Causal Reasoning from Meta-reinforcement Learning

Ishita Dasgupta[*1,4], **Jane Wang**[1],
Silvia Chiappa[1], **Jovana Mitrovic**[1], **Pedro Ortega**[1],
David Raposo[1], **Edward Hughes**[1], **Peter Battaglia**[1],
Matthew Botvinick[1,3], **Zeb Kurth-Nelson**[1,2]

[1]DeepMind, UK
[2]MPS-UCL Centre for Computational Psychiatry, UCL, UK
[3]Gatsby Computational Neuroscience Unit, UCL, UK
[4]Department of Physics and Center for Brain Science, Harvard University, USA

Abstract

Discovering and exploiting the causal structure in the environment is a crucial challenge for intelligent agents. Here we explore whether causal reasoning can emerge via meta-reinforcement learning. We train a recurrent network with model-free reinforcement learning to solve a range of problems that each contain causal structure. We find that the trained agent can perform causal reasoning in novel situations in order to obtain rewards. The agent can select informative interventions, draw causal inferences from observational data, and make counterfactual predictions. Although established formal causal reasoning algorithms also exist, in this paper we show that such reasoning can arise from model-free reinforcement learning, and suggest that causal reasoning in complex settings may benefit from the more end-to-end learning-based approaches presented here. This work also offers new strategies for structured exploration in reinforcement learning, by providing agents with the ability to perform—and interpret—experiments.

1. Introduction

Many machine learning algorithms are rooted in discovering patterns of correlation in data. While this has been sufficient to excel in several areas (Krizhevsky et al., 2012; Cho et al., 2014), sometimes the problems we are interested in are intrinsically causal. Answering questions such as "Does smoking cause cancer?" or "Was this person denied a job due to racial discrimination?" or "Did this marketing campaign cause sales to go up?" require an ability to reason about causes and effects. Causal reasoning may be an essential component of natural intelligence and is present in human babies, rats, and even birds (Leslie, 1982; Gopnik et al., 2001; 2004; Blaisdell et al., 2006; Lagnado et al., 2013).

There is a rich literature on formal approaches for defining and performing causal reasoning (Pearl, 2000; Spirtes et al., 2000; Dawid, 2007; Pearl et al., 2016). We investigate whether such reasoning can be achieved by meta-learning. The approach of meta-learning is to learn the learning (or inference/estimation) procedure itself, directly from data. Analogous (Grant et al., 2018) models that learn causal structure directly from the environment, rather than having a pre-conceived formal theory, have also been implicated in human intelligence (Goodman et al., 2011).

We specifically adopt the "meta-reinforcement learning" method introduced previously (Duan et al., 2016; Wang et al., 2016), in which a recurrent neural network (RNN)-based agent is trained with model-free reinforcement learning (RL). Through training on a large family of structured tasks, the RNN becomes a learning algorithm which generalizes to new tasks drawn from a similar distribution. In our case, we train on a distribution of tasks that are each underpinned by a different causal structure. We focus on abstract tasks that best isolate the question of interest: whether meta-learning can produce an agent capable of causal reasoning, when no notion of causality is explicitly given to the agent.

Meta-learning offers advantages of scalability by amortizing computations and, by learning end-to-end, the algorithm has the potential to find the internal representations of causal structure best suited for the types of causal inference required (Andrychowicz et al., 2016; Wang et al., 2016; Finn et al., 2017). We chose to focus on the RL approach because we are interested in agents that can learn about causes and effects not only from passive observations but also from active interactions with the environment (Hyttinen et al., 2013; Shanmugam et al., 2015).

*Corresponding author: ishitadasgupta@g.harvard.edu

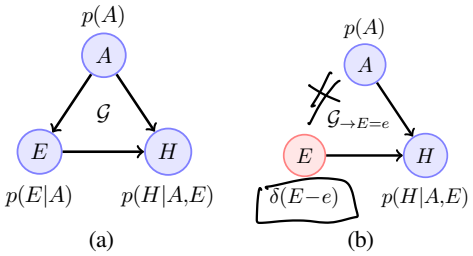

Figure 1. (a): A CBN $\mathcal{G}$ with a confounder for the effect of exercise (E) on heath (H) given by age (A). (b): Intervened CBN $\mathcal{G}_{\to E=e}$ resulting from modifying $\mathcal{G}$ by replacing $p(E|A)$ with a delta distribution $\delta(E-e)$ and leaving the remaining conditional distributions $p(H|E,A)$ and $p(A)$ unaltered.

2. Problem Specification and Approach

We examine three distinct data settings – *observational*, *interventional*, and *counterfactual* – which test different kinds of reasoning.

- In the observational setting (Experiment 1), the agent can only obtain passive observations from the environment. This type of data allows an agent to infer correlations (*associative reasoning*) and, depending on the structure of the environment, causal effects (*cause-effect reasoning*).
- In the interventional setting (Experiment 2), the agent can act in the environment by setting the values of some variables and observing the consequences on other variables. This type of data facilitates estimating causal effects.
- In the counterfactual setting (Experiment 3), the agent first has an opportunity to learn about the causal structure of the environment through interventions. At the last step of the episode, it must answer a counterfactual question of the form "What *would have* happened if a different intervention had been made in the previous timestep?".

Next we formalize these three settings and the patterns of reasoning possible in each, using the graphical model framework (Pearl, 2000; Spirtes et al., 2000; Dawid, 2007). Random variables will be denoted by capital letters (e.g., E) and their values by small letters (e.g., e).

2.1. Causal Reasoning

Causal relationships among random variables can be expressed using *causal Bayesian networks* (CBNs) (see the Supplementary Material). A CBN is a directed acyclic graphical model that captures both *independence* and *causal* relations. Each node X_i corresponds to a random variable, and the joint distribution $p(X_1,\dots,X_N)$ is given by the product of conditional distributions of each node X_i given its parent nodes $\text{pa}(X_i)$, i.e. $p(X_{1:N}\equiv X_1,\dots,X_N)=\prod_{i=1}^{N} p(X_i|\text{pa}(X_i))$.

Edges carry causal semantics: if there exists a directed path from X_i to X_j, then X_i is a *potential cause* of X_j. Directed paths are also called *causal paths*. The *causal effect* of X_i on X_j is the conditional distribution of X_j given X_i restricted to only causal paths.

An example of CBN $\mathcal{G}$ is given in Fig. 1a, where E represents hours of exercise in a week, H cardiac health, and A age. The causal effect of E on H is the conditional distribution restricted to the path $E\to H$, i.e. excluding the path $E\leftarrow A\to H$. The variable A is called a *confounder*, as it confounds the causal effect with non-causal statistical influence. Simply observing cardiac health conditioning on exercise level from $p(H|E)$ (associative reasoning) cannot answer if change in exercise levels cause changes in cardiac health (cause-effect reasoning), since there is always the possibility that correlation between the two is because of the common confounder of age.

Cause-effect Reasoning. The causal effect of $E=e$ can be seen as the conditional distribution $p_{\to E=e}(H|E=e)$[1] on the *intervened* CBN $\mathcal{G}_{\to E=e}$ resulting from replacing $p(E|A)$ with a delta distribution $\delta(E-e)$ (thereby removing the link from A to E) and leaving the remaining conditional distributions $p(H|E,A)$ and $p(A)$ unaltered (Fig. 1b). The rules of do-calculus (Pearl, 2000; Pearl et al., 2016) tell us how to compute $p_{\to E=e}(H|E=e)$ using observations from $\mathcal{G}$. In this case $p_{\to E=e}(H|E=e)=\sum_A p(H|E=e,A)p(A)$[2]. Therefore, do-calculus enables us to reason in the intervened graph $\mathcal{G}_{\to E=e}$ even if our observations are from $\mathcal{G}$. This is the scenario captured by our observational data setting outlined above.

Such inferences are always possible if the confounders are observed, but in the presence of unobserved confounders, for many CBN structures the only way to compute causal effects is by collecting observations directly from the intervened graph, e.g. from $\mathcal{G}_{\to E=e}$ by fixing the value of the variable $E=e$ and observing the remaining variables—we call this process performing an actual intervention in the environment. In our interventional data setting, outlined above, the agent has access to such interventions.

Counterfactual Reasoning. Cause-effect reasoning can be used to correctly answer predictive questions of the type "Does exercising improve cardiac health?" by accounting for causal structure and confounding. However, it cannot answer retrospective questions about what *would have* happened. For example, given an individual i who has died of a heart attack, this method would not be able to answer questions of the type "What would the cardiac health of this individual have been had she done more exercise?". This type of question requires reasoning about a counterfactual world (that did not happen). To do this, we can first use the observations from the factual world and knowledge about the CBN to get an estimate of

[1] In the causality literature, this distribution would most often be indicated with $p(H|\text{do}(E=e))$. We prefer to use $p_{\to E=e}(H|E=e)$ to highlight that intervening on E results in changing the original distribution p, by structurally altering the CBN.

[2] Notice that conditioning on $E=e$ would instead give $p(H|E=e)=\sum_A p(H|E=e,A)p(A|E=e)$.

the specific latent randomness in the makeup of individual i (for example information about this specific patient's blood pressure and other variables as inferred by her having had a heart attack). Then, we can use this estimate to compute cardiac health under intervention on exercise. This procedure is explained in more detail in the Supplementary Material.

2.2. Meta-learning

Meta-learning refers to a broad range of approaches in which aspects of the learning algorithm itself are learned from the data. Many individual components of deep learning algorithms have been successfully meta-learned, including the optimizer (Andrychowicz et al., 2016), initial parameter settings (Finn et al., 2017), a metric space (Vinyals et al., 2016), and use of external memory (Santoro et al., 2016).

Following the approach of (Duan et al., 2016; Wang et al., 2016), we parameterize the entire learning algorithm as a recurrent neural network (RNN), and we train the weights of the RNN with model-free reinforcement learning (RL). The RNN is trained on a broad distribution of problems which each require learning. When trained in this way, the RNN is able to implement a learning algorithm capable of efficiently solving novel learning problems in or near the training distribution (see the Supplementary Material for further details).

Learning the weights of the RNN by model-free RL can be thought of as the "outer loop" of learning. The outer loop shapes the weights of the RNN into an "inner loop" learning algorithm. This inner loop algorithm plays out in the activation dynamics of the RNN and can continue learning even when the weights of the network are frozen. The inner loop algorithm can also have very different properties from the outer loop algorithm used to train it. For example, in previous work this approach was used to negotiate the exploration-exploitation tradeoff in multi-armed bandits (Duan et al., 2016) and learn algorithms which dynamically adjust their own learning rates (Wang et al., 2016; 2018). In the present work we explore the possibility of obtaining a causally-aware inner-loop learning algorithm.

3. Task Setup and Agent Architecture

In our experiments, in each episode the agent interacted with a different CBN $\mathcal{G}$, defined over a set of N variables. The structure of $\mathcal{G}$ was drawn randomly from the space of possible acyclic graphs under the constraints given in the next subsection.

Each episode consisted of T steps, which were divided into two phases: an *information phase* and a *quiz phase*. The information phase, corresponding to the first $T-1$ steps, allowed the agent to collect information by interacting with or passively observing samples from $\mathcal{G}$. The agent could potentially use this information to infer the connectivity and weights of $\mathcal{G}$. The quiz phase, corresponding to the final step T, required the agent to exploit the causal knowledge it collected in the information phase, to select the node with the highest value under a random external intervention.

Causal Graphs, Observations, and Actions

We generated all graphs on $N=5$ nodes, with edges only in the upper triangular of the adjacency matrix (this guarantees that all the graphs obtained are acyclic). Edge weights w_{ji} were uniformly sampled from $\{-1,0,1\}$. This yielded $3^{N(N-1)/2}=59049$ unique graphs. These can be divided into equivalence classes: sets of graphs that are structurally identical but differ in the permutation of the node labels. Our held-out test set consisted of 12 random graphs plus all other graphs in the equivalence classes of these 12. Thus, all graphs in the test set had never been seen (and no equivalent graphs had been seen) during training. There were 408 total graphs in the test set.

Each node, $X_i \in \mathbb{R}$, was a Gaussian random variable. Parentless nodes had distribution $\mathcal{N}(\mu=0.0,\sigma=0.1)$. A node X_i with parents $\text{pa}(X_i)$ had conditional distribution $p(X_i|\text{pa}(X_i)) = \mathcal{N}(\mu=\sum_j w_{ji}X_j,\sigma=0.1)$, where $X_j \in \text{pa}(X_i)$[3].

A root node of $\mathcal{G}$ was always hidden (unobservable), to allow for the presence of an unobserved confounder and the agent could therefore only observe the values of the other 4 nodes. The concatenated values of the nodes, v_t, and and a one-hot vector indicating the external intervention during the quiz phase, m_t, (explained below) formed the observation vector provided to the agent at step t, $o_t=[v_t,m_t]$.

In both phases, on each step t, the agent could choose to take 1 of $2(N-1)$ actions, the first $N-1$ of which were *information actions*, and the second of which were *quiz actions*. Both information and quiz actions were associated with selecting the $N-1$ observable nodes, but could only be legally used in the appropriate phase of the task. If used in the wrong phase, a penalty was applied and the action produced no effect.

Information Phase. In the information phase, an information action $a_t=i$ caused an intervention on the i-th node, setting the value of $X_{a_t}=X_i=5$ (the value 5 was chosen to be outside the likely range of sampled observations, to facilitate learning the causal graph). The node values v_t were then obtained by sampling from $p_{\rightarrow X_i=5}(X_{1:N\setminus i}|X_i=5)$ (where $X_{1:N\setminus i}$ indicates the set of all nodes except X_i), namely from the intervened CBN $\mathcal{G}_{\rightarrow X_{a_t}=5}$ resulting from removing the incoming edges to X_{a_t} from $\mathcal{G}$, and using the intervened value $X_{a_t}=5$ for conditioning its children's values. If a quiz action was chosen during the information phase, it was ignored; namely, the node values were sampled from $\mathcal{G}$ as if no intervention had been made, and the agent was given a penalty of $r_t=-10$ in order to encourage it to take quiz actions during the quiz phase. There was no other reward during the information phase.

[3] We also tested graphs with non-linear causal effects, and larger graphs of size N = 6 (see the Supplementary Material).

The default length an episode for fixed to be $T = N = 5$, resulting in this phase being fixed to a length of $T-1=4$. This was because in the noise-free limit, a minimum of $N-1=4$ interventions, one on each observable node, are required in general to resolve the causal structure and score perfectly on the test phase.

Quiz Phase. In the quiz phase, one non-hidden node X_j was selected at random to be intervened on by the environment. Its value was set to -5. We chose -5 to disallow the agent from memorizing the results of interventions in the information phase (which were fixed to $+5$) in order to perform well on the quiz phase. The agent was informed which node received this external intervention via the one-hot vector m_t as part of the observation from the the final pre-quiz phase timestep, $T-1$. For steps $t<T-1$, m_t was the zero vector. The agent's reward on this step was the sampled value of the node it selected during the quiz phase. In other words, $r_T = X_i = X_{a_T-(N-1)}$ if the action selected was a quiz action (otherwise, the agent was given a penalty of $r_T=-10$).

Active vs Random Agents. Our agents had to perform two distinct tasks during the information phase: a) actively choose which nodes to set values on, and b) infer the CBN from its observations. We refer to this setup as the "active" condition. To better understand the role of (a), we include comparisons with a baseline agent in the "random" condition, whose policy is to choose randomly which observable node it will set values for, at each step of the information phase.

Two Kinds of Learning. The "inner loop" of learning (see Section 2.2) occurs within each episode where the agent is learning from the evidence it gathers during the information phase in order to perform well in the quiz phase. The same agent then enters a new episode, where it has to repeat the task on a different CBN. Test performance is reported on CBNs that the agent has never previously seen, after all the weights of the RNN have been fixed. Hence, the only transfer from training to test (or the "outer loop" of learning) is the ability to discover causal dependencies based on observations in the information phase, and to perform causal inference in the quiz phase.

Agent Architecture and Training

We used a long short-term memory (LSTM) network (Hochreiter and Schmidhuber, 1997) (with 192 hidden units) that, at each time-step t, receives a concatenated vector containing $[o_t, a_{t-1}, r_{t-1}]$ as input, where o_t is the observation[4], a_{t-1} is the previous action (as a one-hot vector) and r_{t-1} the reward (as a single real-value)[5].

[4]'Observation' o_t refers to the reinforcement learning term, i.e. the input from the environment to the agent. This is distinct from observations in the causal sense (referred to as observational data) i.e. samples from a causal structure where no interventions have been carried out.

[5]These are both set to zero for the first step in an episode.

The outputs, calculated as linear projections of the LSTM's hidden state, are a set of policy logits (with dimensionality equal to the number of available actions), plus a scalar baseline. The policy logits are transformed by a softmax function, and then sampled to give a selected action.

Learning was by *asynchronous advantage actor-critic* (Mnih et al., 2016). In this framework, the loss function consists of three terms – the policy gradient, the baseline cost and an entropy cost. The baseline cost was weighted by 0.05 relative to the policy gradient cost. The weighting of the entropy cost was annealed over the course of training from 0.25 to 0. Optimization was done by RMSProp with $\epsilon=10^{-5}$, momentum = 0.9 and decay = 0.95. Learning rate was annealed from 9×10^{-6} to 0, with a discount of 0.93. Unless otherwise stated, training was done for 1×10^7 steps using batched environments with a batch size of 1024.

For all experiments, after training, the agent was tested with the learning rate set to zero, on a held-out test set.

4. Experiments

Our three experiments (observational, interventional, and counterfactual data settings) differed in the properties of the v_t that was observed by the agent during the information phase, and thereby limited the extent of causal reasoning possible within each data setting. Our measure of performance is the reward earned in the quiz phase for held-out CBNs. Choosing a random node in the quiz phase results in a reward of $-5/4=-1.25$, since one node (the externally intervened node) always has value -5 and the others have on average 0 value. By learning to simply avoid the externally intervened node, the agent can earn on average 0 reward. Consistently picking the node with the highest value in the quiz phase requires the agent to perform causal reasoning. For each agent, we took the average reward earned across 1632 episodes (408 held-out test CBNs, with 4 possible external interventions). We trained 8 copies of each agent and reported the average reward earned by these, with error bars showing 95% confidence intervals. The p values based on the appropriate t-test are provided in cases where the compared values are close.

4.1. Experiment 1: Observational Setting

In Experiment 1, the agent could neither intervene to set the value of variables in the environment, nor observe any external interventions. In other words, it only received observations from $\mathcal{G}$, not $\mathcal{G}_{\rightarrow X_j}$ (where X_j is a node that has been intervened on). This limits the extent of causal inference possible. In this experiment, we tested five agents, four of which were learned: "Observational", "Long Observational", "Active Conditional", "Random Conditional", and the "Optimal Associative Baseline" (not learned). We also ran two other standard RL baselines—see the Supplementary Material for details.

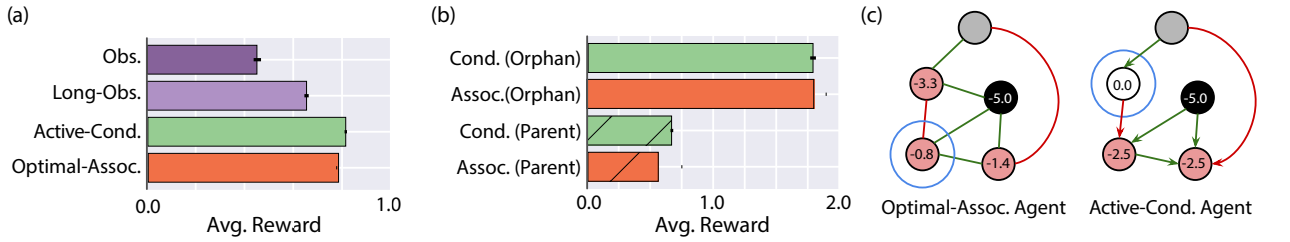

Figure 2. Experiment 1. Agents do cause-effect reasoning from observational data. a) Average reward earned by the agents tested in this experiment. See main text for details. b) Performance split by the presence or absence of at least one parent (Parent and Orphan respectively) on the externally intervened node. c) Quiz phase for a test CBN. Green (red) edges indicate a weight of $+1$ (-1). Black represents the intervened node, green (red) nodes indicate a positive (negative) value at that node, white indicates a zero value. The blue circles indicate the agent's choice. Left panel: The undirected version of $\mathcal{G}$ and the nodes taking the mean values prescribed by $p(X_{1:N\setminus j}|X_j=-5)$, including backward inference to the intervened node's parent. We see that the Optimal Associative Baseline's choice is consistent with maximizing these (incorrect) node values. Right panel: $\mathcal{G}_{\to X_j=-5}$ and the nodes taking the mean values prescribed by $p_{\to X_j=-5}(X_{1:N\setminus j}|X_j=-5)$. We see that the Active-Conditional Agent's choice is consistent with maximizing these (correct) node values.

Observational Agents: In the information phase, the actions of the agent were ignored[6], and the observational agent always received the values of the observable nodes as sampled from the joint distribution associated with $\mathcal{G}$. In addition to the default $T=5$ episode length, we also trained this agent with $4\times$ longer episode length (Long Observational Agent), to measure performance increase with more observational data.

Conditional Agents: The information phase actions corresponded to observing a world in which the selected node X_j is equal to $X_j=5$, and the remaining nodes are sampled from the conditional distribution $p(X_{1:N\setminus j}|X_j=5)$. This differs from intervening on the variable X_j by setting it to the value $X_j=5$, since here we take a conditional sample from $\mathcal{G}$ rather than from $\mathcal{G}_{\to X_j=5}$ (or from $p_{\to X_j=5}(X_{1:N\setminus j}|X_j=5)$), and inference about the corresponding node's parents is possible. Therefore, this agent still has access to only observational data, as with the observational agents. However, on average it receives more diagnostic information about the relation between the random variables in $\mathcal{G}$, since it can observe samples where a node takes a value far outside the likely range of sampled observations. We run active and random versions of this agent as described in Section 3.

Optimal Associative Baseline: This baseline receives the true joint distribution $p(X_{1:N})$ implied by the CBN in that episode and therefore has full knowledge of the correlation structure of the environment[7]. It can therefore do exact associative reasoning of the form $p(X_j|X_i=x)$, but cannot do any cause-effect reasoning of the form $p_{\to X_i=x}(X_j|X_i=x)$. In the quiz phase, this baseline chooses the node that has the maximum value according to the true $p(X_j|X_i=x)$ in that episode, where X_i is the node externally intervened upon, and $x=-5$. This is the best possible performance using only associative reasoning.

[6]These agents also did not receive the out-of-phase action penalties during the information phase since their actions are totally ignored.

[7]Notice that the agent does not know the graphical structure, i.e. it does not know which nodes are parents of which other nodes.

Results

We focus on two questions in this experiment.

(i) Most centrally, do the agents learn to perform cause-effect reasoning using observational data? The Optimal Associative Baseline tracks the greatest reward that can be achieved using only knowledge of correlations – without causal knowledge. Compared to this baseline, the Active-Conditional Agent (which is allowed to select highly informative observations) earns significantly more reward ($p=6\times10^{-5}$, Fig. 2a). To better understand why the agent outperforms the associative baseline, we divided episodes according to whether or not the node that was intervened on in the quiz phase has a parent (Fig. 2b). If the intervened node X_j has no parents, then $\mathcal{G}=\mathcal{G}_{\to X_j}$, and cause-effect reasoning has no advantage over associative reasoning. Indeed, the Active-Conditional Agent performs better than the Optimal Associative Baseline only when the intervened node has parents (hatched bars in Fig. 2b). We also show the quiz phase for an example test CBN in Fig. 2c, where the Optimal Associative Baseline chooses according to the node values predicted by $\mathcal{G}$, whereas the Active-Conditional Agent chooses according the node values predicted by $\mathcal{G}_{\to X_j}$.

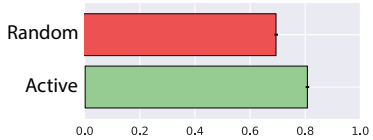

Figure 4. Active and Random Conditional Agents

These analyses allow us to conclude that the agent can perform cause-effect reasoning, using observational data alone – analogous to the formal use of do-calculus. (ii) Do the agents learn to select useful observations? We find that the Active-Conditional Agent's performance is significantly greater than the Random-Conditional Agent (Fig. 4). This indicates that the agent has learned to choose useful data to observe.

For completeness we also included agents that received unconditional observations from $\mathcal{G}$, i.e. the Observational Agents ('Obs' and 'Long-Obs' in Fig. 2a). As expected, these agents

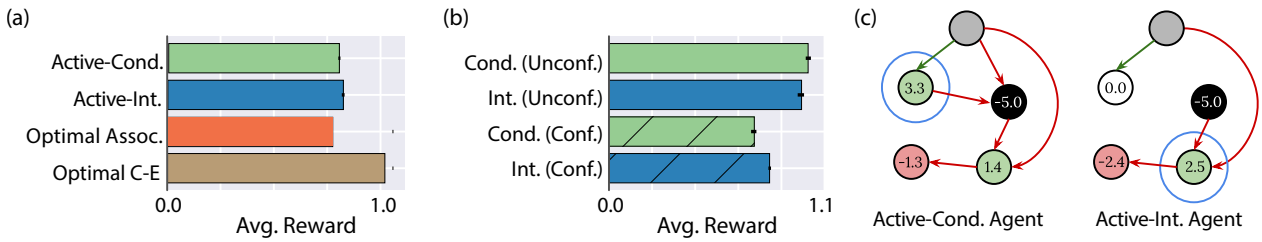

Figure 3. Experiment 2. Agents do cause-effect reasoning from interventional data. a) Average reward earned by the agents tested in this experiment. See main text for details. b) Performance split by the presence or absence of unobserved confounders (abbreviated as Conf. and Unconf. respectively) on the externally intervened node. c) Quiz phase for a test CBN. See Fig. 2 for a legend. Here, the left panel shows the full $\mathcal{G}$ and the nodes taking the mean values prescribed by $p(X_{1:N\setminus j}|X_j=-5)$. We see that the Active-Cond Agent's choice is consistent with choosing based on these (incorrect) node values. The right panel shows $\mathcal{G}_{\to X_j=-5}$ and the nodes taking the mean values prescribed by $p_{\to X_j=-5}(X_{1:N\setminus j}|X_j=-5)$. We see that the Active-Int. Agent's choice is consistent with maximizing on these (correct) node value.

performed worse than the Active-Conditional Agent, because they received less diagnostic information during the information phase. However, they were still able to acquire some information from unconditional samples, and also made use of the increased information available from longer episodes.

4.2. Experiment 2: Interventional Setting

In Experiment 2, the agent receives interventional data in the information phase – it can choose to intervene on any observable node, X_j, and observe a sample from the resulting graph $\mathcal{G}_{\to X_j}$. As discussed in Section 2.1, access to interventional data permits cause-effect reasoning even in the presence of unobserved confounders, a feat which is in general impossible with access only to observational data. In this experiment, we test three new agents, two of which were learned: "Active Interventional", "Random Interventional", and "Optimal Cause-Effect Baseline" (not learned).

Interventional Agents: The information phase actions correspond to performing an intervention on the selected node X_j and sampling from $\mathcal{G}_{\to X_j}$ (see Section 3 for details). We run active and random versions of this agent as described in Section 3.

Optimal Cause-Effect Baseline: This baseline receives the true CBN, $\mathcal{G}$. In the quiz phase, it chooses the node that has the maximum value according to $\mathcal{G}_{\to X_j}$, where X_j is the node externally intervened upon. This is the maximum possible performance on this task.

Results

We focus on two questions in this experiment.

(i) Do our agents learn to perform cause-effect reasoning from interventional data? The Active-Interventional Agent's performance is marginally better than the Active-Conditional Agent ($p = 0.06$, Fig. 3a). To better highlight the crucial role of interventional data in doing cause-effect reasoning, we compare the agent performances split by whether the node that was intervened on in the quiz phase of the episode had unobserved confounders with other variables in the graph (Fig. 3b). In confounded cases, as described in Section 2.1, cause-effect reasoning is impossible with only observational data. We see that the performance of the Active-Interventional Agent is significantly higher ($p = 10^{-5}$) than that of the Active-Conditional Agent in the confounded cases. This indicates that the Active-Interventional Agent (that had access to interventional data) is able to perform additional cause-effect reasoning in the presence of confounders that the Active-Conditional Agent (that had access to only observational data) cannot do. This is highlighted by Fig. 3c, which shows the quiz phase for an example CBN, where the Active-Conditional Agent is unable to resolve the unobserved confounder, but the Active-Interventional Agent is able to.

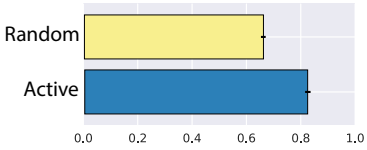

Figure 5. Active and Random Interventional Agents

(ii) Do our agents learn to make useful interventions? The Active-Interventional Agent's performance is significantly greater than the Random-Interventional Agent's (Fig. 5). This indicates that when the agent is allowed to choose its actions, it makes tailored, non-random choices about the interventions it makes and the data it wants to observe.

4.3. Experiment 3: Counterfactual Setting

In Experiment 3, the agent was again allowed to make interventions as in Experiment 2, but in this case the quiz phase task entailed answering a counterfactual question. We explain here what a counterfactual question in this domain looks like. Assume $X_i = \sum_j w_{ji} X_j + \epsilon_i$ where ϵ_i is distributed as $\mathcal{N}(0.0, 0.1)$ (giving the conditional distribution $p(X_i|\text{pa}(X_i)) = \mathcal{N}(\sum_j w_{ji} X_j, 0.1)$ as described in Section 3). After observing the nodes $X_{2:N}$ (X_1 is hidden) in the CBN in one sample, we can infer this latent randomness ϵ_i for each observable node X_i (i.e. *abduction* as described in the Sup-

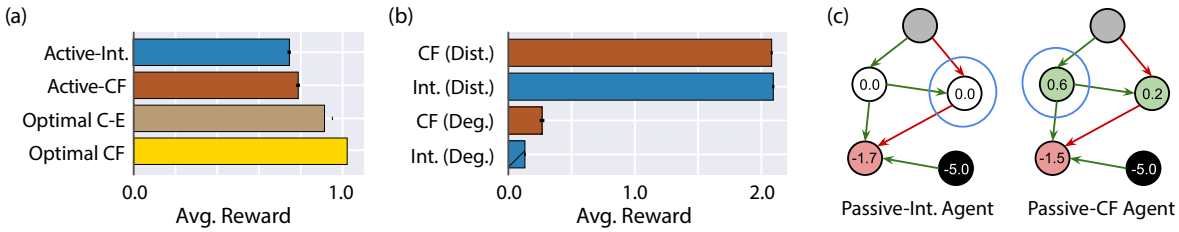

Figure 6. Experiment 3. Agents do counterfactual reasoning. a) Average reward earned by the agents tested in this experiment. See main text for details. b) Performance split by if the maximum node value in the quiz phase is degenerate (Deg.) or distinct (Dist.). c) Quiz phase for an example test-CBN. See Fig. 2 for a legend. Here, the left panel shows $\mathcal{G}_{\rightarrow X_j=-5}$ and the nodes taking the mean values prescribed by $p_{\rightarrow X_j=-5}(X_{1:N\setminus j}|X_j=-5)$. We see that the Active-Int. Agent's choice is consistent with maximizing on these node values, where it makes a random choice between two nodes with the same value. The right panel panel shows $\mathcal{G}_{\rightarrow X_j=-5}$ and the nodes taking the exact values prescribed by the means of $p_{\rightarrow X_j=-5}(X_{1:N\setminus j}|X_j=-5)$, combined with the specific randomness inferred from the previous time step. As a result of accounting for the randomness, the two previously degenerate maximum values are now distinct. We see that the Active-CF. agent's choice is consistent with maximizing on these node values.

plementary Material) and answer counterfactual questions like "What would the values of the nodes be, had X_i instead taken on a different value than what we observed?", for any of the observable nodes X_i. We test three new agents, two of which are learned: "Active Counterfactual", "Random Counterfactual", and "Optimal Counterfactual Baseline" (not learned).

Counterfactual Agents: This agent is exactly analogous to the Interventional agent, with the addition that the latent randomness in the last information phase step $t=T-1$ (where say some $X_p=+5$), is stored and the same randomness is used in the quiz phase step $t=T$ (where say some $X_f=-5$). While the question our agents have had to answer correctly so far in order to maximize their reward in the quiz phase was "Which of the nodes $X_{2:N}$ will have the highest value when X_f is set to -5?", in this setting, we ask "Which of the nodes $X_{2:N}$ would have had the highest value in the last step of the information phase, if instead of having the intervention $X_p=+5$, we had the intervention $X_f=-5$?". We run active and random versions of this agent as described in Section 3.

Optimal Counterfactual Baseline: This baseline receives the true CBN and does exact abduction of the latent randomness based on observations from the penultimate step of the information phase, and combines this correctly with the appropriate interventional inference on the true CBN in the quiz phase.

Results

We focus on two key questions in this experiment.

(i) Do our agents learn to do counterfactual inference? The Active-Counterfactual Agent achieves higher reward than the Active-Interventional Agent ($p = 2\times10^{-5}$). To evaluate whether this difference results from the agent's use of abduction (see the Supplementary Material for details), we split the test set into two groups, depending on whether or not the decision for which node will have the highest value in the quiz phase is affected by the latent randomness, i.e. whether or not the node with the maximum value in the quiz phase changes if the noise is resampled. This is most prevalent in cases where the maximum expected reward is degenerate, i.e. where several nodes give the same maximum reward (denoted by hatched bars in Figure 6b). Here, agents with no access to the randomness have no basis for choosing one over the other, but different noise samples can give rise to significant differences in the actual values that these degenerate nodes have. We see indeed that there is no difference in the rewards received by the Active-Counterfactual and Active-Interventional Agents in the cases where the maximum values are distinct, however the Active-Counterfactual Agent significantly outperforms the Active-Interventional Agent in cases where there are degenerate maximum values.

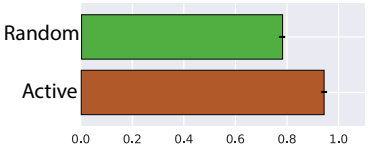

Figure 7. Active and Random Counterfactual Agents

(ii) Do our agents learn to make useful interventions in the service of a counterfactual task? The Active-Counterfactual Agent's performance is significantly greater than the Random-Counterfactual Agent's (Fig. 5). This indicates that when the agent is allowed to choose its actions, it makes tailored, non-random choices about the interventions it makes and the data it wants to observe – even in the service of a counterfactual objective.

5. Summary of Results

In this paper we used the meta-learning to train a recurrent network – using model-free reinforcement learning – to implement an algorithm capable of causal reasoning. Agents trained in this manner performed causal reasoning in three data settings: observational, interventional, and counterfactual. Crucially, our approach did not require explicit encoding of formal principles of causal inference. Rather, by optimizing an agent to perform a task that depended on causal structure, the agent learned im-

plicit strategies to generate and use different kinds of available data for causal reasoning, including drawing causal inferences from passive observation, actively intervening, and making counterfactual predictions, all on held out causal CBNs that the agents had never previously seen.

A consistent result in all three data settings was that our agents learned to perform good experiment design or *active learning*. That is, they learned a non-random data collection policy where they actively chose which nodes to intervene (or condition) on in the information phase, and thus could control the kinds of data they saw, leading to higher performance in the quiz phase than that from an agent with a random data collection policy. Below, we summarize the other keys results from each of the three experiments.

In Section 4.1 and Fig. 2, we showed that agents learned *to perform do-calculus*. In Fig. 2a we saw that, the trained agent with access to only observational data received more reward than the highest possible reward achievable without causal knowledge. We further observed in Fig. 2b that this performance increase occurred selectively in cases where do-calculus made a prediction distinguishable from the predictions based on correlations – i.e. where the externally intervened node had a parent, meaning that the intervention resulted in a different graph.

In Section 4.2 and Fig. 3, we showed that agents learned *to resolve unobserved confounders using interventions* (which is impossible with only observational data). In Fig. 3b we saw that agents with access to interventional data performed better than agents with access to only observational data only in cases where the intervened node shared an unobserved parent (a confounder) with other variables in the graph.

In Section 4.3 and Fig. 6, we showed that agents learned *to use counterfactuals*. In Fig. 6a we saw that agents with additional access to the specific randomness in the test phase performed better than agents with access to only interventional data. In Fig. 6b, we found that the increased performance was observed only in cases where the maximum mean value in the graph was degenerate, and optimal choice was affected by the latent randomness – i.e. where multiple nodes had the same value on average and the specific randomness could be used to distinguish their actual values in that specific case.

6. Discussion and Future Work

To our knowledge, this is the first direct demonstration that causal reasoning can arise out of model-free reinforcement learning. Our paper lays the groundwork for a meta-reinforcement learning approach to causal reasoning that potentially offers several advantages over formal methods for causal inference in complex real world settings.

First, traditional formal approaches usually decouple the problems of *causal induction* (inferring the structure of the underlying model from data) and *causal inference* (estimating causal effects based on a known model). Despite advances in both (Ortega and Stocker, 2015; Lattimore et al., 2016; Bramley et al., 2017; Forney et al., 2017; Sen et al., 2017; Parida et al., 2018), inducing models is expensive and typically requires simplifying assumptions. When induction and inference are decoupled, the assumptions used at the induction step are not fully optimized for the inference that will be performed downstream. By contrast, our model learns to perform induction and inference end-to-end, and can potentially find representations of causal structure best tuned for the required causal inferences. Meta-learning can sometimes even leverage structure in the problem domain that may be too complex to specify when inducing a model (Duan et al., 2016; Santoro et al., 2016; Wang et al., 2016), allowing more efficient and accurate causal reasoning than would be possible without representing and exploiting this structure.

Second, since both the induction and inference steps are costly, formal methods can be very slow at run-time when faced with a new query. Meta-learning shifts most of the compute burden from inference time to training time. This is advantageous when training time is ample but fast answers are needed at run-time.

Finally, by using an RL framework, our agents can learn to take actions that produce useful information—i.e. perform active learning. Our agents' active intervention policy performed significantly better than a random intervention policy, which demonstrates the promise of learning an experimentation policy end-to-end with the causal reasoning built on the resulting observations.

Our work focused on a simple domain because our aim was to test in the most direct way possible whether causal reasoning can emerge from meta-learning. Follow-up work should focus on scaling up our approach to larger environments, with more complex causal structure and a more diverse range of tasks. This opens up possibilities for agents that perform active experiments to support structured exploration in RL, and learning optimal experiment design in complex domains where large numbers of random interventions are prohibitive. The results here are a first step in this direction, obtained using relatively standard deep RL components – our approach will likely benefit from more advanced architectures (e.g. Hester et al., 2017; Hessel et al., 2018; Espeholt et al., 2018) that allow us to train on longer more complex episodes, as well as models which are more explicitly compositional (e.g. Andreas et al., 2016; Battaglia et al., 2018) or have richer semantics (e.g. Ganin et al., 2018) that can more explicitly leverage symmetries in the environment and improve generalization and training efficiency.

7. Acknowledgements

The authors would like to thank the following people for helpful discussions and comments: Neil Rabinowitz, Neil Bramley,

Tobias Gerstenberg, Andrea Tacchetti, Victor Bapst, Samuel Gershman.

References

J. Andreas, M. Rohrbach, T. Darrell, and D. Klein. Neural module networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 39–48, 2016.

M. Andrychowicz, M. Denil, S. Gomez, M. W. Hoffman, D. Pfau, T. Schaul, B. Shillingford, and N. De Freitas. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems*, pages 3981–3989, 2016.

P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

A. P. Blaisdell, K. Sawa, K. J. Leising, and M. R. Waldmann. Causal reasoning in rats. *Science*, 311(5763):1020–1022, 2006.

N. R. Bramley, P. Dayan, T. L. Griffiths, and D. A. Lagnado. Formalizing neuraths ship: Approximate algorithms for online causal learning. *Psychological review*, 124(3):301, 2017.

K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

P. Dawid. Fundamentals of statistical causality. Technical report, University Colledge London, 2007.

Y. Duan, J. Schulman, X. Chen, P. L. Bartlett, I. Sutskever, and P. Abbeel. rl^2: Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779*, 2016.

L. Espeholt, H. Soyer, R. Munos, K. Simonyan, V. Mnih, T. Ward, Y. Doron, V. Firoiu, T. Harley, I. Dunning, et al. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. *arXiv preprint arXiv:1802.01561*, 2018.

C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*, 2017.

A. Forney, J. Pearl, and E. Bareinboim. Counterfactual data-fusion for online reinforcement learners. In *International Conference on Machine Learning*, pages 1156–1164, 2017.

Y. Ganin, T. Kulkarni, I. Babuschkin, S. M. Eslami, and O. Vinyals. Synthesizing programs for images using reinforced adversarial learning. *arXiv preprint arXiv:1804.01118*, 2018.

N. D. Goodman, T. D. Ullman, and J. B. Tenenbaum. Learning a theory of causality. *Psychological review*, 118(1):110, 2011.

A. Gopnik, D. M. Sobel, L. E. Schulz, and C. Glymour. Causal learning mechanisms in very young children: two-, three-, and four-year-olds infer causal relations from patterns of variation and covariation. *Developmental psychology*, 37(5): 620, 2001.

A. Gopnik, C. Glymour, D. M. Sobel, L. E. Schulz, T. Kushnir, and D. Danks. A theory of causal learning in children: causal maps and bayes nets. *Psychological review*, 111(1):3, 2004.

E. Grant, C. Finn, S. Levine, T. Darrell, and T. Griffiths. Recasting gradient-based meta-learning as hierarchical bayes. *arXiv preprint arXiv:1801.08930*, 2018.

M. Hessel, H. Soyer, L. Espeholt, W. Czarnecki, S. Schmitt, and H. van Hasselt. Multi-task deep reinforcement learning with popart. *arXiv preprint arXiv:1809.04474*, 2018.

T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, G. Dulac-Arnold, et al. Deep q-learning from demonstrations. *arXiv preprint arXiv:1704.03732*, 2017.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

A. Hyttinen, F. Eberhardt, and P. O. Hoyer. Experiment selection for causal discovery. *The Journal of Machine Learning Research*, 14(1):3041–3071, 2013.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

D. A. Lagnado, T. Gerstenberg, and R. Zultan. Causal responsibility and counterfactuals. *Cognitive science*, 37(6): 1036–1073, 2013.

F. Lattimore, T. Lattimore, and M. D. Reid. Causal bandits: Learning good interventions via causal inference. In *Advances in Neural Information Processing Systems*, pages 1181–1189, 2016.

A. M. Leslie. The perception of causality in infants. *Perception*, 11(2):173–186, 1982.

V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu. Asynchronous methods for deep reinforcement learning. *CoRR*,

abs/1602.01783, 2016. URL http://arxiv.org/abs/1602.01783.

P. A. Ortega and D. D. Lee A. A. Stocker. Causal reasoning in a prediction task with hidden causes. *37th Annual Cognitive Science Society Meeting CogSci*, 2015.

P. K. Parida, T. Marwala, and S. Chakraverty. A multivariate additive noise model for complete causal discovery. *Neural Networks*, 103:44–54, 2018.

J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

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

A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap. Meta-learning with memory-augmented neural networks. In *International conference on machine learning*, pages 1842–1850, 2016.

R. Sen, K. Shanmugam, A. G. Dimakis, and S. Shakkottai. Identifying best interventions through online importance sampling. *arXiv preprint arXiv:1701.02789*, 2017.

K. Shanmugam, M. Kocaoglu, A. G. Dimakis, and S. Vishwanath. Learning causal graphs with small interventions. In *Advances in Neural Information Processing Systems*, pages 3195–3203, 2015.

P. Spirtes, C. N. Glymour, R. Scheines, D. Heckerman, C. Meek, G. Cooper, and T. Richardson. *Causation, prediction, and search*. MIT press, 2000.

O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, pages 3630–3638, 2016.

J. X. Wang, Z. Kurth-Nelson, D. Tirumala, H. Soyer, J. Z. Leibo, R. Munos, C. Blundell, D. Kumaran, and M. Botvinick. Learning to reinforcement learn. *CoRR*, abs/1611.05763, 2016. URL http://arxiv.org/abs/1611.05763.

J. X. Wang, Z. Kurth-Nelson, D. Kumaran, D. Tirumala, H. Soyer, J. Z. Leibo, D. Hassabis, and M. Botvinick. Prefrontal cortex as a meta-reinforcement learning system. *Nature neuroscience*, 21(6):860, 2018.

Supplementary to Causal Reasoning from Meta-reinforcement Learning

1. Additional Baselines

Figure 1: Reward distribution

We can also compare the performance of these agents to two standard model-free RL baselines. The Q-total Agent learns a Q-value for each action across all steps for all the episodes. The Q-episode Agent learns a Q-value for each action conditioned on the input at each time step $[o_t, a_{t-1}, r_{t-1}]$, but with no LSTM memory to store previous actions and observations. Since the relationship between action and reward is random between episodes, Q-total was equivalent to selecting actions randomly, resulting in a considerably negative reward (-1.247 ± 2.940). The Q-episode agent essentially makes sure to not choose the arm that is indicated by m_t to be the external intervention (which is assured to be equal to -5), and essentially chooses randomly otherwise, giving a reward close to 0 (0.080 ± 2.077).

2. Formalism for Memory-based Meta-learning

Consider a distribution $\mathcal{D}$ over Markov Decision Processes (MDPs). We train an agent with memory (in our case an RNN-based agent) on this distribution. In each episode, we sample a task $m \sim \mathcal{D}$. At each step t within an episode, the agent sees an observation o_t, executes an action a_t, and receives a reward r_t. Both a_{t-1} and r_{t-1} are given as additional inputs to the network. Thus, via the recurrence of the network, each action is a function of the entire trajectory $\mathcal{H}_t = \{o_0, a_0, r_0, \ldots, o_{t-1}, a_{t-1}, r_{t-1}, o_t\}$ of the episode. Because this function is parameterized by the neural network, its complexity is limited only by the size of the network.

3. Abduction-Action-Prediction Method for Counterfactual Reasoning

Pearl et al. (2016)'s "abduction-action-prediction" method prescribes one way to answer counterfactual queries of the type "What would the cardiac health of individual i have been had she done more exercise?", by estimating the specific latent randomness in the unobserved makeup of the individual and by transferring it to the counterfactual world. Assume, for example, the following model for $\mathcal{G}$ of Section 2.1: $E = w_{AE}A + \eta$, $H = w_{AH}A + w_{EH}E + \epsilon$, where the weights w_{ij} represent the known causal effects in $\mathcal{G}$ and ϵ and η are terms of (e.g.) Gaussian noise that represent the latent randomness in the makeup of each individual[1]. Suppose that for individual i we observe: $A = a^i$, $E = e^i$, $H = h^i$. We can answer the counterfactual question of "What if individual i had done more exercise, i.e. $E = e'$, instead?" by: a) *Abduction:* estimate the individual's specific makeup with $\epsilon^i = h^i - w_{AH}a^i - w_{EH}e^i$, b) *Action:* set E to more exercise e', c) *Prediction:* predict a new value for cardiac health as $h' = w_{AH}a^i + w_{EH}e' + \epsilon^i$.

4. Additional Experiments

The purview of the previous experiments was to show a proof of concept on a simple tractable system, demonstrating that causal induction and inference can be learned and implemented via a meta-learned agent. In the following, we scale up to more complex systems in two new experiments.

4.1. Experiment 4: Non-linear Causal Graphs

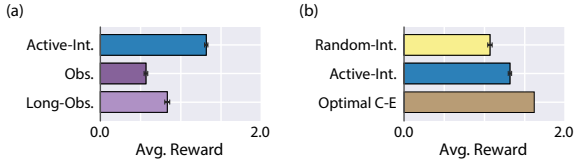

Figure 2: Results for non-linear graphs. (a) Comparing agent performances with different data. (b) Comparing information phase intervention policies.

In this experiment, we generalize some of our results to non-linear, non-Gaussian causal graphs which are more typical of real-world causal graphs and to demonstrate that our results hold without loss of generality on such systems.

Here we investigate causal Bayesian networks (CBNs) with a quadratic dependence on the parents by changing the conditional distribution to $p(X_i|\text{pa}(X_i)) = \mathcal{N}(\frac{1}{N_i}\sum_j w_{ji}(X_j + X_j^2), \sigma)$. Here, although each node is normally distributed given its parents, the joint distribution is not multivariate Gaus-

[1] These are zero in expectation, so without access to their value for an individual we simply use $\mathcal{G}$: $E = w_{AE}A$, $H = w_{AH}A + w_{EH}E$ to make causal predictions.

sian due to the non-linearity in how the means are determined. We find that the Long-Observational Agent achieves more reward than the Observational Agent indicating that the agent is in fact learning the statistical dependencies between the nodes, within an episode[2]. We also find that the Active-Interventional Agent achieves reward well above the best agent with access to only observational data (Long-Observational in this case) indicating an ability to reason from interventions. We also see that the Active-Interventional Agent performs better than the Random-Interventional Agent indicating an ability to choose informative interventions.

4.2. Experiment 5: Larger Causal Graphs

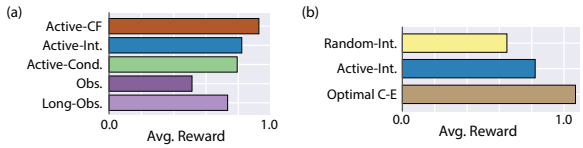

Figure 3: Results for $N = 6$ graphs. (a) Comparing agent performances with different data. (b) Comparing information phase intervention policies.

In this experiment we scaled up to larger graphs with $N = 6$ nodes, which afforded considerably more unique CBNs than with $N = 5$ (1.4×10^7 vs 5.9×10^4). As shown in Fig. 3a, we find the same pattern of behavior noted in the main text where the rewards earned are ordered such that Observational agent $<$ Active-Conditional agent $<$ Active-Interventional agent $<$ Active-Counterfactual agent. We see additionally in Fig. 3b that the Active-Interventional agent performs significantly better than the baseline Random-Interventional agent, indicating an ability to choose non-random, informative interventions.

5. Causal Bayesian Networks

By combining graph theory and probability theory, the causal Bayesian network framework provides us with a graphical tool to formalize and test different levels of causal reasoning. This section introduces the main definitions underlying this framework and explains how to visually test for statistical independence (Pearl, 1988; Bishop, 2006; Koller and Friedman, 2009; Barber, 2012; Murphy, 2012).

A **graph** is a collection of nodes and links connecting pairs of nodes. The links may be directed or undirected, giving rise to **directed** or **undirected graphs** respectively.

A **path** from node X_i to node X_j is a sequence of linked nodes starting at X_i and ending at X_j. A **directed path** is a path whose links are directed and pointing from preceding towards following nodes in the sequence.

[2]The conditional distribution $p(X_{1:N\setminus j}|X_j = 5)$, and therefore Conditional Agents, were non-trivial to calculate for the quadratic case.

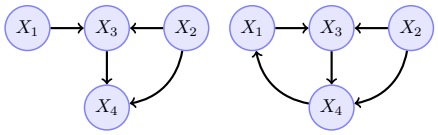

Figure 4: (a): Directed acyclic graph. The node X_3 is a collider on the path $X_1 \to X_3 \leftarrow X_2$ and a non-collider on the path $X_2 \to X_3 \to X_4$. (b): Cyclic graph obtained from (a) by adding a link from X_4 to X_1.

A **directed acyclic graph** is a directed graph with no directed paths starting and ending at the same node. For example, the directed graph in Fig. 4(a) is acyclic. The addition of a link from X_4 to X_1 gives rise to a cyclic graph (Fig. 4(b)).

A node X_i with a directed link to X_j is called **parent** of X_j. In this case, X_j is called **child** of X_i.

A node is a **collider** on a specified path if it has (at least) two parents on that path. Notice that a node can be a collider on a path and a non-collider on another path. For example, in Fig. 4(a) X_3 is a collider on the path $X_1 \to X_3 \leftarrow X_2$ and a non-collider on the path $X_2 \to X_3 \to X_4$.

A node X_i is an **ancestor** of a node X_j if there exists a directed path from X_i to X_j. In this case, X_j is a **descendant** of X_i.

A **graphical model** is a graph in which nodes represent random variables and links express statistical relationships between the variables.

A **Bayesian network** is a directed acyclic graphical model in which each node X_i is associated with the conditional distribution $p(X_i|\text{pa}(X_i))$, where $\text{pa}(X_i)$ indicates the parents of X_i. The joint distribution of all nodes in the graph, $p(X_{1:N})$, is given by the product of all conditional distributions, i.e. $p(X_{1:N}) = \prod_{i=1}^{N} p(X_i|\text{pa}(X_i))$.

When equipped with causal semantic, namely when describing the process underlying the data generation, a Bayesian network expresses both causal and statistical relationships among random variables—in such a case the network is called **causal**.

Assessing statistical independence in Bayesian networks. Given the sets of random variables $\mathcal{X}, \mathcal{Y}$ and $\mathcal{Z}$, $\mathcal{X}$ and $\mathcal{Y}$ are statistically independent given $\mathcal{Z}$ if all paths from any element of $\mathcal{X}$ to any element of $\mathcal{Y}$ are **closed** (or **blocked**). A path is closed if at least one of the following conditions is satisfied:

(i) There is a non-collider on the path which belongs to the conditioning set $\mathcal{Z}$.

(ii) There is a collider on the path such that neither the collider nor any of its descendants belong to $\mathcal{Z}$.

References

D. Barber. *Bayesian reasoning and machine learning*. Cambridge University Press, 2012.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

K. P. Murphy. *Machine Learning: a Probabilistic Perspective*. MIT Press, 2012.

J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

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