36届 ICML 论文集（2019） 14. Padella, A., de Leoni, M., Dogan, O., Galanti, R.：可解释的流程规范性分析（2022） 15. Qafari, M.S., van der Aalst, W.M.P.：利用结构方程模型进行流程挖掘中的根因分析。载于：BPM国际研讨会，修订版精选论文（2020） 16. Qafari, M.S., van der Aalst, W.M.P.：流程挖掘中的案例级反事实推理。载于：CAiSE论坛论文集（2021） 17. Romano, Y., Sesia, M., Candes, E.：具有有效自适应覆盖率的分类。NeurIPS（2020） 18. Rubin, D.B.：估计随机与非随机研究中干预措施的因果效应。《教育心理学杂志》 19. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.：近端策略优化算法。arXiv 预印本 arXiv:1707.06347（2017） 20. Sutton, R.S., Barto, A.G.：强化学习：导论。麻省理工学院出版社（2018） 21. Teinemaa, I., Tax, N., de Leoni, M., Dumas, M., Maggi, F.M.：基于告警的处方性流程监控。载于：BPM论坛（2018） 22. Wager, S., Athey, S.：利用随机森林估计和推断异质性处理效应。《美国统计学会杂志》（2018） 23. Weinzierl, S., Dunzer, S., Zilker, S., Matzner, M.：用于推荐下一最佳行动的规范性业务流程监控。载于：BPM论坛（2020）