Causal Confusion in Imitation Learning

Pim de Haan
UC Berkeley

Dinesh Jayaraman
UC Berkeley

Sergey Levine
UC Berkeley

Abstract

Behavioral cloning provides a simple and general mechanism for training policies for autonomous agents; it reduces policy learning to supervised learning, where a standard discriminative model, such as a deep net, is trained to predict expert actions given observations. Such discriminative models are non-causal: the training procedure is unaware of the particular causal structures of the underlying dynamical system. In this paper, we point out that ignoring causality in this manner is particularly damaging in imitation learning. In particular, it leads to a surprising "causal confusion" phenomenon: fixed-capacity policies that are trained to fixed training error on the same demonstrations with access to more information can actually yield worse performance. We investigate how this problem arises and how its occurrence may be predicted for different behavioral cloning tasks. Further, we propose a solution to combat causal confusion, which involves first inferring a distribution over potential causal models consistent with the behavioral cloning objective, then identifying the correct hypothesis through "intervention". Our approach permits intervention in the form either of expert queries or of policy execution in the environment. Our experiments both verify the presence of causal confusion in behavioral cloning on various benchmark domains and validate our solution against DAgger and other baselines and ablations.

1 Introduction

Imitation learning allows for control policies to be learned directly from example demonstrations provided by human experts. The benefits of imitation learning are clear: it is easy to implement, and exploiting expert knowledge largely reduces or completely removes the need for extensive interaction with the environment during training [34, 2, 1, 14].

However, imitation learning suffers from a fundamental problem: distributional shift [5, 35]. Unlike supervised learning, training and testing states are drawn from different distributions, induced by the expert and learned policies, respectively. Therefore, the training objective of imitating expert behavior along the imitation trajectories is not perfectly aligned with the true objective of performing the task correctly. This general problem is widely acknowledged [5, 35, 36]. Despite this, naïve behavioral cloning continues to be yield excellent results in practice [29, 34, 30, 2, 26]. This might lead one to conclude that, for many practical problems, behavioral cloning is a viable method for imitation.

In this paper, we identify a specific and very problematic manifestation of distributional shift that might at first come as a surprise: "causal confusion." Correlates of expert actions in the demonstration set are impossible to distinguish from true causes. While this problem exists in standard supervised learning too, causal confusion is particularly pronounced and important to address in behavior cloning for two reasons: (i) the temporal structure of sequential action, where future observations are effects of current actions, induces complex causal structures, and (ii) the aforementioned distributional shift makes correct causal models particularly valuable, as we will show.

NIPS 2018 Workshop on Causal Learning.

To illustrate, consider behavior cloning to train a neural network to drive a car. Consider two models: (i) model A, whose input is an image of the dashboard and windshield, and (ii) model B, with identical architecture, whose input is the same image but with the dashboard blacked out (see Fig 1). Both cloned policies achieve low training losses, but when tested on the road, model B drives well, while model A struggles. It turns out that the dashboard has an indicator light that comes on instantaneously to indicate when the brake is applied. Model A wrongly learns that it must apply the brake whenever it sees the brake light on—even though the brake light is the *effect* of braking, treating it as the cause is sufficient to achieve low training loss.

Figure 1: Causal confusion: *more* information yields worse imitation learning performance. Here, an imitator with access to the full scene including the brake indicator etc. (left) performs worse than an imitator whose inputs are partially ablated (right). See text.

This illustrates a classic symptom of causal confusion: access to *more information* leads a *fixed-size model* to exhibit *worse generalization performance* in the presence of distributional shift. This is neither standard overfitting, where larger models memorize training data and fail to generalize to other samples from the same distribution, nor is it related to any optimization difficulties associated with access to more information—it occurs even when both models are trained to the same training error.

Our first contribution in this paper is to point out the causal confusion problem in imitation learning and investigate its symptoms. We show that simply adding a little bit of additional information to the observation vector can cause this problem to appear in a number of simple benchmark control domains. Fortunately, sequential decision making offers a way to conduct interventional queries to resolve causal confusion, by letting the learned model control the system and observe outcomes. Based on this, our second contribution is to propose a solution to overcome this causal confusion issue by learning the correct causal model, even when using complex deep neural network models. We devise an efficient intervention strategy by first performing variational inference over causal models to infer a diverse hypothesis set, and then use targeted intervention to efficiently search over this hypothesis set, either by querying an expert, or by executing behaviorally cloned policies corresponding to different hypotheses in the environment.

2 Related Work

Imitation learning. Imitation learning through behavioral cloning dates back to Pomerleau *et al*, 1989 [34] and remains popular today [29, 30, 2, 8, 26]. The distributional shift problem, wherein a cloned policy encounters unfamiliar states during autonomous execution, has been addressed by a host of approaches [5, 35, 36, 19, 13]. Most closely related to us amongst these are [5, 35, 36], with which we share the idea of iteratively querying an expert based on states encountered by some intermediate cloned policy. Our solution differs in the following ways: (i) it specifically targets causal confusion rather than general distributional shift—in other words, while these above approaches attempt to minimize distributional shift between demonstrations and policy execution by collecting new "on-policy" demonstrations, our approach attempts to be robust to distributional shift by learning the correct causal model, (ii) rather than relying on a large number of expert queries to retrain policies, our approach uses expert queries as causal interventions on the state to disambiguate among various trained models—it thus requires many fewer expert queries, and (iii) it is also flexible enough to substitute expert queries with environment rewards when queryable experts are not available. In our experiments, we compare against the popular DAgger [36] approach to combat distributional shift.

Causal inference. Causal inference is the general problem of deducing cause-effect relationships among variables [41, 31, 32, 40, 6, 42]. "Causal discovery" approaches allow causal inference from pre-recorded observations under constraints [43, 12, 22, 10, 23, 24, 20, 9, 27, 45]. Observational causal inference is known to be impossible in general [31, 33]. We operate in the interventional regime [44, 7, 39, 38], well-suited to imitation learning, where a user may "experiment" to discover causal structures by assigning values to some subset of the variables of interest and observing the effects on the rest of the system.

Causal inference in the context of imitation learning is to our knowledge unstudied, despite the fact that, as we will show, ignoring causal structure is particularly problematic in this context. In [16], cause-effect reasoning is applied to infer high-level intentions of demonstrators from their actions, given a pre-specified causal graph from intentions to actions. In contrast, we focus on causal relationships among observations and actions, and discover the causal graph.

3 Identifying Causal Confusion

Consider the graph in Fig 2, showing the underlying dynamics of imitation learning among states/observations $X^t = \{X_i^t\}_{i=1}^n$ and expert actions A^t over time t. At time t, the expert's decisions A^t are based on an unknown subset of the state variables X^t. We refer to variables in this subset as "causes" and the others as "nuisance variables". A confounding variable $Z^t = [X^{t-1}, A^{t-1}]$ influences each state variable in X^t, so that some nuisance variables may still be correlated with A^t among (X^t, A^t) pairs from demonstration trajectories. In the car example, the dashboard light is a nuisance variable, influenced by the previous expert action, which is contained in Z^t.

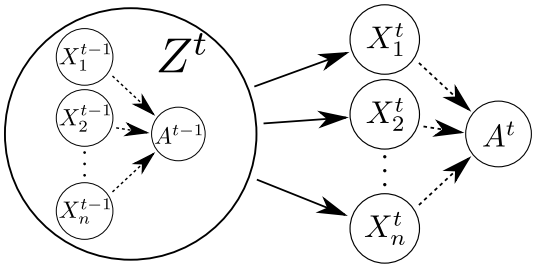


Figure 2: A graph of the underlying causal dynamics of imitation learning. Parents of a node represent its causes. State variables $\{X_i^t\}_{i=1}^n$ are fully observed.

A standard supervised imitation learner might learn a model that relies on nuisance variables to predict actions. Note that nuisance variables are *true* (not spurious) correlates; such a model would generalize well to held-out (X^t, A^t) pairs from demonstrations. However, as motivated in Sec 1, imitation-learned policies face a distributional shift problem when deployed. In the graph of Fig 2, the distribution of Z^t shifts; past states X^{t-1} and actions A^{t-1} are generated by the imitation policy rather than the expert policy. This affects the relationship of the optimal (expert) action A^t to nuisance variables, but not to true causes.

Thus, in order to be robust to distributional shift, a policy must rely solely on true causes of expert actions and ignore nuisance variables. Following [31], this amounts to inferring the interventional query $p(A^t|do(X^t))$. In other words, if some external mechanism were able to "intervene" on the state X^t to assign it some value independent of its parent nodes, what would the distribution of A^t be? Inferring this interventional query directly determines robustness to distributional shift. In the car example, modeling the interventional query would imply "setting" the brake light to on or off and observing the expert's behaviour. This would yield a clear signal unobstructed by confounders: the expert ignores the brake light, and activates the brakes solely based on causes observable through the windshield. This can be formalised in the language of functional causal models [31].

Functional Causal Models: A functional causal model (FCM) over a set of variables $\{Y_i\}_{i=1}^n$ is a tuple (G, θ_G) containing a graph G over $\{Y_i\}_{i=1}^n$, and deterministic functions $f_i(\cdot; \theta_G)$ with parameters θ_G describing how the causes of each variable Y_i determine it:

$$Y_i = f_i(Y_{\text{Pa(i;G)}}, E_i; \theta_G), \tag{1}$$

where E_i is a stochastic noise variable that represents all external influences on Y_i, and $\text{Pa}(i; G)$ denote the indices of parent nodes of Y_i, which correspond to its causes.

An intervention on Y_i to set its value may now be represented by a structural change in this graph to produce the "mutilated graph" $G_{\bar{Y}_i}$, in which incoming edges to Y_i are removed. For a more thorough overview of FCMs, see [31].

Proposition 1. *Let the expert's functional causal model be $(G^*, \theta^*_{G^*})$, with causal graph $G^* \in \mathcal{G}$ and function parameters $\theta^*_{G^*}$. We assume some faithful*[1] *learner $(\hat{G}, \theta_{\hat{G}}), \hat{G} \in \mathcal{G}$ that agrees on the interventional query:*

$$\forall X, A : p_{G^*, \theta^*_{G^*}}(A|do(X)) = p_{\hat{G}, \theta_{\hat{G}}}(A|do(X))$$

[1] A causal model is faithful when all conditional independence relationships if the distribution are represented in the graph.

Then it must be that $G^* = \hat{G}$.[2]

Proof. For graph G, define the index set of state variables that are independent of the action in the mutilated graph $G_{\bar{X}}$:

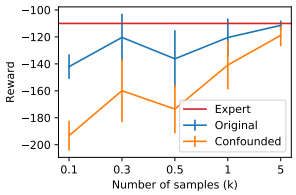
$$I_G = \{i | X_i \perp\!\!\!\perp_{G_{\bar{X}}} A\}$$

From the assumption of matching interventional queries and the assumption of stability, it follows that: $I_{G^*} = I_{\hat{G}}$. From the graph, we observe that $I_G = \{i | (X_i \to A) \notin G\}$ and thus $G^* = \hat{G}$. □

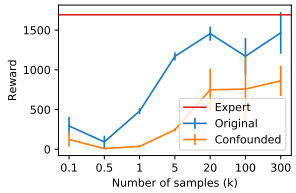
In other words, in keeping with our intuition above, correctly modeling interventional queries requires learning the correct causal graph G. In realistic learning scenarios, direct intervention such as setting the state of a car's brake light may be impossible. However, as we will argue in Sec 4.2, interaction with the environment may be a viable indirect substitute for intervention.

Causal confusion testbeds: To study causal confusion further, we first construct tasks that exhibit causal confusion. We start from widely studied benchmark control tasks from OpenAI Gym [3] and simply append the previous action to the observation vector. This is a proxy for scenarios like our car example, in which traces of past actions are observable in the state. Producing causal confusion in this manner relies on actions at past times being predictive of future actions, so that the past action is an effective nuisance variable. This is a common feature of many control tasks, where optimal actions vary smoothly over time. We select three such tasks from Gym: MountainCar, Hopper, and Walker2d.

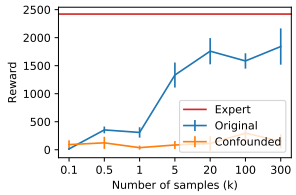
For each task, we first train "expert" policies through reinforcement learning (Q-learning [28] for MountainCar, TRPO [37] for others). Then, we train two imitation policies with identical architectures on the same state-action pairs from demonstrations from the experts, yielding near-zero validation error on held-out demonstration data. The input to the first imitator (CONFOUNDED) is the augmented observation vector containing the previous action nuisance variable. The second imitator (ORIGINAL) receives the original observation vector augmented with a random noise variable in place of the nuisance variable. Finally, we train each imitator on different-sized datasets.



(a) MountainCar



(b) Hopper



(c) Walker2D

Figure 3: Diagnosing causal confusion: net reward (y-axis) vs number of training samples (x-axis) for ORIGINAL and CONFOUNDED, compared to expert reward (mean and stdev over 5 runs). Also see Appendix A.

Fig 3 shows the total reward received by the different policies for Mountain Car, Hopper, and Walker2d. ORIGINAL yields rewards tending towards expert performance as the size of the imitation dataset increases. CONFOUNDED improves much more slowly in MountainCar and Hopper, and fails to improve at all in Walker2d. Overall, the results are clear: in all three cases, *adding* information leads to the *inferior* performance of CONFOUNDED, compared with the control setting, ORIGINAL. As Figure 9 in the appendix shows, this difference is not due to different training/validation losses on the expert demonstrations—for example, in Walker2d, CONFOUNDED yields lower validation loss than ORIGINAL on held-out demonstration samples, but yields much lower rewards. This not only validates the existence of the causal confusion problem which we have motivated and described above, but also provides us with testbeds for a solution. In particular, we will evaluate causal confusion resolution on the modified MountainCar and Hopper environments.

4 Resolving Causal Confusion Through Discovery and Intervention

We propose to infer the expert's causal model in two stages. First, we propose a causal discovery approach to find the set of all functional causal models, parameterized by deep neural networks,

[2]For notational simplicity, we drop time t from the superscript when exclusively discussing states and actions from the same time.

consistent with the expert demonstrations (Sec 4.1). Then, we perform targeted interventions to efficiently search over the hypothesis set for the correct causal model (Sec 4.2).

4.1 Variational Inference for Causal Discovery

The problem of discovering causal graphs from passively observed data is called causal discovery. The PC algorithm [41] is arguably the most widely used and easily implementable causal discovery algorithm. In the case of Fig 2, the PC algorithm would imply the absence of the arrow $X_i^t \to A^t$, if the conditional independence relation $A^t \perp\!\!\!\perp X_i^t | Z$ holds, which can be tested by measuring the mutual information. However, the PC algorithm relies on *faithfulness* of the causal graph i.e. conditional independence must imply d-separation in the graph. However, faithfulness is easily violated in a Markov Decision Process. If for some i, X_i^t is a cause of the expert's action A^t (the arrow $X_i \to A^t$ should exist), but X_i is the result of a deterministic function of Z^t, then always $A^t \perp\!\!\!\perp X_i^t | Z$ and the PC algorithm would wrongly conclude that the arrow $X_i \to A^t$ is absent.[3]

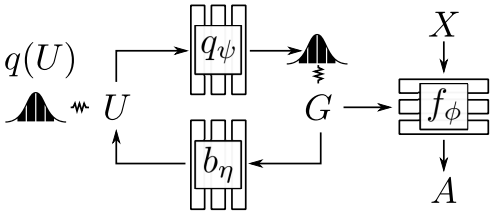


Figure 4: Training architecture for variational inference-based causal discovery as described in Eq 4.1. The policy network f_ϕ represents a mixture of policies, one corresponding to each value of the binary causal graph structure variable G. This variable in turn is sampled from the distribution $q_\psi(G|u)$ produced by an inference network from an input latent U. Further, a network b_η regresses back to the latent U to enforce that G should not be independent of U.

We take a Bayesian approach to causal discovery [12] from demonstrations. Recall from Sec 3 that the expert's actions A are based on an unknown subset of the state variables $\{X_i\}_{i=1}^n$. Each X_i may either be a cause or not, so there are 2^n possible graphs. We now define a variational inference approach to infer a distribution over functional causal models (graphs and associated parameters) such that its modes are consistent with the demonstration data D.

While Bayesian inference is intractable, variational inference can be used to find a distribution that is close to the true posterior distribution over models. We parameterize the structure G of the causal graph as a vector of n correlated Bernoulli random variables G_k, each indicating the presence of a causal arrow from X_k to A. We assume a variational family with a point estimate θ_G of the parameters corresponding to graph G and use a latent variable model to describe the correlated Bernoulli variables, with a standard normal distribution $q(U)$ over latent random variable U:

$$q_\psi(G, \theta) = q_\psi(G)[\theta = \theta_G] = \int q(U) \prod_{k=1}^{n} q_\psi(G_k|U)[\theta = \theta_G] dU \tag{2}$$

We now optimise the evidence lower bound (ELBO):

$$\arg\min_q D_{KL}(q_\psi(G, \theta)|p(G, \theta|D)) =$$
$$\arg\max_\psi \sum_i \mathbb{E}_{U\sim q(U), G_k \sim q_\psi(G_k|U)} \left[\log p(A_i|X_i, G, \theta_G) + \log p(G)\right] + \mathcal{H}_q(G). \tag{3}$$

The terms in this ELBO are intuitive:

- Likelihood: $p(A_i|X_i, G, \theta_G)$ is the likelihood of the observations X under the FCM (G, θ_G). It is modelled by a single neural network $f_\phi([X \odot G, G])$, where $\odot$ is the element-wise multiplication, $[\cdot, \cdot]$ denotes concatenation and ϕ are neural network parameters.
- Entropy Regularizer: $\mathcal{H}_q$ acts as a regularizer to prevent the graph distribution from collapsing to the maximum a-posteriori estimate. It is intractable to directly maximize entropy, but a tractable variational lower bound can be formulated. Using the product rule of entropies, we may write:

$$\mathcal{H}_q(G) = \mathcal{H}_q(G|U) - \mathcal{H}_q(U|G) + \mathcal{H}_q(U) = \mathcal{H}_q(G|U) + I_q(U; G)$$

[3] More generally, faithfulness places strong constraints on the expert graph. For example, a visual state may contain unchanging elements such as the car frame in Fig 1, which are by definition deterministic functions of the past. As another example, goal-conditioned tasks must include a constant goal in the state variable at each time, which once again has deterministic transitions, violating faithfulness.

In this expression, $\mathcal{H}_q(G|U)$ promotes diversity of graphs, while $I_q(U;G)$ encourages correlation among $\{G_k\}$. $I_q(U;G)$ can be bounded below using the same variational bound used in InfoGAN [4], with a variational distribution b_η: $I_q(U;G) \geq \mathbb{E}_{U,G\sim q_\psi} \log b_\eta(U|G)$. Thus, during optimization, in lieu of entropy, we maximize the following lower bound:

$$\mathcal{H}_q(G) \geq \mathbb{E}_{U,G\sim q}\left[-\sum_k \log q_\psi(G_k|U) + \log b_\eta(U|G)\right]$$

- Prior over graph structures: The prior $p(G)$ over graph structures is set to prefer graphs with fewer causes for action A—it is thus a sparsity prior.

Note that G is a discrete variable, so we cannot use the reparameterization trick [18]. Instead, we use the Gumbel Softmax trick [15, 25] to compute gradients for training a small network $q_\psi(G_k|U)$. Note that this does not affect f_ϕ, which can be trained with standard backpropagation.

Note that the loss of Eq 3 is easily interpretable independent of the formalism of variational Bayesian causal discovery. A mixture of predictors f_ϕ is jointly trained, each paying attention to diverse sparse subsets (identified by G) of the inputs. This is related to variational dropout [17]. Once this model is trained, $q_\psi(G)$ represents the hypothesis distribution over graphs, and $\pi_G(x) = f_\phi([x \odot G, G])$ represents the imitation policy corresponding to a graph G. Fig 4 shows the architecture.

4.2 Targeted Intervention Through Expert Queries and Environment Rewards

The discovery posterior over hypotheses inferred in Sec 4.1 now functions as the intervention prior for causal inference through intervention. We propose two variants for targeted intervention to compute the likelihood $p(\mathcal{O}|G)$, for optimality random variable $\mathcal{O}$, of each hypothesis:

- Intervention by expert queries: This is the standard intervention approach applied to imitation learning i.e., intervene on X_t to assign it a value, and observe the expert response A. This requires an interactive expert, as in [35], but the expert can be queried efficiently.
- Intervention by policy execution: In the absence of a queryable expert, environmental rewards r_t may be used to conduct indirect intervention in the imitation learning setting. The policies π_G corresponding to different hypotheses G are evaluated in the environment and rewards r_t are collected. The likelihood is proportional to $\exp \sum_t r_t$. The intuition is simple: environmental rewards from policy rollouts contain information about optimal expert policies even when those experts are not queryable.

Once the likelihood is determined by intervention, it is combined with the intervention prior $p_0(G)$ using Bayes rule to compute the final posterior distribution over functional causal models $p(G|\mathcal{O})$.

Note that both above intervention approaches evaluate individual hypotheses in isolation, but the number of hypotheses grows exponentially in the number of state variables. To handle larger states, we infer a graph distribution $\pi(G)$, by assuming an energy based model with a linear energy $Q(G) = \langle w, G\rangle + b$, so the graph distribution is $\pi(G) = \prod_i \pi(G_i) = \prod_i \text{Bernoulli}(G_i|\sigma(w_i/\tau))$, where σ is the sigmoid, which factorizes in independent factors. The independence assumption is sensible as we are interested in the mode of this distribution, instead of in capturing all modes, as during the discovery phase. $Q(G)$ is inferred from linear regression on dataset $(G, \mathcal{L}(G)+\log p_0(G))$, where $p_0(G)$ is the posterior from the causal discovery. G can be sampled arbitrarily.

For intervention by policy execution, $\mathcal{L}(G)$ is the reward obtained by executing policy π_G. The current graph distribution is used to sample G. For intervention by expert queries, $\mathcal{L}(G)$ is the mean-squared error of the policy $f_\phi([\cdot \odot G, G])$ on expert query data. In this case, the number of G's for which we evaluate the policy is only constrained by computational considerations. In practice, it may be feasible to execute the policy for all G.

The above method can be formalized within the reinforcement learning framework [21]. As we show in Appendix C, the energy-based model can be seen as an instance of Soft Q-Learning [11].

5 Causal Inference Evaluation

We now use the modified MountainCar and Hopper environments from Sec 1 as our testbeds to evaluate the solution described in Sec 4. We compare against three baselines: (i) DAGGER [36], which

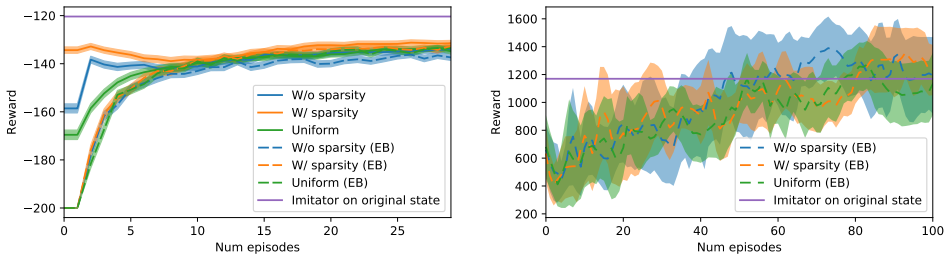


Figure 6: Reward of selected model after intervention by policy execution. (Left) MountainCar (Right) Hopper.

attempts to minimize distributional shift through iterative imitation policy training and on-policy dataset aggregation, (ii) UNIFORM, which ablates out the effect of the hypothesis generation scheme described in Sec 4.1, instead sampling hypotheses uniformly at random for intervention, and (iii) W/O SPARSITY PRIOR, which sets the prior over graphs $p(G)$ to be uniform in Eq 3. We also study the effect of intervention with a learned linear energy based model (EB). For Hopper, we found that only the energy based method yielded successful intervention posteriors, due to the higher dimensionality. We evaluate each approach by plotting net returns (sum of rewards) vs. intervention cost. Intervention cost may be measured either in terms of number of expert queries, or the number of policy rollouts.

Causal discovery. In modified MountainCar, there are exactly eight possible causal graphs G, corresponding to each of three state variables (position, velocity, previous action) either causing or not causing the next action. We represent these in binary form – e.g. the true causal model is "110" (position and velocity cause next action, and previous action is ignored). Fig 5 shows the discovered posterior using the causal discovery approach of Sec 4.1. The true causal model is assigned the second highest posterior after the model that accesses all three observations when we take the prior over graphs to be uniform. When the sparsity prior is included, the true causal model is the mode of the posterior. Modified Hopper has 14 state variables (11 original state variables corresponding to joint states, plus 3 past action variables), and the number of candidate graphs is $2^{14} = 16,384$. In our Hopper experiments, the highest probabilities were assigned to confounded models, but the true causal model was also assigned a significantly above-average probability as expected, ranked 743 out of $16,384$. We use this discovery posterior as the prior for causal inference during intervention (see Sec 4.2).

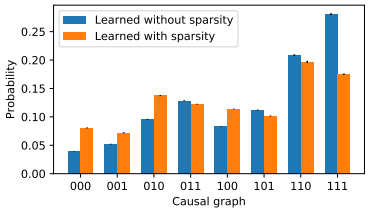


Figure 5: Posterior after causal discovery of different hypotheses on MountainCar.

Intervention by policy execution. In this experiment, we evaluate two methods of intervention through policy execution. In the first, we sample and execute the hypotheses learned in the first stage in order of their discovery posteriors, highest first. The model with the highest mean reward over these intervention rollouts is maintained as the "selected" model. In the second method, we learn the energy based model through Soft Q-Learning. Fig 6 plots the mean reward computed vs. number of intervention rollouts, comparing sampling hypotheses in order of the discovery posterior against sampling in random order. As Fig 6 shows, all methods converge to a model yielding similar reward, which we verified to be the correct causal models of MountainCar and Hopper.

For the non-energy based model, with MountainCar, a small difference is noticeable in the used prior. The uniform prior requires about four runs on average, while the learned prior without sparsity needs only two, as the causal model is assigned the second highest probability by the discovery posterior. The learned prior with sparsity selects the causal model even without intervention.

These results show that our method successfully performs causal inference within a few trials. They also validate that indirect intervention through policy execution is indeed a valid intervention approach, as argued in Sec 4.2.

Intervention by expert queries. Next, we perform direct intervention by querying the expert on samples from trajectories produced by the different causal graphs, where the number of trajectories per graph is weighed by the discovery posterior. Of these trajectories, the states are sampled on which the graphs disagree most. This allows us to compute the intervention likelihood and posterior over hypotheses after each query, shown in Fig 7. In this setting, we can also directly compare to DAgger [36].

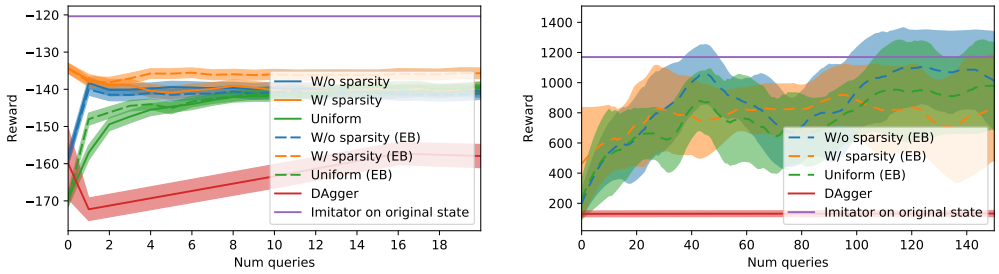


Figure 7: Reward of selected model after intervention by expert queries. (Left) MountainCar (Right) Hopper.

Our approach not only correctly identifies the causal model efficiently, but also exceeds the rewards achieved by a policy trained with DAgger, while using far fewer expert queries. In Appendix B, we show that DAgger requires hundreds of queries to achieve similar rewards for MountainCar and tens of thousands for Hopper.

Of the two intervention approaches, policy execution converges to better final rewards—we believe this is because optimizing for agreement with the expert is only a proxy for optimizing for reward. For example, two quite different policies may both be optimal. Finally, for both environments, UNIFORM is actually feasible and does not perform very poorly in both intervention settings—however, this would not be feasible in larger problems with more state variables, where the benefits of causal inference would be clearer.

VAE-encoded latent space. Does our approach continue to work with more complex state spaces? We encode MountainCar states into a 10-D latent space through a variational autoencoder [18]. First, does the causal confusion effect persist? Comparing ORIGINAL-VAE and CONFOUNDED-VAE in Fig 8 shows that indeed, it does. Next, does our method successfully relieve causal confusion? CAUSAL-VAE (EB) is our energy-based approach without sparsity applied with the VAE-encoded confounded state. As Fig 8 shows, our method improves the performance of CONFOUNDED-VAE to be on par with the imitator on the true non-VAE-encoded state space. This result holds promise for the ability of our approach to handle more general cases of causal confusion.

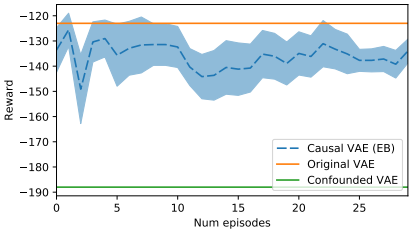


Figure 8: Rewards intervention by policy execution with VAE encoded state on MountainCar.

6 Conclusions

We have identified and described a basic problem in imitation learning through behavior cloning: causal confusion. We have shown how to identify this problem, and constructed simple tasks exhibiting the problem by modifying commonly used control benchmarks. Further, we have proposed a causally motivated approach to solve this problem by first constructing a mix of hypotheses through variational inference, then using targeted interventions to evaluate and select hypotheses. Our experiments on MountainCar and Hopper show the promise of this approach, compared to the widely used DAgger approach for behavior cloning.

For these settings, the advantages of the causal discovery phase were not very clear, since for these environments it is feasible to train policies for all hypotheses jointly. For this same reason, we found that the variational inference approach for causal discovery proposed in Sec 4.1 did not provide gains over an "offline" causal discovery approach that simply assigned higher discovery posterior probabilities to graphs that yielded lower errors after training the policy network f_ϕ. However, we expect that for problems that are higher dimensional, learning the causal discovery graph distribution jointly with the policies as in Sec 4.1 would yield larger gains. On the other hand, the benefits of our targeted intervention approach (Sec 4.2) are already very clear even in these lower-dimensional problems.

For future work we will attempt to identify and solve causal confusion in visual domains. Additionally, we will explore the problem in settings such as driving in which the nuisance variables arise more naturally in the state space, as identified earlier in [29], where behavior cloning failed due to causal confusion when provided with state history information.

Acknowledgments: We would like to thank Karthikeyan Shanmugam and Shane Gu for pointers to prior work early in the project, and Yang Gao, Abhishek Gupta, Marvin Zhang, Alyosha Efros, and Roberto Calandra for helpful discussions in various stages of the project.

References

[1] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and autonomous systems*, 57(5):469–483, 2009.

[2] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

[3] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[4] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[5] Hal Daumé, John Langford, and Daniel Marcu. Search-based structured prediction. *Machine learning*, 75(3):297–325, 2009.

[6] Frederick Eberhardt. Introduction to the foundations of causal discovery. *International Journal of Data Science and Analytics*, 3(2):81–91, 2017.

[7] Frederick Eberhardt and Richard Scheines. Interventions and causal inference. *Philosophy of Science*, 74(5):981–995, 2007.

[8] Alessandro Giusti, Jerome Guzzi, Dan Ciresan, Fang-Lin He, Juan Pablo Rodriguez, Flavio Fontana, Matthias Faessler, Christian Forster, Jurgen Schmidhuber, Gianni Di Caro, Davide Scaramuzza, and Luca Gambardella. A machine learning approach to visual perception of forest trails for mobile robots. *IEEE Robotics and Automation Letters*, 2016.

[9] Olivier Goudet, Diviyan Kalainathan, Philippe Caillou, David Lopez-Paz, Isabelle Guyon, Michele Sebag, Aris Tritas, and Paola Tubaro. Learning functional causal models with generative neural networks. *arXiv preprint arXiv:1709.05321*, 2017.

[10] Isabelle Guyon, Constantin Aliferis, Greg Cooper, André Elisseeff, Jean-Philippe Pellet, Peter Spirtes, and Alexander Statnikov. Design and analysis of the causation and prediction challenge. In *Causation and Prediction Challenge*, pages 1–33, 2008.

[11] Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. *arXiv preprint arXiv:1702.08165*, 2017.

[12] David Heckerman, Christopher Meek, and Gregory Cooper. *A Bayesian Approach to Causal Discovery*, pages 1–28. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[13] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, pages 4565–4573, 2016.

[14] Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, 50(2):21, 2017.

[15] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[16] Garrett Katz, Di-Wei Huang, Rodolphe Gentili, and James Reggia. Imitation learning as cause-effect reasoning. In *Artificial General Intelligence*, pages 64–73. Springer, 2016.

[17] Diederik P Kingma, Tim Salimans, and Max Welling. Variational dropout and the local reparameterization trick. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2575–2583. Curran Associates, Inc., 2015.

[18] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[19] Michael Laskey, Jonathan Lee, Roy Fox, Anca Dragan, and Ken Goldberg. Dart: Noise injection for robust imitation learning. *arXiv preprint arXiv:1703.09327*, 2017.

[20] Thuc Le, Tao Hoang, Jiuyong Li, Lin Liu, Huawen Liu, and Shu Hu. A fast pc algorithm for high dimensional causal discovery with multi-core pcs. *IEEE/ACM transactions on computational biology and bioinformatics*, 2016.

[21] Sergey Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review. *CoRR*, abs/1805.00909, 2018.

[22] D. Lopez-Paz, R. Nishihara, S. Chintala, B. Schölkopf, and L. Bottou. Discovering causal signals in images. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*, pages 58–66, Piscataway, NJ, USA, July 2017. IEEE.

[23] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. In *Advances in Neural Information Processing Systems*, pages 6446–6456, 2017.

[24] Marloes H Maathuis, Diego Colombo, Markus Kalisch, and Peter Bühlmann. Predicting causal effects in large-scale systems from observational data. *Nature Methods*, 7(4):247, 2010.

[25] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

[26] Jeffrey Mahler and Ken Goldberg. Learning deep policies for robot bin picking by simulating robust grasping sequences. In *Conference on Robot Learning*, pages 515–524, 2017.

[27] Jovana Mitrovic, Dino Sejdinovic, and Yee Whye Teh. Causal inference via kernel deviance measures. *arXiv preprint arXiv:1804.04622*, 2018.

[28] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[29] Urs Muller, Jan Ben, Eric Cosatto, Beat Flepp, and Yann L Cun. Off-road obstacle avoidance through end-to-end learning. In *Advances in neural information processing systems*, pages 739–746, 2006.

[30] Katharina Mülling, Jens Kober, Oliver Kroemer, and Jan Peters. Learning to select and generalize striking movements in robot table tennis. *The International Journal of Robotics Research*, 32(3):263–279, 2013.

[31] Judea Pearl. *Causality*. Cambridge university press, 2009.

[32] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. MIT press, 2017.

[33] Jonas Peters, Joris M Mooij, Dominik Janzing, and Bernhard Schölkopf. Causal discovery with continuous additive noise models. *The Journal of Machine Learning Research*, 15(1):2009–2053, 2014.

[34] Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. In *Advances in neural information processing systems*, pages 305–313, 1989.

[35] Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 661–668, 2010.

[36] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635, 2011.

[37] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International Conference on Machine Learning*, pages 1889–1897, 2015.

[38] Rajat Sen, Karthikeyan Shanmugam, Alexandros G Dimakis, and Sanjay Shakkottai. Identifying best interventions through online importance sampling. *arXiv preprint arXiv:1701.02789*, 2017.

[39] Karthikeyan Shanmugam, Murat Kocaoglu, Alexandros G Dimakis, and Sriram Vishwanath. Learning causal graphs with small interventions. In *Advances in Neural Information Processing Systems*, pages 3195–3203, 2015.

[40] Peter Spirtes. Introduction to causal inference. *Journal of Machine Learning Research*, 11(May):1643–1662, 2010.

[41] Peter Spirtes, Clark N Glymour, Richard Scheines, David Heckerman, Christopher Meek, Gregory Cooper, and Thomas Richardson. *Causation, prediction, and search*. MIT press, 2000.

[42] Peter Spirtes and Kun Zhang. Causal discovery and inference: concepts and recent methodological advances. In *Applied informatics*, volume 3, page 3. SpringerOpen, 2016.

[43] Mark Steyvers, Joshua B Tenenbaum, Eric-Jan Wagenmakers, and Ben Blum. Inferring causal networks from observations and interventions. *Cognitive science*, 27(3):453–489, 2003.

[44] Simon Tong and Daphne Koller. Active learning for structure in bayesian networks. Citeseer, 2001.

[45] Yixin Wang and David M Blei. The blessings of multiple causes. *arXiv preprint arXiv:1805.06826*, 2018.

A Results on Diagnosis of Causal Confusion

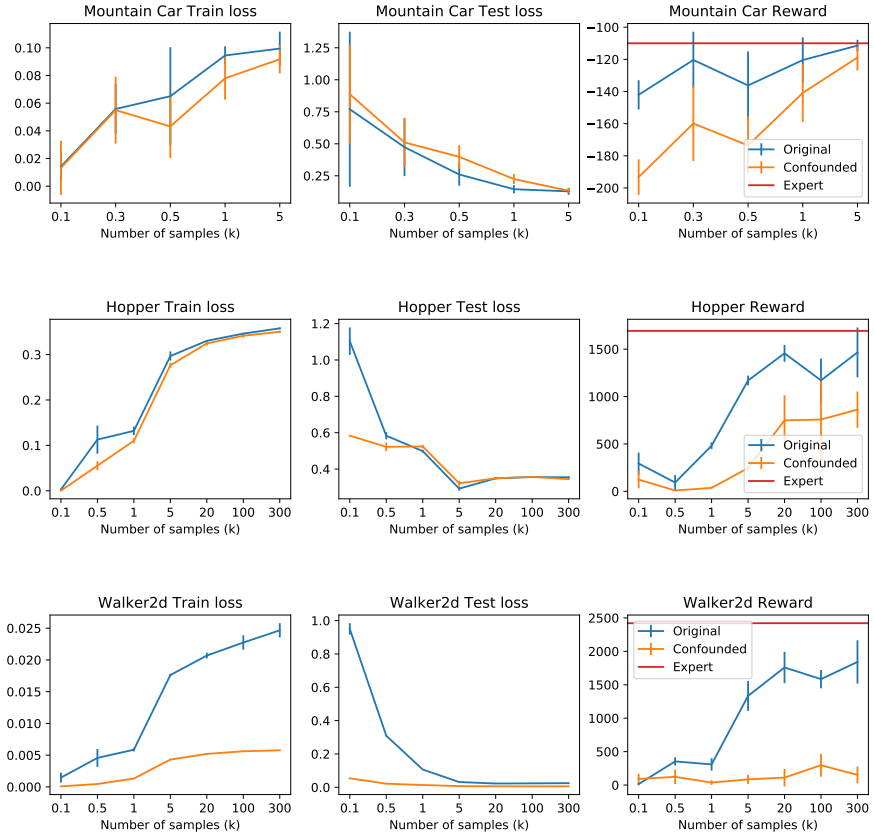
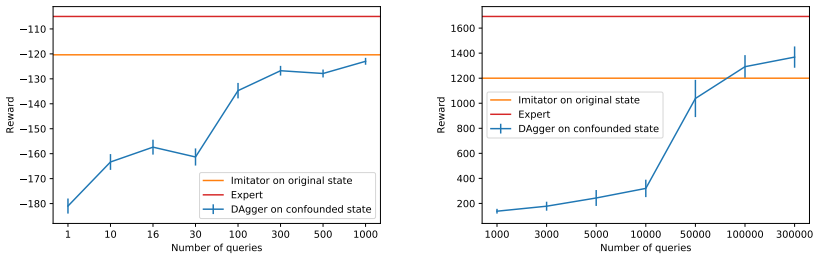


Figure 9: An expanded version of Fig 3 in the main paper, demonstrating diagnosis of the causal confusion problem in three settings. Here, the final reward, shown in Fig 3 is shown in the third column. Additionally, we also show the behavior cloning training loss (first column) and validation loss (second column) on trajectories generated by the expert. The x-axis for all plots is the number of training examples used to train the behavior cloning policy.

In Fig 9 we show the causal confusion in several environments. We observe that while training and validation losses for behavior cloning are frequently near-zero for both the original and confounded policy, the confounded policy consistently yields significantly lower reward when deployed in the environment. This confirms the causal confusion problem.

B Results of DAgger



(a) MountainCar (b) Hopper

Figure 10: DAgger results trained on the confounded state.

The results in Fig 10 show that DAgger requires hundreds of samples before reaching rewards comparable to the rewards achieved by a non-DAgger imitator trained on the original state.

C Intervention posterior inference as reinforcement learning

Given a method of evaluating the likelihood $p(\mathcal{O}|G)$ of a certain graph G to be optimal and a prior $p_0(G)$, we wish to infer the posterior $p(G|\mathcal{O})$. The number of graphs is finite, so we can compute this posterior exactly. However, there may be very many graphs, so that impractically many likelihood evaluations are necessary. Only noisy samples from the likelihood can be obtained, as in the case of intervention through policy execution, where the reward is noisy, this problem is exacerbated.

If on the other hand, a certain structure on the policy is assumed, the sample efficiently can be drastically improved, even though policy can no longer be exactly inferred. This can be done in the framework of Variational Inference. For a certain variational family, we wish to find, for some temperature τ:

$$\pi(G) = \arg\min_{\pi(G)} D_{KL}(\pi(G)||p(\mathcal{O}|G)) = \arg\min_{\pi(G)} \mathbb{E}_\pi \left[\log p(\mathcal{O}|G) + \log p_0(G)\right] + \tau\mathcal{H}_\pi(G) \quad (4)$$

The variational family we assume is the family of independent distributions:

$$\pi(G) = \prod_i \pi_i(G_i) = \prod_i \text{Bernoulli}(G_i|\sigma(w_i/\tau)) \quad (5)$$

Eq 4 can be interpreted as a 1 step entropy-regularized MDP with reward $\tilde{r} = \log p(\mathcal{O}|G) + \log p_0(G)$ [21]. It can be optimized through a policy gradient, but this would require many likelihood evaluations. More efficient is to use a value based method. The independence assumption translates in a linear Q function: $Q(G) = \langle w, G\rangle + b$, which can be simply learned by linear regression on off-policy pairs $(G, \tilde{r})$. In Soft Q-Learning [11] it is shown that the policy that maximizes Eq 4 is $\pi(G) \propto \exp Q(G)/\tau$, which can be shown to coincide in our case with Eq 5:

$$
\begin{aligned}
\pi(G) &= \frac{\exp(\langle w, G\rangle + b)/\tau}{\sum_{G'} \exp(\langle w, G'\rangle + b)/\tau} \\
&= \frac{\exp\langle w, G\rangle/\tau}{\sum_{G'} \exp(\langle w, G'\rangle/\tau} \\
&= \frac{\prod_i \exp(w_i G_i/\tau)}{\sum_{G'} \prod_i \exp(w_i G'_i/\tau)} \\
&= \frac{\prod_i \exp(w_i G_i/\tau)}{\prod_i \sum_{G'_i} \exp(w_i G'_i/\tau)} \\
&= \prod_i \frac{\exp(w_i G_i/\tau)}{\sum_{G'_i} \exp(w_i G'_i/\tau)} \\
&= \prod_i \frac{\exp(w_i G_i/\tau)}{1 + \exp w_i/\tau} = \prod_i \text{Bernoulli}(G_i|\sigma(w_i/\tau))
\end{aligned}
$$

where in the fourth line we used the identity $\sum_{G_1,G_2,\ldots} \prod_i \to \prod_i \sum_{G_i}$, which is commonly used in statistical physics.