Online learning of causal structure in a dynamic game situation

Yue Gao (ygao@cs.cornell.edu)
Department of Computer Science, Cornell University
Ithaca, NY, 14853 USA

Eyal Nitzany (ein3@cornell.edu)
Program in Computational Biology and Medicine, Cornell University
Ithaca, NY, 14853 USA

Shimon Edelman (edelman@cornell.edu)
Department of Psychology, Cornell University
Ithaca, NY, 14853 USA

Abstract

Agents situated in a dynamic environment with an initially unknown causal structure, which, moreover, links certain behavioral choices to rewards, must be able to learn such structure incrementally on the fly. We report an experimental study that characterizes human learning in a controlled dynamic game environment, and describe a computational model that is capable of similar learning. The model learns by building up a representation of the hypothesized causes and effects, including estimates of the strength of each causal interaction. It is driven initially by simple guesses regarding such interactions, inspired by events occurring in close temporal succession. The model maintains its structure dynamically (including omitting or even reversing the current best-guess dependencies, if warranted by new evidence), and estimates the projected probability of possible outcomes by performing inference on the resulting Bayesian network. The model reproduces the human performance in the present dynamical task.

Keywords: Temporal learning, causality, structure learning, Dynamic Bayesian graphical model, STDP.

Introduction

There are many types of cues that an agent can use to learn the causal structure of its interactions with the environment, such as prior knowledge (which constrains the hypothesis space), statistical relations, intervention, and temporal ordering (Lagnado et al., 2007). Among these, temporal ordering is particularly intriguing. First, proximity among cues appears to play a central role in learning structure in time and space (Goldstein et al., 2010). Second, in causal learning, temporal ordering, similarly to intervention, carries with it information regarding the direction of causality, which is crucial for prediction. Finally, putative causal relationships between ordered events that occur in close temporal proximity can be registered by relatively well-understood computational mechanisms akin to those that support synaptic modification in nervous systems.

Both the learning of causal structure (as in model selection) and the modification of its parameters (as in classical schemes such as ΔP, $PowerPC$, and Rescorla-Wagner) can be put on a rational basis (Griffiths and Tenenbaum, 2009; Holyoak and Cheng, 2011). Given the special appeal of temporal proximity as a cue to causal structure and strength, a simple, incremental, heuristic approach to causal learning based on this cue is, however, worth exploring — particularly if such an approach proves effective in dealing with dynamical scenarios where irrelevant variables abound and where the model may need to be modifiable on the fly. In this paper, we describe a dynamical causal Bayesian model that uses temporal proximity among cues to learn its structure and parameters from a continuous stream of observations and action-related rewards in a computer game-related scenario.

Dynamic causal Bayesian modeling

Consider a dynamic situation described by a set of binary variables $\mathcal{X} = \{X_1, X_2, \ldots, X_n\}$, whose values occasionally change over time, and where $X_i = 1$ indicates the presence of some item or feature and $X_i = 0$ its absence. The causal relationships among the variables in $\mathcal{X}$, if any, are initially unknown; each one could be a cause, an effect, or neither. Our approach integrates bottom-up, event-driven learning with top-down revisions as dictated by the model's self-maintained track record in predicting impending events or the outcomes of actions.

The graph representing the model's current hypothesis regarding the causal relationships over the members of $\mathcal{X}$, which serve as its vertices, has initially no edges. As time progresses, edges are added to the graph according to the temporal order of the observed events and any interventions (the outcomes of model's own actions). As new edges, corresponding to pairwise hypothesized causal dependencies, are added, the model attempts to integrate the subgraphs they form — "twigs" (Figure 1, left) each consisting of a pair of vertices joined by a directed edge — into a larger (eventually, global) structure.

Note that the twigs are supposed to capture causal "strength," which we operationalize via a Hebb-like learning mechanism (detailed below), while the final causal model is intended to support probabilistic inference, by being treated as a Bayesian network. The $Union$ operation (Figure 2, left), which combines twigs

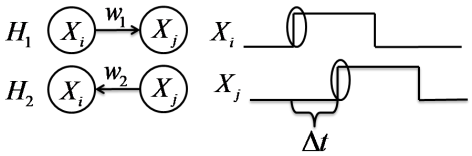

Figure 1: *Left:* the two possible $Twig$ structures for two variables X_i and X_j. The weights w_i represent causal strength. *Right:* a new $Twig$ is added to the model if a prior event involving some X_i is found within a short time window Δt of a triggering event involving X_j.

Figure 2: *Left:* the $Union$ operation. In this example, there are four possible $DCBN$ outcomes: causal chains U_1 and U_2, a common effect structure U_3, and a common cause structure U_4. *Right:* for each case where a binary effect is driven by a real-valued "strength" link, the $Union$ operation adds a hidden *softmax* node, as illustrated here for U_3 by the frame drawn around $R_{i,j}$ and E (Lu et al., 2008).

into a dynamic causal Bayesian network ($DCBN$), mediates between these two aspects of the model by inserting as needed "hidden" variables that convert real-valued strength variables into probability distributions (Figure 2, right).

As the network forms, the model becomes ready for generating predictions (inference). Given the state of observations at time t, it can be used to predict the most likely value for variables of interest at a later time. During this phase, inference is alternated with learning, with the latter being driven by the model's monitoring of its own predictions and by comparing those to the observed outcomes. The resulting changes may include the model's representation of the causal structure of the environment: for instance, the direction of some of the twigs (cause-and-effect subgraphs) may be reversed.

Structure and strength learning. We now proceed to describe the operation of the model in some detail, starting with twig learning. Every elementary subgraph, or $Twig = \{C, E, w\}$, consists of a single cause C, a single effect E, and the strength or weight w of their causal connection. Initially, no connections between variables exist. Similarly to how humans seem to handle causal cues such as temporal ordering and proximity of notable events (Lagnado et al., 2007), the model only seeks to form a $Twig$ when some item or feature appears on the scene (i.e., a variable changes state from 0 to 1). The model then scans the recent past, up to a duration of ΔT, for a potential cause of the event at hand, in the form of the change in some other variable's value. Any variable that is on the record as having changed its value (either from 0 to 1 or from 1 to 0) is labeled as a potential causes for the event, forming a $Twig$ (Figure 1).

The weight w is modified via Hebbian learning, specifically, spike timing dependent plasticity (STDP; Caporale and Dan, 2008). This family of temporally asymmetric Hebbian rules affords quick (exponential) learning, as well as unlearning (in "negative" trials, in which the purported effect precedes the cause):

$$\begin{aligned} \Delta w_+ &= A_+ \exp(-\Delta t/\Delta T) \\ \Delta w_- &= A_- \exp(\Delta t/\Delta T) \\ w(t+1) &= \min\left(w\left(t\right) + \Delta w,\ w_{max}\right) \end{aligned} \tag{1}$$

where the $+$ and $-$ subscripts denote positive and negative trials (E following or preceding C, respectively). We set $A_+ = 1$ and $A_- = 0.5$, thus giving more weight to positive evidence. If some $Twig$ elements share a common variable, the model attempts to combine them by applying the $Union$ operation, as in $Twig(X_i, X_j) \bigcup Twig(X_i, X_k) \Rightarrow DCBN(X_i, X_j, X_k)$ (Figure 2), adding, as needed, hidden variables, as described below.

Learning the *softmax* parameters. To integrate a representation of causal strength into a probabilistic (Bayesian) model, we follow Lu et al. (2008) by endowing the model with internal states, or hidden variables: R_i *and* R_j in Figure 2, right. The state of each R_i is related to that of its parent node X_i through a Gaussian distribution parameterized by the weight w_i:

$$P(R_i \mid w_i, X_i) \propto e^{-(R_i - w_i X_i)/2\sigma_i^2} \tag{2}$$

The binary effect variable $E = e_i$, e_i being the ith discrete value of E, is driven, in turn, by $\mathbf{R}$ through a *softmax* function:

$$P(E = e_i \mid \mathbf{R}) = \frac{\exp\left(\mathbf{w}\left(:, i\right)^T \mathbf{R} + \mathbf{b}\left(i\right)\right)}{\sum_j \exp\left(\mathbf{w}\left(:, j\right)^T \mathbf{R} + \mathbf{b}\left(j\right)\right)} \tag{3}$$

where $\mathbf{R}$ is the vector that comprises R_i and R_j, and $\mathbf{w}$ and $\mathbf{b}$ are parameters that are learned as the model is exposed to data, using an iteratively reweighted least squares (IRLS) algorithm (Green, 1984).

Inference. We illustrate the inference process, in which the model is used to generate predictions for some

Algorithm 1 Dynamic causal Bayesian model (DCBN)

```
INITIAL LEARNING
Given: variables 𝒳; window ΔT.
Note: |𝒳| = n is the number of variables.
Note: t is the current time.
for i = 1 → n do
    if X_i^t == 1 and X_i^{t−1} == 0 then
        for j = 1 → n do
            if X_j preceded X_i by Δt < ΔT then
                if No Twig(X_i, X_j, w_ij) exists then
                    Compute w_ij (eq. 1);
                    Create Twig(X_i, X_j, w_ij);
                else
                    Update w_ij (eq. 1);
                end if
            end if
        end for
    end if
    Compute Union over Twigs to form DCBN;
    Train softmax (eq. 3) for hidden variables;
end for
INFERENCE AND FURTHER LEARNING
while True do
    Perform inference on DCBN;
    if inference deviates from observation then
        Modify Twig weights w (eq. 1);
        for every Twig do
            if w < 0 then
                Reverse the edge;
                Re-learn w;
            end if
        end for
        If structure changed, recompute Union;
    end if
    Retrain softmax parameters;
end while
```

variable values, given others, on an example with an effect E that depends on two causes, $X_{1,2}$ (Figure 2, right). Given the values of $X_{1,2}$, inference requires integration over the hidden variables:

$$P(E \mid w_1, w_2, X_1, X_2) = \tag{4}$$
$$= \iint P(E \mid R_1, R_2) \prod_{i=1}^{2} P(R_i \mid w_i, X_i)\, P(X_i)\, dR_1 dR_2$$

Because R_i are unobserved and continuous and their descendants are discrete, exact inference is impossible (Lerner et al., 2001; Murphy, 1999). As an approximation, we sample each R_i, conditioned on its parent X_i and weight w_i. Specifically, if $X_i = 1$, we sample from the Gaussian distribution associated with it (eq. 2); if $X_i = 0$, we sample from a zero-mean Gaussian distribution, which is the same for all the variables.

We then discretize the integral, with a step size of 0.1 and lower and upper bounds set to $L = \min_{X_i}(-4\sigma_i)$ and $U = \max_{X_i}(w_i X_i + 4\sigma_i)$, respectively:

$$P(E \mid w_1, w_2, X_1, X_2) = \tag{5}$$
$$= \sum_{R_1=L}^{U} \sum_{R_2=L}^{U} P(E \mid R_1, R_2) \prod_{i=1}^{2} P(R_i \mid w_i, X_i)\, P(X_i)$$

When the predicted value of E yielded by the inference step matches the observed value with a high confidence, the model is not modified. Every time the prediction falters, the model learns; both its structure and its parameters can be modified, as described in Algorithm 1.

The experiments

To evaluate the model, we tested it in an experiment that involved learning in a dynamically unfolding game situation.[1] For the same experiment, we also collected performance data from human subjects.

Most of the published studies of causal learning to date have been conducted in somewhat artificial behavioral settings. In many studies, the task consists of a series of trials, in each of which the subject is presented with a few stimuli — often just two or three items on a blank screen, along with choices that can be made via a key press (e.g., Steyvers, Tenenbaum, Wagenmakers, and Blum, 2003). More elaborate tasks may involve a contraption that displays a few objects whose behaviors may be causally interlinked (e.g., Kushnir, Gopnik, Lucas, and Schulz, 2010). The narrative context that defines the task for the subjects is often couched in causal language (as in "Can you tell what makes the box go?"). In comparison, in the present study the behavioral task involved an arguably more natural situation: playing a computer game, which unfolds in real time, and requires that the subject drive down a track surrounded by various objects, while attempting to accumulate rewards.

The experimental platform we used is an adaptation of a car-racing computer game.[2] The virtual environment through which the subject is driving consists of tracks surrounded by scenes whose composition is controlled. It is flexible enough to support various types of cues to causal structure, including interventions (Lagnado et al., 2007). Moreover, because the game can be played against another subject or against a computer program, it affords the study of social effects in learning (Goldstein et al., 2010).[3]

[1] In a separate study, we used the model to replicate successfully some of the standard effects in causal learning, such as forward and backward blocking (Holyoak and Cheng, 2011).

[2] http://supertuxkart.sourceforge.net/ (public domain). We modified the game to support Wiimote and to incorporate our tracks, scenes, and reporting. The modified code is available upon request.

[3] Note that instructions that the subject receives from the experimenter may be considered a kind of social cue.

Figure 3: A typical game scene, as presented to the subjects as part of the instructions for the experiment.

scene type	*crate*	*dog*	*cat*	*fox*	box contents
1	1	[0]	1	0	$[plunger]_{t+}$
2	1	[**1**]	1	0	$[cake]_{t+}$
3	1	[0]	1	1	$[plunger]_{t+}$
4	1	[**1**]	1	1	$[cake]_{t+}$
5	0	0	1	0	$[plunger]_{t+}$
6	0	**1**	1	0	$[cake]_{t+}$

Table 1: The six scene types in the experiment, with the presence or absence of various objects indicated by 1/0. When *crate* is present, *dog* is hidden inside it (bracketed, $[\cdot]$). The contents of the surprise box (*cake* or *plunger*) become visible only if the subject actively "takes" it (signified by $[\cdot]_{t+}$). Note that *dog* perfectly predicts $[cake]_{t+}$, but subjects who miss the significance of *crate* will be unable to distinguish between scenes 1 and 2, or 3 and 4.

The behavioral experiment

Given the novelty and the potential difficulty of the dynamical learning scenario, in this study we opted for a maximally simple dependency to be learned: a single causal link between two variables. Each scene in the experiment could include any or all of the following objects: a dog, a cat, a fox, and a crate (Figure 3). In addition, in each scene there was a "surprise" box which, if the subject chose to "take" it, revealed the reward: a cake or a plunger, depending on the appearance of other objects in the scene. The subjects were instructed to collect as many cakes as possible, while refraining from taking plungers. Altogether, each subject encountered 252 scenes: 6 different racetracks × 3 laps × 14 scenes drawn at random for each track from among the scene types listed in Table 1.

The subjects' task is best *seen*[4] as learning a directed or causal (rather than an undirected or merely associative) relationship, for two reasons: the asymmetry in the temporal structure of each scene encounter and in the functional significance of its components. First, the reward never co-occurred with any of the other variables: rather, it always followed them temporally, and then only if the subject actively intervened by opening the surprise box. Second, it made no sense for the subjects to hypothesize symmetrical functional roles for the reward and for the other variables, given that their goal was formulated exclusively in terms of the reward. In any case, no causal language was used in the instructions given to subjects, which makes the present experimental set up arguably more natural as a platform for exploring simple learning in the wild than those that explicitly require the subjects to seek causal explanations for behavioral outcomes.

Eighteen subjects, recruited online from the Cornell University subject pool, participated in the study for course credit. The dependent variable, Correct, was defined as equal to 1 in trials where the subject opened a box with *cake* or refrained from opening a box with *plunger*. A mixed model analysis using the *lmer* package (Bates, 2005), with a binomial linking function, and with Subject, Track, and Scene as random factors, yielded a significant effect of Lap on Correct ($z = 7.53$, $p = 5.1 \times 10^{-14}$). Averaged over tracks, the subjects' Correct rate reached 0.61 in the third lap. The far from perfect performance is understandable, given that the inconsequential parts of the game environment (such as a stable with horses, bales of hay, etc.), as well as of the surprise-box scenes themselves, made it difficult for subjects to home in on the truly predictive variable (*dog*). Moreover, in scene types 1 through 4, the dog appears inside a crate and is thus not visible, unless the subject drives through the crate (something that few subjects ventured to do).

The evolution of subjects' performance over time is illustrated for each scene type in Figure 4, right. Debriefing indicated that subjects generally assumed correctly that the contents of the surprise box could be anticipated by noting which of the other objects were present in the scene. Many of the subjects did not, however, allow for the possibility that a cause may be hidden, which prompted them to invent incorrect explanations for the difference between scene types (1,2) and (3,4). Some subjects also tried to find patterns in the irrelevant variables such as the distances among objects, the curving of the track, and the location of the box with respect to other items.

Modeling results

We simulated the behavioral experiment by feeding the model incrementally the same sequence of observations

[4]The question of what the relationship between *dog* and *cake* in this simple scenario really *is*, causal or associative, is best avoided, given the philosophical issues surrounding causality (Schaffer, 2009). Somewhat paradoxically, the distinction is easier for more complex networks of dependencies among variables, such as those explored by Blaisdell et al. (2006).

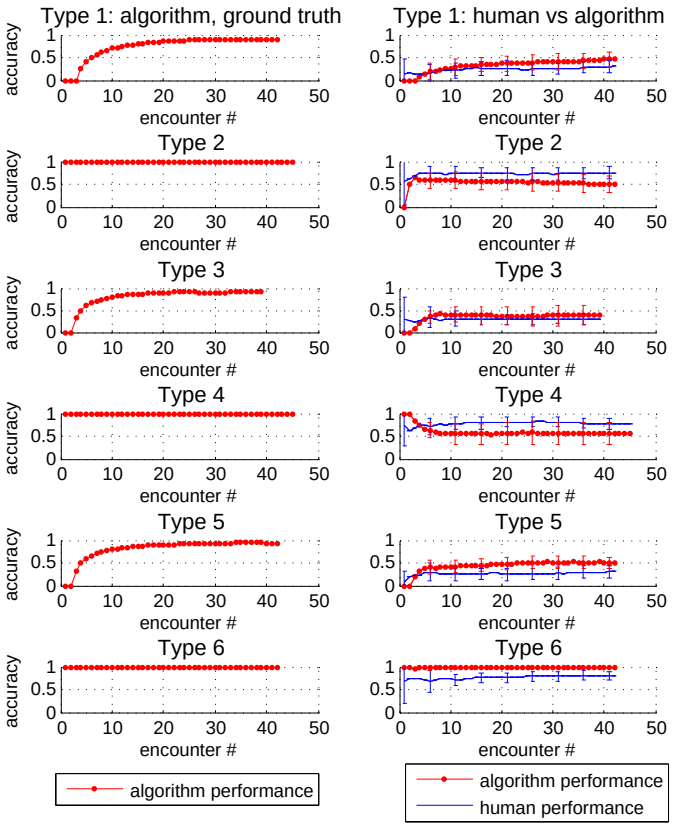

Figure 4: *Left:* the performance of the algorithm on ground-truth data, for each of the six scene types. *Right:* the performance of 18 runs of the algorithm (filled circles) and of the 18 subjects subjects in a real run (means with 95% confidence intervals). The ups and downs in the algorithm's performance over time are due to its sensitivity to the order of scene appearance (batch algorithms do not exhibit this behavior).

encountered by the human subjects, namely, the values of the four variables listed in Table 1, plus, in the cases where the model decided to open the surprise box, the value of reward. In the first lap (14 scenes; the "initial learning" phase in Algorithm 1), the model was set to open every box. Subsequently, if the model's decision whether or not to open the box could be made with 95% confidence,[5] it chose the recommended action; otherwise it flipped a coin. If the decision was to open the box, the model used the outcome to adjust its parameters; if not, it simulated an outcome by adding to the predicted value of the reward a random number (distributed uniformly in $[0, 1]$) and used that outcome to learn.[6]

As can be seen in Figure 4, left, when fed ground-truth data, the model learned quickly and reliably.[7] More to the point, when presented with the real sequence of observations, it generally behaved similarly to human subjects (Figure 4, right), reaching a comparable level of performance: 0.66 accuracy in the third lap. As with the human subjects, the effect of Lap was significant ($z = 2.56$, $p = 0.01$). In Figure 4, right, in those cases where the dog was hidden from view (scene types 1 through 4; see Table 1), the human subjects performed poorly, and the algorithm too converged to a chance-level performance.

Conclusions

Similarly to some other recent studies and models of causal learning (Lu et al., 2008; Lagnado and Speekenbrink, 2010; Bonawitz et al., 2011), the present work focuses on sequential learning and inference. There are also important differences. First, our behavioral setup uses a dynamic video game that subjects readily relate to. Second, the model we develop is rooted in some basic intuitions regarding how animals learn the causal structure of dynamic situation: (1) the importance of close temporal succession of events and outcomes, (2) the utility of neural-like mechanisms that may register it, and (3) a heuristic approach to bootstrapping causal learning from very simple pairwise dependencies gleaned from the data. In those respects, the algorithm we offer is a special-purpose model rather than a general learner.

To ascertain that subjects in our game scenario engage in causal learning and inference, rather than in memorization of contextual cues they believe to be associated with particular outcomes, future experiments will need to include explicit intervention-based tests (cf. Blaisdell, Sawa, Leising, and Waldmann, 2006), including having the subjects manipulate the variables of their choice to test any hypotheses that they may have formed. It would also be interesting to analyze the evolution over time of the subjects' choices in opening or avoiding reward boxes: early in the experiment, it is rational to open boxes, so as to gather data; as the subjects develop an ability to predict the reward, they should become more choosy. This sequential behavior can then be compared to that of the model (Bonawitz et al., 2011).

The model itself can be improved and extended in several ways. For instance, as it is tested on learning

[5] As decided by a binomial test with a confidence interval of $\hat{p} \pm z_{1-\alpha/2}\sqrt{\hat{p}(1-\hat{p})/n}$, where $\hat{p}$ is the sample proportion of successes in the observed sequence of trials and $z_{1-\alpha/2}$ is the $1-\alpha/2$ percentile of a standard normal distribution, with α being the error percentile and n the sample size. For a 95% confidence level, $\alpha = 5\%$ and $z_{1-\alpha/2} = 1.96$.

[6] Without some such mechanism, the model would have no way of recovering from a string of "don't open" decisions — a problem that is peculiar to models that intersperse learning with inference.

[7] In comparison, a straightforward model selection approach based on maximum likelihood or AIC/BIC optimization, implemented with the Bayes Network Toolbox for Matlab (Murphy, 2001), trained incrementally on the ground truth data, did not converge to the right causal graph for this experiment.

tasks that involve more complex causal structure than that in the present study, it may be necessary to include methods for detecting and "defusing" loops that would otherwise complicate inference. Furthermore, the model can be made to incorporate additional cues to causal structure, in particular, interventions (Steyvers et al., 2003), global contextual cues, and factors such as eligibility traces (Izhikevich, 2007) that would allow it to learn from such cues across multiple time scales. Finally, if equipped with a vision front end and real-valued outputs, a model rooted in the present approach may employ reinforcement learning (Fox et al., 2008; Hosoya, 2009) to master driving around the track and competing directly with a human participant.

Acknowledgments. We thank Tamar Kushnir and the members of her lab for valuable comments on the present project. EN was supported by Cornell University's Tri-Institutional Training Program in Computational Biology and Medicine.

References

Bates, D. (2005). Fitting linear mixed models in R. *R News 5*, 27–30.

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

Bonawitz, E., S. Denison, A. Chen, A. Gopnik, and T. L. Griffiths (2011). A simple sequential algorithm for approximating bayesian inference. In L. Carlson, C. Hölscher, and T. F. Shipley (Eds.), *Proceedings of the 33rd Annual Conference of the Cognitive Science Society*, pp. 2463–2468.

Caporale, N. and Y. Dan (2008). Spike timing-dependent plasticity: A Hebbian learning rule. *Annual Review of Neuroscience 31*, 25–46.

Fox, C. N., N. Girdhar, and K. N. Gurney (2008). A causal Bayesian network view of reinforcement learning. In *Proceedings of the 21th International Florida Artificial Intelligence Research Society Conference, FLAIRS-21*, pp. 109–110.

Goldstein, M. H., H. R. Waterfall, A. Lotem, J. Halpern, J. Schwade, L. Onnis, and S. Edelman (2010). General cognitive principles for learning structure in time and space. *Trends in Cognitive Sciences 14*, 249–258.

Green, P. J. (1984). Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. *J. Royal Stat. Soc. Ser. B 46*, 149–192.

Griffiths, T. L. and J. B. Tenenbaum (2009). Theory-based causal induction. *Psychological Review 116*, 661–716.

Holyoak, K. J. and P. W. Cheng (2011). Causal learning and inference as a rational process: the new synthesis. *Annual Review of Psychology 62*, 135–163.

Hosoya, H. (2009). A motor learning neural model based on Bayesian network and reinforcement learning. In *Proceedings of International Joint Conference on Neural Networks*, Atlanta, GA. IEEE.

Izhikevich, E. M. (2007). Solving the distal reward problem through linkage of STDP and dopamine signaling. *Cerebral Cortex 17*, 2443–2452.

Kushnir, T., A. Gopnik, C. Lucas, and L. Schulz (2010). Inferring hidden causal structure. *Cognitive science 34*(1), 148–160.

Lagnado, D. A. and M. Speekenbrink (2010). The influence of delays in real-time causal learning. *The Open Psychology Journal 3*, 184–195.

Lagnado, D. A., M. R. Waldmann, Y. Hagmayer, and S. A. Sloman (2007). Beyond covariation: cues to causal structure. In A. Gopnik and L. Schulz (Eds.), *Structure*, pp. 1–48. Oxford University Press.

Lerner, U., E. Segal, and D. Koller (2001). Exact inference in networks with discrete children of continuous parents. In *Proc. 17th Conf. on Uncertainty in Artificial Intelligence*, pp. 319–238.

Lu, H., R. R. Rojas, T. Beckers, and A. Yuille (2008). Sequential causal learning in humans and rats. In B. C. Love, K. McRae, and V. M. Sloutsky (Eds.), *Proceedings of the 30th Annual Conference of the Cognitive Science Society*, pp. 188–195.

Murphy, K. (1999). A variational approximation for bayesian networks with discrete and continuous latent variables. In *Proc. UAI*, Volume 99, pp. 457–466.

Murphy, K. P. (2001). The Bayes Net Toolbox for Matlab. *Computing Science and Statistics 33*, 331–351.

Schaffer, J. (2009). The metaphysics of causation. In E. N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy* (Spring 2009 ed.).

Steyvers, M., J. B. Tenenbaum, E. J. Wagenmakers, and B. Blum (2003). Inferring causal networks from observations and interventions. *Cognitive Science 27*, 453–489.