迈向基于不变因果学习算法在观测数据上的适用性

博尔哈·格雷罗·桑蒂利安马德里理工大学（UPM），
ETSIINF，西班牙 b.guerrero@alumnos.upm.es

2023年3月30日

摘要

机器学习可以借助因果发现来实现可解释性，也可以借助因果推断来提升泛化能力。在这一研究方向中，研究人员提出了一些用于分布外（OOD）泛化的不变学习算法，通过利用多个训练环境寻找不变关系。其中一些算法侧重于因果发现，例如不变因果预测（ICP），它可以找出目标变量的因果父节点；另一些则直接提供能在 OOD 环境中良好泛化的因果最优预测器，例如不变风险最小化（IRM）。这类算法假定存在多个环境，并将其视为因果推断背景下不同干预的体现。然而，在处理观测数据和现实应用时，通常无法获得此类环境。本文提出了一种高效生成这些环境的方法。我们通过在模拟数据上实现 ICP，评估这一无监督学习问题的性能。我们还展示了如何将 ICP 与我们的方法结合，高效地应用于因果发现。最后，我们提出了改进版方法，并将其与 ICP 结合，用于包含多个协变量的数据集；在此类数据集上，ICP 和其他因果发现方法通常会出现性能下降。

关键词 因果机器学习 · 因果发现 · 分布外（OOD）泛化 · 可解释性

1 引言

当前的机器学习方法存在一个根本性问题：它们局限于模式识别，并通过最小化训练误差来学习复杂的预测规则。因此，机器学习系统依赖数据中的偏差，这些偏差基于可能存在的虚假相关性，却无法理解数据中的因果关系。这导致其泛化能力不足，难以在未见过的情境中进行预测。因果关系是实现分布外泛化、构建能够回答干预问题的更稳健系统的关键。当前机器学习方法的另一个重要局限是可解释性不足。同样是因为它们并未真正理解数据中的原因与结果。尽管基于模型解释的方法（如 LIME [22] 和 SHAP [17] ）有助于解决这一问题，但它们并未真正回答数据如何表现，只说明模型如何表现。要克服这一局限，因果关系同样至关重要。由此可见，推断因果关系非常重要。通常，标准做法是开展随机对照试验：将一个群体随机分配并接受干预（改变某个感兴趣的变量），而另一个群体则不接受干预。通过这类实验收集数据，并用于衡量该干预的因果效应。然而，这类实验可能难以开展，甚至无法开展；也可能因成本、时间及其他限制而不值得开展。在这些情况下，我们会使用观测数据，并应用定量方法发现因果关系，这一过程称为因果发现。

因果发现是一个具有挑战性的领域，通常依赖领域知识，而非定量工具。然而，如果缺乏关于可能因果关系的先验知识，我们就需要运用数学方法。因果发现以结构因果模型（SCM）[3],[24],[19] 和因果图[16], [9], [27], [23], [25]的理论为基础。因果图是一种有向无环图，用于表示变量之间的依赖关系。结构因果

模型还规定了变量之间依赖关系的函数形式。在基于图模型的计算方法中，我们可以区分基于约束和基于评分的方法，以及基于函数因果模型的方法。PC 和快速因果推断（FCI）[26] 属于基于约束的方法，通过在一个初始的合理因果图上检验条件独立性来开展工作。另一方面，贪婪等价搜索（GES）[5] 旨在从零开始构建因果图，并检查待添加的因果关系是否能优化预先定义的评分函数。此外，还有基于非高斯模型 [25] 和非线性模型[12],[21] 的结构因果模型算法；这些算法通常能够区分不同的有向无环图，但代价是需要对结构因果模型作出额外假设。然而，总的来说，这些方法存在局限，因为多个因果结构可能满足相同的条件独立关系。因此，它们通常只能给出一组合理的因果图，而其中未必包含真实的因果图。

近年来，研究热潮开始关注因果关系的不变性，以服务于因果发现和因果推断。其中较早的一项工作是不变因果预测（ICP）[20] ，这是一种因果发现算法，其基础是考察目标变量的所有直接原因。该方法试图利用因果关系相较于虚假相关所具有的众所周知的不变性来推断因果关系。具体而言，该方法的假设是：在不同干预下，给定一组直接因果预测变量时，目标变量的条件分布必须保持不变。在此背景下，研究者定义了环境的概念，即不同的数据集，每个数据集都代表一种特定干预。随后，算法迭代考察特征子集的多种组合，找出其中具有不变性的组合，并将其视为目标变量合理的因果父节点。最后，这些合理原因集合的交集（这些原因在所有环境中均具有预测性）是实际直接原因的一个子集。

不变风险最小化（IRM）[2] 是另一种因果发现算法，其推断依据同样是利用因果关系的不变性。IRM也基于在多个环境中进行训练这一理念。不过，IRM的重点并非在因果图中找出目标变量的因果父节点，而是旨在使预测模型在未见过的环境中实现分布外泛化。IRM方法通过优化一个惩罚函数来获得数据表示，使模型在所有环境中都能达到最佳表现。后续相关研究沿用了这一理念，最终目标是在不同机器学习任务中提升分布外泛化能力[4], [7],[13], [1],[14]。其中一些研究聚焦于非线性等特定情形[11] 以及鲁棒性，尤其是在数据分布发生显著变化的场景中[10],[13],[18],[15]。

在这项工作的绝大部分内容中，环境这一概念都发挥着核心作用。通常假设数据中已给定环境。然而，结合现实情境，尤其是在处理观测数据时，情况通常并非如此，因此无法应用这类算法。近期的一些研究，特别是在深度学习领域，已引入划分方法来生成环境 [28]。此外，ISL[6] 首次将无监督学习，尤其是 K-means 算法，纳入不变学习算法中。
我们提出了一种基于“监督聚类”的方法，用于生成环境。这些环境可与 ICP 结合或单独用于因果发现，也可与另一种基于因果不变性的算法结合。我们严格解读了 ICP 和 IRM 等原始研究对给定环境所作的假设。因此，这为实现这类算法敞开了大门。该方法适用于观测数据以及一般的现实应用。我们的方法在包含连续、类别型和二元特征的混合数据集上表现良好，即使给定数据中没有显著的分布偏移时也如此；这种情况往往比相反情况更具挑战性。我们将该方法与 ICP 结合，在多个环境未知的合成因果数据集上评估其性能，以检验该方法的实际效果。

2 方法论

正如我们所见，在处理观测数据时，通常我们并不知道应用不变性所需的环境。因此，我们需要一种方法，为给定的观测数据集生成该环境。

2.1 生成环境

在因果发现和 ICP 算法的背景下，我们可以定义一组输入或协变量 X，以及一个目标变量 Y，并研究其因果父节点。随后，在不变因果算法中，研究人员采用的主要假设是，这些环境代表不同的来源，可能对应于随机实验中的不同干预。根据定义，这些环境之间的协变量分布会发生变化，这在机器学习文献中称为协变量偏移。通常，这一术语指训练集与测试集之间的偏移，而在此处的语境中，它指的是不同环境之间的偏移。

更严格地说，其他研究人员将这些环境定义为不仅输入发生变化，而且输入与目标之间的联合分布也发生变化，即 P(X, Y)；这种情况通常称为数据集偏移。由于这一假设是成功应用 ICP 的必要条件，因此我们在生成环境时以此为目标。本文提出了一种基于对该假设的严格解释的方法，以确保环境之间存在数据集偏移。该假设可以写作

当 $j \neq k$ 且 $\forall e \in E$ 时，满足 $P(X_i, Y)_{ej} \neq P(X_i, Y)_{ek}$ 。一种宽泛的解释是，在不同环境之间生成协变量偏移。第二种解释可能并非最优，因此更有可能无法发现现有的因果关系。

换言之，这些环境是未知的潜在聚类。鉴于我们已经阐明了在这些环境之间生成数据集偏移的必要性，因此我们提出了一种监督聚类方法，即一种采用监督方法的无监督分类。通过这种方式，我们能够获得输入与感兴趣的目标变量之间具有不同联合分布的聚类。然而，并非所有监督模型都能提供可识别的聚类。在此，我们提出将决策树作为能够同时满足这两个要求的算法。首先，决策树可以作为一种监督算法进行拟合，并且能够最大化每次分裂所生成节点之间的异质性。换言之，它能够在考虑输入及其与输出之间关系的基础上，最大化节点间的方差，而这正是我们希望实现的目标。另一方面，各次分裂均由特征上的阈值明确定义，这些阈值被称为分裂条件。因此，我们能够获得相应的聚类，其中每个最终节点都构成一个环境。

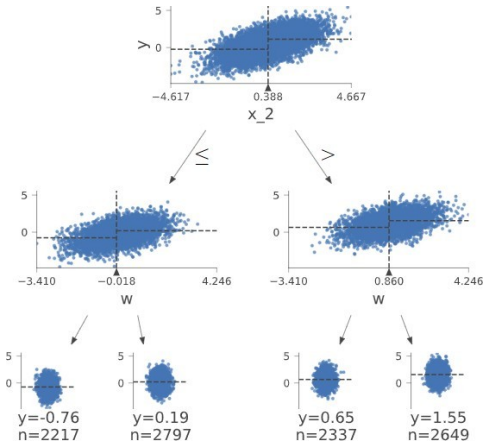

图 1：决策树聚类。

一个重要方面是每个变量都能满足这一要求。因此，环境的生成以每个输入变量和感兴趣的目标为条件。由此，对于给定的固定环境数 E，我们将针对 X 中每个 X_i ，在以 Y 为条件的情况下生成 E 个环境。这样，我们就能保证对每个协变量检验《因果关系》。关于簇的数量，我们倾向于选择中等数量，而非较少数量。简单来说，这是因为结果更加稳健，也比例如仅有两个环境更有保障。另一方面，我们尚未发现使用大量环境具有特别的优势。

2.2 理论框架

生成环境后，我们就可以利用因果不变性这一性质，在观测数据上进行因果发现。我们回顾 [20] 最初定义的不变性假设，并讨论可识别因果预测变量的概念。将 E 定义为所有可能干预的空间，因此 E 包含所有的

各个环境 e。如上所述，我们有变量 $(X)_e$、$(Y)_e$ ，它们的联合分布通常取决于环境 $e \in E$。

不变预测的命题1：存在一个维度为 p 的系数向量 $\gamma\star$ ，其中 p 是 y 的真实因果父节点的数量，该向量满足对所有 $e \in E$ ：

$$(Y)_e = \mu + (X)_e\gamma \star + (\epsilon)_e, \quad \epsilon_e \sim F_\epsilon, \quad \epsilon_e \perp\!\!\!\perp X_e \in S^\star \tag{1}$$

其中，ϵ_e 是均值为零、方差有限且在所有环境 $e \in E$ 中具有相同分布 $F \in$ 的误差。因此，在结构方程模型中，对 Y 具有直接因果效应的变量构成集合 $S^\star$ ，该集合满足命题1。因此，不变预测假设的第一个推论是，给定包含真实

Y 的因果父节点 it 系数向量在不同环境之间保持不变。第二个推论是, e
模型的残差 在不同环境中必须具有相等的均值，并且在严格的解释下，方差也必须相等。
更正式地说，从概率论的角度来看，我们可以将不变因果性质定义为：在给定因果父节点的条件下，目标变量的条件概率在不同环境之间相等。那么，我们为因果发现提出的完整理论框架必须满足两个条件：

1. 不同环境中目标变量与输入变量的联合分布不一致，有时也称为数据集偏移。

$$P(X_i, Y)_{ej} \neq P(X_i, Y)_{ek}, \quad j \neq k, \quad \forall e \in E \tag{2}$$

其中，i = =l, 2, ..., N，N 为协变量的数量；j、k = = l, 2, ..., K，K 为.....

环境。

2. 给定因果父节点时目标变量条件概率相等。

$$P(X_i|Y)_{ej} = P(X_i|Y)_{ek}, \quad j \neq k, \quad \forall e \in E \tag{3}$$

其中，i =l,2,...,N，N 为协变量的数量；j、k = l,2,...,K，K 为

环境的数量。

由于我们的方法针对每个变量分别生成相应的环境，以确保满足条件 1，因此还必须针对相应的环境 e 单独检查条件 2；只有同时满足这两个条件时，我们才认为 X_i 是目标变量 Y 的因果父节点。

2.3 算法

理解理论框架后，我们可以定义算法的步骤。给定一个数据集，我们选择一个目标变量 Y 进行研究，并将其余协变量记为 X，步骤如下：

1. 对每个 X_i 和 Y 拟合一棵决策树，并固定环境数量 K；对于二元或类别型目标变量，将其用作分类器，否则用作回归器。2. 保留决策树为每个变量定义的阈值，以构成环境。由此，对于每个 X_i，我们得到 K 个环境 ($e_1i, e_2i, \ldots, e_ki$)。我们可以进行双样本KS检验，以检查环境之间的数据集偏移。3. 对于输入变量的所有可能子集：(a) 使用所有环境以及该子集中指定的输入特征训练模型，并获取残差。对于线性情形，我们拟合线性回归，其估计参数为 $\beta(S)$，残差为 $R = Y - X\beta(S)^*$。(b) 使用双样本T检验检验零假设，即每个环境中的残差均值相等；合并所有结果时，使用Bonferroni校正。对于方差相等的情形，使用F检验和Bonferroni校正。最后，将两个p值合并为其最小值的两倍；若最终p值低于置信水平 α，则拒绝该子集。4. 找出所有未被拒绝的子集中均包含的特征，作为潜在因果预测变量。仅将变量 X_i 中那些在相关环境 ($e_1i, e_2i, \ldots, e_ki$) 中属于因果父节点候选项的变量，选作目标变量 Y 的因果父节点。

对于包含多个协变量的情况，ICP 的性能可能会下降，因为其保守特性使问题更具挑战性。针对这些情况，我们提出了同一算法的另一种版本，旨在构建更稳健的流程来寻找因果父节点的候选变量。其步骤上的差异如下：

3. 在生成输入变量的所有可能子集之前，我们先将输入变量的数量限制为 N=5 ，然后生成所有可能的子集。现在，这些子集中的每一个都成为我们的原始输入，并按照算法原始版本中的步骤 3 进行处理。4. 由于我们进行了多次检查，因此提出一种投票方法。该方法通过统计每个可能候选项的数量（遵循之前相同的逻辑），并使用新的置信区间来保留最终选定的父节点。例如，如果我们选择 α = o.1，那么只有在至少 90 % 的检查中都被选为候选项的候选项，才会被接受为因果父节点。

正如我们将在下一节看到的，对于包含多个协变量且因果关系更复杂的情况，与算法的原始版本相比，这个第二版本已证明能够取得更好的结果。

3 数值结果

在本节中，我们区分两种实验。由于生成环境时我们要解决的是无监督学习问题，因此需要一种方法来评估该方法。为此，我们将使用环境，按照上一节描述的算法应用 ICP。本实验中使用的 ICP 实现是 Python 库 ICPy [8] 的一个改编版本。随后，我们将该方法应用于模拟数据，并通过真正率（TPR）和假发现率（FDR）衡量因果发现能力。作为这些指标的基准，我们还实现了其他传统的因果发现算法，包括 FCI 和 LINGAM。本次比较有两个目的：首先，它使我们能够检验 ICP´发现因果关系的能力；其次，也是更重要的一点，它能让我们了解算法在生成环境时是否高效。
在第一个实验中，我们生成了 3 个不同的数据集。部分协变量通过高斯过程或二项过程随机生成，其余变量则通过结构方程模型（SEM）和高斯噪声生成。这些数据集从简单到复杂，涵盖了不同的设置。具体而言，第二个数据集包含通过同时影响原因和结果的变量来衡量混杂因素的情况。第二个数据集还包含二元变量和分类变量，我们的方法针对这些变量进行了适配，以确保其能够良好运行，包括方法的第一部分以及 ICP算法。

然后，我们对每个数据集进行5次模拟，并应用我们的方法。在这里，我们对每个数据集中的每个变量迭代应用该方法。因此，每个变量都会成为我们感兴趣的目标变量，以尝试重建完整的因果图。这样，我们便能以更具挑战性的方式测试该算法并检验其可靠性。此外，我们还可以轻松地将其与其他因果发现算法进行比较。最后，我们计算5次模拟的平均真正率和假发现率，作为稳健的指标。表1和表2分别列出了本实验的真正率和假发现率。

	ICP	FCI	LINGAM
数据集1	1.0	1.0	1.0
数据集2	1.0	1.0	1.0
数据集3	1.0	1.0	1.0

表1：实验1的真正率比较

	ICP	FCI	LINGAM
数据集1	0	0.08	0.16
数据集2	0	0	0.14
数据集3	0.04	0	0.20

表2：实验1的假发现率比较。

关于结果的解读，真正率（TPR）表示算法检测出的图中因果父节点所占的百分比，其中 1 为最佳情况，0 为最差情况。而假发现率（FDR）表示变量被选为因果父节点、但实际上并非因果父节点的频率，其中 0 为最佳情况，1 为最差情况。从表 1 的结果可以看出，ICP 能够与其他方法相媲美，其真正率达到 100%，意味着它几乎能找到所有变量的因果预测变量。表 2 中的结果也显示，最高比率约为 4%，而 FCI 为 8%，LINGAM 为 20%。基于这些结果，我们可以认为，生成这些环境能够有效检验数据中因果关系的不变性。

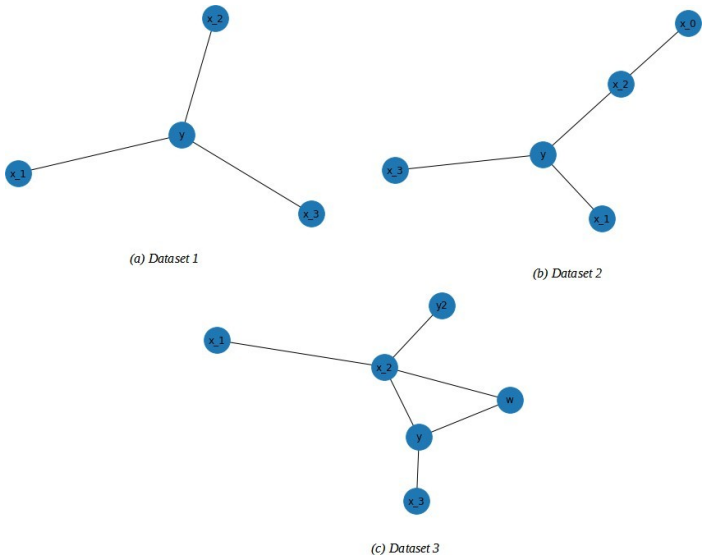

图2：ICPv1 在数据集1、2和3上发现的因果图。

在第二次实验中，我们生成了两个新的数据集。遵循与之前相同的逻辑，这些数据集更加复杂，包含更多的协变量和因果关系。众所周知，随着协变量数量的增加，因果发现算法的性能会下降。另一方面，众所周知，ICP 也存在同样的问题。检验这些局限性正是第二次实验的目标。我们将第一种方法称为 ICPv1，第二种方法称为 ICPv2。

鉴于这些局限性以及初步结果，我们开发了算法的另一个版本，正如我们已在第3节中所述。该版本的算法基于相同的概念，其主要目标是在包含多个协变量的数据集中具有更强的鲁棒性。我们将其纳入本次实验的比较，并讨论表3和表4中的结果。

	FCI	LINGAM	ICPv1	ICPv2
数据集5	0.73	1.0	.73	1.0
数据集6	0.87	1.0	0.6	1.0

表3：实验2的真正率比较。

	FCI	LINGAM	ICPv1	ICPv2
数据集5	0	0.21	0.11	0.08
数据集6	0	0.38	0.1	0.21

表4：实验2的假发现率比较。

从表3和表4可以看出，在这种情况下，先前版本的算法无法高效地进行因果发现，表现为真正率和假发现率较低。类似地，FCI和LINGAM的性能也大幅下降，同样表现不佳。最后可以看到，所提出的新版本在这些复杂图上取得了良好的指标。

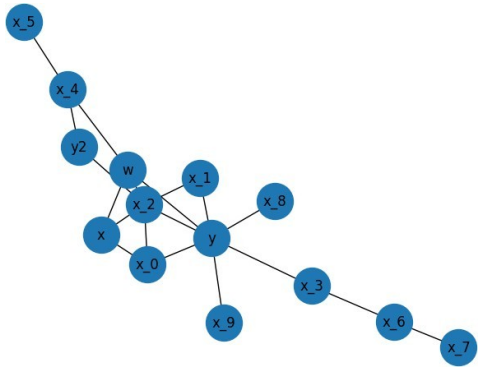

图3：ICPv2 在数据集5上发现的因果图。

4 结论

根据这些结果，我们可以认为，我们的方法适合在观测数据上高效生成环境。这为应用基于不变因果性质的算法打开了大门；这是本文的主要贡献之一，可用于在观测数据上进行因果推断、因果发现，或单纯改进机器学习预测（如IRM）。我们还看到，该方法可以与ICP算法成功结合，用于在观测数据上进行因果发现。最后，我们提出了ICP的新版本，可与我们的方法结合使用，适用于具有多个协变量和复杂因果关系的数据集。这个新版本表现出良好的真正率和假发现率，并已证明优于其他因果发现方法，因此可以成为一种因果发现的替代方法。与其他因果发现算法一样，我们的方法无法确定因果关系的方向，这是一个局限，也可以作为未来研究的目标。

参考文献

[1] Kartik Ahuja、Karthikeyan Shanmugam、Kush Varshney 和 Amit Dhurandhar。不变风险最小化博弈。发表于国际机器学习会议，第145–155页。PMLR，2020年。[2]Martin Arjovsky、Léon Bottou、Ishaan Gulrajani 和 David Lopez-Paz。不变风险最小化。*arXiv*预印本 *arXiv:1907.02893*，2019年。[3] Kenneth A Bollen。含潜变量的结构方程，第210卷。John Wiley & Sons，1989年。[4] Patricia W Cheng 和 Hongjing Lu。因果不变性是构建世界表征的一项必要约束：因果力量不变性的泛化。《牛津因果推理手册》，第65–84页，2017年。[5] Max Chickering。统计高效贪心等价搜索。发表于人工智能不确定性会议，第241–249页。PMLR，2020年。[6] Yunhao Ge、Sercan Ö Arik、Jinsung Yoon、Ao Xu、Laurent Itti 和 Tomas Pfister。通过不变结构学习实现更好的泛化与因果可解释性。*arXiv*预印本 *arXiv:2206.06469*，2022年。[7] AmirEmad Ghassami、Saber Salehkaleybar、Negar Kiyavash 和 Kun Zhang。利用回归不变性学习因果结构。神经信息处理系统进展，第30卷，2017年。[8] Jan Gleixner。Invariantcausalprediction Python库icpy。2020年。[9] Sander Greenland、Judea Pearl 和 James M Robins。流行病学研究中的因果图示。《流行病学》，第37–48页，1999年。[10] Christina Heinze-Deml 和 Nicolai Meinshausen。条件方差惩罚与领域偏移鲁棒性。*arXiv*预印本 *arXiv:1710.11469*，2017年。[11] Christina Heinze-Deml、Jonas Peters 和 Nicolai Meinshausen。非线性模型的不变因果预测。《因果推断杂志》，6(2)，2018年。[12] Patrik Hoyer、Dominik Janzing、Joris M Mooij、Jonas Peters 和 Bernhard Schölkopf。利用加性噪声模型进行非线性因果发现。神经信息处理系统进展，第21卷，2008年。[13] Niki Kilbertus、Giambattista Parascandolo 和 Bernhard Schölkopf。反因果学习中的泛化。

*arXiv*预印本 *arXiv:1812.00524*，2018年。Masanori Koyama 和 Shoichiro Yamaguchi。何时在分布外泛化问题中不变性才有用？*arXiv*预印本 *arXiv:2008.01883*，2020年。[14] David Krueger、Ethan Caballero、Joern-Henrik Jacobsen、Amy Zhang、Jonathan Binas、Dinghuai Zhang、Remi Le Priol 和 Aaron Courville。通过风险外推实现分布外泛化（REx）。发表于国际机器学习会议，第5815–5826页。PMLR，2021年。[15] Steffen L Lauritzen 和 David J Spiegelhalter。图结构上概率的局部计算及其在专家系统中的应用。英国皇家统计学会学报：*B*辑（方法论），50(2):157–194，1988年。[16] Scott M Lundberg 和 Su-In Lee。解释模型预测的统一方法。神经信息处理系统进展，第30卷，2017年。[17] Nicolai Meinshausen。从分布鲁棒性的角度看因果关系。发表于*2018*年 *IEEE* 数据科学研讨会（*DSW*），第6–10页。IEEE，2018年。[18] Judea Pearl。《因果关系》。剑桥大学出版社，2009年。[19] Jonas Peters、Peter Bühlmann 和 Nicolai Meinshausen。利用不变预测进行因果推断：识别与置信区间。英国皇家统计学会学报：*B*辑（统计方法论），第947–1012页，2016年。[20] Jonas Peters、Joris M Mooij、Dominik Janzing 和 Bernhard Schölkopf。利用连续加性噪声模型进行因果发现。2014年。[21] Marco Tulio Ribeiro、Sameer Singh 和 Carlos Guestrin。“我为什么应该相信你？”：解释任意分类器的预测。发表于第*22*届 *ACM SIGKDD* 国际知识发现与数据挖掘会议论文集，第1135–1144页，2016年。[22] Thomas S Richardson 和 James M Robins。单世界干预图（SWIG）：统一反事实方法与因果关系的图方法。统计与社会科学中心，华盛顿大学系列工作论文，128(30):2013，2013年。[23] James M Robins、Miguel Angel Hernan 和 Babette Brumback。边际结构模型与流行病学中的因果推断。《流行病学》，第550–560页，2000年。[24] Shohei Shimizu、Patrik O Hoyer、Aapo Hyvärinen、Antti Kerminen 和 Michael Jordan。用于因果发现的线性非高斯无环模型。机器学习研究杂志，7(10)，2006年。[25] Peter Spirtes。用于因果推断的随时算法。发表于国际人工智能与统计学研讨会，第278–285页。PMLR，2001年。[26] Peter Spirtes、Clark N Glymour、Richard Scheines 和 David Heckerman。《因果、预测与搜索》。麻省理工学院出版社，2000年。[27] Nanyang Ye、Jingxuan Tang、Huayu Deng、Xiao-Yun Zhou、Qianxiao Li、Zhenguo Li、Guang-Zhong Yang 和 Zhanxing Zhu。对抗不变学习。发表于*2021*年 *IEEE/CVF*计算机视觉与模式识别会议（*CVPR*），第12441–12449页。IEEE，2021年。

A 附录

A.1 数值结果的实验设置

数据集根据数值结果部分指定的标准生成。具体而言，五个数据集根据以下方程进行模拟：

数据集*1*

$X_1 = N(0.8, 1)$

$X_2 = N(0.5, 1)$

$Y = 0.7X_1 + 0.4X_2 + N(0, 1)$

$X_3 = 0.5Y + N(0, 1)$

数据集*2*

$X_1 =$ 二项分布$(6, 0.5)$

$X_2 =$ 二项分布$(0.5, 1)$

$Y = 0.7X_1 + 0.4X_2 + N(0, 1)$

$X_3 = 0.5Y + N(0, 1)$

数据集*3*

$X_1 = N(0.2, 0.5)$

$W = N(0.5, 1)$

$X_2 = 0.6W + 0.4X_1 + N(0, 1)$

$Y = 0.5X_2 + 0.5W + N(0, 1)$

$Y_2 = 0.7X_2 + N(0.2, 1)$

$X_3 = 0.3Y + N(0, 1)$

数据集*4*

$X = N(0.2, 0.8)$

$X_1 = N(0, 1.1)$

$X_0 = 0.5X + N(0, 0.5)$

$W = X + N(0, 1)$

$X_2 = 0.5X_1 + 0.5W + 0.5X_0$

$Y = 0.5X_2 + 0.5W + N(0, 1)$

$Y_2 = 0.7X_2 + N(0.2, 1)$

$X_3 = 0.5Y + N(0, 1)$

$X_4 = W + N(0, 1)$

$X_5 = 0.8X_4 + N(0, 1)$

数据集*5*

$X = N(0.2, 0.8)$

$X_1 = N(0, 1.1)$

$X_9 = N(0.4, 0.75)$

$X_0 = 0.5X + N(0, 0.5)$

$W = X + N(0, 1)$

$X_2 = 0.5X_1 + 0.5W + 0.5X_0$

$Y = 0.5X_2 + 0.5W + 0.5X_9 + N(0, 1)$

$Y_2 = 0.7X_2 + N(0.2, 1)$

$X_3 = 0.5Y + N(0, 1)$

$X_4 = W + N(0, 1)$

$X_5 = 0.8X_4 + N(0, 1)$

$X_6 = X_3 + N(0, 1)$

$X_7 = 0.1X_6 + N(0.2, 0.5)$