Causal Imitation Learning under Temporally Correlated Noise

Gokul Swamy [1] **Sanjiban Choudhury** [2] **J. Andrew Bagnell** [3 1] **Zhiwei Steven Wu** [1]

Abstract

We develop algorithms for imitation learning from policy data that was corrupted by temporally correlated noise in expert actions. When noise affects multiple timesteps of recorded data, it can manifest as spurious correlations between states and actions that a learner might latch on to, leading to poor policy performance. To break up these spurious correlations, we apply modern variants of the *instrumental variable regression* (IVR) technique of econometrics, enabling us to recover the underlying policy *without* requiring access to an interactive expert. In particular, we present two techniques, one of a generative-modeling flavor (`DoubIL`) that can utilize access to a simulator, and one of a game-theoretic flavor (`ResiduIL`) that can be run entirely offline. We find both of our algorithms compare favorably to behavioral cloning on simulated control tasks.

1. Introduction

Much of the theory of imitation learning (IL) tells us that with enough demonstrations, we should be able to accurately recover the expert's policy. A long line of work (Ross et al., 2011; Sun et al., 2019; Spencer et al., 2021; Swamy et al., 2021) has derived performance bounds that seem to imply that if infinite-sample training error is driven to zero, value equivalence to the expert policy should follow. However, when we actually apply IL algorithms on large datasets, we sometimes see them produce manifestly incorrect estimates of the expert's policy (Muller et al., 2006; Codevilla et al., 2019; de Haan et al., 2019; Bansal et al., 2018; Kuefler et al., 2017). One possible reason for this phenomenon is that empirically, we might only have access to recordings of the expert that are corrupted by *temporally correlated noise* (TCN). For example, a quadcopter pilot might have been flying under persistent wind or an expert driver might have been using a car with sticky brakes. More generally, we might expect that for a variety of sequential prediction tasks, observational data might have noise that is not independently distributed across timesteps.

The downstream effect of TCN (more formally, an unobserved confounder) is temporal correlations in the recorded actions that do not have their true cause in the recorded state. Consider again our quadcopter pilot demonstrating how to fly straight on a rather windy day. If we directly fed these swerve-filled trajectories to the learner, they might learn to reproduce the deviations, producing trajectories that deviate *even further* from a straight path in a test-time windy environment. At a more abstract level, these sorts of inconsistent policy estimates can result from temporal correlations between pairs of actions (e.g. the persistent wind affecting the observed heading) being reflected in the state (e.g. the quadcopter position) leading to spurious correlations between state and action that the learner might unfortunately latch onto (e.g. turning further left when on the left).

What then should we hope to learn in these confounded settings? Given we do not have access to the unobserved confounder, a reasonable choice is to ensure that we match the behavior of an expert with access to the same information we have. Put differently, we should strive to produce actions matching those proposed by an expert *queried* about our current observation. While applying an interactive imitation learning algorithm like DAgger (Ross et al., 2011) would allow us to collect a dataset uncorrupted by confounding (as we directly observe deconfounded expert actions), a queryable expert is not a realistic assumption for many domains. We therefore focus on approaches that operate on the basis of a fixed set of demonstrations. We base our algorithms on *instrumental variable regression* (IVR) (Angrist et al., 1996), a technique from econometrics for dealing with confounding in recorded data. The high-level idea of IVR is to leverage an *instrument*, a source of random variation independent of the confounder, to deconfound inputs to a learning procedure via conditioning on the instrument. In dynamical systems, history can act as this source of variation, as it is unaffected by future confounding (Hefny et al., 2015). Our key insight is that ***we can leverage past states as instruments to break the spurious correlation between states and actions caused by an unobserved confounder***.

[1] Carnegie Mellon University [2] Cornell University [3] Aurora Innovation. Correspondence to: Gokul Swamy <gswamy@cmu.edu>.

Proceedings of the 39th International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

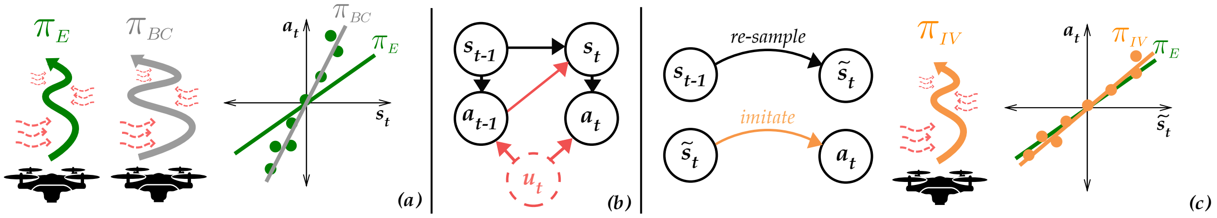

Figure 1. (a) When temporally correlated noise (e.g. wind) affects expert actions, standard imitation learning approaches like behavioral cloning can amplify this noise, leading to poor test-time performance. *(b)* TCN u_t affects both the input (s_t) and output (a_t) of our learning procedure. This breaks a cardinal assumption of regression-based approaches like behavioral cloning, rendering them inconsistent. *(c)* We can re-simulate state transitions from a past state, producing fresh samples ($\tilde{s}_t$). We can then regress from these sampled states to observed expert actions to recover the expert's policy as the noise on inputs and outputs is no longer correlated.

The contributions of our work are three-fold:

1. We formalize confounding in imitation learning. We construct a structural causal model that captures confounding from temporally correlated noise. We derive a test to detect whether TCN is present in a dataset.

2. We present a unified derivation of modern instrumental variable regression techniques. We show how two recent extensions of the classical IVR technique share a common structure. We extend the theoretical analysis of these previous works by deriving accuracy bounds.

3. We provide two novel algorithms to deal with confounding in imitation learning. We build upon modern IVR to derive two algorithms that are consistent under TCN:

- `DoubIL` is a generative modeling approach that can use a simulator for reduced sample complexity.
- `ResiduIL` is a game-theoretic and simulator-free approach.

We derive performance bounds for policies produced by these algorithms under TCN. We then validate their performance on simulated control tasks. We also empirically investigate how the persistence of the confounder impacts policy performance.

2. Related Work

Imitation Learning. Broadly speaking, imitation learning approaches can be grouped into three classes: offline, online, and interactive. Our work is most similar to offline imitation learning algorithms (e.g. Behavioral Cloning (Pomerleau, 1989)) that operate purely on collected data. Unlike previous work however, we consider the effect of unobserved confounding. Our work shares the goal of interactive imitation learning algorithms (e.g. DAgger (Ross et al., 2011), AggreVaTe (Ross & Bagnell, 2014)), in that we seek to match what the expert would do at a particular state, rather than what is in the corrupted demonstration. Importantly, we focus on matching expert actions on *expert* rollouts, rather than on *learner* rollouts, as one usually does in interactive IL. Because of the unobserved confounders, the recorded expert actions and the output of an expert query would not match. Because we are only focusing on expert rollouts, we do not need an interactive expert.

Zhang et al. (2020); Kumor et al. (2021) consider imitation learning through the lens of causal inference and derive a structural condition on the inputs to the learner's policy for recovering the expert's policy. Because we only consider additive TCN, we are still able to identify causal effects without satisfying this condition – see Sec. 5.1 of (Pearl, 1995) for more discussion of this point. While (Zhang et al., 2020; Kumor et al., 2021) give general learnability conditions, we derive efficient algorithms with performance guarantees for a specific subclass of feasible problems that are of practical interest.

Lastly, we note that we focus only on matching the actions of a deterministic expert in this work – we leave matching arbitrary expert moments (Swamy et al., 2021) to future work.

Inertia Effects in Imitation Learning. Several authors have empirically observed a latching behavior in policies trained via imitation learning, where learned policies tend to inappropriately repeat the same action (Muller et al., 2006; Codevilla et al., 2019; de Haan et al., 2019; Bansal et al., 2018; Kuefler et al., 2017). We seek to provide a plausible explanation and correction for the phenomenon reported in these works. We note that when attempting to explain these sorts of *inertia effects*, de Haan et al. (2019) propose causal confounding as the root cause of the learner's error. However, as pointed out by Spencer et al. (2021), there is no actual confound in the theoretical or empirical examples of the work of de Haan et al. (2019), merely a high degree of covariate shift. This is because the learner observes all of the variables that influenced the expert action. We instead consider the setting with unobserved TCN.

SETTING	STATE	EXPERT ACTION	OBSERVED ACTION	CONFOUNDER
Quadcopter Flying	Position	Intended Heading	Actual Heading	Persistent Wind
Product Pricing	Demand	Profit Margin	Price	Raw Materials Cost
ICU Treatment	Symptoms (e.g. heartburn)	Intent to Treat	Patient Treated	Comorbidity (e.g. fever)
Shared Autonomy	User State	Intended Action	Executed Action	Assistance

Table 1. Several examples of TCN that can lead to inconsistent estimates of the expert's policy. The first was noted empirically by Ng et al. (2003) and examples related to the next two rows have been observed by Wright (1928); Desautels et al. (2017); Soo et al. (2019).

Instrumental Variable Regression. The classical approach to instrumental variable regression (Wright, 1928) is a two-stage least squares procedure (e.g. in Angrist et al. (1996)'s textbook). We focus on the more general nonlinear setting and instead base our approaches on the more recent DEEPIV (Hartford et al., 2017) and AGMM (Dikkala et al., 2020). We present extensions to the work in these papers, including a unified derivation of both methods and error analysis for DEEPIV. Prior work (Bradtke & Barto, 1996; Hefny et al., 2015; Chen et al., 2021) has considered using past states as an instrument for reinforcement learning. We instead focus on imitation learning and derive algorithms with policy performance bounds that factor in the strength of the past state instrument.

3. A Brief Review of Instruments in Causal Modeling

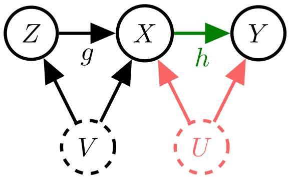

Figure 2. The structural causal model (SCM) considered in IVR. We are interested in finding h, the causal relationship from X to Y, even though there is an unobserved confounder, U. We do so by leveraging the effect of Z, which provides randomness independent of U.

We begin by discussing the concept of an instrument. Let X, Y, and Z be random variables on (potentially infinite) sample spaces $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$. Assume that X, Y, and Z have the causal, rather than statistical, dependency structure in Fig. 2. Given a dataset of (x, y, z) tuples, we are interested in determining the causal relationship between X and Y, $\mathbb{E}[Y|do(x)]$, where $do(\cdot)$ is the interventional operator of Pearl et al. (2016). Intuitively, $\mathbb{E}[Y|do(x)]$ is the expected value of Y when we *intervene* and set $X = x$, rather than observe such an X. In Fig. 2, $h(x) = \mathbb{E}[Y|do(x)]$. Because of the presence of an unobserved confounder, U, that affects both X and Y, standard regression (e.g. Ordinary Least Squares or OLS) generically produces inconsistent estimates. Coarsely, this occurs because OLS will overestimate the influence of the parts of X that are affected by the confounder. If we only have observational data or are unable to perform randomized control trials, a canonical technique to recover h is IVR (Wright, 1928; Angrist et al., 1996; Winship & Morgan, 1999). Formally, an *instrument* Z must satisfy three structural conditions:

1. *Unconfounded Instrument*: $Z \perp\!\!\!\perp U$ – i.e. independent randomization from confounder.
2. *Exclusion*: $Z \perp\!\!\!\perp Y|X, U$ – i.e. no extraneous paths.
3. *Relevance*: $Z \not\perp\!\!\!\perp X$ – i.e. conditioning has an effect.

Z satisfies these three conditions in the SCM of Fig. 2. [1] Without loss of generality, we assume that $\mathbb{E}[U] = 0$. We further assume that noise U enters additively to Y,[2] and write out the following equations:

$$X = g(Z, U, V), \quad Y = h(X) + U. \tag{1}$$

We can now concisely derive a set of *conditional moment restrictions* (CMR):

$$0 = \mathbb{E}[U] = \mathbb{E}[U|z] = \mathbb{E}[Y - h(X)|z] \tag{2}$$
$$\Rightarrow \forall z \in \mathcal{Z},\ \mathbb{E}[Y|z] = \mathbb{E}[h(X)|z]. \tag{3}$$

In words, these constraints are saying that a necessary condition for recovery of $h(x)$ is that for all values of Z, the actual and predicted expected values of $Y|z$ are equal.

How can we find a predictor that satisfies the CMR? Let us first consider the setting with linear relationships between all variables. Then, one can recover $h(x) = \beta x$ by computing $\beta = \mathbb{E}[ZY]/\mathbb{E}[ZX]$. This is equivalent to the Two-Stage Ordinary Least Squares (2SLS) procedure (Angrist et al., 1996), in which one first regresses from Z to X and then regresses from the predicted $\hat{X}$ to Y, returning the latter coefficients. Intuitively, the first stage of this procedure is

[1] The inclusion of V makes our model a generalization of the standard IVR model, so we confirm the validity of the instrument in Appendix A.

[2] Without this assumption, one can only upper/lower bound $h(x)$ (Balke & Pearl, 2013).

aggregating Xs based on some $z \in \mathcal{Z}$ so that the particular instantiation of U that was correlating X and Y in the observational data has its effect "washed out" in the $\hat{X}$s. Thus, regression from $\hat{X}$ to Y is consistent.

For the more general, nonlinear problem, we can derive an appropriate loss function for finding an $\widehat{h}$ that approximately satisfies the CMR. If we have finite samples and can therefore only estimate conditional expectations up to some tolerance, it is natural to relax the CMR to

$$\begin{array}{ll} \min_{\widehat{h} \in \mathcal{H}, \boldsymbol{\delta}} & \frac{1}{2}\mathbb{E}_z[\delta_z^2] \\ \text{s.t.} & |\,\mathbb{E}[Y - \widehat{h}(X)|z]| \leq \delta_z, \quad \delta_z \geq 0, \quad \forall z \in \mathcal{Z}, \end{array} \tag{4}$$

where the δ_z are slack variables. Then, the Lagrangian (with the natural $P(z)$-weighted inner product that captures how often each we expect each z to occur) is

$$L(\widehat{h}, \boldsymbol{\delta}, \boldsymbol{\lambda}) = \sum_{z \in \mathcal{Z}} P(z)\lambda_z(\mathbb{E}[Y - \widehat{h}(X)|z] - \delta_z) + P(z)\frac{1}{2}\delta_z^2, \tag{5}$$

where $\boldsymbol{\lambda}$ is the vector of Lagrange multipliers. By the stationarity component of the KKT conditions, we know that

$$\nabla_{\delta_z} L(\widehat{h}, \boldsymbol{\delta}, \boldsymbol{\lambda}) = -P(z)\lambda_z + P(z)\delta_z = 0, \tag{6}$$

implying that $\delta_z = \lambda_z$. Plugging this back in, we can simplify the Lagrangian to

$$L(\widehat{h}, \boldsymbol{\lambda}) = \sum_{z \in \mathcal{Z}} P(z)\lambda_z \mathbb{E}[Y - \widehat{h}(X)|z] - P(z)\frac{1}{2}\lambda_z^2. \tag{7}$$

We refer to (7) as the *Regularized Lagrangian*. Now, solving for the optimal Lagrange multipliers via stationarity, we arrive at the expression

$$\nabla_{\lambda_z} L(\widehat{h}, \boldsymbol{\lambda}) = P(z)\mathbb{E}[Y - \widehat{h}(X)|z] - P(z)\lambda_z = 0, \tag{8}$$

which implies that the optimal λ_z is equal to $\mathbb{E}[Y - \widehat{h}(X)|z]$. Plugging this back into (7) produces the loss function

$$L(\widehat{h}) = \frac{1}{2}\sum_{z \in \mathcal{Z}} P(z)\mathbb{E}[Y - \widehat{h}(X)|z]^2 = \text{PRMSE}^2(\widehat{h}). \tag{9}$$

This expression is the square of the *Projected Root Mean Squared Error* (PRMSE) of Chen & Pouzo (2012). To recap, by minimizing (9), we are attempting to find an $\widehat{h}$ that approximately satisfies the CMR. Minimizing PRMSE is a necessary condition for recovering $\mathbb{E}[Y|do(X)]$. For it to be a sufficient condition, one needs the natural identifiability assumptions – we refer interested readers to Chen & Pouzo (2012) for a more thorough discussion.

3.1. Generative Modeling Approach

How should we minimize the PRMSE then? One option is learning the distribution $P(X|z) = g(z)$, passing samples from it to a candidate $\widehat{h}$, and trying to match $\mathbb{E}[Y|z]$. This is a generalization of the 2SLS procedure to nonlinear functions. The nonlinearity of the second stage means that one cannot simply compute the first moment of the $P(X|z)$ distribution, which is recovered by linearly regressing from X to Z in the 2SLS procedure. One instead needs to learn the entire $g(z) = P(X|z)$. Such an approach was first proposed by Hartford et al. (2017), and amounts to first learning a $g(z)$ (e.g. via maximum likelihood estimation) and then solving

$$\min_{\widehat{h} \in \mathcal{H}} \mathbb{E}_Z\left[(\mathbb{E}[Y|z] - \mathbb{E}_{\hat{x} \sim g(z)}[\widehat{h}(\hat{x})])^2\right]. \tag{10}$$

We note that this approach suffers from a "double-sample" issue (Baird, 1995) where multiple independent samples of $g(z)$ are required to compute gradients of $\widehat{h}$. To see this, note that the gradient with respect to $\widehat{h}$ of (10) is

$$\mathbb{E}_Z\left[(\mathbb{E}[Y|z] - \mathbb{E}_{\hat{x} \sim \hat{g}(z)}[\widehat{h}(\hat{x})])(-\mathbb{E}_{\hat{x} \sim \hat{g}(z)}[\frac{\partial}{\partial \widehat{h}}\widehat{h}(\hat{x})])\right]. \tag{11}$$

Notice that $\hat{x}$ appears under two *separate* expectations that are then multiplied together. Therefore, to get an unbiased estimate of this product, a minimum of two samples of $\hat{x}$ are required, one for each expectation.

The work of Hartford et al. (2017) did not have theoretical analysis regarding the effect of errors in a learned $g(z)$ upon attempts to learn $h(x)$. We prove the following error bound in Appendix A:

Theorem 3.1. *Assume we learn a $g(z)$ s.t.*

$$\max_{\widehat{h} \in \mathcal{H}} \mathbb{E}_Z[(\mathbb{E}_{x \sim g(z)}[\widehat{h}(x)] - \mathbb{E}_{x \sim P(X|z)}[\widehat{h}(x)])^2] \leq \delta. \tag{12}$$

Then, optimizing (10) to value ϵ corresponds to recovering a $\widehat{h}(x)$ s.t. PRMSE$(\widehat{h}) \leq \sqrt{\delta} + \sqrt{\epsilon}$.

3.2. Game-Theoretic Approach

One can also proceed by instead solving the two-player zero-sum game with the Regularized Lagrangian (7) as the payoff function. Denoting by $f \in \mathcal{F} = \{\mathcal{Z} \to \mathbb{R}\}$ the function that maps z's to their Lagrange multipliers, we can write this game as

$$\min_{\widehat{h} \in \mathcal{H}} \max_{f \in \mathcal{F}} \mathbb{E}[2(Y - \widehat{h}(X))f(Z) - f(Z)^2]. \tag{13}$$

This game is the core objective of the AGMM method of Dikkala et al. (2020). Importantly, one does not need to learn a generative model of $P(X|z)$ for these sorts of game-theoretic approaches. We prove the following theorem in Appendix A:

Theorem 3.2. *Assume that $\mathcal{H}$ and $\mathcal{F}$ are bounded, closed under negation, convex, compact, $h \in \mathcal{H}$, and that $\forall \widehat{h} \in \mathcal{H}$, $f(z) = \mathbb{E}[Y - \widehat{h}(X)|z] \in \mathcal{F}$. Then, an ϵ-approximate Nash equilibrium of (13) corresponds to recovering an $\widehat{h}(x)$ s.t. PRMSE$(\widehat{h}) \leq \sqrt{\epsilon}$.*

One can find such an equilibrium via a standard reduction to no-regret online learning (Freund & Schapire, 1997).

In summary, one can frame nonlinear IVR as a generative modeling or game-theoretic problem, leading to different error characteristics. We now turn our attention to applying these methods to imitation learning with unobserved confounders.

4. Causal Confounding in Imitation Learning

Let us introduce some IL-specific notation. We use $\Delta(S)$ to mean the set of distributions over S and focus on a Markov Decision Process (MDP) parameterized by $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, r, T \rangle$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow \Delta(\mathcal{S})$ is the transition operator, $r : \mathcal{S} \times \mathcal{A} \rightarrow [-1, 1]$ is the reward function, and T is the horizon of the problem. Let $J(\pi) = \mathbb{E}_{\tau \sim \pi}[\sum_{t=1}^{T} r(s_t, a_t)]$ denote the *value* of policy π, $\Pi \subseteq \{\mathcal{S} \rightarrow \Delta(\mathcal{A})\}$ be the policy class we optimize over and d_π be the visitation distribution of policy π. In the presence of unobserved TCN, the trajectories generated by the expert can be captured by the structural causal model (SCM) in Fig. 3.

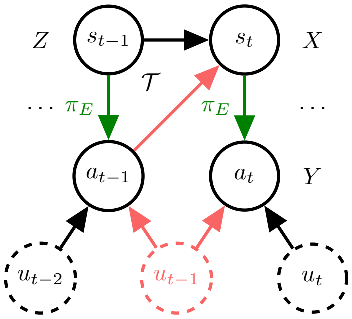

Figure 3. An SCM that captures TCN. The confounding ($U = u_{t-1}$) is mediated via the dynamics into the state, introducing spurious correlations between states ($X = s_t$) and actions ($Y = a_t$). To break the confounding, we can utilize the past state as an instrument ($Z = s_{t-1}$).

We use u_{t-1} to denote the confounder at timestep t. See Table 1 for several examples. The confounder perturbs the past action and travels through the dynamics to influence the current state. The same confounder also perturbs the current action, leading to spurious correlations between the recorded state and action. This correlative effect is also visible in the structural equations corresponding to Fig. 3,

$$\begin{aligned} X &= s_t && (14)\\ &= \mathcal{T}(s_{t-1}, a_{t-1}) && (15)\\ &= \mathcal{T}(s_{t-1}, \pi_E(s_{t-1}) + u_{t-1} + u_{t-2}) && (16)\\ Y &= a_t = \pi_E(s_t) + u_t + u_{t-1}. && (17)\end{aligned}$$

Note the shared red term between input X and output Y. Fig. 3 also tells us that $Z = s_{t-1}$ satisfies the three conditions to make it a valid instrument for countering the effects of $U = u_{t-1}$. Intuitively, the past state is independent of the current confounder, allowing it to function as an independent source of randomness. One can imagine longer time-scale correlations between actions than just the one-step connection in Fig. 3 – our approaches naturally extend to this setting by using a state further back in the past as the instrument. However, this also means that the past state instrument is less predictive of the current state – we discuss the implications of this point further in Sec. 6.

5. What would the Expert $do(\cdot)$?: Algorithms for Causal Imitation Learning

Essentially, standard imitation learning approaches like behavior cloning attempt to match $\mathbb{E}[a|s]$, the average observed action in the data at state s. An approach based on IVR instead attempts to match the *interventional* effect of the expert policy, $\mathbb{E}[\pi_E(s)|s] = \mathbb{E}[a|do(s)]$. Conceptually, $\mathbb{E}[a|do(s)]$ is telling us what the expert would do on average if we intervened and *placed* them in state s. Because of the unobserved TCN, $\mathbb{E}[a|do(s)]$ differs from $\mathbb{E}[a|s]$.

We note that $\mathbb{E}[a|do(s)]$ is the answer we would get by averaging responses from a queryable expert in interactive approaches like DAgger (Ross et al., 2011). However, as we are only interested in the result of queries on states from expert demonstrations, we are able to compute $\mathbb{E}[a|do(s)]$ via IVR and do not require access to a queryable expert.

We now present two approaches for causal imitation learning that can be seen as applications of the generative modeling and game-theoretic approaches of Sec. 3. At their core, both algorithms are attempting to minimize a PRMSE objective,

$$\min_{\pi \in \Pi} \mathbb{E}_{(s,s',a') \sim d_{\pi_E}}[(\mathbb{E}[a' - \pi(s')|s])^2], \tag{18}$$

instead of the usual offline IL objective,

$$\min_{\pi \in \Pi} \mathbb{E}_{(s,a) \sim d_{\pi_E}}[(a - \pi(s))^2]. \tag{19}$$

Matching symbols with Sec. 3 tells us that minimizing (18) corresponds to recovering $\mathbb{E}[a|do(s)]$. We now discuss the performance implications of approximately doing so.

5.1. From PRMSE to Performance

For deriving performance bounds, we assume the same distribution of TCN affects the learner at test time.[3] Our goal in this setting is therefore to eliminate the effect of the

[3] Under a different noise distribution (or no noise at all), we might do better or worse than the demonstrator. For example, on a less windy day, we are likely to do better than the quadcopter pilot at flying straight. If we make an additional *overlap* assumption that we see data at all parts of the state space the learner reaches under

confounder so at test time we do not needlessly reproduce its effects (e.g. the increased swerving in our quadcopter example). This is why we focus on minimizing (18) instead of (19). We emphasize that under TCN, minimizing the (19) to 0 would not recover the expert's policy.

We now define two key concepts. First, let a confounder distribution $P(U)$ be c-Total Variation stable (Bassily et al., 2021) if

$$\|a-b\|_2 \le \delta \Rightarrow d_{TV}(a+U, b+U) \le c\delta. \tag{20}$$

This property is satisfied by a wide variety of distributions (e.g. for standard normal random variables, $c = 1/2$). Second, in the IL setting, the measure of ill-posedness of a problem (Dikkala et al., 2020; Chen & Pouzo, 2012) is

$$\kappa(\Pi) = \sup_{\pi \in \Pi} \frac{\sqrt{\mathbb{E}_{s \sim d_{\pi_E}}[(\pi_E(s) - \pi(s))^2]}}{\sqrt{\mathbb{E}_{s,s',a' \sim d_{\pi_E}}[\mathbb{E}[a' - \pi(s')|s]]^2}} \tag{21}$$

$$= \sup_{\pi \in \Pi} \frac{\text{RMSE}(\pi)}{\text{PRMSE}(\pi)}. \tag{22}$$

We leverage these two definitions in the following bound on policy performance.

Theorem 5.1. *Assume $P(U)$ is c-TV Stable temporally correlated noise, π_E is deterministic, and let $\kappa(\Pi)$ be the measure of the ill-posedness of the problem. Then, PRMSE$(\pi) \le \epsilon \Rightarrow J(\pi_E) - J(\pi) \le c\kappa(\Pi)\epsilon T^2$.*

We prove this statement in Appendix A. Intuitively, $\kappa(\Pi)$ measures the strength of the past state as an instrument. To build intuition, first consider the extreme case where $s' = s$. Then, $\kappa(\Pi) = 1$. As the past state becomes a weaker instrument, $\kappa(\Pi) > 1$. Thus, if the confounding affects multiple timesteps, we would expect $\kappa(\Pi)$ to grow as one needs to reach further back in time to find a valid instrument, leading to a looser performance bound. We investigate the effect of the length of confounding on the ill-posedness of the problem empirically in Sec. 6.

5.2. With a Simulator: DoubIL

```
Algorithm 1 DoubIL
  Input: Dataset $\mathcal{D}_E$ of expert trajectories, Policy class $\Pi$,
  Simulator $\widehat{\mathcal{T}}$
  Output: Trained policy $\pi_2$
  $\pi_1 = \arg\min_{\pi \in \Pi} \mathbb{E}_{s,a \sim \mathcal{D}_E}[-\log \pi(a|s)]$
  $\mathcal{D}_{IV} = \{(\widetilde{s'} \sim \widehat{\mathcal{T}}(s, \pi_1(s)), a') | \forall (s, a') \in \mathcal{D}_E\}$
  $\pi_2 = \arg\min_{\pi \in \Pi} \mathbb{E}_{s,a \sim \mathcal{D}_{IV}}[(a - \pi(s))^2]$
```

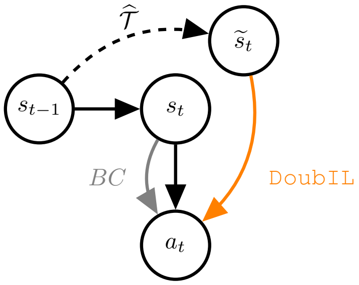

Figure 4. DoubIL deconfounds inputs to the second stage regression by re-sampling state transitions via simulator $\widehat{\mathcal{T}}$.

Algorithm 1 can be seen as a variation of generative modeling approach of Sec. 3 and Hartford et al. (2017) where one leverages knowledge of one factor of the $P(X|z)$ distribution and just learns the other factor. Via the Markov assumption, we can factorize $P(X|z) = P(S'|s) = \sum_{a \in \mathcal{A}} P(a|s)\mathcal{T}(s,a)$. Assuming access to a simulator $\widehat{\mathcal{T}}$ that closely approximates the true transition dynamics, we can focus on learning just the $P(a|s)$ component: the standard imitation learning task. Notably, this first-stage policy is biased as it includes the effect of the confounder: $P(a|s) = P(U + \pi_E(s)|s)$. However, when we use it to simulate transitions, the next states that are drawn no longer have the particular instantiation of the confounder present in the recorded dataset's next actions. Using a tilde to denote a fresh draw from a distribution, simulated states are drawn from

$$\widetilde{s}_t \sim \widehat{\mathcal{T}}(s_{t-1}, \pi_1(s_{t-1})) \tag{23}$$

while the observed next actions are drawn from

$$a_t \sim \pi_E(\mathcal{T}(s_{t-1}, \pi_E(s_{t-1}) + u_{t-1} + u_{t-2})) + u_{t-1} + u_t. \tag{24}$$

Notice that there are no shared noise terms. This allows us to apply standard imitation learning to this new dataset of $(\widetilde{s}_t, a_t)$ to learn a causally consistent policy. This is because $\mathbb{E}[a_t|\widetilde{s}_t] = \mathbb{E}[a_t|do(s_t)]$. The two applications of imitation learning lead us to term this algorithm DoubIL. To derive a PRMSE bound, we can translate the guarantee of Theorem 3.1 to our factored context:

Lemma 5.2. *Assume we learn a $\pi_1(s)$ s.t.*

$$\max_{\pi \in \Pi} \mathbb{E}_{s_{t-1}}[(\mathbb{E}_{s_t \sim \widehat{\mathcal{T}}(s_{t-1}, \pi_1(s_{t-1}))}[\pi(s_t)] \tag{25}$$

$$- \mathbb{E}_{s_t \sim P(s_t|s_{t-1})}[\pi(s_t)])^2] \le \delta. \tag{26}$$

Then, optimizing the second-stage MSE to ϵ corresponds to recovering a π_2 s.t.

$$\textit{PRMSE}(\pi_2) = \sqrt{\mathbb{E}_{s \sim d_{\pi_E}}[\mathbb{E}[\pi_2(s') - \pi_E(s')|s]^2]} \tag{27}$$

$$\le \sqrt{\delta} + \sqrt{\epsilon} \tag{28}$$

a different noise distribution, driving (18) to 0 would imply value equivalence to the expert that has its actions affected by this new TCN distribution. Thus, we learn a policy that is *value-equivariant* to the expert under a change of TCN.

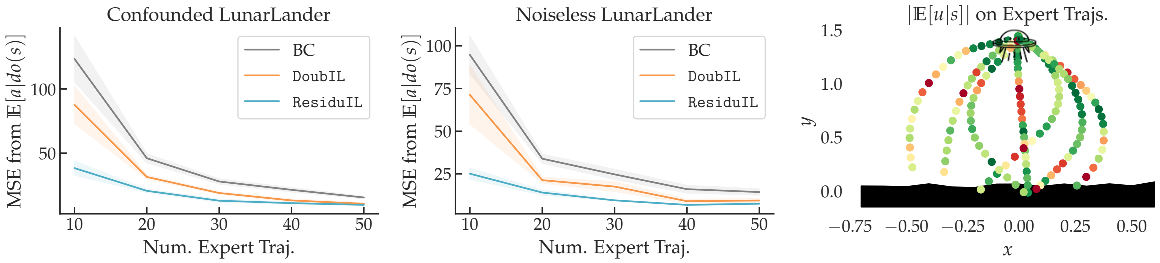

Figure 5. We train behavioral cloning, DoubIL, and ResiduIL on trajectories from a modified LunarLander environment, computing standard errors across four runs. **Left:** DoubIL and ResiduIL are better able to match $\pi_E(s) = \mathbb{E}[a|do(s)]$ on states from expert rollouts. **Center:** The policies learned by our algorithms generalize better than those produced by behavioral cloning to the state distribution of the expert on the noiseless problem ($u_t = 0$). **Right:** We can compare the results of behavioral cloning to one of our causal IL procedures to identify areas of the state space where the effect of confounding is strong (the red dots).

We prove this lemma in Appendix A. Combining this lemma with Theorem 5.1 allows one to derive a policy performance bound of

$$J(\pi_E) - J(\pi_{\texttt{DoubIL}}) \leq c\kappa(\Pi)(\sqrt{\delta} + \sqrt{\epsilon})T^2 \qquad (29)$$

under TCN. We note that one could simply learn the mapping $P(s'|s)$ but this can be far less sample efficient than merely learning a policy when $|\mathcal{A}| \leq |\mathcal{S}|$, as is often true in practice. [4]

5.3. Without state re-sampling: ResiduIL

```
Algorithm 2 ResiduIL
  Input: Dataset $\mathcal{D}_E$ of expert trajectories, Policy class $\Pi$, Discriminator class $\mathcal{F}$, Learning rate $\eta$
  Output: Trained policy $\pi$
  Set $\pi \in \Pi$, $f \in \mathcal{F}$, $\widetilde{g}_\pi = 0$, $\widetilde{g}_f = 0$
  while $\pi$ not satisfactory do
    $L(\pi, f) = \mathbb{E}_{(s,s',a') \sim \mathcal{D}_E}[2(a' - \pi(s'))f(s) - f(s)^2]$
    $g_\pi = \nabla_\pi L(\pi, f)$, $g_f = \nabla_f L(\pi, f)$
    $\pi \leftarrow \pi - \eta(2g_\pi - \widetilde{g}_\pi)$
    $f \leftarrow f + \eta(2g_f - \widetilde{g}_f)$
    $\widetilde{g}_\pi \leftarrow g_\pi$, $\widetilde{g}_f \leftarrow g_f$
  end while
```

Algorithm 2 is the direct application of the game-theoretic approach of Sec. 3 and Dikkala et al. (2020) to imitation learning. We term it ResiduIL because the adversary attempts to predict the residual between the learner and the expert's actions while the learner attempts to minimize this residual. This algorithm can be run completely offline (i.e. without access to a simulator). We use the Optimistic Mirror Descent approach of Syrgkanis et al. (2015) to find approximate Nash equilibria in our experiments. Once again, we can extend our past results to the IL setting.

Lemma 5.3. *An ϵ-approximate equilibrium for the policy player corresponds to recovering a policy π s.t PRMSE$(\pi) \leq \sqrt{\epsilon}$.*

This lemma dovetails with Theorem 5.1 to prove that

$$J(\pi_E) - J(\pi_{\texttt{ResiduIL}}) \leq c\kappa(\Pi)\sqrt{\epsilon}T^2 \qquad (30)$$

under TCN (Appendix A). We now turn our attention to validating these guarantees empirically.

6. Experiments

We test DoubIL and ResiduIL on a slightly modified version of the OpenAI Gym (Brockman et al., 2016) LunarLander-v2 environment against a behavioral cloning baseline. We generate demonstrations by simulating rollouts of an expert policy trained via PPO (Schulman et al., 2017), adding fresh Gaussian noise to the expert's action as well as cached noise from the last timestep. The latter noise is the confounder. We refer interested readers to Appendix B for full details, including hyperparameters. We see that both of our methods are able to more closely match $\mathbb{E}[a|do(s)]$ than behavioral cloning, especially in the low-data regime (Fig. 5, left). We also measure the MSE on states from deconfounded expert rollouts – while there are no simple guarantees on this state distribution, we see that our methods generalize better than BC empirically (Fig. 5, middle).

At this point, one might wonder how, given a dataset of expert demonstrations, one detects whether there is unobserved confounding in the data. We can answer this question by comparing the results of behavioral cloning and either

[4] A natural question at this point might be whether a standard on-policy moment-matching algorithm (Swamy et al., 2021) that uses repeated calls to a simulator to perform rollouts would also be able to learn a value-equivalent policy from TCN-corrupted demonstrations. Notice that our simulator does not add in TCN when simulating an action. Thus, if π_E were rolled out in this simulator, one would not expect it to produce trajectories similar to the demonstrations. This means that the true expert policy is not an equilibrium strategy of the moment-matching game, so one shouldn't expect the learned policy to perform at the expert's level.

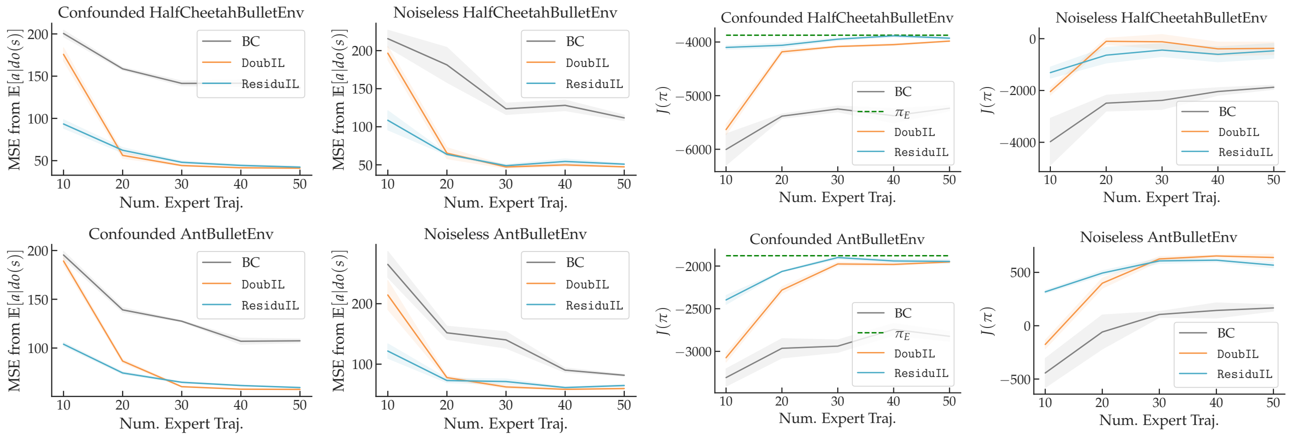

Figure 6. We train behavioral cloning, `ResiduIL`, and `DoubIL` on trajectories from the HalfCheetahBulletEnv and AntBulletEnv environments, computing standard errors across four runs. We see both of our approaches out-perform behavioral cloning on all metrics.

of our above algorithms. We prove the follow result in Appendix A:

Lemma 6.1. *Assume* $\pi_{BC}(s) = \mathbb{E}[a|s]$ *and* $\pi_{IV}(s) = \mathbb{E}[a|do(s)]$. *Then,* $\mathbb{E}[u|s] = \pi_{BC}(s) - \pi_{IV}(s)$.

The implication of this lemma is that comparing the outputs of IVR-based procedures to behavioral cloning can help us detect causal confounding – if they greatly differ with a sufficiently sized dataset, there is likely temporally correlated noise in our data. Moreover, the states where they differ represent the parts of the state space where the influence of the confounder is highest. Fig. 5, right, is an empirical example of how the test of Lemma 6.1 can be used to identify areas of the state space where the effect of the confounder is especially strong (e.g. the center).

We next consider the HalfCheetahBulletEnv and AntBulletEnv environments (Coumans & Bai, 2016–2019). Similar to the previous set of experiments, we train an expert via SAC (Haarnoja et al., 2018) and use Gaussian noise as the confounder – see Appendix B for more details. In Fig. 6, We see `ResiduIL` and `DoubIL` significantly out-perform behavioral cloning across all metrics and nearly match expert performance with enough data on both confounded MDPs. This further corroborates our theory, which argues that behavioral cloning will not be able to consistently estimate the expert's policy under TCN. In contrast, our methods are able to achieve value equivalence to the expert policy.

6.1. The Effect of TCN Persistence on Ill-Posedness

For linear problems, we can bound $\kappa(\Pi)$ (the measure of ill-posedness) via an eigenvalue ratio (Dikkala et al., 2020). Extending our previous model to include the effect of the last H confounders ($a_t = \pi_E(s_t) + \sum_{j=t-H}^{t} u_j.$), we arrive at the bound

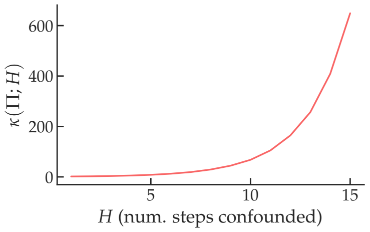

Figure 7. We compute $\kappa(\Pi)$ for an LQG problem where we vary the number of steps a confounder sticks around for.

$$\kappa(\Pi; H) \leq \sqrt{\frac{\lambda_{max}(\mathbb{E}[s_t s_t^T])}{\lambda_{min}(\mathbb{E}[\mathbb{E}[s_t|s_{t-H}]\mathbb{E}[s_t|s_{t-H}]^T])}}. \quad (31)$$

We compute this quantity empirically for a linear-quadratic problem with Gaussian confounding and plot results in Fig. 7. As expected, we see that increasing the length of confounding leads to weaker instruments as one has to use states further back in time. Theorem 5.1 therefore tells us that we should expect a larger performance gap between the learner and expert. We refer interested readers to Appendix B for full experimental setup details.

7. Conclusion

We present a model that captures confounding in imitation learning and derive two algorithms, `DoubIL` and `ResiduIL`, that are able to utilize history as an instrument to mitigate the effects of temporally correlated noise. We prove performance bounds and validate their empirical efficacy under TCN. We further consider how the persistence of TCN affects the performance of IVR-based imitation learning methods. We release our code at **`https://github.com/gkswamy98/causal_il`**.

8. Acknowledgements

GS thanks Daniel Kumor, Allie Del Giorno, Swaminathan Gurumuthy, Jonathan Spencer, and Keegan Harris for feedback on this work. ZSW is supported in part by the NSF FAI Award #1939606, a Google Faculty Research Award, a J.P. Morgan Faculty Award, a Facebook Research Award, an Okawa Foundation Research Grant, and a Mozilla Research Grant. GS is supported by his family and friends.

References

Angrist, J. D., Imbens, G. W., and Rubin, D. B. Identification of causal effects using instrumental variables. *Journal of the American statistical Association*, 91(434): 444–455, 1996.

Baird, L. Residual algorithms: Reinforcement learning with function approximation. In *Machine Learning Proceedings 1995*, pp. 30–37. Elsevier, 1995.

Balke, A. and Pearl, J. Counterfactual probabilities: Computational methods, bounds and applications, 2013.

Bansal, M., Krizhevsky, A., and Ogale, A. S. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *CoRR*, abs/1812.03079, 2018. URL http://arxiv.org/abs/1812.03079.

Bassily, R., Nissim, K., Smith, A., Steinke, T., Stemmer, U., and Ullman, J. Algorithmic stability for adaptive data analysis. *SIAM Journal on Computing*, (0):STOC16–377, 2021.

Bradtke, S. J. and Barto, A. G. Linear least-squares algorithms for temporal difference learning. *Machine learning*, 22(1):33–57, 1996.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. Openai gym, 2016.

Chen, X. and Pouzo, D. Estimation of nonparametric conditional moment models with possibly nonsmooth generalized residuals. *Econometrica*, 80(1):277–321, 2012.

Chen, Y., Xu, L., Gulcehre, C., Paine, T. L., Gretton, A., de Freitas, N., and Doucet, A. On instrumental variable regression for deep offline policy evaluation, 2021.

Codevilla, F., Santana, E., López, A. M., and Gaidon, A. Exploring the limitations of behavior cloning for autonomous driving. *CoRR*, abs/1904.08980, 2019. URL http://arxiv.org/abs/1904.08980.

Coumans, E. and Bai, Y. Pybullet, a python module for physics simulation for games, robotics and machine learning. http://pybullet.org, 2016–2019.

de Haan, P., Jayaraman, D., and Levine, S. Causal confusion in imitation learning. *Advances in Neural Information Processing Systems*, 32:11698–11709, 2019.

Desautels, T., Das, R., Calvert, J., Trivedi, M., Summers, C., Wales, D. J., and Ercole, A. Prediction of early unplanned intensive care unit readmission in a uk tertiary care hospital: a cross-sectional machine learning approach. *BMJ open*, 7(9):e017199, 2017.

Dikkala, N., Lewis, G., Mackey, L., and Syrgkanis, V. Minimax estimation of conditional moment models, 2020.

Freund, Y. and Schapire, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, 2018.

Hartford, J., Lewis, G., Leyton-Brown, K., and Taddy, M. Deep iv: A flexible approach for counterfactual prediction. In *International Conference on Machine Learning*, pp. 1414–1423. PMLR, 2017.

Hefny, A., Downey, C., and Gordon, G. J. Supervised learning for dynamical system learning. *Advances in neural information processing systems*, 28:1963–1971, 2015.

Kakade, S. and Langford, J. Approximately optimal approximate reinforcement learning. In *IN PROC. 19TH INTERNATIONAL CONFERENCE ON MACHINE LEARNING*, pp. 267–274, 2002.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kuefler, A., Morton, J., Wheeler, T., and Kochenderfer, M. Imitating driver behavior with generative adversarial networks. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 204–211. IEEE, 2017.

Kumor, D., Zhang, J., and Bareinboim, E. Sequential causal imitation learning with unobserved confounders. 2021.

Muller, U., Ben, J., Cosatto, E., Flepp, B., and Cun, Y. L. Off-road obstacle avoidance through end-to-end learning. In *Advances in neural information processing systems*, pp. 739–746. Citeseer, 2006.

Ng, A. Y., Kim, H. J., Jordan, M. I., Sastry, S., and Ballianda, S. Autonomous helicopter flight via reinforcement learning. In *NIPS*, volume 16. Citeseer, 2003.

Pearl, J. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995.

Pearl, J., Glymour, M., and Jewell, N. P. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016.

Pomerleau, D. A. Alvinn: An autonomous land vehicle in a neural network. 1989.

Ross, S. and Bagnell, J. A. Reinforcement and imitation learning via interactive no-regret learning, 2014.

Ross, S., Gordon, G. J., and Bagnell, J. A. A reduction of imitation learning and structured prediction to no-regret online learning, 2011.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms, 2017.

Soo, A., Zuege, D. J., Fick, G. H., Niven, D. J., Berthiaume, L. R., Stelfox, H. T., and Doig, C. J. Describing organ dysfunction in the intensive care unit: a cohort study of 20,000 patients. *Critical Care*, 23(1):1–15, 2019.

Spencer, J., Choudhury, S., Venkatraman, A., Ziebart, B., and Bagnell, J. A. Feedback in imitation learning: The three regimes of covariate shift, 2021.

Sun, W., Vemula, A., Boots, B., and Bagnell, J. A. Provably efficient imitation learning from observation alone, 2019.

Swamy, G., Choudhury, S., Bagnell, J. A., and Wu, Z. S. Of moments and matching: A game-theoretic framework for closing the imitation gap, 2021.

Syrgkanis, V., Agarwal, A., Luo, H., and Schapire, R. E. Fast convergence of regularized learning in games. In Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper/2015/file/7fea637fd6d02b8f0adf6f7dc36aed93-Paper.pdf.

Winship, C. and Morgan, S. L. The estimation of causal effects from observational data. *Annual review of sociology*, 25(1):659–706, 1999.

Wright, P. G. *Tariff on animal and vegetable oils*. Macmillan Company, New York, 1928.

Zhang, J., Kumor, D., and Bareinboim, E. Causal imitation learning with unobserved confounders. *Advances in Neural Information Processing Systems*, 33, 2020.

A. Proofs

A.1. Proof of Validity of Instrument

Proof. We check the instrument conditions in order:

1. *Unconfounded Instrument*: $Z \perp\!\!\!\perp U$: The $Z \to X \leftarrow U$, $V \to X \leftarrow U$, and $X \to Y \leftarrow U$ triples are blocked by standard d-separation rules (Pearl et al., 2016). All paths from Z to U must pass through one of these triples so $Z \perp\!\!\!\perp U$.
2. *Exclusion*: $Z \perp\!\!\!\perp Y|X, U$: The $Z \to X \to Y$, $X \leftarrow U \to Y$, and $V \to X \to Y$ triples are blocked by standard d-separation rules. All paths from Z to Y must pass through one of these triples so $Z \perp\!\!\!\perp Y|X, U$.
3. *Relevance*: $Z \not\perp\!\!\!\perp X$: There is a $Z \to X$ edge, which is assumed to be non-degenerate.

Thus, Z is a valid instrument for determining the causal relationship between X and Y. □

A.2. Proof of Theorem 3.1

Proof. We simplify notation for clarity in our proof. Consider two vectors of the same dimension, $\mathbf{a}$ and $\mathbf{b}$. Assume that $\sum_i^N a_i^2 \le \epsilon$ and $\sum_i^N b_i^2 \le \delta$. This implies that $\|\mathbf{a}\|_2 \le \sqrt{\epsilon}$ and $\|\mathbf{b}\|_2 \le \sqrt{\delta}$. Then, by the triangle inequality, $\|\mathbf{a} - \mathbf{b}\|_2 \le \|\mathbf{a}\|_2 + \|\mathbf{b}\|_2 \le \sqrt{\epsilon} + \sqrt{\delta}$. Setting $a_i = \sqrt{P(z)}(\mathbb{E}[Y|z] - \mathbb{E}_{\hat{x}\sim g(z)}[\widehat{h}(\hat{x})])$ and $b_i = \sqrt{P(z)}(\mathbb{E}_{\hat{x}\sim g(z)}[\widehat{h}(\hat{x})] - \mathbb{E}[\widehat{h}(x)|z])$ proves that

$$\max_{\widehat{h}\in\mathcal{H}} \mathbb{E}_Z[(\mathbb{E}_{x\sim g(z)}[\widehat{h}(x)] - \mathbb{E}_{x\sim P(X|z)}[\widehat{h}(x)])^2] \le \delta, \tag{32}$$

$$\mathbb{E}_z[(\mathbb{E}[Y|z] - \mathbb{E}_{\hat{x}\sim g(z)}[\widehat{h}(\hat{x})])^2] \le \epsilon \tag{33}$$

$$\Rightarrow PRMSE(\widehat{h}) = \sqrt{\mathbb{E}_z[(\mathbb{E}[Y|z] - \mathbb{E}_{x\sim P(X|z)}[\widehat{h}(x)])^2]} \le \sqrt{\epsilon} + \sqrt{\delta} \tag{34}$$

□

A.3. Proof of Theorem 3.2

Proof. Recall (13):

$$\min_{h\in\mathcal{H}} \max_{f\in\mathcal{F}} \mathbb{E}[2(Y - h(X))f(Z) - f^2(Z)] \tag{35}$$

An ϵ-approximate equilibrium is an $(\widehat{h}, \widehat{f})$ pair such that:

$$\max_{f\in\mathcal{F}} \mathbb{E}[2(Y - \widehat{h}(X))f(Z) - f^2(Z)] - \frac{\epsilon}{2} \tag{36}$$

$$\le \mathbb{E}[2(Y - \widehat{f}(X))\widehat{f}(Z) - \widehat{f}^2(Z)] \tag{37}$$

$$\le \min_{h\in\mathcal{H}} \mathbb{E}[2(Y - h(X))\widehat{f}(Z) - \widehat{f}^2(Z)] + \frac{\epsilon}{2} \tag{38}$$

Taking the derivative w.r.t $f(z)$ of the payoff and setting it equal to 0, we arrive at

$$2P(z)\mathbb{E}[Y - \widehat{h}(X)|z] - 2P(z)f(z) = 0 \Rightarrow f(z) = \mathbb{E}[Y - \widehat{h}(X)|z]. \tag{39}$$

Plugging this back into (45) gives us the inequality

$$\mathbb{E}_Z[\mathbb{E}[Y - \widehat{h}(X)|z]^2] - \frac{\epsilon}{2} \le \min_{h\in\mathcal{H}} \mathbb{E}[2(Y - h(X))\widehat{f}(Z) - \widehat{f}^2(Z)] + \frac{\epsilon}{2}. \tag{40}$$

Assuming we are in the realizable setting (e.g. $h(x) = \mathbb{E}[Y|do(x)] \in \mathcal{H}$), $\min_{h\in\mathcal{H}} \mathbb{E}[2(Y - h(X))\widehat{f}(Z) - \widehat{f}^2(Z)] \le 0$. Thus, we can write that:

$$\mathbb{E}_Z[\mathbb{E}[Y - \widehat{h}(X)|z]^2] - \frac{\epsilon}{2} \le \frac{\epsilon}{2} \Rightarrow \text{PRMSE}(\widehat{h}) \le \sqrt{\epsilon}. \tag{41}$$

□

We note that Theorem 3.2 follows somewhat directly from the main theorems of (Dikkala et al., 2020) but that it was not stated in this precise form in their work.

A.4. Proof of Lemma 5.2

Proof. Notice that

$$\max_{\pi \in \Pi} \mathbb{E}_{s_{t-1}}[(\mathbb{E}_{s_t \sim \widehat{\mathcal{T}}(s_{t-1}, \pi_1(s_{t-1}))}[\pi(s_t)] - \mathbb{E}_{s_t \sim P(s_t|s_{t-1})}[\pi(s_t)])^2] \leq \delta \tag{42}$$

can be re-written as

$$\max_{\pi \in \Pi} \mathbb{E}_Z[(\mathbb{E}_{x \sim g(z)}[\pi(x)] - \mathbb{E}_{x \sim P(X|z)}[\pi(x)])^2] \leq \delta. \tag{43}$$

Thus, the proof of Theorem 5.2 holds as written. □

A.5. Proof of Lemma 5.3

An ϵ-approximate equilibrium for the policy player is a π such that

$$\max_{f \in \mathcal{F}} \mathbb{E}[2(a_t - \pi(s_t))f(s_{t-1}) - f^2(s_{t-1})] - \frac{\epsilon}{2} \leq \min_{\pi \in \Pi} \mathbb{E}[2(a_t - h(s_t))\widehat{f}(s_{t-1}) - \widehat{f}^2(s_{t-1})] + \frac{\epsilon}{2}. \tag{44}$$

With a change of notation, we can re-write this as:

$$\max_{f \in \mathcal{F}} \mathbb{E}[2(Y - \pi(X))f(Z) - f^2(Z)] - \frac{\epsilon}{2} \leq \min_{\pi \in \Pi} \mathbb{E}[2(Y - h(X))\widehat{f}(Z) - \widehat{f}^2(Z)] + \frac{\epsilon}{2}. \tag{45}$$

Thus, the proof of Theorem 3.2 holds as written.

A.6. Proof of Theorem 5.1

Proof. By definition,

$$\text{PRMSE}(\pi) = \sqrt{\mathbb{E}_{s \sim d_{\pi_E}}[\mathbb{E}[a' - \pi(s')|s]]^2} = \epsilon. \tag{46}$$

Recall that the measure of ill-posedness of the problem (Dikkala et al., 2020; Chen & Pouzo, 2012) can be defined as

$$\kappa(\Pi) = \sup_{\pi \in \Pi} \frac{\sqrt{\mathbb{E}_{s \sim d_{\pi_E}}[(\pi_E(s) - \pi(s))^2]}}{\sqrt{\mathbb{E}_{s,s',a' \sim d_{\pi_E}}[\mathbb{E}[a' - \pi(s')|s]]^2}} = \sup_{\pi \in \Pi} \frac{\text{RMSE}(\pi)}{\text{PRMSE}(\pi)} \tag{47}$$

Directly,

$$\text{RMSE}(\pi) \leq \epsilon \kappa(\Pi). \tag{48}$$

We repeat the definition of total variation stability of a distribution $P(U)$:

$$\|a - b\|_2 \leq \delta \Rightarrow d_{TV}(a + U, b + U) \leq c\delta. \tag{49}$$

We proceed by noting that TV-stability implies that $\forall s \in \mathcal{S}$,

$$d_{TV}(\pi(s) + U, \pi_E(s) + U) \leq c\,\|\pi(s) - \pi_E(s)\| \tag{50}$$

$$\Rightarrow d_{TV}(\pi(s) + U, \pi_E(s) + U)^2 \leq c^2\,\|\pi(s) - \pi_E(s)\|^2 \tag{51}$$

$$\Rightarrow \mathbb{E}_{s \sim d_{\pi_E}}[d_{TV}(\pi(s) + U, \pi_E(s) + U)^2] \leq c^2 \mathbb{E}_{s \sim d_{\pi_E}}[\|\pi(s) - \pi_E(s)\|^2] = c^2 \text{MSE}(\pi). \tag{52}$$

By Jensen's inequality,

$$\mathbb{E}_{s \sim d_{\pi_E}}[d_{TV}(\pi(s) + U, \pi_E(s) + U)]^2 \leq \mathbb{E}_{s \sim d_{\pi_E}}[d_{TV}(\pi(s) + U, \pi_E(s) + U)^2] \leq c^2 \text{MSE}(\pi). \tag{53}$$

Taking the square root of both sides, we arrive at

$$\mathbb{E}_{s \sim d_{\pi_E}}[d_{TV}(\pi(s) + U, \pi_E(s) + U)] \leq c\,\text{RMSE}(\pi) \leq c\kappa(\Pi)\epsilon. \tag{54}$$

Lastly, we apply the Performance Difference Lemma of (Kakade & Langford, 2002) as follows:

$$\begin{aligned} J(\pi_E) - J(\pi) &= T\mathbb{E}_{s,a\sim d_{\pi_E}}[Q^\pi(s,a) - \mathbb{E}_{a'\sim\pi(s)}[Q^\pi(s,a')]] && (55) \\ &= T\mathbb{E}_{s,a\sim d_{\pi_E}}[Q^\pi(s,\pi_E(s)+u+\widetilde{u}_1) - \mathbb{E}[Q^\pi(s,\pi(s)+u+\widetilde{u}_2)]] && (56) \\ &\leq T^2\mathbb{E}_{s\sim d_{\pi_E}}[d_{TV}(\pi(s)+U, \pi_E(s)+U)] && (57) \\ &\leq c\kappa(\Pi)\epsilon T^2. && (58) \end{aligned}$$

We use the fact that the same u would be added to both the learner and the expert's actions and that rewards are in the range $[-1,1]$ in the third step.

□

A.7. Proof of Lemma 6.1

Proof.

$$\mathbb{E}[a_t|do(s_t)] = \mathbb{E}[\pi_E(s_t) + u_t + u_{t-1}|do(s_t)] = \pi_E(s_t) + \mathbb{E}[u_t] + \mathbb{E}[u_{t-1}] = \pi_E(s_t) \tag{59}$$

$$\mathbb{E}[a_t|s_t] = \mathbb{E}[\pi_E(s_t) + u_t + u_{t-1}|s_t] = \pi_E(s_t) + \mathbb{E}[u_t] + \mathbb{E}[u_{t-1}|s_t] = \pi_E(s) + \mathbb{E}[u_{t-1}|s_t] \tag{60}$$

$$\pi_{BC}(s) - \pi_E(s) = \mathbb{E}[a_t|s_t] - \mathbb{E}[a_t|do(s_t)] = \mathbb{E}[u_{t-1}|s_t] = \mathbb{E}[u|s] \tag{61}$$

□

B. Experiment Details

B.1. LunarLander Experiments

For ease of simulation, we remove the legs from the LunarLander vehicle (the joints connecting them to the main body have a state that is not recorded in the observed state), remove the dispersion noise, and generate trajectories with a fixed ground layout.

For all learned functions, we use two-layer ReLu MLPs with 256 hidden units. We use the Adam optimizer (Kingma & Ba, 2014) for behavioral cloning and `DoubIL` and use the optimistic variant for `ResiduIL`. We apply a weight decay of 1e-3 to all. We train all methods for 50k steps.

Parameter	Value
Learning Rate	3e-4
Batch Size	128

Table 2. Parameters for behavioral cloning.

For computational ease, we only learn the mean of $P(a|s)$ for `DoubIL` and add fresh, appropriately scaled normal noise on-top of it to simulate drawing actions. For more complex noise models, one would need to use a moment matching algorithm (Swamy et al., 2021) in the first stage.

Parameter	Value
Learning Rate	3e-4
Batch Size	128
Num. Samples for $\mathbb{E}$	4

Table 3. Parameters for `DoubIL`.

For implementing the "double samples" for the gradient, we compute $\mathbb{E}_1[a' - \pi(s')|s]$ and $\mathbb{E}_2[a' - \pi(s')|s]$ using independent samples. Then, we apply a stop-gradient operator to the former expectation before taking a product between the expectations and averaging over s:

$$L(\pi) = \mathbb{E}_s[\text{🛑}(\mathbb{E}_1[a' - \pi(s')|s])\mathbb{E}_2[a' - \pi(s')|s]]. \tag{62}$$

This loss function has the correct gradient as it uses independent samples for computing the two expectations.

Parameter	Value
Learning Rate	5e-5
Batch Size	128
BC Regularizer Weight	5e-2
f norm Penalty	1e-3
Adam βs	0, 1e-2

Table 4. Parameters for `ResiduIL`.

We use Gaussian noise with $\sigma = 0.5$.

B.2. PyBullet Experiments

We increase the weight decay for all networks to 5e-3. We keep the same parameters for `DoubIL` (except for increasing the number of samples for $\mathbb{E}$ to 8 for AntBulletEnv) and `BC` as for the LunarLander experiments. We use the following parameters for `ResiduIL`.

Parameter	Value
Learning Rate	5e-5
Batch Size	128
BC Regularizer Weight	0
f norm Penalty	1e-3
Adam βs	0, 1e-2

Table 5. Parameters for `ResiduIL`.

We use Gaussian noise with $\sigma = 2$ for AntBulletEnv and $\sigma = 3$ for HalfCheetahBulletEnv.

B.3. LQG Experiments

We compute the optimal policy for the following canonical linear system via solving a Discrete-Time Algebraic Ricatti Equation via the standard iterative method:

$$x_t = Ax_{t-1} + Bu_{t-1} \tag{63}$$

$$J(K) = \sum_t^T x_t^T Q x_t + (Kx_t)^T R K x_t \tag{64}$$

$$A = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 0.5(\Delta T)^2 \\ \Delta T \end{bmatrix}, Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, R = \begin{bmatrix} 0.1 \end{bmatrix}, \Delta T = 0.1$$

This is the dynamics of a "sliding brick on a frozen lake." We then simulate rollouts of 200 timesteps with u_t being drawn i.i.d. from the standard normal distribution. We confound actions with the sum of confounders going H steps back:

$$a_t = K^* s_t + \sum_{j=t-H}^{t} u_j. \tag{65}$$

We simulate 1000 such rollouts to compute (31) empirically. We calculate $\mathbb{E}[X|z] = \mathbb{E}[s_t|s_{t-H}] = (A + BK^*)^H s_{t-H}$ analytically instead of via samples due to the small value of the quantity in comparison to the variance of the noise.