A Probabilistic Theory of Abductive Reasoning

Nicolas A. Espinosa Dice[1][a], Megan L. Kaye[1][b], Hana Ahmed[1,2][c], and George D. Montañez[1][d]

[1]*AMISTAD Lab, Department of Computer Science, Harvey Mudd College, Claremont, CA, USA*

[2]*Scripps College, Claremont, CA, USA*

{*nespinosadice, mkaye, hahmed, gmontanez*}*@hmc.edu*

Keywords: abductive logic, machine learning, creative abduction, creativity, graphical model, Bayesian network, probabilistic abduction

Abstract: We present an abductive search strategy that integrates creative abduction and probabilistic reasoning to produce plausible explanations for unexplained observations. Using a graphical model representation of abductive search, we introduce a heuristic approach to hypothesis generation, comparison, and selection. To identify creative and plausible explanations, we propose 1) applying novel structural similarity metrics to a search for simple explanations, and 2) optimizing for the probability of a hypothesis' occurrence given known observations.

1 INTRODUCTION

Imagine that one morning you step outside to find that the grass is wet, ruining your new shoes. Could rain have caused the wet grass? However, you cannot recall whether yesterday was cloudy. How likely is it to have rained last night if there were no clouds?

Now imagine an alternative scenario, in which you are a medical student studying the causes and symptoms of tuberculosis. You learn that if a patient has an abnormal x-ray, there are several possible factors, including lung cancer and tuberculosis. How can you determine which diagnosis to give in light of the x-ray results? What relevant information is available to help you decide a best explanation?

For a final example, imagine arriving at work to find that information on your company's database has been corrupted. Your boss is responsible for fixing the deficiency that allowed this data corruption to occur. Overwhelmed by the vast number of possible explanations for the data corruption, your boss tasks you, a database engineer, with identifying plausible causes of the issue.

These tasks require *abductive inference*: creating and identifying *hypotheses* (causes) that are the most promising explanations for the *observed effects*. You can make use of *background information*, prior occurrences similar in nature to this one, where the effects and causes were successfully identified. However, you also acknowledge the possibility of unfamiliar causes, which are beyond the scope of prior information and your current knowledge.

For example, in our database problem, corrupted data is the unexplained observed effect. Your primary task reflects that of abductive inference, which is a strategy for discovering hypotheses that are worthy of *further investigation*, which Schurz refers to as the *strategical function* of abductive inference (Schurz, 2008). What constitutes further investigation depends on the application of abductive inference; it can be broadly defined as any work that provides more information about the causes or effects. Additionally, the *search space*—the space of all available hypotheses—can be significantly large, necessitating an effective *search strategy* for finding promising hypotheses within reasonable time and computational costs.

Despite significant advances in machine learning research over the last three decades, traditional supervised learning models are ill-equipped to handle the aforementioned example problems. Supervised learning models emulate *inductive* inference, in which hypotheses are causal rules that best fit the known data. Unlike abduction, the primary function of inductive inference is *justificational*, specifically the justifica-

[a] https://orcid.org/0000-0001-7802-6196
[b] https://orcid.org/0000-0001-5422-8244
[c] https://orcid.org/0000-0003-4532-0334
[d] https://orcid.org/0000-0002-1333-4611

tion of the conjectured conclusion. Induction serves little strategical function because the range of possible conclusions is restricted by the methods of generalizing prior observed cases.

In this paper, we present a novel abductive search method capable of handling the examples previously described. Our model first uses *abductive search* for *hypothesis identification*. To limit redundancy in the abductive search results, we introduce two distinct similarity metrics that compare causal structures of variables. Additionally, to account for possible unfamiliar causes, we implement *hypothesis generation* in our model as a method of generating *novel explanations*—hypothesized causes that are not necessarily observed in the background information. Finally, while abductive "confirmation" does not indicate whether an abduced hypothesis logically precedes the observed effects, our model utilizes a *hypothesis comparison* method to compare hypotheses based on the likelihood of the explanation.

Cox et al. used abduction with surface deduction to generate novel hypotheses from Horn-clauses, and suggested extending this method's application to abduction from directed graphs (Cox et al., 1992). Our abductive search model relies on Reichenbach's Common Cause Principle rather than surface deduction for hypothesis generation, and uses edit-distance and Jaccard-based reasoning to distinguish redundant hypotheses. This combination of creative and probabilistic abduction with similarity-based reasoning for abductive search is distinct from the approach of Cox et al. (Cox et al., 1992). The use of Reichenbach's Common Cause Principle is inspired by Schurz's theory on common cause abduction (Schurz, 2008). While Schurz seems to deny the potential usefulness of integrating common cause and Bayesian reasoning, we introduce a form of Bayesian confirmation that provides probabilistic explanations for the hypotheses discovered through common cause abduction. Likewise, while our abductive model checks consistency and simplicity similarly to Reiter's heuristic diagnosis model (Reiter, 1987), we rely on Bayesian conditioning during hypothesis generation and comparison, which strengthens the plausibility of our model's conjectured hypotheses.

2 GRAPHICALLY MODELING ABDUCTION

Graphical models are tools for integrating logical and probabilistic reasoning in order to represent rational processes and causal relationships. Developed by Pearl, they comprehensively account for complexity and uncertainty within a dataset (Pearl, 1998). A probabilistic graphical model is composed of nodes representing random variables, and edges connecting the nodes to indicate conditional independence or dependence.

An abductive search problem can be represented in a directed acyclic graph (DAG), in which a directed edge from one node (the "parent") to another (the "child") represents a causal relationship between them. For edges of a DAG that are weighted with the conditional probability $P(\text{child} \mid \text{parent})$ of the child variable given the parent, the weight speaks to the causal relationship's influential strength.

2.1 Bayesian Networks

We adapt the definition of *Bayesian network* from (Feldbacher-Escamilla and Gebharter, 2019), and make use of conventional notation: *Sets* of objects, including *sets of sets*, are represented by boldfaced uppercase letters (e.g., **S**). *Variables* are represented by upper-case letters (e.g., X), and their respective realizations are represented by corresponding lower-case letters (e.g., x). Additionally, a *directed edge* between two variables is represented by an arrow, $\rightarrow$, where the parent node is at the arrow's tail and the child node is at the tip (e.g., $X_i \rightarrow X_j$).

Following the definitions in (Feldbacher-Escamilla and Gebharter, 2019), $B\langle \boldsymbol{V}, \boldsymbol{E}, P \rangle$ is a Bayesian network such that $\boldsymbol{V}$ is a set of *random variables*, $\boldsymbol{E}$ is a set of *directed edges*, and P is a *probability distribution* over $\boldsymbol{V}$.

For all $X_i \in \boldsymbol{V}$, $\boldsymbol{Par}(X_i)$ is the set of X_i's *parents*:

$$\boldsymbol{Par}(X_i) = \{X_j \in \boldsymbol{V} \mid X_j \rightarrow X_i\}. \qquad (1)$$

The set of X_i's *children* is defined as

$$\boldsymbol{Ch}(X_i) = \{X_j \in \boldsymbol{V} \mid X_i \rightarrow X_j\}. \qquad (2)$$

We define the set of X_i's *descendants* to be

$$\boldsymbol{Des}(X_i) = \{X_j \in \boldsymbol{V} \mid X_i \rightarrow ... \rightarrow X_j\}, \qquad (3)$$

and the set of X_i's *ancestors* to be

$$\boldsymbol{Anc}(X_i) = \{X_j \in \boldsymbol{V} \mid X_j \rightarrow ... \rightarrow X_i\}. \qquad (4)$$

Within the context of this paper, all variables in $\boldsymbol{V}$ are discrete. To properly incorporate continuous variables into the model, the discretization approach presented in (Chen et al., 2017) can be used with a discretization runtime of $O(r \cdot n^2)$, where r is the number of class variable instantiations. Furthermore, Freidman et al. present a method of discretizing continuous variables while learning the structure of the Bayesian network using background information, that is, data denoting the values of previous instantiations of variables in $\boldsymbol{V}$ (Friedman et al., 1996).

Additionally, we define a set of observed nodes

$$\boldsymbol{O} = \boldsymbol{O_E} \cup \boldsymbol{O_O}, \tag{5}$$

where

$$\boldsymbol{O_E} = \{O_{E_1}, \dots, O_{E_l}\} \tag{6}$$

is the set of nodes representing *observed effects* that require explanation. Subsequently,

$$\boldsymbol{O_O} = \{O_{O_1}, \dots, O_{O_j}\} \tag{7}$$

is the set of nodes representing *observations* that do not require explanations. Sets $\boldsymbol{O_E}$ and $\boldsymbol{O_O}$ are disjoint, namely, $\boldsymbol{O_E} \cap \boldsymbol{O_O} = \emptyset$. Furthermore, $\boldsymbol{U}$ is the set of *unobserved nodes*, such that

$$\boldsymbol{U} = \boldsymbol{V} - \boldsymbol{O}. \tag{8}$$

A *hypothesis* $\boldsymbol{H}$ will take the form

$$\boldsymbol{H} = \{h_1, \dots, h_m\}, \tag{9}$$

where $\boldsymbol{H} \subseteq \boldsymbol{U}$. An *explanation* refers to hypotheses that are causally related to a given set of observed effects.

2.1.1 Node Marginal Probability Distribution

We wish to calculate probabilities of n proposed nodes $v_i \in \{v_1, \dots, v_n\}$ given a set of known nodes $\boldsymbol{O} = \{x_1, \dots, x_m\}$ and unknown nodes $\boldsymbol{U} = \{U_1, \dots, U_\ell\}$. Because there are unknown nodes—random variables with unknown values—we need to account for all potential outcomes. Thus, we calculate the marginal probability distribution:

$$P(v_1, \dots, v_n \mid x_1, \dots, x_m) = \frac{P(v_1, \dots, v_n, x_1, \dots, x_m)}{P(x_1, \dots, x_m)} \tag{10}$$

To calculate the marginal probability distribution that accounts for all potential outcomes of $U_1, \dots, U_\ell$, we take a sum over all possible values. Thus, the numerator of the full equation is:

$$P(v_1, \dots, v_n, x_1, \dots, x_m) = \sum_{U_1 \in \{u_1, \neg u_1\}} \cdots \sum_{U_\ell \in \{u_\ell, \neg u_\ell\}} P(v_1, \dots, v_n, x_1, \dots, x_m, U_1, \dots, U_\ell) \tag{11}$$

To complete the equation, the denominator follows the same computation.

3 Abduction as a Search Strategy

We will demonstrate how to use abductive reasoning in a *best-first search* for explanations. Schurz defines a *best-first explanation* within the search space of an abductive model as one that meets the following criteria (Schurz, 2008):

1. The hypothesis is the most justifiable out of all candidate hypotheses.
2. The children/successors are the most plausible of all the hypothesis' successors.

We expand on Schurz's definition by adding the following third criterion:

3. The hypothesis is a *common cause* or *distant common cause* (an ancestor node) of all given observed effects.

Addition of this third criterion for potential hypotheses is based on *Reichenbach's Common Cause Principle* (CCP). The CCP is cited by Schurz as the justification basis for creative abduction (Schurz, 2008), and it is defined as follows:

Definition 3.1 (Reichenbach's Common Cause Principle)**.** For two properties A and B that are 1) correlated, and 2) unrelated by a conditional relationship, there must exist some *common cause* C such that A and B are both causal effects of C.

Our method relies on this principle during hypothesis selection. Given observed effects, we target a common cause (a hypothesis) that is the most promising explanation of the observed effects.

3.1 Problem Definition

Using abduction as a search strategy, we model a creative abductive solution for the following search problem adapted from (Feldbacher-Escamilla and Gebharter, 2019).

Given:

- A set of observed effects $\boldsymbol{O_E}$.
- A set of known or background data $\boldsymbol{O_O}$.

Find:
A solution with the following elements (Prendinger and Ishizuka, 2005):

- A candidate hypothesis $\boldsymbol{H_C}$ that is causally related to all $O_{O_i} \in \boldsymbol{O_O}$ and all $O_{E_i} \in \boldsymbol{O_E}$.
- A causal rule denoting that $\boldsymbol{H_C}$ is a potential set of causes for $\boldsymbol{O_E}$.
- The necessary condition that $\boldsymbol{H_C} \cap \boldsymbol{O}$ is consistent for all $O_{O_i}, O_{E_i} \in \boldsymbol{O}$.

4 Hypothesis Identification, Generation, and Comparison

Due to the vast search space of possible causes, the model's first component, *hypothesis identification*, must be completed using an abductive search strategy. Hypothesis identification serves a strategical function in the model by identifying possible common causes of the observed effects. Treating possible causes as hypotheses, we use the term *abductive search* because we optimize our search for $P(\boldsymbol{O_E} \mid \boldsymbol{H})$, the probability of the observed effects occurring given the hypothesis. $P(\boldsymbol{O_E} \mid \boldsymbol{H})$ measures the *fit* of a hypothesis, or the degree to which hypothesis $\boldsymbol{H}$ explains observed effects $\boldsymbol{O_E}$. Abductive inference, by definition, is agnostic towards how probable a hypothesis is, and instead optimizes for how well they explain the observed effects. From a probabilistic reasoning perspective, this is akin to optimizing for $P(\boldsymbol{O_E} \mid \boldsymbol{H})$. Thus, by optimizing our search for how well hypotheses explain the observed effects, we are modeling abduction. Additionally, optimizing the search for $P(\boldsymbol{O_E} \mid \boldsymbol{H})$ allows for the consideration of *surprising* (unlikely) hypotheses that if true, sufficiently explain the observed effects.

Having identified the set of possible common causes, each cause is treated as a candidate hypothesis and evaluated by a comparison function that optimizes for $P(\boldsymbol{H} \mid \boldsymbol{O_E})$, the probability that the hypothesis is true given the observed effects occurring. By Bayes' Theorem, we see

$$P(\boldsymbol{H} \mid \boldsymbol{O_E}) \propto P(\boldsymbol{O_E} \mid \boldsymbol{H})P(\boldsymbol{H}). \tag{12}$$

Thus, by optimizing for $P(\boldsymbol{H} \mid \boldsymbol{O_E})$, the probability of the hypothesis is taken into account. Through hypothesis comparison, our model incorporates a justificational function. The best performing candidate hypotheses are added to the set of *promising hypotheses*, denoted $\boldsymbol{H_P}$.

4.1 Hypothesis Identification and Abductive Search For Possible Common Causes

This section discusses the search for possible common causes of a set of observed effects. For clarity, in probability functions, we treat possible common causes as possible hypotheses.

The abductive search attempts to model the following

$$\underset{\boldsymbol{H} \subseteq \boldsymbol{U}}{\operatorname{argmax}} P(\boldsymbol{O_E} \mid \boldsymbol{H}), \tag{13}$$

in cases where $\boldsymbol{O_E}$ is given. Because edges in a Bayesian network are weighted with conditional probabilities, the hypotheses that optimize $P(\boldsymbol{O_E} \mid \boldsymbol{H})$ will generally be connected to more of the observed effects by an edge or directed path. So, rather than computing $P(\boldsymbol{O_E} \mid \boldsymbol{H})$ for all of the possible hypotheses, we can instead search for possible hypotheses that maximize the number of observed effects to which they are connected.

To begin the search for such possible hypotheses, we can apply CCP to the observed effects if and only if all variables $O_{E_i} \in \boldsymbol{O_E}$ satisfy the conditions of CCP. Specifically, for all $O_{E_i}, O_{E_j} \in \boldsymbol{O_E}$ such that $O_{E_i} \neq O_{E_j}$, O_{E_i} and O_{E_j} must be correlated and unrelated by a conditional relationship. For now, we assume that $\boldsymbol{O_E}$ satisfies the criteria for CCP, and we will later demonstrate how to handle the two possible cases in which CCP criteria are not satisfied.

By CCP, there exists some common cause C such that O_{E_i} and O_{E_j} are both effects of C, for all $O_{E_i}, O_{E_j} \in \boldsymbol{O_E}$ where $O_{E_i} \neq O_{E_j}$. This means there exists some *directed path* from C to O_i and O_j.

Definition 4.1 (Directed Path)**.** A *directed path* from X_i to X_j, where $X_i, X_j \in \boldsymbol{V}$, is a set of edges $\boldsymbol{E}_{X_i,X_j}$ such that either $(X_i, X_{\alpha_1}), \ldots, (X_{\alpha_k}, X_j) \in \boldsymbol{E}_{X_i,X_j}$ where $\alpha_1, \ldots, \alpha_k$ represent arbitrary node indices of the graph for some $k \in \mathbb{N}$, or $(X_i, X_j) \in \boldsymbol{E}_{X_i,X_j}$.

Therefore, the set of common causes of the observed effects, $\boldsymbol{C}(\boldsymbol{O_E})$, must be a subset of the set of variables with a directed path to $\boldsymbol{O_E}$.

Definition 4.2 (Singleton Complete Explanations)**.** A *singleton complete explanation* is a variable in $\boldsymbol{Anc}(\boldsymbol{O_E})$ with a directed path to every variable in $\boldsymbol{O_E}$. The set of singleton complete explanations is given by

$$\boldsymbol{C_P}(\boldsymbol{O_E}) := \bigcap_{O_{E_i} \in \boldsymbol{O_E}} \boldsymbol{Anc}(O_{E_i}). \tag{14}$$

We refer to the singleton explanations in $\boldsymbol{C_P}(\boldsymbol{O_E})$ as possible common causes because $P(\boldsymbol{O_E} \mid \boldsymbol{H})$—where $\boldsymbol{H}$ is the cause—has not yet been computed. Calculating this marginal probability is the only method of verifying a variable or set of variables as an actual common cause. Each possible cause in this set is considered a *complete explanation* because there exists a directed path from the nodes composing the explanation to each observed effect.

However, we must also consider cases where $\boldsymbol{O_E}$ does not satisfy the CCP criteria. Specifically, there are two possible cases in which the CCP criteria is not satisfied by $\boldsymbol{O_E}$.

Case 1 Suppose that there exists some distinct pairs of variables $O_{E_i}, O_{E_j} \in \boldsymbol{O_E}$ such that O_{E_i} and O_{E_j} are conditionally related. In such a case, there must be a directed path between O_{E_i} and O_{E_j}. Consequently, since the Bayesian network is acyclic, then without a loss of generality, $O_{E_i} \in \boldsymbol{Des}(O_{E_j})$. Therefore, it is possible that O_{E_j} explains O_{E_i}, meaning that O_{E_j} causes O_{E_i}. So, O_{E_i} can be removed from $\boldsymbol{O_E}$ and added to $\boldsymbol{O_O}$. Thus, we would maintain the condition that all pairs of distinct variables in $\boldsymbol{O_E}$ are unrelated by an edge or directed path. However, if we are not certain that O_{E_j} explains O_{E_i}, then we can leave O_{E_i}, O_{E_j} in $\boldsymbol{O_E}$.

Case 2 Suppose there exists some distinct pairs of variables $O_{E_i}, O_{E_j} \in \boldsymbol{O_E}$ such that O_{E_i} and O_{E_j} are uncorrelated. In this case, because O_{E_i} and O_{E_j} are observed effects, it may be impossible to find a single common cause explaining both nodes. Therefore, we must consider cases where the best explanation is a hypothesis containing multiple variables.

There may exist three distinct types of possible common causes:

1. Multivariate subsets of $\boldsymbol{Anc}(\boldsymbol{O_E})$ that are *complete explanations*.
2. Multivariate subsets of $\boldsymbol{Anc}(\boldsymbol{O_E})$ that are *partial explanations*.
3. Multivariate subsets of $\boldsymbol{Anc}(\boldsymbol{O_E})$ that are *novel explanations*.

Note that the possible common causes are now multivariate sets rather than singleton sets.

When the CCP criteria is not satisfied, there may not exist a single common cause of all the observed effects. In such cases, we must instead consider explanations that incorporate multiple variables.

Definition 4.3 (Multivariate Complete Explanations)**.** A *multivariate complete explanation* consists of multiple nodes whose joint set of descendants contains the set of observed effects as a subset. The set of multivariate complete explanations is given by

$$\{\boldsymbol{S} \subseteq \boldsymbol{Anc}(\boldsymbol{O_E}) \mid \boldsymbol{O_E} \subseteq \boldsymbol{Des}(\boldsymbol{S})\}. \tag{15}$$

We must also consider the existence of an observed effect whose explanation is beyond the scope of the model. This could occur when $\boldsymbol{O_E}$ contains noisy observed effects that cannot be sufficiently explained by the model.

A simple example of noisy observed effects in Bayesian networks are *root nodes*: nodes with empty ancestor sets. Since the ancestor sets of root nodes are empty, there cannot exist any causes or hypotheses that explain the root nodes. In such cases, the root nodes in $\boldsymbol{O_E}$ given by

$$\boldsymbol{O_R} = \{O_{E_i} \in \boldsymbol{O_E} \mid \boldsymbol{Par}(O_{E_i}) = \emptyset\}, \tag{16}$$

would be removed from $\boldsymbol{O_E}$ and added to $\boldsymbol{O_O}$.

If a noisy observation is not a root node, it remains in $\boldsymbol{O_E}$. To handle such cases, we include possible causes that do not have a directed path from the possible cause to every observed effect.

Definition 4.4 (Multivariate Partial Explanations)**.** A *partial explanation* consists of multiple nodes whose joint set of descendants contains a subset of $\boldsymbol{O}_E$. The set of partial explanations is given by

$$\{\boldsymbol{S} \subseteq \boldsymbol{Anc}(\boldsymbol{O}_E) \mid \exists O_{E_i} \in \boldsymbol{O_E}, O_{E_i} \in \boldsymbol{Des}(\boldsymbol{S})\}. \tag{17}$$

Lastly, we account for observed effects that descend from unfamiliar causes: causes of a given set of observed effects that were not observed in the background information. In these cases, we develop a hypothesis generation method to generate novel explanations of the unique causes.

Definition 4.5 (Novel Explanations)**.** A *novel explanation* is an explanation found through hypothesis generation.

Hypothesis generation refers to the introduction of new edges in the Bayesian network for the purpose of creating common causes of the observed effects. Generating an edge between two nodes entails the development of a causal relationship between them. The set of *generated edges* $\boldsymbol{E_{H_G}}$ of a hypothesis $\boldsymbol{H}$ will take the form

$$\boldsymbol{E_{H_G}} := \{(h_{\alpha_1}, h_{\beta_1}), ..., (h_{\alpha_g}, h_{\beta_g})\}. \tag{18}$$

However, the model is faced with a vast search space of nodes to generate new edges between, necessitating incorporation of bias in the search. Specifically, in searching for a set of unobserved nodes to generate edges between, we optimize for the number of observed effects that are descendants of the given set of unobserved nodes.

4.1.1 Implementation

Because there is uncertainty as to whether $\boldsymbol{O_E}$ satisfies the CCP criteria, we must include multivariate complete explanations, multivariate partial explanations, and novel explanations in the set of possible common causes.

Algorithm 1 identifies the sets of singleton complete explanations, multivariate complete explanations, and multivariate partial explanations, and it returns their union, defined as $\boldsymbol{C_P}(\boldsymbol{O_E})^+$. Algorithm 2 then uses $\boldsymbol{C_P}(\boldsymbol{O_E})^+$ to generate novel explanations, where $\boldsymbol{C_{P_G}}(\boldsymbol{O_E})$ is the set of novel explanations.

Algorithm 1 is motivated by the *Apriori algorithm* (Agrawal and Srikant, 1994) for inferring causal relations between sales items from large transaction datasets. The Apriori algorithm relies on the *apriori property*, a relational invariant between sets and subsets, in order to improve efficiency. We leverage a similar property in the following algorithm.

Let k represent the size of the candidate possible hypotheses set and k_T be a hyperparameter specifying

Algorithm 1: Computing Partial and Complete Explanations, $\boldsymbol{C_P}(\boldsymbol{O_E})^+$

```
Set $\boldsymbol{C_P}(\boldsymbol{O_E})^+ = \emptyset$;
Set $\boldsymbol{R}_{Prev} = \boldsymbol{Anc}(\boldsymbol{O_E})$;
Set $\boldsymbol{R}_{Curr} = \emptyset$;
for $k = k_T, k_T - 1, k_T - 2, \ldots, 1$ do
    if $\boldsymbol{R}_{Prev}$ is empty then
        Return $\boldsymbol{C_P}(\boldsymbol{O_E})^+$;
    end
    for $\boldsymbol{S} \subseteq \boldsymbol{R}_{Prev}$ such that $|\boldsymbol{S}| = k$ and
      $sim(\boldsymbol{R}_{Curr}, \boldsymbol{S}) < S_T$ do
        if $\frac{|\boldsymbol{O_E} \cap \boldsymbol{Des}(\boldsymbol{S})|}{|\boldsymbol{O_E}|} \geq P_T$ then
            Set $\boldsymbol{C_P}(\boldsymbol{O_E})^+ = \boldsymbol{C_P}(\boldsymbol{O_E})^+ \cup \{\boldsymbol{S}\}$;
            Set $\boldsymbol{R}_{Curr} = \boldsymbol{R}_{Curr} \cup \boldsymbol{S}$;
        end
    end
    Set $\boldsymbol{R}_{Prev} = \boldsymbol{R}_{Curr}$;
    Set $\boldsymbol{R}_{Curr} = \emptyset$;
end
Return $\boldsymbol{C_P}(\boldsymbol{O_E})^+$;
```

the maximum size k to be considered. The set $\boldsymbol{R}_{Curr}$ keeps track of variables for which the algorithm will compute smaller subsets. Whether a set of variables is added to $\boldsymbol{R}_{Curr}$ is determined by measuring the similarity of $\boldsymbol{R}_{Curr}$ to the new set. If the new set is above a certain threshold, specified by hyperparameter S_T, then the new set does not substantially add to the existing connection between the nodes in $\boldsymbol{R}_{Curr}$ and the observed effects, nor does the new set significantly increase the ability of $\boldsymbol{R}_{Curr}$ to explain the observed effects. In such a case, we ignore that set. Finally, P_T is a hyperparameter that specifies the percentage of the observed effects that must be connected to a possible hypothesis.

Next, we use $\boldsymbol{C_P}(\boldsymbol{O_E})^+$ to compute novel explanations. Specifically, Algorithm 2 computes $\boldsymbol{N}(\boldsymbol{O_E})$, a set of tuples that associates hypotheses with their corresponding generated edges. More precisely, for the members of $\boldsymbol{N}(\boldsymbol{O_E})$, the first element in each tuple is a hypothesis and the second element in the tuple is the hypothesis' corresponding set of generated edges, $\boldsymbol{E_{H_G}}$. Consequently, the set of novel explanations, $\boldsymbol{C_{P_G}}(\boldsymbol{O_E})$, is the set of the first elements in the tuples in $\boldsymbol{N}(\boldsymbol{O_E})$. Additionally, note that $\boldsymbol{C_P}(\boldsymbol{O_E})^+ = \boldsymbol{C_{P_G}}(\boldsymbol{O_E})$, but each hypothesis in $\boldsymbol{C_{P_G}}(\boldsymbol{O_E})$ contains a corresponding set of generated edges that defines new edges between nodes, resulting in a new explanation. As a technical note, the implementation of Algorithm 2 only iterates over partial explanations in $\boldsymbol{C_P}(\boldsymbol{O_E})^+$, since complete explanations contain nodes that are already connected to all observed effects.

Algorithm 2: Computing Novel Explanations

```
Set $\boldsymbol{N}(\boldsymbol{O_E}) = \emptyset$;
for $\boldsymbol{H} \in \boldsymbol{C_P}(\boldsymbol{O_E})^+$ do
    Set $\boldsymbol{O}_{Exc} = \boldsymbol{O_E} - (\boldsymbol{O_E} \cap \boldsymbol{Des}(\boldsymbol{H}))$;
    Set $\boldsymbol{E_{H_G}} = \emptyset$;
    for $O_i \in \boldsymbol{O}_{Exc}$ do
        Set $H_{Max} = \text{argmax}_{H_i \in \boldsymbol{H}} \text{sim}(O_i, H_i)$;
        Set $\boldsymbol{E_{H_G}} = \boldsymbol{E_{H_G}} \cup \{(H_{Max}, O_i)\}$;
    end
    Set $\boldsymbol{N}(\boldsymbol{O_E}) = \boldsymbol{N}(\boldsymbol{O_E}) \cup \{(\boldsymbol{H}, \boldsymbol{E_{H_G}})\}$;
end
Return $\boldsymbol{N}(\boldsymbol{O_E})$;
```

4.2 Hypothesis Comparison

Once we have identified the set of potential common causes $\boldsymbol{C_P}(\boldsymbol{O_E})^+$, we refine it by optimizing for $P(\boldsymbol{H} \mid \boldsymbol{O})$, the likelihood of a hypothesis, which serves a justificational function in the model.

It is important to note that $P(\boldsymbol{O} \mid \boldsymbol{H})$ is not a verification measure of the hypothesis' occurrence in a given situation, but it is rather a justificational component for the model's output set $\boldsymbol{C_P}(\boldsymbol{O_E})^+$. Each $\boldsymbol{H} \in \boldsymbol{C_P}(\boldsymbol{O_E})^+$ is theoretical and therefore unverifiable by an abductive model, but we can estimate a hypothesis' promise given the observed facts with the measure $P(\boldsymbol{O} \mid \boldsymbol{H})$.

4.2.1 Comparing Identified and Generated Hypotheses

In order to compare selected and generated hypotheses, we defined a cost function that optimizes for $P(\boldsymbol{H} \mid \boldsymbol{O})$, while creating a negative "cost" for generated edges.

The comparison function is

$$F(\boldsymbol{H}, \boldsymbol{E_{H_G}}, \boldsymbol{O}) := P(\boldsymbol{H} \mid \boldsymbol{O}) - \alpha c(\boldsymbol{E_{H_G}}), \tag{19}$$

where $c(\boldsymbol{E_{H_G}})$ is the cost of generating edges, defined below, α is a weighting hyperparameter, and $\boldsymbol{E_{H_G}}$ is the set of generated edges of hypothesis $\boldsymbol{H}$. The cost of generating edges is defined by

$$c(\boldsymbol{E_{H_G}}) := \sum_{(X_i, X_j) \in \boldsymbol{E_{H_G}}} (1 - \text{sim}(X_i, X_j)), \tag{20}$$

where $\text{sim}(X_i, X_j)$ is the similarity of X_i compared to X_j, according to the similarity metrics defined in Section 5. Note that for each hypothesis under consideration, the probability distribution for the particular hypothesis is updated such that for all $(O_i, H_i) \in \boldsymbol{E_{H_G}}$, $P(O_i \mid H_i) = 1$.

For all candidate hypotheses $\boldsymbol{H} \in \boldsymbol{C_P}(\boldsymbol{O_E})^+$, $F(\boldsymbol{H}, \boldsymbol{E_{H_G}}, \boldsymbol{O})$ is computed, and the hypotheses with the highest P_Z percent of scores are added to the set of promising hypotheses, $\boldsymbol{H_P}$, where P_Z is a hyperparameter used to control the selectivity of the process.

5 Similarity Metrics

In *hypothesis comparison*, we incorporate bias towards simple explanations. Simplicity in the graphical model is indicated by *structural dissimilarity*, as similar hypotheses can be deemed redundant. To gauge structural similarity between hypotheses, we introduce two metrics that determine node similarity: 1) *graph edit distance*, on the basis of edge weight, and 2) the *Jaccard Index of Variables*, on the basis of common descendants. In the search for plausible explanations within a probabilistic Bayesian network, disfavoring similarity reduces redundancy among hypotheses and produces simpler explanations. Intuitively, this bias reflects an innate understanding that objects with similar properties and behaviors produce similar outcomes, and vice versa.

5.1 Similarity Metric: Jaccard Index of Variables

The *Jaccard Index* is defined as

$$J(A,B) := \frac{|A \cap B|}{|A \cup B|}, \tag{21}$$

where A and B are sets.

The model uses what we define as the *Jaccard Index of Variables*, a Jaccard Index-inspired metric of similarity between variables in $\boldsymbol{V}$. But rather than relying solely on set cardinalities, we use the probability distribution P. Thus, we adapt the Jaccard Index to compute similarity between two variables A and B in the Bayesian network based on their children. The children of A and B are denoted as $\boldsymbol{Ch}(A) = \boldsymbol{A_C}$ and $\boldsymbol{Ch}(B) = \boldsymbol{B_C}$. The Jaccard Index of Variables is defined as

$$\begin{aligned}\mathrm{sim}_{\mathrm{Jaccard}}(A,B) &:= \frac{(\boldsymbol{Ch}(A) \cap \boldsymbol{Ch}(B))}{(\boldsymbol{Ch}(A) \cup \boldsymbol{Ch}(B))}, && (22)\\ &= \frac{(\boldsymbol{A_C} \cap \boldsymbol{B_C})}{(\boldsymbol{A_C} \cup \boldsymbol{B_C})}, && (23)\end{aligned}$$

where

$$\begin{aligned}(\boldsymbol{A_C} \cup \boldsymbol{B_C}) &:= |\boldsymbol{A_C} - \boldsymbol{B_C}| + (\boldsymbol{A_C}) + (\boldsymbol{B_C})\\ &\quad - (\boldsymbol{A_C} \cap \boldsymbol{B_C}), && (24)\\ (\boldsymbol{A_C} \cap \boldsymbol{B_C}) &:= \sum_{C \in \boldsymbol{A_C} \cap \boldsymbol{B_C}} \min(P(C|A), P(C|B)) && (25)\end{aligned}$$

and

$$\begin{aligned}(\boldsymbol{A_C}) &:= \sum_{C \in \boldsymbol{A_C} \cap \boldsymbol{B_C}} P(C \mid A), && (26)\\ (\boldsymbol{B_C}) &= \sum_{C \in \boldsymbol{A_C} \cap \boldsymbol{B_C}} P(C \mid B). && (27)\end{aligned}$$

5.2 Similarity Metric: Edit-Distance

We refer to *edit-distance* (Bunke, 1997) as the cost of operations required to transform one graphical structure into another. Edit-distance is the basis of our *edit-distance based similarity* metric, which is defined as

$$\begin{aligned}\mathrm{sim}_{\mathrm{Edit}}(A,B) &:= \frac{c(A,B)}{|\boldsymbol{Ch}(A)|} && (28)\\ &= \frac{c(A,B)}{|\boldsymbol{A_C}|}, && (29)\end{aligned}$$

where $c(A,B)$ is a *cost function* to measure the cost of graph changing operations such that

$$c(A,B) := \sum_{C \in \boldsymbol{A_C} \cap \boldsymbol{B_C}} \|P(C \mid A) - P(C \mid B)\| + |\boldsymbol{A_C} - \boldsymbol{B_C}|, \tag{30}$$

where $\|P(C \mid A) - P(C \mid B)\|$ is the absolute value of the difference $P(C \mid A) - P(C \mid B)$.

5.3 Similarity of Sets of Variables

In addition to computing the similarity of variables, we can compute the similarity of sets of variables. Specifically, for some similarity metric $\mathrm{sim}(\cdot,\cdot)$, we can compute

$$\mathrm{sim}(\boldsymbol{A},\boldsymbol{B}) := \frac{\sum_{X_i \in \boldsymbol{A}} \mathrm{sim}(X_i, X_j^*)}{|\boldsymbol{A}|} \tag{31}$$

where $X_j^* := \mathrm{argmax}_{X_j \in \boldsymbol{B}} \mathrm{sim}(X_i, X_j)$. The algorithm for computing the similarity of sets of variables is given below in Algorithm 3.

Algorithm 3: Computing Similarity of Sets of Variables, $\mathrm{sim}(\boldsymbol{A},\boldsymbol{B})$

```
Set total = 0;
for $X_i \in \boldsymbol{A}$ do
    Set $X_j^* = \mathrm{argmax}_{X_j \in \boldsymbol{B}} \mathrm{sim}(X_i, X_j^*)$;
    Set total = total + $\mathrm{sim}(X_i, X_j^*)$;
end
Set $\mathrm{sim}(\boldsymbol{A},\boldsymbol{B})$ = total/$|\boldsymbol{A}|$;
Return $\mathrm{sim}(\boldsymbol{A},\boldsymbol{B})$;
```

6 APPLICATIONS

6.1 Database Corruption

Using the hypothesis identification, generation, and comparison methods previously described, we can demonstrate hypothesis selection within the example of a corrupted database, which we discussed in Section 1.

First, we will construct a graphical model to represent the situation. Figure 1 gives a hypothetical graphical representation of the corrupted database example to illustrate our process, with nodes that are related by an arbitrary conditional probability table.

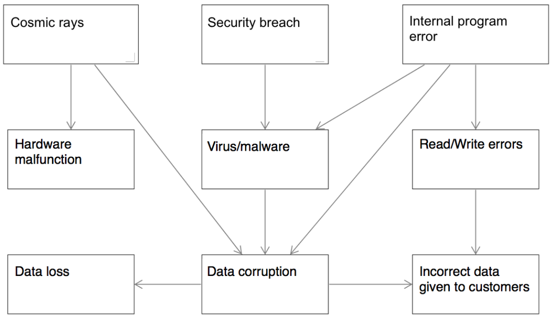

Figure 1: Data Corruption Example DAG

After inputting background information and observed effects/surprising phenomena to the current model, we can identify every potential hypothesis—both probable and improbable. For example, we could have observed Hardware Malfunction in conjunction with Data Corruption.

Some of these hypotheses could be overly-complicated and redundant. For example, a possible explanation could be the simultaneous occurrence of an Internal Program Error and a Virus/Malware. Another possible explanation could that only a Virus/Malware caused Data Corruption. These two hypotheses share common components and effects, but we can reduce the complexity of our hypothesis by simply selecting Virus/Malware. In these cases of redundant hypotheses, our model utilizes the similarity metrics to identify the simple hypothesis.

Our first step, which we refer to as observation-testing, is calculating $P(\boldsymbol{O} \mid \boldsymbol{H_i})$ for some hypothesis $\boldsymbol{H_i}$. Given a potential hypothesis, we will calculate the probabilities of the observed effects. However, due to the graph structure, we need to calculate the marginal probability, which takes into account all relevant nodes, regardless of whether they are in our hypothesis set or not. We will then choose the top $n\%$ of these hypotheses to move on to the second phase. Having found hypotheses that best explain the observed effects, the second step is hypothesis refinement: calculating $P(\boldsymbol{H_i} \mid \boldsymbol{O})$ for some hypothesis $\boldsymbol{H_i}$. Given the set of hypotheses from observation-testing, we will then identify and choose the hypothesis that is most probable given our background information. As a result of this step, we are given the most probable hypothesis that also adequately explains our observed effects.

Suppose that our first step selected two potential hypotheses to move forward into stage two: Cosmic Rays, which yielded a probability of .93, and Virus/Malware, which yielded a probability of .71. However, during phase two, suppose Virus/Malware yielded a probability of .67, while Cosmic Rays yielded a probability of .002. Virus/Malware would have a higher final probability and would therefore be chosen as our final hypothesis.

We continue the data corruption example to demonstrate hypothesis generation. Consider a new situation where we are given Internal Program Error and Security Breach as potential explanations for the observed occurrences of both Hardware Malfunction and Read/Write Errors. Note that while Internal Program Error is a parent node to Read/Write Errors, it is not an ancestor for Hardware Malfunction. Also, Security Breach is not an ancestor for Hardware Malfunction nor Read/Write Errors. This means that we have no common cause hypothesis for the observed effects. In this case, the model will generate a common cause hypothesis using the edge generation process described in Section 4.1.

An edge will be introduced to connect a hypothesis node with a new child node that is also an observed effect. In the current example, neither of our potential hypotheses would cause a Hardware Malfunction, yet a Hardware Malfunction has been observed. So, two edges are generated: one connecting Security Breach to Hardware Malfunction, and one connecting Internal Program Error to Hardware Malfunction. Security Breach and Internal Program are now novel common cause hypotheses, and will be reevaluated using the observation-testing and hypothesis refinement methods previously described.

6.2 The Wet Grass Network

As an additional example, let us consider the Wet Grass Bayesian network (Bayes Server, 2020), shown in Figure 2. Using our similarity methods, the algorithm successfully avoids choosing both “S” (Sprinkler) and “R” (Rain) to increase the probability when the two are not given. However, when either “S” or “R” is an observation, the algorithm chooses the cor-

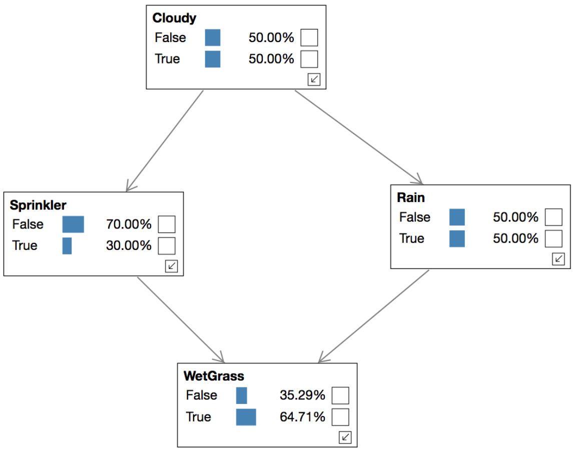

Figure 2: Wet Grass Bayes Net

responding node to increase the probability of grass being wet. However, it appears to strongly weight hypotheses that perform well in the first stage, especially if the graph is small. As an example, when given "W" (WetGrass) while observing "C" (Cloudy), the algorithm chooses "S" as the cause, as it performs better in the first phase and there are only two hypotheses, "S" and "R", and the algorithm chooses the top half for these testing purposes. Overall, this system chooses simple hypotheses with no extraneous information that aim to increase probability of the observed effects, while also taking into account the probability of the hypotheses.

6.3 Discussion

While our algorithms produce reasonable explanations for small graphs, it remains to be seen how well these methods scale to exponentially larger networks, and how they adapt to non-Bayesian probability theories or graphical models. As such, we view this work as a preliminary investigation into using probabilistic structures to develop abductive explanations, in comparison to the large body of previous work in abductive computation which has focused primarily on symbolic methods and formal logic (Ng and Mooney, 1992; Mooney, 2000; Juba, 2016; Ignatiev et al., 2019).

7 CONCLUSION

The purpose of our abductive search model is to develop plausible explanations for surprising phenomena. Approaching this scenario as a search problem, we are interested in finding our search target, which is a set of the most promising potential hypothetical causes for the unexplained observed events. The fit of a hypothesis relative to known data is measured by $P(\boldsymbol{O} \mid \boldsymbol{H})$, which is the probability of the observed events occurring given that the hypothesis is also true. So, we want a set of hypotheses $\boldsymbol{C_P} = \boldsymbol{H}_1, \dots, \boldsymbol{H_m}$ that, for each $\boldsymbol{H_i} \in \boldsymbol{C_P}$, 1) optimizes the measure of fit $P(\boldsymbol{O} \mid \boldsymbol{H_i})$, and 2) has a high $P(\boldsymbol{H_i} \mid \boldsymbol{O})$ value relative to other hypotheses.

Having established criteria for a search target, we apply an abductive search strategy to find these promising potential hypotheses. We deemed abduction as the most effective form of inference for addressing such problems, including the data corruption one, where a) known information is incomplete and b) a set of novel hypotheses are the search target. We used graphical models—specifically, Bayesian networks—to represent both the causal relationships within a search space and the elements of abductive search.

We present hypothesis *selection*, *generation*, and *comparison* as the primary methods for finding promising potential hypotheses. Our hypothesis selection criteria is based upon Reichenbach's Common Cause Principle. In the case that no common cause hypothesis exists, we rely on hypothesis generation to produce novel potential common cause hypotheses. These generated hypotheses can be a) multivariate hypotheses, b) partial explanations, or c) generated edges. Then, having obtained the set of all potential hypothetical causes, we subject the individual elements to hypothesis comparison, selecting the hypothesis that maximizes $P(\boldsymbol{H} \mid \boldsymbol{O})$.

Future research on probabilistic abduction's explanatory capabilities can be conducted with regards to search algorithms in general, which are often viewed as black-box methods. We also see Bayes' Theorem as a tool for bridging theoretical abductive search methods—such as those presented in this paper, those in (Schurz, 2008), and those in (Cox et al., 1992)—with machine learning classification from incomplete or fuzzy data.

Other future work includes analyzing algorithmic scaling and implementing methods such as dynamic programming to reduce algorithm runtime, as well as further exploring effective hypothesis generation methods by examining the effects of edge generation on hidden variables in a Bayesian network and the implications of successive edge generation on our model's predictive accuracy.

Acknowledgements

This research was supported in part by Harvey Mudd College and the National Science Foundation under

Grant No. 1950885. Any opinions, findings or conclusions expressed are the authors' alone, and do not necessarily reflect the views of Harvey Mudd College or the National Science Foundation.

REFERENCES

Agrawal, R. and Srikant, R. (1994). Fast Algorithms for Mining Association Rules. *Proceeding of the 20th VLDB Conference*, pages 487–499.

Bayes Server (2020). Live Examples. https://www.bayesserver.com/. Accessed: 2020/09/25.

Bunke, H. (1997). On a relation between graph edit distance and maximum common subgraph. *Pattern Recognition Letters*, 18(8):689–694.

Chen, Y.-C., Wheeler, T. A., and Kochenderfer, M. J. (2017). Learning discrete Bayesian networks from continuous data. *Journal of Artificial Intelligence Research*, 59:103–132.

Cox, P., Knill, E., and Pietrzykowski, T. (1992). Abduction in logic programming with equality. In *Proceedings of the Eighth International Conference on Fifth Generation Computer Systems*.

Feldbacher-Escamilla, C. J. and Gebharter, A. (2019). Modeling creative abduction Bayesian style. *European Journal for Philosophy of Science*, 9(1):9.

Friedman, N., Goldszmidt, M., et al. (1996). Discretizing continuous attributes while learning Bayesian networks. In *Proceeding of the Thirteenth International Conference on Machine Learning*, pages 157–165.

Ignatiev, A., Narodytska, N., and Marques-Silva, J. (2019). Abduction-based explanations for machine learning models. In *Proceeding of the Thirty-Third AAAI Conference on Artificial Intelligence*, volume 33, pages 1511–1519.

Juba, B. (2016). Learning abductive reasoning using random examples. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 999–1007. Citeseer.

Mooney, R. J. (2000). Integrating abduction and induction in machine learning. *Abduction and Induction*, pages 181–191.

Ng, H. T. and Mooney, R. J. (1992). A First-Order Horn-Clause Abductive System and Its Use in Plan Recognition and Diagnosis. Technical report, Department of Computer Sciences, The University of Texas at Austin.

Pearl, J. (1998). Graphical models for probabilistic and causal reasoning. *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, 1:367–389.

Prendinger, H. and Ishizuka, M. (2005). A creative abduction approach to scientific and knowledge discovery. *Knowledge-Based Systems*, 18(7):321–326.

Reiter, R. (1987). A theory of diagnosis from first principles. *Artificial intelligence*, 32(1):57–95.

Schurz, G. (2008). Patterns of abduction. *Synthese*, 164(2):201–234.