Learning Abductive Reasoning Using Random Examples

Brendan Juba[*]
Washington University in St. Louis
bjuba@wustl.edu

Abstract

We consider a new formulation of *abduction* in which degrees of "plausibility" of explanations, along with the rules of the domain, are *learned* from concrete examples (settings of attributes). Our version of abduction thus falls in the *"learning to reason"* framework of Khardon and Roth. Such approaches enable us to capture a natural notion of "plausibility" in a domain while avoiding the extremely difficult problem of specifying an explicit representation of what is "plausible."

We specifically consider the question of which syntactic classes of formulas have efficient algorithms for abduction. We find that the class of k-DNF explanations can be found in polynomial time for any fixed k; but, we also find evidence that even weak versions of our abduction task are intractable for the usual class of *conjunctions*. This evidence is provided by a connection to the usual, inductive PAC-learning model proposed by Valiant. We also consider an exception-tolerant variant of abduction. We observe that it is possible for polynomial-time algorithms to tolerate a few adversarially chosen exceptions, again for the class of k-DNF explanations. All of the algorithms we study are particularly simple, and indeed are variants of a rule proposed by Mill.

1 Introduction

Abduction is the process of passing from an observation to a plausible explanation or diagnosis. For example, to understand a story in which a man is holding a gun in a bank, one must "abduce" that (perhaps) the man wishes to rob the bank. This is not a sound inference, of course – the man could be a guard, the man could be seeking to place the firearm in a safe deposit box, etc. – but it represents at least a highly plausible explanation for the given facts. Unlike the usual forms of inference of deduction and induction, abduction as a form of inference was only brought to prominence relatively recently, by Pierce (1931). It was then promoted as a core task in AI by Charniak and McDermott (1985). It has since been observed that problems as diverse as diagnosis (Reggia 1983; Reiter 1987), image understanding (Cox and Pietrzykowski 1986; Poole 1990), natural language understanding (Hobbs et al. 1990), planning (Eshghi 1988; Missiaen, Bruynooghe, and Denecker 1995), and plan recognition (Charniak and McDermott 1985) all involve abduction. We will discuss applications slightly further in Section 5.

As we will review, the task of abduction itself has already been formalized in (at least) three distinct ways, and we propose a new formalization of abduction using *examples*; the examples here consist of settings of the various attributes, e.g., encoding a concrete "scene" or "episode" in the image units of Valiant's neuroidal model (2000a), or more abstractly, as in the entries in a database. We assume these examples to have been drawn independently from a common unknown distribution D, modeling the frequency with which such scenes occur in a domain. We formulate the task as searching for a *conditional distribution*: in addition to this data, we are given a Boolean query that we wish to "explain" in a sense we will elaborate on below, and a class of Boolean formulas $\mathcal{H}$ over a distinguished set of attributes of the data A. A indicates the attributes that we wish to allow in our explanations, for example, attributes that took values prior to the condition to be explained, and might therefore be predictive. We then seek to find a (hypothesis) formula h in the class $\mathcal{H}$ using only the attributes in A such that

(i) *Approximate validity:* the query is (almost) always true on examples drawn from the conditional distribution $D|h$, i.e., the distribution over assignments induced by D given that h is satisfied, and

(ii) *Plausibility:* the probability of h being true under D is at least some minimum value μ^*. We will often seek to find a h attaining as large a plausibility μ as possible.

So, h is an "explanation" in the sense that the query empirically follows from h, and h holds sufficiently often. For example, in our model, the query might indicate whether or not the key facts of the story – the gun in the bank – are present in the examples, and the property h to be found represents the desired "explanation." A slightly more precise example that we will return to later involves formulating a diagnosis. For example, we may have various attributes about the state of a car, and the query may be something like $\mathsf{key\_turned} \wedge \neg\mathsf{engine\_running}$, for which we may seek an explanation such as $h = \mathsf{key\_turned} \wedge \neg\mathsf{gas\_in\_tank}$: although $\Pr[\mathsf{key\_turned} \wedge \neg\mathsf{gas\_in\_tank}]$ may be low, it occa-

[*] Work partially performed while the author was affiliated with Harvard University and supported by ONR grant number N000141210358. Currently supported by an AFOSR Young Investgator award.

sionally may happen, and surely

$$\Pr\left[\begin{array}{r|r}\mathsf{key\_turned}\wedge & \mathsf{key\_turned}\wedge \\ \neg\mathsf{engine\_running} & \neg\mathsf{gas\_in\_tank}\end{array}\right]=1.$$

So, $\mathsf{key\_turned} \wedge \neg\mathsf{gas\_in\_tank}$ may be a sufficiently plausible explanation for $\mathsf{key\_turned} \wedge \neg\mathsf{engine\_running}$. We will return to this example when we discuss our algorithms.

We can refer directly to notions of (approximate) "validity" and "entailment" in this model because we assume that we have completely specified examples, on which the various formulas can be evaluated directly, much as in model-based reasoning (Kautz, Kearns, and Selman 1995).[1] We stress that all of the entailment constraints of the domain are thus properties of the distribution D, and are therefore represented implicitly in the problem formulation by the examples drawn from D.

In other words, this entailment relation underlying our abductive reasoning task must be *learned* from the examples drawn from D. We seek computationally efficient algorithms with a "PAC" guarantee: that with high *probability* (over the examples) both conditions are *approximately* satisfied. (We will define the task more formally in Section 2.) Our model thus belongs to the *learning to reason* framework of Khardon and Roth (1997b). Indeed, Khardon and Roth suggested in that work that such a learning formulation of abduction was possible, although they did not actually present a formalization of abduction in their framework. In a later work, Roth (1996) also briefly indicated that various aspects of an abduction task could be carried out in neural networks (and thus learned) but again did not elaborate on the semantics of the task. Similarly, in their work on model-based reasoning, Kautz et al. (1995) likewise both considered model-based abduction and indicated that model-based reasoning could use random examples, but again did not actually formally specify the semantics of the task. Our work is thus (perhaps surprisingly) the first to explicitly consider this formulation of abduction.

It is already widely appreciated that learning is a highly effective alternative to explicit knowledge engineering. Indeed, machine learning (e.g., from examples) has been far more effective than traditional knowledge engineering at acquiring robust representations across a variety of domains and tasks. Relatedly, Valiant (1994; 2000a) argued that learned representations should enable systems to better cope with an open world, and thus learning should be used as a basis for robust cognition. But, the main motivation for treating abduction itself as a learning task as we do is that it provides a means of efficiently capturing natural, domain-specific notions of the "plausibility" of explanations.

In particular here, we avoid the need to explicitly represent a (prior) distribution. Both the probabilistic version of the "set covering" model of abduction presented by Bylander et al. (1991) and the models of abduction based on probabilistic graphical models (Pearl 1988; Poole 1993) were Bayesian models that depended on assigning some such *prior probabilities* to the various *explanations.* In these models, "plausibility" of the proposed conditions is evaluated in terms of these probabilities. The need to estimate such a prior was deemed to be one of the main drawbacks of these previous probabilistic models of abduction (McIlraith 1998), as good priors are hard to estimate. In this way, our formulation thus differs crucially from these previous models. This is also how our formulation differs from, for example that of Hobbs et al. (1990), in which explicit weights or costs (without necessarily having a probabilistic interpretation) are attached to the various literals to be used in an explanation. Hobbs et al. briefly suggest that an interpretation in terms of conditional probabilities might be used to obtain such weights, but the assignment of specific weights to the literals is problematic unless they refer to events that are for example either disjoint or uncorrelated. This is a recurring limitation of the proposals that attach weights or probabilities directly to attributes.

Moreover, in the non-probabilistic logic-based or logic programming (Denecker and Kakas 2002) approaches, the usual syntactic criteria, such as minimizing the number of literals as done in ATMS (Reiter and de Kleer 1987), appear to serve essentially a proxy for some other kind of unspecified domain-specific "plausibility" notion, by appealing to something like Occam's razor. McIlraith (1998) nicely discusses the problems that may arise with these approaches, for example they are highly representation-dependent. Previous works on combining learning and abduction simply used such syntactic minimization criteria for plausibility (Thompson and Mooney 1994; Flach and Kakas 2000). Another probabilistic formulation, proposed by Bacchus et al. (1996) proposed to use a maximum-entropy distribution over the attributes as a prior, which is essentially similar to these syntactic criteria. In particular, it is also representation language dependent and may be simply inappropriate.

Our Results

The main question we consider in this work is, for which classes of hypothesis formulas $\mathcal{H}$ do efficient algorithms abduce explanations in our new model?[2] We find that a particularly simple algorithm abduces k-DNF explanations. We further generalize this algorithm to provide some weak *"exception tolerance"*: if the best k-DNF explanation only gives the query conditional probability $1-\epsilon$ for some $\epsilon > 0$, then the exception-tolerant algorithm finds a k-DNF that gives the query conditional probability $1 - O(n^k\epsilon)$ when there are n attributes in the vocabulary. That is, the probability of counterexamples to the "explanation" we find may be $O(n^k)$ times greater than that of the best possible explanation.[3] Thus, we see that this new abductive reasoning task

[1] We note that the role of *proofs* in such models is that they serve as a means to decide entailment under incomplete information. We leave the extension of this model to partial information as a subject for future work.

[2] We do not limit the class of representations that the query is drawn from, apart from assuming that it can be evaluated efficiently on an example. The complexity of the query representation appears to be largely irrelevant in this model.

[3] Although this $O(n^k)$ increase is indeed somewhat large, we stress that it should be contrasted with the state-of-the-art in such exception-tolerant supervised learning of k-DNFs, which similarly

is feasible for some natural classes of explanations, even in the presence of some noise or rare exceptions.

We also stress that both the algorithms and the query representations we use in these results are particularly simple, and interpretable by humans. In particular, the algorithms, which essentially eliminate terms when they encounter (too many) bad examples for these terms, follow a classical human strategy for identifying hypotheses proposed by Mill (1843, Book III, Chapter 8). We therefore view our algorithms and the representations we produce as being cognitively plausible, although our model did not strictly require it.

On the other hand, we find that abducing the usual class of *conjunctions* as explanations is likely to be intractable, even if we allow a richer representation as an explanation: Any algorithm that finds explanations whenever there exists a conjunction that is an explanation would yield an algorithm for PAC-learning DNF formulas in the standard PAC-learning model (Valiant 1984). This has been the central open problem in PAC-learning since the model was first proposed by Valiant (1984), and recent results by Daniely et al. (2014) and Daniely and Shalev-Shwartz (2014) show that the problem may be intractable, given a new assumption about the hardness of refuting random k-SAT instances (stronger than Feige's assumption (2002)). Since most of the usual classes of representations can either be expressed by k-DNF formulas or can themselves express conjunctions, this result together with our algorithms essentially settles the question of which representations have efficient algorithms in our model.

2 Abduction for k-DNF Explanations

In this work, we are seeking to find explicit representations of "explanations" of possibly low, but non-negligible probability for which conditioned on the corresponding event, some given *query* condition to be *explained* or *diagnosed* is (almost) always satisfied. Our probability distributions will not be given explicitly, but instead will be represented by examples drawn from the distributions in question. Formally, we focus on the following class of problems:

Definition 1 (Abduction) *For a* representation class $\mathcal{H}$ *of Boolean formulas over propositional attributes* $x_1, \ldots, x_n$*, the (proper)* abduction *task is as follows. We are given as input* m *independent examples* $x^{(1)}, \ldots, x^{(m)}$ *from an arbitrary distribution* D *over* $\{0,1\}^n$ *(assignments to the* n *attributes), a* query *formula* $c(x)$ *over* $x_1, \ldots, x_n$*, and an* alphabet $A \subseteq \{x_1, \ldots, x_n\}$*, for which there exists* $h^* \in \mathcal{H}$ *only using attributes in* A *such that* $\Pr[c(x) = 1 | h^*(x) = 1] = 1$ *and* $\Pr[h^*(x) = 1] \geq \mu$*. Then, with probability* $1-\delta$*, in time polynomial in* $n, 1/\mu, 1/\epsilon$*, and* $1/\delta$*, we find an* explanation $h \in \mathcal{H}$ *only using attributes in* A *such that*

1. $\Pr[c(x) = 1 | h(x) = 1] \geq 1 - \epsilon$ *and*

2. $\Pr[h(x) = 1] \geq \Omega\left(\left((1-\epsilon)\frac{\mu}{n}\right)^d\right)$ *for some* $d \in \mathbb{N}$*.*

(We will write events $c(x) = 1$ *as* c *and* $h(x) = 1$ *as* h*.)*

So, in our car diagnosis example, suppose $\mathcal{H}$ is the class of 2-DNF formulas, and for the query $c = \mathsf{key\_turned} \wedge \neg\mathsf{engine\_running}$ we omit the attribute $\mathsf{engine\_running}$ from A in order to avoid producing unenlightening formulas that might otherwise "diagnose c" with properties that use $\neg\mathsf{engine\_running}$, such as c itself. Then, $\mathsf{key\_turned} \wedge \neg\mathsf{gas\_in\_tank}$, being a conjunction of two literals (a *term* of size two), indeed counts as a 2-DNF and furthermore does not mention $\mathsf{engine\_running}$. Therefore, as argued in the Introduction, we expect that this formula would be a suitable explanation $h \in \mathcal{H}$ solving the abduction task.

Remarks on the Definition

Naturally, the running time and sample complexity generally depend polynomially on the number of attributes n, probability of observing the condition μ, and degree of approximation ϵ desired. As with PAC-learning, we will actually obtain running times that only depend polynomially on $\log 1/\delta$ rather than $1/\delta$ (but in general we might be satisfied with the latter). Furthermore, we could consider an *"improper"* version of the problem, finding representations from some larger, possibly more expressive class than the $\mathcal{H}$ containing the "optimal" hypothesis h^*. The form of the representation is naturally important for some applications (we will discuss an example in Section 5), though, and so in this work we focus primarily on the proper version of the problem.

We formulated the second condition to include a relaxed notion of plausibility, in which the probability of the diagnosis/explanation only needs to be polynomially related to the promised minimum probability of an explanation μ; we also allowed the loss of some polynomial factors in the dimension n, and multiplicative $1-\epsilon$ losses in the approximation. Our positive results, for k-DNFs, *do not* require any such notion of approximation—whenever a condition h^* that is satisfied with probability at least μ exists, our algorithms will actually find a condition h that is also satisfied with probability at least μ. But, the value of such an abstract definition is in the power it grants to establish (1) connections to other models, in which case, it is quite typical to obtain a $(1-\epsilon)$ multiplicative approximation, hence the use of ϵ as a generic "approximation" parameter, and (2) *negative* results, in which we would like the broadest possible definition. Indeed, we will see that for some extremely simple representations – specifically, conjunctions – the abduction task is unfortunately unlikely to have efficient algorithms, even in this very liberal sense in which we allow arbitrary polynomial dependence on the dimension n and the optimum probability μ.

We note that we could also have formulated the second condition as finding a h that makes $\Pr[h|c]$ suitably large, that is, that h is sufficiently "plausible" given that c is known or presumed to be true.[4] We show in Appendix A that this formulation of the abduction task is essentially equivalent to the definition we use.

suffers a $O(n^{k/3})$ blow-up of the error (Awasthi, Blum, and Sheffet 2010).

[4] We thank a reviewer for suggesting this formulation.

The Elimination Algorithm for Abduction

Valiant (1984) gave an analysis showing that the "elimination algorithm" (Algorithm 1), a simple method essentially proposed by Mill (1843, Book III, Chapter 8), is a PAC-learning algorithm for disjunctions. We note that the same algorithm also can be used to identify a k-DNF explanation (in our sense) quite efficiently.

```
input : Examples $x^{(1)}, \ldots, x^{(m)}$, query $c$, alphabet $A$.
output: A $k$-DNF over attributes from $A$.
begin
    Initialize $h$ to be the disjunction over all terms of
    at most $k$ literals on the alphabet $A$.
    for $i = 1, \ldots, m$ do if $c(x^{(i)}) = 0$ then
        forall the $T \in h$ do if $T(x^{(i)}) = 1$ then
        Remove $T$ from $h$.

    end

    return $h$.
end
```

Algorithm 1: Elimination algorithm

Theorem 2 *The abduction task for k-DNF explanations can be solved by the elimination algorithm using* $O(\frac{1}{\mu\epsilon}(n^k + \log 1/\delta))$ *examples, obtaining a k-DNF h with* $\Pr[h] \geq \mu$.

Proof: For the given sample size, it is immediate that the elimination algorithm runs in polynomial time. We thus need only establish correctness.

Initially every term of size at most k on A is contained in h, so h contains all of the terms of h^*. We claim that this invariant is maintained: since $\Pr[c|h^*] = 1$, whenever $c(x^{(i)}) = 0$, it must be that every term of h^* is falsified on $x^{(i)}$. Thus, these terms are not removed during any iteration, so they are included in the final k-DNF h. It therefore follows that since $\Pr[h^*] \geq \mu$ and $\{x : h^*(x) = 1\} \subseteq \{x : h(x) = 1\}$, $\Pr[h] \geq \mu$.

We now bound the probability that the algorithm terminates with a h such that $\Pr[c|h] < 1 - \epsilon$. We note that for such h, $\Pr[\neg c \wedge h] > \epsilon \Pr[h]$. For any h with $\Pr[h] \geq \mu$, it follows that $\Pr[\neg c \wedge h] > \mu\epsilon$. Thus, for $N \overset{\text{def}}{=} \sum_{i=1}^{k} \binom{2n}{i} \leq \left(\frac{2en}{k}\right)^k$ (the number of terms of size at most k), after $m = \frac{1}{\mu\epsilon}(N \ln 2 + \ln 1/\delta)$ examples, the probability that such a h is falsified when $c(x^{(i)}) = 0$ for all i is at most $(1 - \mu\epsilon)^m \leq e^{-\ln(2^N/\delta)} = \delta/2^N$. So, by a union bound over the (at most) 2^N such k-DNFs, we find that the probability of any of them being false on every example is at most δ. Since the h we output is guaranteed by construction to be false for every $x^{(i)}$ where $c(x^{(i)}) = 0$, we find that with probability $1 - \delta$, $\Pr[c|h] \geq 1 - \epsilon$ as needed. ■

For our example problem of generating a k-DNF diagnosis for key_turned ∧ ¬engine_running given a set of example situations, the elimination algorithm simply rules out all of the terms which were true in examples where either key_turned was false or engine_running was true. For $k \geq 2$, this will yield a disjunction of terms that includes the term key_turned ∧ ¬gas_in_tank, possibly among some others that either are possible explanations or never occurred in our examples. For example, perhaps wiper_fluid_low ∧ ¬tire_pressure_ok never occurred in any example. It is intuitively unlikely to be an *explanation* of the query, but without examples in which wiper_fluid_low ∧ ¬tire_pressure_ok actually occurred, it remains in the h we return as a *possible* explanation (that has not yet been ruled out). The disjunction of such terms is our maximally plausible condition.

We note that we could modify the algorithm to also eliminate such spurious terms that never occurred in any example; so, in our example scenario, this will only leave those terms that have been observed to be satisfied when both key_turned and ¬engine_running hold. Then as long as this modified algorithm is provided with $O(\frac{n^k}{\mu\epsilon}(k \log n + \log 1/\delta))$ examples, it still returns explanations with plausibility at least $(1 - \epsilon)\mu$. Indeed, letting N continue to denote the number of terms of size at most k, if a term is true with probability greater than $\mu\epsilon/N$ over D, then $\frac{N}{\mu\epsilon}(\log N + \log 2/\delta)$ examples only fail to include an example of such a term being satisfied with probability at most $\delta/2N$. Thus, with probability $1 - \delta/2$, the only terms of the optimal h^* that may be deleted are only individually satisfied with probability $\epsilon\mu/N$ over D—in aggregate, with probability at most $\epsilon\mu$. Hence with probability at least $(1 - \epsilon)\mu$ over D, h^* must be satisfied by some term that is included in the returned h. The rest of the analysis is the same.

3 Exception-Tolerant Abduction

We now consider a variant of our basic model in which no explanation entails the target condition with conditional probability 1. For example, we may be looking for an explanation for the observation that "Tweety flies," but no representation in our class may possibly perfectly explain "flying"—we may have that 99.99% of the birds we have observed fly, but as this is less than 100%, our earlier method cannot propose flying if we have seen enough examples of birds, since an encounter with a counterexample causes terms of the desired explanation to be deleted. Or, in our car example, suppose that we would like to identify a condition that entails engine_running. The *qualification problem* (McCarthy 1980) is that it is generally impossible to identify a condition that strictly entails such things. But, in the vast majority of cases, we expect that when key_turned holds, so does engine_running. Formally, this is true if ¬gas_in_tank and other, perhaps even unformalized conditions are indeed relatively rare occurrences, say occurring no more than 0.1% of the time in total. We would like an alternative framework that allows us to find such slightly imperfect explanations.

We will *not* assume any additional structure on the errors, e.g., that they are the result of independent noise. This "agnostic" formalization captures the philosophy that the world may actually be described precisely by complicated rules, but we wish to merely find an explanation that is often sufficient. This is one possible PAC-learning style solution to the qualification problem in general (see works by Roth (1995) and Valiant (1994; 1995; 2006) for more on this aspect).

We observe that a simple variant of the elimination algorithm achieves a weak kind of exception tolerance: suppose that we only delete the literals that are true on more than $8\mu\epsilon m$ examples where c is false out of the m total examples. (8 is a convenient constant larger than 1.) Then:

Theorem 3 *If there is a k-DNF h^* with probability at least $\mu/4$ and at most 4μ that gives c conditional probability $1-\epsilon$, then given $\Omega(\frac{1}{\mu\epsilon}(\log\frac{n^k}{\delta}))$ examples, we find that the above ϵ-tolerant elimination algorithm obtains a k-DNF explanation with probability at least that of h^* of being satisfied, under which c is true with probability $1-O(n^k\epsilon)$.*

So, for example if key_turned $\wedge$ ¬engine_running only holds $\sim 0.1\%$ of the time, and our car is described by ~ 100 possible terms (say pairs of ~ 10 attributes), then we could obtain a rule such as key_turned as an explanation for engine_running that is good except with probability $\sim 10\%$ (i.e., $\sim 10^{-3}\times 10^2 = 10^{-1}$). Of course we would like a better dependence on the size of our vocabulary, for example matching the dependence of $n^{k/3}$ achieved by Awasthi et al. (2010) for agnostic PAC-learning for k-DNFs; we suggest this as a natural direction for future work. Nevertheless, we again find that our new formulation of abduction also permits a rather simple algorithm to find the same simple kind of explanations, but now moreover features some robustness to rare counterexamples, as might occur in a complex open world.

The proof will require the Chernoff bound:

Theorem 4 (Chernoff bound) *Let $X_1,\ldots,X_m$ be independent random variables taking values in $[0,1]$, such that $\mathbb{E}[\frac{1}{m}\sum_i X_i]=p$. Then for $\gamma\in[0,1]$,*

$$\Pr\left[\frac{1}{m}\sum_i X_i > (1+\gamma)p\right] \le e^{-mp\gamma^2/3}$$

$$\text{and } \Pr\left[\frac{1}{m}\sum_i X_i < (1-\gamma)p\right] \le e^{-mp\gamma^2/2}$$

Proof of Theorem 3: We first observe that for the "ideal" k-DNF h^* for which $\Pr[c|h^*]\ge 1-\epsilon$ and $4\mu\ge\Pr[h^*]\ge\mu/4$, $\Pr[\neg c\wedge h^*]\le\Pr[h^*]\epsilon\le 4\mu\epsilon$. So, no term of h^* may be true when c is false with probability greater than $4\mu\epsilon$. Given more than $\frac{3}{4\mu\epsilon}\ln\frac{N}{\delta}$ examples (where N is the number of terms of size at most k), it follows from the Chernoff bound that the probability that any one of these (at most N) terms is true when c is false in more than a $8\mu\epsilon$ fraction of the examples is at most $\frac{\delta}{N}$.

At the same time, taking $\gamma=1/2$, we find that the probability that any term T for which $\Pr[\neg c\wedge T]\ge 16\mu\epsilon$ remains in our k-DNF after $\frac{8}{16\mu\epsilon}\ln\frac{N}{\delta}$ examples is also at most $\frac{\delta}{N}$. Noting that no term T can give $\neg c\wedge T$ probability both greater than $16\mu\epsilon$ and less than $4\mu\epsilon$ simultaneously, we can simply take a union bound over the appropriate event for all of the at most N terms to find that the overall probability of any occurring is at most δ. When none occur, the algorithm obtains a k-DNF h that contains all of the terms of h^* – and so $\Pr[h]\ge\Pr[h^*]\ge\mu/4$ – and moreover (by a union bound over the terms) gives $\Pr[\neg c\wedge h]\le 16N\mu\epsilon$. Hence, for this h, $\Pr[\neg c|h]\le 64N\epsilon$. Thus, $\Pr[c|h]\ge 1-64N\epsilon$. ■

Obtaining a value for μ. Although we have placed a stronger condition on h^* – we now require an *upper* bound on its probability in addition to a lower bound – we now argue that we can find a satisfactory estimate of μ by repeatedly running the algorithm as follows. We start by making a conservative "guess" of $\mu_1 = 1/4$; certainly, $\Pr[h]\le 1 = 4\mu_1$. In general, given a guess μ_i, we run our algorithm using μ_i to obtain a candidate h. We then check to see that h is true on at least a $\mu_i/2$-fraction out of $m\ge\frac{12}{\mu_i}\ln\frac{1}{\delta_i}$ examples for $\delta_i=\delta/i(i+1)$. If so, then we return h, and if not, we put $\mu_{i+1}=\mu_i/4$ and repeat.

This works for the following reason. First, the Chernoff bound guarantees that when $\Pr[h]<\mu_i/4$, h will not pass except with probability δ_i; so if h does pass, there is some h with $\Pr[h]\ge\mu_i/4$ (with probability $1-\delta_i$). Moreover, the Chernoff bound also guarantees that h *will* pass if $\Pr[h]\ge\mu_i$ (except with probability δ_i). So, overall with probability $1-\sum_i\delta_i\ge 1-\delta$, the tests will fail until some iteration i^* in which $\Pr[h^*]\ge\mu_{i^*}/4$ for some h^*; we know that since iteration i^*-1 failed, for any h, $\Pr[h]<\mu_{i^*-1}=4\mu_{i^*}$. We may or may not obtain a h that passes on this iteration, but if we do, our estimate μ_{i^*} suffices for the guarantee of our algorithm to ensure that h is a solution to the abduction task. Regardless, on iteration i^*+1, we have that $\Pr[h^*]\ge\mu_{i^*}/4=\mu_{i^*+1}$, and then the Chernoff bound guarantees that such an h will pass (except with probability δ_{i^*+1}), and also $\Pr[h^*]\le\mu_{i^*}=4\mu_{i^*+1}$ (with probability $1-\delta_{i^*}$), so again μ_{i^*+1} is a sufficiently close estimate that our guarantee applies and h is a solution to the abduction task.

4 Abduction for Conjunctions Solves Hard Learning Problems

Abduction is frequently cast as the task of finding explanations given as a *conjunction* of literals. It is therefore natural to ask whether our abduction task for conjunctions is tractable. Unfortunately, we obtain evidence that it is not:

Theorem 5 *If the abduction task for conjunctions can be solved (even improperly), then DNF is PAC-learnable in polynomial time.*

We remind the reader of the definition of PAC-learning:

Definition 6 (PAC-learning (Valiant 1984)) *A class $\mathcal{C}$ of representations of Boolean predicates is said to be* (improperly) PAC-learnable *if the following kind of polynomial time algorithm exists. Given access to examples drawn independently from an unknown distribution D together with the evaluation of some unknown $h^*\in\mathcal{C}$ on the examples and input parameters ϵ and δ, it returns an efficiently evaluable hypothesis h such that with probability $1-\delta$ over the examples, for x drawn from D, $\Pr[h(x)=h^*(x)]\ge 1-\epsilon$.*

The proof is actually quite similar to the analogous result for agnostic learning of conjunctions by Kearns et al. (1994): Recall that a *weak learning* algorithm is a PAC-learning algorithm that merely produces a hypothesis h such that $\Pr[h=h^*]\ge 1/2+1/\text{poly}(n,|h^*|)$ (i.e., for some arbitrary polynomial in the size of the examples and representation h^*). A famous result by Schapire (1990) showed how to efficiently reduce PAC-learning to weak learning by *"boosting."*

Proof of Theorem 5: Suppose our examples are of the form (x, b) where $b = \varphi(x)$ for some DNF φ. We will show how to obtain a weak learner for φ from an algorithm for abducing conjunctions, thus obtaining a PAC-learning algorithm by boosting.

Suppose φ has size s. If φ is satisfied with probability greater than $1/2 + 1/s$ or less than $1/2 - 1/s$, then the constant functions will do as weak learners, so assume φ is satisfied with probability $1/2 \pm 1/s$. Then, we see that some term T in φ is satisfied with probability at least $1/2s - 1/s^2 \overset{\text{def}}{=} \mu$. We note that $\Pr[\varphi|T] = 1$. An algorithm for abducing conjunctions (with $\epsilon = 1/4$) for $c(x, b) = b$ and $A = \{x_1, \ldots, x_n\}$ therefore finds a hypothesis $\tilde{T}$ such that $\Pr[\tilde{T}] \geq 1/p(1/\mu, n, 4/3) \geq 1/p'(n, s)$ for some polynomial $p'(n, s)$ and $\Pr[\varphi|\tilde{T}] \geq 3/4$.

Our weak learner is now as follows: if $\Pr[\varphi|\neg\tilde{T}] \geq 1/2$, we use the constant 1, and otherwise we use $\tilde{T}$. Note that this is equivalent to a hypothesis that predicts according to a majority vote on $\neg\tilde{T}$ (and predicts 1 on $\tilde{T}$). We note that it is correct with probability at least

$$\frac{1}{2}(1 - \Pr[\tilde{T}]) + \Pr[\varphi|\tilde{T}]\Pr[\tilde{T}] \geq \frac{1}{2} + \left(\frac{3}{4} - \frac{1}{2}\right)\Pr[\tilde{T}]$$
$$\geq \frac{1}{2} + \frac{1}{4p'(n, s)}$$

which is sufficient for weak learning. ■

As a consequence, we obtain evidence that abducing conjunctons is hard: recent work by Daniely et al. (2014) and Daniely and Shalev-Shwartz (2014) has established that learning DNF is hard under a plausible conjecture:

Theorem 7 (Daniely and Shalev-Shwartz 2014) *If there is some $f : \mathbb{N} \rightarrow \mathbb{N}$ such that $f \rightarrow \infty$ for which no polynomial-time algorithm can refute random k-SAT instances with $n^{f(k)}$ clauses, then there is no polynomial-time PAC-learning algorithm for DNF.*

The premise of the theorem is a strengthening of Feige's hypothesis (Feige 2002), which was that a linear number of constraints are hard to refute. (Note that the state of the art requires $n^{k/2}$ clauses (Coja-Oghlan, Cooper, and Frieze 2010); as we add more constraints, it becomes *easier* to find a refutation.) Thus, as a corollary of Theorem 5, we find:

Corollary 8 *If there is some $f : \mathbb{N} \rightarrow \mathbb{N}$ such that $f \rightarrow \infty$ for which no polynomial-time algorithm can refute random k-SAT instances with $n^{f(k)}$ constraints, then there is no algorithm that solves the abduction task for conjunctions.*

So, although this hypothesis is new and largely untested, it still provides some complexity-theoretic evidence that we should not expect to find a tractable algorithm for abducing conjunctions in our new model. This result essentially settles the question of which (natural) representations can be produced as explanations in our new model: Most natural knowledge representations are either expressible by a k-DNF, and thus fall within the scope of our earlier algorithms, or can themselves express conjunctions, and thus seem to be outside the scope of any algorithm on account of Theorem 5.

5 Applications and Further Motivations

Our use of k-DNF representations for abduction is a bit unusual. To our knowledge, only Inoue (2012) has previously considered abducing DNF representations. We therefore conclude with a brief discussion of some notable potential applications.

Goal Formulation

Our results seem particularly suitable for the following application in planning: consider an essentially propositional formulation of planning such as in STRIPS or a factored (PO)MDP. Suppose we are given a collection of example traces of the agent interacting with the (possibly nondeterministic) environment and a (possibly complex) goal predicate. Then our algorithm, given the traces as examples together with the goal as the query c and an alphabet comprising all of the state attributes (for example), can identify a k-DNF condition that is satisfied moderately often, that entails the goal is satisfied (when such a k-DNF exists). This k-DNF can then be provided as a representation of the goal for planning. This is particularly useful if c does not have a representation as a k-DNF—for example, perhaps we do not have an attribute for it and the actual representation of c requires large terms.

As an informal example, if our goal is "be in San Diego and have a car and eat fish tacos," then perhaps a suitable 2-DNF subgoal is "be on an airplane" and "the airplane is bound for SAN," or "be on a train" and "the train is bound for San Diego," or "drive a car" and "hold a map to San Diego," since we'd anticipate that in the traces where any of these terms is satisfied, the agent reached San Diego and then obtaining the car and eating fish tacos followed as a matter of course. Providing "be in San Diego and have a car and eat fish tacos" to an algorithm for our abduction task as the query c will obtain such a 2-DNF as h if we have a suitable set of example traces, in which the agent sometimes actually arrives in San Diego via such means.

Note that some applications for SAT-solvers/resolution theorem-proving in planning require DNF goals: Suppose we want to use resolution (equivalently, clause-learning SAT-solvers (Beame, Kautz, and Sabharwal 2004)) to *prove* that a fixed plan in a given nondeterministic environment satisfies a goal, along the lines of the "plan test" step of Castellini et al. (2003) (such a step also appears in Juba (2016)). Then, since resolution is a proof system for DNFs (by refuting their negations, which are CNFs), we would need to represent the goal by a DNF. Moreover, DNFs of low width (i.e., few literals per clause) are generally considered preferable: In addition to a preference for short clauses being a standard heuristic in clause learning SAT solvers (but see Zhang et al. (2001)), SAT solvers are guaranteed to be efficient whenever a small-width proof exists (Atserias, Fichte, and Thurley 2011); but, the input must have small width in order for such a proof to exist. A k-DNF representation is thus a good fit: we need a DNF in order to apply such planning techniques, and we would generally prefer a DNF with at most k literals per clause (as found by our algorithm), if one can be found.

Selection of Preconditions

The problems we consider here arise in Valiant's Robust Logic framework (Valiant 2000b), which was a proposal to capture certain kinds of common sense reasoning (Valiant 1994; 1995; 2006); the problem also arises in a similar probabilistic formalization of common sense reasoning by Roth (1995), developed further by Khardon and Roth (1997a). Roughly, the issue is that Valiant and Roth show that the famous non-monotonic effects of common sense reasoning can be captured naturally by a probabilistic semantics, in which the incorporation of new knowledge is captured by filtering the examples used for reasoning, referred to by Valiant as "applying a precondition." So for example, one makes the nonmonotonic inference that birds fly as follows: given that a scene x^* concerns a bird (say, has $\mathsf{bird} = 1$ for some such attribute) we *filter* our set of examples X to obtain $X|_{\mathsf{bird}} = \{x \in X : x_{\mathsf{bird}} = 1\}$. Now, in this set $X|_{\mathsf{bird}}$, we decide $\mathsf{can\_fly}$ by examining the *fraction* of $x \in X|_{\mathsf{bird}}$ for which $\mathsf{can\_fly} = 1$, and asserting $\mathsf{can\_fly} = 1$ if this fraction is sufficiently large.[5] The condition bird used to filter the examples is the precondition for the scene x^* here.

In these works, the precondition is simply assumed to be given; *how* it would be selected is not explicitly considered. Another work by Valiant (1994, p.164) informally suggests that such "context" might be simply given by the attributes that are currently firing in the neuroidal model. But, this view leads to problems: the specific, irrelevant details of the scene may never have been encountered before, leaving no examples to perform the common sense reasoning. The problems we formalize here might be viewed as the problem of *proposing* a candidate precondition (relative to some desired property) that has enough data to offer meaningful predictions.

We can formalize such a problem as:

Definition 9 (Precondition search) *The* (optimal) precondition search *task for a class of representations $\mathcal{H}$ over n attributes is as follows. Given an* observation *example $x^* \in \{0,1\}^n$ and a* query *representation c for some arbitrary distribution D over $\{0,1\}^n$ such that $D(x^*) > 0$, if there exists some $h^* \in \mathcal{H}$ such that $h^*(x^*) = 1$, $\Pr[c(x) = 1|h^*(x) = 1] = 1$, and $\Pr[h^*(x) = 1] \geq \mu$, using examples drawn from D, find an $h \in \mathcal{H}$ in time polynomial in n, $1/\epsilon$, $1/\mu$, and $1/\delta$ such that with probability $1-\delta$ over the examples, $h(x^*) = 1$, $\Pr[c(x) = 1|h(x) = 1] \geq 1-\epsilon$, and $\Pr[h(x) = 1] \geq \mu$. Such an h is said to be a* precondition *for x^* relative to c.*

Theorem 10 *The elimination algorithm solves optimal precondition search for k-DNFs.*

Proof: Notice, the disjunction of all $h \in \mathcal{H}$ for which $\Pr[\neg c \wedge h] = 0$ has probability equal to at least that of any $h^* \in \mathcal{H}$ that maximizes $\Pr[h^*]$. So, since $\mathcal{H}$ is the class of k-DNFs, h^* must be equal to the disjunction over all such $h \in \mathcal{H}$ with probability 1 over the distribution over examples D. In particular, since $D(x^*) > 0$, if x^* satisfies any $h \in \mathcal{H}$, x^* must satisfy h^*. Moreover, our analysis of the elimination algorithm in the proof of Theorem 2 also shows that whenever x^* satisfies h^*, it also satisfies the h we obtain, as h contains all of the terms of h^*. So, the elimination algorithm is also solving this precondition selection problem relative to a query for k-DNFs. ■

In a standard example, we wish to select a precondition with respect to which we will make judgments about whether or not a bird Tweety flies in some particular example scene x^*. Let's suppose that we want to determine if there are preconditions supporting the conclusions that Tweety flies or Tweety does not fly—so we will search for preconditions that would support the conclusion $c_1 = \mathsf{can\_fly}$ and that would support $c_2 = \neg\mathsf{can\_fly}$. Now, further suppose that Tweety is a penguin, so $x^*_{\mathsf{bird}} = 1$ and $x^*_{\mathsf{penguin}} = 1$. We can now find preconditions h_1 for c_1 and h_2 for c_2—say $h_1 = \mathsf{bird}$ and $h_2 = \mathsf{penguin}$ (we'll suppose that $\mathsf{can\_fly}$ is unknown in x^*). In both cases, the abduced condition is more specific than either c_1 or c_2 in that it (approximately) entails the query in each case, and yet it is general as possible with respect to $\mathcal{H}$, given that it is true of the specific scene x^* and (approximately) entails the query. So, intuitively, the preconditions give "evidence" for or against $\mathsf{can\_fly}$ in the scene x^*. We might prefer $\neg\mathsf{can\_fly}$ since, within the filtered set of examples $X|_{\mathsf{bird}}$, we could again search for a precondition for $\neg\mathsf{can\_fly}$ which would turn up the precondition $\mathsf{penguin}$; but for $X|_{\mathsf{penguin}}$, there does not exist any precondition for the query $\mathsf{can\_fly}$, since any such h will eliminate all examples in $X|_{\mathsf{penguin}}$. This suggests a connection to argument semantics (Dung 1995) (see also Michael (2015)), which we leave for future work.

This application is related to the selection of a *"reference class"* for estimation, a problem that featured prominently in Reichenbach's theory of probability (1949). Our use of disjunctive representations here does not follow Reichenbach's suggestion or its refinements by Kyburg (1974) or Pollock (1990), to find a *most specific* reference class, and relatedly, to *disallow disjunctive classes*; see Bacchus et al. (1996) for a discussion of this approach and some problems it encounters. Alas, the most natural representations for reference classes are arguably conjunctions, which Theorem 5 suggests may be out of reach.

Acknowledgements

We thank Leslie Valiant, Madhu Sudan, Or Sheffet, Sam Wiseman, Kilian Weinberger, and Sanmay Das for conversations that helped direct this work, and Loizos Michael, Dan Roth, and the anonymous reviewers for their feedback.

A An Alternative Formulation

Recall that we defined the abduction task as, given a target condition $c(x)$, finding a $h \in \mathcal{H}$ that gives $\Pr[c(x) = 1|h(x) = 1] \geq 1-\epsilon$ and $\Pr[h(x) = 1] \geq \mu$ (or in many cases, simply maximizing $\Pr[h(x) = 1]$ subject to the first constraint). We could have formulated this instead as a prob-

[5] Naturally, this is only interesting for example scenes x^* in which $\mathsf{can\_fly}$ is unknown, but we won't treat incomplete information in depth here. Valiant and Roth simply assume that x^* is missing attributes, whereas the "training" examples X are complete. See Michael (2010; 2014) and Juba (2013) for a fuller treatment.

lem of computing a "maximum posterior" hypothesis, i.e., given the "data" that $c(x) = 1$.[6]

Definition 11 (MP Abduction) *For a* representation class *$\mathcal{H}$ of Boolean formulas over propositional attributes $x_1, \ldots, x_n$, the* MP abduction *task is as follows. We are given as input m independent examples $x^{(1)}, \ldots, x^{(m)}$ from an arbitrary distribution D over $\{0,1\}^n$, a* query *formula $c(x)$ over $x_1, \ldots, x_n$, and an* alphabet *$A \subseteq \{x_1, \ldots, x_n\}$, for which there exists $h^* \in \mathcal{H}$ only using attributes in A such that $\Pr[c(x) = 1 | h^*(x) = 1] = 1$ and $\Pr[h^*(x) = 1 | c(x) = 1] \geq \alpha$. Then, with probability $1 - \delta$, in time polynomial in $n, 1/\alpha, 1/\Pr[c(x) = 1], 1/\epsilon$, and $1/\delta$, we find an* explanation *$h \in \mathcal{H}$ only using attributes in A such that*

1. $\Pr[c(x) = 1 | h(x) = 1] \geq 1 - \epsilon$ *and*

2. $\Pr[h(x) = 1 | c(x) = 1] \geq \Omega\left(\left((1-\epsilon)\frac{\alpha \Pr[c(x)=1]}{n}\right)^d\right)$ *for some $d \in \mathbb{N}$.*

We included factors of $\Pr[c]$ in the time and approximation bounds which were not present in the original parameters; it was not necessary since given $\Pr[c|h] \geq 1 - \epsilon$ (i.e., condition 1), $\Pr[c] \geq \Pr[c \wedge h] \geq (1 - \epsilon)\Pr[h]$, and thus $(1 - \epsilon)\mu \leq \Pr[c]$, so any factor of $\Pr[c]$ in such a bound can be replaced by additional factors of $(1 - \epsilon)\mu$. Similar calculations show that this formulation is essentially equivalent to the "prior plausibility" formulation of abduction that we originally proposed:

Theorem 12 *Optimal solutions to MP abduction solve the original abduction task with plausibility $\Pr[h(x) = 1] \geq (1-\epsilon)\mu$ and optimal solutions to the original abduction task solve MP abduction with $\Pr[h(x) = 1 | c(x) = 1] \geq (1 - \epsilon)\alpha$; more generally, solutions to the MP abduction task are solutions to the original task and vice-versa.*

Proof: First suppose that $h \in \mathcal{H}$ satisfies the common condition 1, that $\Pr[c|h] \geq 1 - \epsilon$. Now, we write

$$\Pr[h(x) = 1 | c(x) = 1] \overset{\text{def}}{=} \alpha(h).$$

Equivalently, $\Pr[h \wedge c] = \alpha(h)\Pr[c]$. Now, notice: we can rewrite $\Pr[h] = \Pr[h \wedge c] + \Pr[\neg c \wedge h]$, so $\Pr[h] - \Pr[\neg c \wedge h] = \Pr[h \wedge c]$. Therefore, $\Pr[h] - \Pr[\neg c \wedge h] = \alpha(h)\Pr[c]$, where $\Pr[c]$ is fixed independent of h (since c is given). We can rewrite the first condition, $\Pr[c|h] \geq 1 - \epsilon$ as $\Pr[\neg c \wedge h] \leq \epsilon \Pr[h]$; so, since the first condition is satisfied,

$$(1 - \epsilon)\Pr[h] \leq \Pr[h] - \Pr[\neg c \wedge h] \leq \Pr[h].$$

Hence, $(1 - \epsilon)\Pr[h] \leq \alpha(h)\Pr[c] \leq \Pr[h]$ and so choosing h to maximize $\Pr[h]$ is equivalent to choosing h to maximize $\alpha(h)$ (and hence, $\Pr[h|c]$) within a multiplicative $1-\epsilon$ factor. That is, optimal solutions to MP abduction solve the original abduction task with plausibility $\Pr[h] \geq \alpha(h^*)\Pr[c] \geq (1-\epsilon)\mu$ and an optimal solution to the original abduction task solves MP abduction with $\alpha(h) \geq (1 - \epsilon)\Pr[h^*]/\Pr[c] \geq (1 - \epsilon)\alpha(h^*) = (1 - \epsilon)\alpha$.

For the final part, we transform the inequalities by applying $f(p) = \frac{1}{C}\left((1-\epsilon)\frac{p}{n}\right)^d$ for appropriate C and d since f is monotonic for $p \in [0, 1]$. Given a solution to MP abduction with $\alpha(h) \geq \frac{1}{C}\left((1-\epsilon)\frac{\alpha \Pr[c]}{n}\right)^d$, using $\Pr[h] \geq \alpha(h)\Pr[c]$ and $\alpha \Pr[c] \geq (1 - \epsilon)\mu$, we get

$$\Pr[h] \geq \frac{1}{C}\left((1-\epsilon)\frac{\alpha \Pr[c]}{n}\right)^d \Pr[c] \geq \frac{1}{C}\left((1-\epsilon)\frac{\mu}{n}\right)^{2d}$$

and h is thus a solution to the standard abduction task. Likewise, given an approximate solution to the standard abduction, i.e., h with $\Pr[h] \geq \Omega\left(((1-\epsilon)(\mu/n))^d\right)$, using $\alpha(h)\Pr[c] \geq (1 - \epsilon)\Pr[h]$ and $\mu \geq \alpha \Pr[c]$, we get

$$\alpha(h)\Pr[c] \geq (1-\epsilon)\frac{1}{C}\left((1-\epsilon)\frac{\alpha \Pr[c]}{n}\right)^d.$$

So, $\alpha(h) \geq \Omega\left(((1-\epsilon)\alpha \Pr[c]/n)^{d+1}\right)$, and therefore h is a solution to MP abduction. ■

References

Atserias, A.; Fichte, J. K.; and Thurley, M. 2011. Clause-learning algorithms with many restarts and bounded-width resolution. *JAIR* 40:353–373.

Awasthi, P.; Blum, A.; and Sheffet, O. 2010. Improved guarantees for agnostic learning of disjunctions. In *Proc. 23rd COLT*, 359–367.

Bacchus, F.; Grove, A. J.; Halpern, J. Y.; and Koller, D. 1996. From statistical knowledge bases to degrees of belief. *Artificial Intelligence* 87:75–143.

Beame, P.; Kautz, H.; and Sabharwal, A. 2004. Towards understanding and harnessing the potential of clause learning. *JAIR* 22:319–351.

Bylander, T.; Allemang, D.; Tanner, M. C.; and Josephson, J. R. 1991. The computational complexity of abduction. *Artificial Intelligence* 49:25–60.

Castellini, C.; Giunchiglia, E.; and Tacchella, A. 2003. SAT-based planning in complex domains: Concurrency, constraints, and nondeterminism. *Artificial Intelligence* 147(1–2):85–117.

Charniak, E., and McDermott, D. 1985. *Introduction to Artificial Intelligence*. Reading, MA: Addison-Wesley.

Coja-Oghlan, A.; Cooper, C.; and Frieze, A. 2010. An efficient sparse regularity concept. *SIAM J. Discrete Math.* 23(4):2000–2034.

Cox, P., and Pietrzykowski, T. 1986. Causes for events: their computation and applications. In *Proc. 8th Int'l Conf. Automated Deduction*, 608–621.

Daniely, A., and Shalev-Shwartz, S. 2014. Complexity theoretic limtations on learning DNF's. arXiv:1404.3378.

[6] MP abduction resembles the Bayesian "maximum a posteriori" (MAP) inference, but note that we *do not* have a prior distribution over *representations,* only a distribution over attributes that may or may not satisfy representations. Naturally, if the examples uniquely satisfied one representation, these would be equivalent, but in general there may be many representations satisfied by any given example.

Daniely, A.; Linial, N.; and Shalev-Shwartz, S. 2014. From average case complexity to improper learning complexity. In *Proc. 46th STOC*, 441–448.

Denecker, M., and Kakas, A. 2002. Abduction in logic programming. In *Computational Logic: Logic Programming and Beyond*, volume 2407. Berlin: Springer. 402–437.

Dung, P. M. 1995. On the acceptability of arugments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. *Artificial Intelligence* 77(2):321–357.

Eshghi, K. 1988. Abductive planning with event calculus. In *Proc. 5th Int'l Logic Programming Conf.*, 562–579.

Feige, U. 2002. Relations between average case complexity and approximation complexity. In *Proc. 34th STOC*, 534–543.

Flach, P. A., and Kakas, A. C. 2000. *Abduction and Induction: Essays on their relation and integration*. Springer.

Hobbs, J.; Stickel, M.; Appelt, D.; and Martin, P. 1990. Interpretation as abduction. Technical Report 499, SRI, Menlo Park, CA.

Inoue, K. 2012. DNF hypotheses in explanatory induction. In Muggleton, S. H.; Tamaddoni-Nezhad, A.; and Lisi, F. A., eds., *Proc. ILP 2011*, volume 7207 of *LNCS*. Berlin: Springer. 173–188.

Juba, B. 2013. Implicit learning of common sense for reasoning. In *Proc. 23rd IJCAI*, 939–946.

Juba, B. 2016. Integrated common sense learning and planning in POMDPs. Forthcoming.

Kautz, H.; Kearns, M.; and Selman, B. 1995. Horn approximations of empirical data. *Artificial Intelligence* 74(1):129–145.

Kearns, M. J.; Schapire, R. E.; and Sellie, L. M. 1994. Towards efficient agnostic learning. *Machine Learning* 17(2-3):115–141.

Khardon, R., and Roth, D. 1997a. Defaults and relevance in model based reasoning. *Artificial Intelligence* 97(1-2):169–193.

Khardon, R., and Roth, D. 1997b. Learning to reason. *J. ACM* 44(5):697–725.

Kyburg, H. E. 1974. *The Logical Foundations of Statistical Inference*. Dordrecht: Reidel.

McCarthy, J. 1980. Circumscription – a form of non-monotonic reasoning. *Artificial Intelligence* 13(1–2):27–39. Available at http://www-formal.stanford.edu/jmc/circumscription.html.

McIlraith, S. A. 1998. Logic-based abductive inference. Technical Report KSL-98-19, Knowledge Systems Laboratory.

Michael, L. 2010. Partial observability and learnability. *Artificial Intelligence* 174(11):639–669.

Michael, L. 2014. Simultaneous learning and prediction. In *Proc. 14th KR*, 348–357.

Michael, L. 2015. Jumping to conclusions. In *Proc. 2nd DARe (IJCAI'15 workshop)*.

Mill, J. S. 1843. *A System of Logic, Ratiocinative and Inductive*, volume 1. London: John W. Parker.

Missiaen, L.; Bruynooghe, M.; and Denecker, M. 1995. CHICA, a planning system based on event calculus. *J. Logic and Computation* 5(5).

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

Pierce, C. S. 1931. Elements of logic. In Hartshorn, C., et al., ed., *Collected Papers of Charles Sanders Pierce*. Harvard University Press.

Pollock, J. L. 1990. *Nomic Probabilities and the Foundations of Induction*. Oxford: Oxford University Press.

Poole, D. 1990. A methodology for using a default and abductive reasoning system. *Int'l J. Intelligent Sys.* 5:521–548.

Poole, D. 1993. Probabilistic Horn abduction and Bayesian networks. *Artificial Intelligence* 64(1):81–129.

Reggia, J. 1983. Diagnostic expert systems based on a set-covering model. *Int'l J. Man-Machine Studies* 19(5):437–460.

Reichenbach, H. 1949. *Theory of Probability*. Berkeley, CA: University of California Press.

Reiter, R., and de Kleer, J. 1987. Foundations for assumption-based truth maintenance systems: Preliminary report. In *Proc. AAAI-87*, 183–188.

Reiter, R. 1987. A theory of diagnosis from first principles. *Artificial Intelligence* 32:57–95.

Roth, D. 1995. Learning to reason: the non-monotonic case. In *Proc. 14th IJCAI*, volume 2, 1178–1184.

Roth, D. 1996. A connectionist framework for reasoning: Reasoning with examples. In *Proc. AAAI-96*, 1256–1261.

Schapire, R. E. 1990. The strength of weak learnability. *Machine Learning* 5(2):197–227.

Thompson, C. A., and Mooney, R. J. 1994. Inductive learning for abductive diagnosis. In *Proc. AAAI-94*, 664–669.

Valiant, L. G. 1984. A theory of the learnable. *Communications of the ACM* 18(11):1134–1142.

Valiant, L. G. 1994. *Circuits of the Mind*. Oxford: Oxford University Press.

Valiant, L. G. 1995. Rationality. In *Proc. 8th COLT*, 3–14.

Valiant, L. G. 2000a. A neuroidal architecture for cognitive computation. *J. ACM* 47(5):854–882.

Valiant, L. G. 2000b. Robust logics. *Artificial Intelligence* 117:231–253.

Valiant, L. G. 2006. Knowledge infusion. In *Proc. AAAI-06*, 1546–1551.

Zhang, L.; Madigan, C. F.; Moskewicz, M. W.; and Malik, S. 2001. Efficient conflict driven learning in a Boolean satisfiability solver. In *Proc. IEEE/ACM Int'l Conf. on Computer Aided Design (ICCAD'01)*, 279–285.