A Defeasible Reasoning Model of Inductive Concept Learning from Examples and Communication

Santiago Ontañón[a,c], Pilar Dellunde[a,b], Lluís Godo[a], Enric Plaza[a]

[a] *IIIA - CSIC, Artificial Intelligence Research Institute, Spanish Council for Scientific Research, Campus UAB, 08193 Bellaterra, Catalonia (Spain),*
{santi,pilar,godo,enric}@iiia.csic.es
[b] *Universitat Autònoma de Barcelona, 08193 Bellaterra, Catalonia (Spain),*
pilar.dellunde@uab.cat
[c] *Computer Science Department, Drexel University, Philadelphia (USA),*
santi@cs.drexel.edu

Abstract

This paper introduces a logical model of inductive generalization, and specifically of the machine learning task of inductive concept learning (ICL). We argue that some inductive processes, like ICL, can be seen as a form of defeasible reasoning. We define a *consequence relation* characterizing which hypotheses can be induced from given sets of examples, and study its properties, showing they correspond to a rather well-behaved non-monotonic logic. We will also show that with the addition of a preference relation on inductive theories we can characterize the inductive bias of ICL algorithms. The second part of the paper shows how this logical characterization of inductive generalization can be integrated with another form of non-monotonic reasoning (argumentation), to define a model of multiagent ICL. This integration allows two or more agents to learn, in a consistent way, both from induction and from arguments used in the communication between them. We show that the inductive theories achieved by multiagent induction plus argumentation are sound, i.e. they are precisely the same as the inductive theories built by a single agent with all data.

Keywords:
Induction, Logic, Argumentation, Machine Learning, Concept Learning

1. Introduction

Inductive generalization is the basis for machine learning methods which learn general hypotheses from examples. However, with the exception of a few isolated proposals [1, 2, 3, 4], there has been little effort towards specific logical models of inductive generalization. The lack of a formal logical model of induction may have hindered the development of approaches that combine induction with other forms of logical reasoning.

In this paper we do not tackle induction in its more general definition, but limit ourselves to inductive generalization, and specifically, to the common task of inductive concept learning (ICL), which is the most well studied induction problem in machine learning. We will argue that inductive generalization is a form of defeasible reasoning, and define an inductive consequence relation (denoted by $\mid\!\sim$) characterizing which hypotheses can be induced from given sets of examples, and show its logical properties.

Relationships between inductive reasoning and non-monotonic reasoning have already been established by Flach in [1, 5], where he presents a logical analysis of induction and considers several postulates for a general inductive consequence relation along with representation theorems in terms of preferential models, in the tradition of non-monotonic reasoning[1] [7]. However, while the work of Flach aims at defining general rationality postulates for induction in general, our focus is on characterizing a particular form of induction (ICL), which allows us to develop a more concrete model (see B for an in-depth comparison of our proposal with Flach's). Moreover, Flach presents a logical characterization of induction focusing on *hypothesis generation* rather than *hypothesis selection*, i.e. intending to model which are the valid hypotheses one can induce from a set of examples, but not which of those hypotheses is the best one. In this paper, within the framework of ICL, we go one step further and propose that *hypothesis selection* can also be logically characterized by means of a preference relation on inductive theories (suitable sets of hypotheses), and propose some preference relations which capture the typical biases used in ICL algorithms (like parsimony or margin maximization).

There are two main implications of defining a logical model of inductive generalization. First, it allows for a better understanding of ICL algorithms, and second, it facilitates the integration of inductive reasoning with other

[1] A similar work for abductive reasoning is that of Pino-Pérez and Uzcátegui [6].

forms of logical reasoning, as we will show by integrating ICL with computational argumentation to define a model of multiagent ICL. This paper extends the preliminary work in [8], modeling inductive generalization as a non-monotonic logic, extending the properties satisfied, and using preference relations to model bias in ICL.

The second part of this paper presents an integration of two non-monotonic forms of reasoning: induction and argumentation. This integration shows the advantage of having a logical model of induction. For instance, a multiagent induction system such as [9] already introduced the idea of integrating inductive learning and argumentation in an implemented systems, but lacked any formal grounding for such an integration. In particular, in this paper we present a model of multiagent ICL obtained by directly integrating our inductive consequence relation with computational argumentation. In this approach, argumentation is used to model the communication between agents, and ICL models their internal learning processes.

The remainder of this paper is organized as follows. Section 2 introduces the problem of inductive concept learning as typically framed in the machine learning literature. Then, Section 3 introduces a logical model of induction and proposes an inductive consequence relation, while Section 4 deals with preferences over inductive theories. In Section 5 we recall basic notions of computational argumentation and we introduce the notion of argumentation-consistent induction. Next, Sections 6 and 7 define a model of multiagent ICL by integrating our logical model of ICL with computational argumentation. The paper closes discussing some related work and with the conclusions. We have also included two appendices: Appendix A contains a generalization of Theorem 1 to n agents, and Appendix B provides more details on the comparison of our inductive consequence relation with Flach's previous work.

2. Background

Inductive concept learning (ICL) [10] using inductive techniques is not defined formally in the literature of machine learning; rather it is usually defined as a task, as follows:

Given:

1. A space X of instances
2. A space of hypotheses or generalizations H, modeled as a set of mappings $h : X \to \{0, 1\}$

3. A target concept c, modeled as a partially known mapping $c : X \to \{0, 1\}$
4. A set D of training examples (for which c is known), where a training example is a pair $\langle x_i, c(x_i)\rangle$

Find a hypothesis $h \in H$ such that $h(x) = c(x)$ for each instance x in the set of training examples D

This strictly Boolean definition is usually weakened to allow the equality $h(x) = c(x)$ not being true for all examples in D but just for a percentage, and the difference is called the *error* of the learnt hypothesis.

Another definition of inductive concept learning is that used in Inductive Logic Programing (ILP) [11], where the background knowledge, in addition to the examples, has to be taken into account. Nevertheless, ILP also defines ICL as a task to be achieved by an algorithm, as follows:

Given:

1. A set of positive E^+ and negative E^- examples of a predicate p
2. A set of Horn rules (background knowledge) B
3. A space of hypotheses H (a sublanguage of Horn logic language)

Find A hypothesis $h \in H$ such that

- $\forall e \in E^+ : B \wedge h \models e$ (h is complete)
- $\forall e \in E^- : B \wedge h \not\models e$ (h is consistent)

These definitions, although widespread, are unsatisfactory and leave several issues without a precise characterization. For example, the space of hypotheses H is usually expressed only by conjunctive formulas. However, most concepts need more than one conjunctive formula (more than one generalization) but this is "left outside" of the definition and is explained as part of the strategy of an inductive algorithm. For instance, the set-covering strategy [12] consists of finding one definition that covers only part of the positive examples in D, proceeding then to eliminate the covered examples to obtain a new D' that will be used in the next step. Another example is that, typically, smaller hypotheses are preferred to longer hypotheses; but again, that is left out of the definition.

In this paper our goal is not to provide a definition of the task of inductive concept learning, but to provide a logical characterization of the inductive inference processes required for performing such task.

3. Inductive Generalization for Concept Learning

Inductive generalization can be seen as having two main components: hypothesis generation and hypothesis selection [1]. We will model the former using an *inductive consequence relation*, that defines which statements are valid inductive consequences of given a set of examples, and the later using a *preference relation*, which determines which of those statements are "better" than others. This section formally defines our inductive consequence relation.

3.1. An Inductive Consequence Relation

In order to present our model of inductive concept learning, let us start by describing our language. There are three basic elements in our language: examples, hypotheses (or generalizations) and background knowledge. We will use fragments of first-order logic as the representation language for these elements. Given that we focus on inductive concept learning, hypotheses will basically be classification rules (i.e. rules which classify an example as either belonging to the target concept or not). Therefore, in the rest of this paper, the hypotheses induced from examples will be called *rules*.

We will use a distinguished unary predicate C to denote the target concept. Thus, we will write $C(a)$ when the example identified by the constant a belongs to the target concept, and $\neg C(a)$ otherwise. Our formulas will be of two kinds: examples, and rules.

- *Examples* will be conjunctions of the form $\varphi(a) \wedge C(a)$, where a is a constant, $\varphi(x)$ is an arbitrary formula with x being its only free variable. A *positive example* of C will be of the form $\varphi(a) \wedge C(a)$; a *negative example* of C will be of the form $\varphi(a) \wedge \neg C(a)$.
- *Rules* will be universally quantified formulas of the form $(\forall x)(\varphi(x) \rightarrow C(x))$, where $\varphi(x)$ is an arbitrary formula with x being its only free variable.

The set of examples will be noted by $\mathcal{L}_e$ and the set of rules by $\mathcal{L}_r$, and the set of all formulas of our language will be $\mathcal{L} = \mathcal{L}_e \cup \mathcal{L}_r$. In what follows, we will use the symbol $\vdash$ to denote derivation in classical first order logic. By *background knowledge* we will refer to a finite set of formulas $K \subset \mathcal{L}_r$.

Let us assume that the similarity type of our first-order language is finite (that is, we have a finite number of constants, predicates and function

symbols). We fix a finite number of variables and we assume that all the variables contained in the formulas (either in examples or in rules) are among these. Without loss of generality we can also assume that in each formula (either in examples or in rules) there are not different quantifier blocks with the same variable. Moreover, we can assume also that, the variable x does not occur quantified in $\phi(x)$. For instance, we will not allow formulas like $\phi(x) := (\forall y)Ryx \wedge (\exists y)(\forall x)Txy$. Under these assumptions, using the fact that every first-order formula is logically equivalent to a prenex formula with the same free variables, it is not difficult to check that there are only a finite number of (example and rule) formulas modulo logical equivalence (see for instance [13, Chap. 2]). Therefore, we will assume that both $\mathcal{L}_e$ and $\mathcal{L}_r$ are finite.

The previously defined notation allows us to define an inductive consequence relation between examples and rules. For simplicity we will write $\alpha \to \beta$ as a shorthand for the formula $(\forall x)(\alpha(x) \to \beta(x))$.

Definition 1. (Covering) *Given background knowledge K, we say that a rule $\alpha \to C$* covers an example $\varphi(a) \wedge C(a)$ *(or $\varphi(a) \wedge \neg C(a)$) when $\varphi(a) \wedge K \vdash \alpha(a)$.*

Definition 2. (Inductive Consequence) *Given background knowledge K, a set of examples $\Delta \subseteq \mathcal{L}_e$ and a rule $r = \alpha \to C$, the* inductive consequence *$\Delta \mid\sim_K \alpha \to C$ holds iff:*
*1) **(Explanation)** r covers at least one positive example of C in Δ,*
*2) **(Consistency)** r does not cover any negative example of C in Δ*

Notice that if we have two conflicting examples in Δ of the form $\varphi(a) \wedge C(a)$ and $\psi(b) \wedge \neg C(b)$, and $\varphi(a)$ is a less specific description than $\psi(a)$ (i.e. $K \vdash \psi(a) \to \varphi(a)$) then no rule $\alpha \to C$ covering the example $\varphi(a) \wedge C(a)$ can be inductively derived from Δ. The next definition identifies when a set of examples is free of these kind of conflicts.

Definition 3. (Consistent Set of Examples) *A set of examples Δ is said to be* consistent with respect to a concept C and background knowledge K *when: if $\varphi(a) \wedge C(a)$ and $\psi(b) \wedge \neg C(b)$ belong to Δ, then both $K \nvdash \varphi \to \psi$ and $K \nvdash \psi \to \varphi$.*

Definition 4. (Inducible Rules) *Given a consistent set of examples Δ and background knowledge K, the set of all rules that can be induced from Δ and K is $IR_K(\Delta) = \{(\varphi \to C) \in \mathcal{L}_r \mid \Delta \mid\sim_K \varphi \to C\}$.*

Notice that if Δ contains examples for a given concept C and also examples of $\neg C$, the set $IR_K(\Delta)$ will contain both rules that conclude C and rules that conclude $\neg C$. In general, $IR_K(\Delta)$ contains rules that conclude every concept for which there are examples in Δ. Also, notice that since $\mathcal{L}_r$ is finite, $IR_K(\Delta)$ must also be finite. Next we show some interesting properties of the inductive consequence $\mid\!\sim_K$.

Some formalizations of defeasible reasoning as non-monotonic logics, such as [14] and [7], consider *Reflexivity*, *Left Logical Equivalence* and *Right Weakening* the basic properties without which a system should not be considered a logical system, while others, such as [15], consider *Reflexivity* and *Cut* to be the basic properties of a logical system. Since our consequence relation is defined between two different sets of formulas (examples and rules), most of these properties do not directly apply to our setting. Nevertheless, it is interesting to check whether the principles underlying these properties hold for our consequence relation.

Intuitively speaking, the *Left Logical Equivalence* property expresses the requirement that logically equivalent formulas have exactly the same consequences. In our framework, in order to evaluate this principle, we need to define first the notion of *equivalent sets of examples*.

Definition 5. (Equivalent Sets of Examples) *Given background knowledge K, and two sets of examples $\Delta = \Delta^+ \cup \Delta^-$ and $\Gamma = \Gamma^+ \cup \Gamma^-$, where*

$$\begin{aligned}
\Delta^+ &= \{\varphi_0(a_0) \wedge C(a_0), \ldots, \varphi_k(a_k) \wedge C(a_k)\} \\
\Delta^- &= \{\varphi_{k+1}(a_{k+1}) \wedge \neg C(a_{k+1}), \ldots, \varphi_n(a_n) \wedge \neg C(a_n)\} \\
\Gamma^+ &= \{\phi_0(b_0) \wedge C(b_0), \ldots, \phi_l(b_l) \wedge C(b_l)\} \\
\Gamma^- &= \{\phi_{l+1}(b_{l+1}) \wedge \neg C(b_{l+1}), \ldots, \phi_m(b_m) \wedge \neg C(b_m)\},
\end{aligned}$$

we say that Δ is equivalent to Γ *modulo* K, $(\Delta \equiv_K \Gamma)$, *iff*

1. *For every $i \leq k$, there is $j \leq l$ such that $K \vdash \varphi_i \rightarrow \phi_j$*
2. *For every $j \leq l$, there is $i \leq k$ such that $K \vdash \phi_i \rightarrow \varphi_j$*
3. *For every $i > k$, there is $j > l$ such that $K \vdash \varphi_i \rightarrow \phi_j$*
4. *For every $j > l$, there is $i > k$ such that $K \vdash \phi_i \rightarrow \varphi_j$*

Now we can show that, after suitable reformulations, *Left Logical Equivalence* and the rest of above mentioned properties are satisfied.

Proposition 1. *The inductive consequence relation $\mid\!\sim_K$ satisfies the following properties:*

1. *Reflexivity: Assume that* Δ *is consistent w.r.t.* C *and* K*. If* $\varphi(a) \wedge C(a) \in \Delta$*, then* $\Delta \mid\!\sim_K \varphi \rightarrow C$*.*
2. *Left Logical Equivalence: If* $\Delta \mid\!\sim_K \alpha \rightarrow C$ *and* $\Delta \equiv_K \Delta'$*, then* $\Delta' \mid\!\sim_K \alpha \rightarrow C$*.*
3. *Right Logical Equivalence: If* $K \vdash \beta \leftrightarrow \alpha$ *and* $\Delta \mid\!\sim_K \alpha \rightarrow C$*, then* $\Delta \mid\!\sim_K \beta \rightarrow C$*.*
4. *Cut: If* $\Delta \cup \{\varphi(a) \wedge C(a), \phi(b) \wedge C(b)\} \mid\!\sim_K \alpha \rightarrow C$ *and* $K \vdash \varphi \rightarrow \phi$ *then* $\Delta \cup \{\varphi(a) \wedge C(a)\} \mid\!\sim_K \alpha \rightarrow C$*.*
5. *Cautious Monotonicity: If* $\Delta \mid\!\sim_K \alpha \rightarrow C$ *and* $\Delta \mid\!\sim_K \beta \rightarrow C$*, for every new constant* b*,* $\Delta \cup \{\alpha(b) \wedge C(b)\} \mid\!\sim_K \beta \rightarrow C$*.*
6. *Cautious Right Weakening: If* $K \vdash \alpha \rightarrow \beta$ *and* $\Delta \mid\!\sim_K \beta \rightarrow C$*, and* $\alpha \rightarrow C$ *covers some positive example in* Δ*, then* $\Delta \mid\!\sim_K \alpha \rightarrow C$*.*

Proof.

1. Since $\varphi(a) \wedge C(a) \in \Delta$ and we obviously have $\varphi(a) \wedge K \vdash \varphi(a)$, explanation trivially holds. Now assume $\psi(a) \wedge \neg C(a) \in \Delta$. Then, since Δ is consistent w.r.t. C and K, $\psi(a) \wedge K \nvdash \varphi(a)$, hence consistency also holds.
2. By definition of covering, if a rule $\alpha \rightarrow C$ covers a positive example of Δ, say $\varphi(a) \wedge C(a)$, it covers any other example $\phi(b) \wedge C(b) \in \Delta'$ such that $K \vdash \varphi \rightarrow \phi$. By definition of equivalent sets of examples (modulo K), at least one of such examples belongs to Δ'. An analogous argument holds for the negative examples.
3. By definition of covering, two logically equivalent rules (modulo K) cover exactly the same positive and negative examples.
4. The reason is that, if the rule $\alpha \rightarrow C$ covers the positive example $\phi(b) \wedge C(b)$, since $K \vdash \varphi \rightarrow \phi$, then $\alpha \rightarrow C$ also covers the positive example $\varphi(a) \wedge C(a)$.
5. By Definition 2 adding a positive example for an induced rule maintains the validity of that rule.
6. By Definition 2 the rule $\alpha \rightarrow C$ clearly satisfies the explanation property. Moreover, $\alpha \rightarrow C$ satisfies also the consistency property: otherwise, since $K \vdash \varphi \rightarrow \phi$, the rule $\beta \rightarrow C$ will cover a negative example, contrary to our assumption.

□

The first property, *Reflexivity*, transforms (or *lifts*) every example $e \in \Delta$ into a rule r_e where constants have been substituted by variables. This *lifting* is usually called in ICL literature the "single representation trick," by which an example in the language of instances (here $\mathcal{L}_e$) is transformed into an expression in the language of hypotheses (here $\mathcal{L}_r$).

The *Right Logical Equivalence* property expresses that one may replace logically equivalent formulas by one another on the right of the $\mid\!\sim_K$. The *Cut* property expresses the fact that one may, in his way towards a plausible conclusion, first add a hypothesis to the facts he knows to be true and prove the plausibility of his conclusion from this enlarged set of facts and then infer inductively this added hypothesis from the facts. Notice that the validity of *Cut* does not imply monotonicity. Nevertheless, we have seen that a form of *Cautious Monotonicity* holds for our relation.

Observe also that the inductive consequence relation $\mid\!\sim_K$ does not satisfy *Right Weakening*: If $K \vdash \alpha \to \beta$ and $\Delta \mid\!\sim_K \beta \to C$, then $\Delta \mid\!\sim_K \alpha \to C$. The reason is that, since α is more specific than β, the rule $\alpha \to C$ may cover no positive example. *Right Weakening* expresses the fact that one must be ready to accept as plausible consequences all that is logically implied by what one thinks are plausible consequences. We have proposed instead a *Cautious Right Weakening* property as the one that is relevant in our model.

Let us now analyze some additional properties, which are specially relevant for inductive concept learning.

Proposition 2. *The inductive consequence relation $\mid\!\sim_K$ satisfies the following properties:*

1. *If $\Delta \mid\!\sim_K \alpha \to C$ and $K \vdash \alpha \to \varphi$ then $\Delta \not\mid\!\sim_K \varphi \to \neg C$.*
2. *If $\Delta \mid\!\sim_K \alpha \to C$ and $K \vdash \varphi \to \alpha$ then $\Delta \not\mid\!\sim_K \varphi \to \neg C$.*
3. *Falsity Preserving: let $r = \alpha \to C$ be such that it covers a negative example from Δ, hence $r \notin IR_K(\Delta)$; then $r \notin IR_K(\Delta \cup \Delta')$ for any further set of examples Δ'.*
4. *Positive Monotonicity: $\Delta \mid\!\sim_K \alpha \to C$ implies $\Delta \cup \{\varphi(a) \wedge C(a)\} \mid\!\sim_K \alpha \to C$.*
5. *Negative Monotonicity: if $\varphi(a) \wedge \neg C(a) \in \Delta$, $\Delta \mid\!\sim_K \alpha \to C$ implies $\Delta \setminus \{\varphi(a) \wedge \neg C(a)\} \mid\!\sim_K \alpha \to C$.*
6. *Positive Non-monotonicity: if $\varphi(a) \wedge C(a) \in \Delta$, $\Delta \mid\!\sim_K \alpha \to C$ does not imply $\Delta \setminus \{\varphi(a) \wedge C(a)\} \mid\!\sim_K \alpha \to C$.*
7. *Negative Non-monotonicity: $\Delta \mid\!\sim_K \alpha \to C$ does not imply $\Delta \cup \{\varphi(a) \wedge \neg C(a)\} \mid\!\sim_K \alpha \to C$, but it implies $\Delta \cup \{\varphi(a) \wedge \neg C(a)\} \not\mid\!\sim_K \alpha \to \neg C$.*

8. *Generalization: if* $\Delta = \{\varphi(a) \wedge C(a)\}$ *and* $\Delta \mid\!\sim_K \alpha \to C$ *then* $K \vdash \varphi \to \alpha$.
9. *If* $\Delta_1 \cup \Delta_2 \mid\!\sim_K \alpha \to C$ *then either* $\Delta_1 \mid\!\sim_K \alpha \to C$ *or* $\Delta_2 \mid\!\sim_K \alpha \to C$, *that is,* $IR_K(\Delta_1 \cup \Delta_2) \subseteq IR_K(\Delta_1) \cup IR_K(\Delta_2)$.

Proof.

1. Let us assume that $K \vdash \alpha \to \varphi$ and $\Delta \mid\!\sim_K \varphi \to \neg C$. Then, by Consistency, for all $\psi(a) \wedge C(a) \in \Delta$ we have $\psi(a) \wedge K \nvdash \varphi(a)$, and hence $\psi(a) \wedge K \nvdash \alpha(a)$ as well. Then, clearly $\Delta \not\mid\!\sim_K \alpha \to C$.
2. Let us assume now that $K \vdash \varphi \to \alpha$ and $\Delta \mid\!\sim_K \varphi \to \neg C$. Then, by Explanation, there exists $\psi(a) \wedge \neg C(a) \in \Delta$ such that $\psi(a) \wedge K \vdash \varphi(a)$. But then we have $\psi(a) \wedge K \vdash \alpha(a)$ as well, so again $\Delta \not\mid\!\sim_K \alpha \to C$.
3. Notice that if r covers a negative example of Δ, that particular example will remain in $\Delta \cup \Delta'$.
4. This property is stronger than Cautious Monotonicity, and follows by the same argument.
5. It is direct consequence that if $\alpha \to C$ follows from Δ, it cannot cover any negative example.
6. Removing a positive example invalidates an inductive inference when that example is the only one covered the rule.
7. $\Delta \mid\!\sim_K \alpha \to C$ does not imply $\Delta \cup \{\varphi(a) \wedge \neg C(a)\} \mid\!\sim_K \alpha \to C$ because nothing prevents $\varphi(a) \wedge K \vdash \alpha(a)$ to hold. The fact that $\Delta \cup \{\varphi(a) \wedge \neg C(a)\} \not\mid\!\sim_K \alpha \to \neg C$ is there a consequence of Properties 3 and 1.
8. If Δ consists of only one positive example $\varphi(a) \wedge C(a)$, the only way for α to cover $\varphi(a)$ is that α (classically) follows from φ.
9. Let $r \in IR_K(\Delta_1 \cup \Delta_2)$ (see Definition 4). It means that r at least covers a positive example $e^+ \in \Delta_1 \cup \Delta_2$ and covers no negative example of $\Delta_1 \cup \Delta_2$, so it covers no negative example of both Δ_1 and Δ_2. Now, if $e^+ \in \Delta_1$ then clearly $r \in IR_K(\Delta_1)$; otherwise, if $e^+ \in \Delta_2$, then $r \in IR_K(\Delta_2)$, hence in any case $r \in IR_K(\Delta_1) \cup IR_K(\Delta_2)$.

□

Let us now examine the intuitive interpretation of the properties in Proposition 2 from the point of view of ICL; for this purpose we will reformulate some notions into the vocabulary commonly used in ICL.

Properties 1 and 2 state that by generalizing (resp. specializing) an induced rule will never conclude the negation of the target concept. Property

3 states the well known fact that induction is falsity preserving, i.e. once we know some induced rule is not valid, it will never be valid again by adding more examples to Δ. Property 4 states that adding a positive example e^+ does not invalidate any existing induced rule, i.e. $IR_K(\Delta)$ does not decrease; notice that it can increase since $IR_K(\Delta \cup \{e^+\})$ might have induced rules that explain e^+ that were not in $IR_K(\Delta)$. Property 5 states that no negative example can be covered if $\alpha \rightarrow C$ follows from Δ. Property 6 states that when we remove the only positive example covered by the rule, we invalidate the inductive inference.

Property 7 states that adding a negative example e^- might invalidate existing induced rules in $IR_K(\Delta)$, i.e. $IR_K(\Delta \cup \{e^-\}) \subseteq IR_K(\Delta)$. This is related to Property 3, since once a negative example defeats an induced rule r, we know r will never be valid regardless of how many examples are added to Δ. Property 8 states a generalization property, in the case where Δ consists of only one positive example. Property 9 states that the rules that can be induced from the union of two sets of examples are a subset of the union of the rules that can be induced from each of the sets.

Actually, a few of the mentioned properties in Propositions 1 and 2 suffice to fully characterize the inductive consequence relation $\mid\!\sim_K$, as we will show presently. For the sake of simplicity, we will assume that we don't have any background knowledge K.

Proposition 3. (Characterization) *Let $\mid\!\approx$ be a relation between consistent sets of examples for a concept C and rules satisfying the following properties:*

(P1) Reflexivity*: if $\varphi(a) \wedge C(a) \in \Delta$ then $\Delta \mid\!\approx \varphi \rightarrow C$*

(P2) Generalization*: if $\Delta = \{\varphi(a) \wedge C(a)\}$ and $\Delta \mid\!\approx \alpha \rightarrow C$ then $\vdash \varphi \rightarrow \alpha$*

(P3) Negative Monotonicity*: if $\Delta \mid\!\approx \alpha \rightarrow C$ and $\varphi(a) \wedge \neg C(a) \in \Delta$, then $\Delta \setminus \{\varphi(a) \wedge \neg C(a)\} \mid\!\approx \alpha \rightarrow C$*

(P4) *If $\Delta_1 \cup \Delta_2 \mid\!\approx \alpha \rightarrow C$ then either $\Delta_1 \mid\!\approx \alpha \rightarrow C$ or $\Delta_2 \mid\!\approx \alpha \rightarrow C$*

(P5) *If $\Delta \mid\!\approx \alpha \rightarrow C$ and $\vdash \alpha \rightarrow \varphi$ then $\Delta \not\mid\!\approx \varphi \rightarrow \neg C$.*

Then, it holds that $\Delta \mid\!\approx \alpha \rightarrow C$ iff $\alpha \rightarrow C$ covers at least one positive example of C and does not cover any negative example of C in Δ, as required by Definition 2.

Proof. In what follows, given a consistent set of examples Δ and a concept C, we will denote by Δ^+ its subset of positive examples for C in Δ, and by Δ^- its set of negative examples. Assume $\Delta \mid\!\approx \alpha \to C$, we have to prove that (i) $\alpha \to C$ covers some positive example in Δ and (ii) $\alpha \to C$ does not cover any negative example.

(i) Using (P3) one can remove all negative examples from Δ but still preserving the consequence, i.e. we have $\Delta^+ \mid\!\approx \alpha \to C$. Now, using (P4), we conclude that there must exist at least one positive example $\varphi(a) \wedge C(a) \in \Delta^+$ such that $\{\varphi(a) \wedge C(a)\} \mid\!\approx \alpha \to C$. Finally, by (P2), one has that $\vdash \varphi \to \alpha$.

(ii) By contraposition. Assume $\alpha \to C$ covers a negative example $\psi(b) \wedge \neg C(b) \in \Delta^-$, and hence $\vdash \psi \to \alpha$. By (P1), we have $\Delta \mid\!\approx \psi \to \neg C$, and since $\vdash \psi \to \alpha$, by (P5) we also have $\Delta \not\mid\!\approx \alpha \to C$, contradiction. □

3.2. Inductive Theories

The notions of inductive consequence and inducible rules allow us to define the idea of an *inductive theory* for a given concept as a set of inducible rules which, together with the background knowledge, explain all the positive examples.

Definition 6. (Inductive Theory) *An* inductive theory T *for a concept* C*, w.r.t.* Δ *and* K*, is a subset* $T \subseteq IR_K(\Delta)$ *such that all the rules in* T *conclude* C*, and for all* $\varphi(a) \wedge C(a) \in \Delta$*, it holds that* $T \cup K \cup \{\varphi(a)\} \vdash C(a)$*.* T *is* minimal *if there is no* $T' \subset T$ *that* T' *is an inductive theory for* C*.*

Since rules concluding C in $IR_K(\Delta)$ do not cover any negative example of C, if T is an inductive theory for C w.r.t. Δ and K, and $\psi(a) \wedge \neg C(a) \in \Delta$ for some constant a, then it holds that $T \cup K \cup \{\psi(a)\} \nvdash C(a)$. Observe that, in the case that Δ is a consistent set of examples, the existence of inductive theories is guaranteed due to the reflexivity property: the set of all rules obtained lifting examples is an inductive theory. Notice also that the notion of inductive theory is relevant for ICL because an inductive machine learning algorithm has as output a specific inductive theory.

3.3. Exemplification

The *Zoology* data set is a standard machine learning dataset containing 101 instances of animals associated with an animal family (fish, insect, mammal, etc.). The goal is to learn general descriptions of each of the families by induction. For exemplification purposes, we will use *mammal* as our target

concept, represented by m. The Zoology dataset, as available from the UCI machine learning repository, has no background knowledge so $K = \emptyset$. Let us now consider three animals in *Zoology* (an aardvark, an antelope and a bass):

$$\begin{aligned}
e_1 := \quad & hair(a_1) \wedge milk(a_1) \wedge predator(a_1) \wedge toothed(a_1) \\
& \wedge backbone(a_1) \wedge breathes(a_1) \wedge fourlegged(a_1) \\
& \wedge catsize(a_1) \wedge m(a_1) \\
& = \varphi_1(a_1) \wedge m(a_1) \\
e_2 := \quad & hair(a_2) \wedge milk(a_2) \wedge toothed(a_2) \wedge backbone(a_2) \\
& \wedge breathes(a_2) \wedge fourlegged(a_2) \wedge tail(a_2) \\
& \wedge catsize(a_2) \wedge m(a_2) \\
& = \varphi_2(a_2) \wedge m(a_2) \\
e_3 := \quad & eggs(a_3) \wedge aquatic(a_3) \wedge predator(a_3) \wedge fins(a_3) \\
& \wedge backbone(a_3) \wedge toothed(a_3) \wedge tail(a_3) \wedge \neg m(a_3) \\
& = \varphi_3(a_3) \wedge \neg m(a_3)
\end{aligned}$$

Given $\Delta = \{\varphi_1(a_1) \wedge m(a_1), \varphi_2(a_2) \wedge m(a_2), \varphi_3(a_3) \wedge \neg m(a_3)\}$, to illustrate $\mid\!\sim_K$, we consider several hypotheses:

$$\begin{aligned}
r_1 := \quad & (\forall x)(hair(x) \wedge milk(x) \rightarrow m(x)) \\
r_2 := \quad & (\forall x)(toothed(x) \wedge backbone(x) \rightarrow m(x)) \\
r_3 := \quad & (\forall x)(tail(x) \wedge domestic(x) \rightarrow m(x)) \\
r_4 := \quad & (\forall x)(fourlegged(x) \rightarrow m(x))
\end{aligned}$$

- $\Delta \mid\!\sim_K r_1$, because both $\varphi_1(a_1) \vdash hair(a_1) \wedge milk(a_1)$ and $\varphi_2(a_2) \vdash hair(a_2) \wedge milk(a_2)$ (thus satisfying the explanation condition) and $\varphi_3(a_3) \nvdash hair(a_3) \wedge milk(a_3)$ (thus satisfying the consistency condition).
- $\Delta \not\mid\!\sim_K r_2$, because $\varphi_1(a_1) \vdash toothed(a_1) \wedge backbone(a_1)$, hence it satisfies explanation, but $\varphi_3(a_3) \vdash toothed(a_3) \wedge backbone(a_3)$, and thus it's not consistent.
- $\Delta \not\mid\!\sim_K r_3$, because $\varphi_1(a_1) \nvdash tail(a_1) \wedge domestic(a_1)$ and $\varphi_2(a_2) \nvdash tail(a_2) \wedge domestic(a_2)$, i.e. it does not satisfy the explanation condition. So, even if r_3 satisfies the consistency condition, it does not explain any example.
- $\Delta \mid\!\sim_K r_4$, because both $\varphi_1(a_1) \vdash fourlegged(a_1)$ and $\varphi_2(a_2) \vdash fourlegged(a_2)$ (thus satisfying the explanation condition) and $\varphi_3(a_3) \nvdash fourlegged(a_3)$ (thus satisfying the consistency condition).

In this example, the sets $T_1 = \{r_1\} \subseteq IR_K(\Delta)$, $T_2 = \{r_4\} \subseteq IR_K(\Delta)$ and $T_3 = \{r_1, r_4\} \subseteq IR_K(\Delta)$ are inductive theories of m w.r.t. Δ. Clearly, only T_3 is not minimal.

4. Preference over Inductive Consequences

Although many rules can be inductive consequences of a given set of examples, ICL algorithms have a set of preferences and inductive biases that make them prefer some rules over the rest, or some inductive theories over the rest. For example, rules that cover more positive examples are preferred over rules that cover less examples, shorter rules are preferred over longer rules, and hypotheses with larger *margins* are preferred over those with smaller margins [16]. In our model of inductive generalization we incorporate these criteria by means of a preference relation.

Depending on the bias we want to model, the preference relation might be defined over rules or over inductive theories. In either case, since preference might depend on both the set of examples Δ and background knowledge K, we will note our preference relation by $\geq_{\Delta,K}$.

When the preference is expressed over rules, we write $r_1 \geq_{\Delta,K} r_2$ when r_1 is at least as preferred as r_2 ($r_1 >_{\Delta,K} r_2$ when r_1 is strictly preferred to r_2). In general, this preference relation is only assumed to be a partial preorder in the set $IR_K(\Delta)$.

Definition 7. (Preferred Rules) *The set of* preferred rules $IR_K^{\geq}(\Delta) = \{r \in IR_K(\Delta) \mid \nexists r' \in IR_K(\Delta) : r' >_{\Delta,K} r\}$ *is the subset of inducible rules that are maximally preferred.*

When the preference is expressed over inductive theories, we write $T_1 \geq_{\Delta,K} T_2$ when T_1 is at least as preferred as T_2. Again, in general, this preference relation is assumed to be only a partial preorder on the set of possible inductive theories.

Given that ICL algorithms ultimately return inductive theories, if the preference is expressed over rules, we are interested in having a preference over inductive theories, which can be defined as follows.

Definition 8. (Preference over Inductive Theories) *Given a preference $\geq_{\Delta,K}$ over rules, an inductive theory T is* preferred *over another theory T', denoted $T \geq_{\Delta,K} T'$, if there exist $r \in T, r' \in T'$ such that $r \geq_{\Delta,K} r'$, and for each $r \in T$ there is no $r' \in T'$ such that $r' >_{\Delta,K} r$.*

Having a preference relation $\geq_{\Delta,K}$ on inductive theories allows us to define the following concepts of preferred and ideal inductive theories.

Definition 9. (Preferred Inductive Theory) *We say that an inductive theory T is* (maximally) preferred *with respect to $\geq_{\Delta,K}$ if there is no other inductive theory $T' \subseteq IR_K(\Delta)$ such that $T' >_{\Delta,K} T$.*

Definition 10. (Ideal Inductive Theory) *We say that an inductive theory T is* ideal *with respect to $\geq_{\Delta,K}$ if it is both maximally preferred w.r.t. $\geq_{\Delta,K}$ and minimal.*

We remark that in the previous definition the term "ideal theory" neither carries any implicit meaning of being a "best" theory according to some unspecified criterion nor any other mathematical or algebraic meaning, it is just a shorthand to denote an inductive theory that is minimal and maximally preferred (according to a given preference relation).

Next, we present two examples of how some typical biases of ICL techniques can be expressed using our framework.

4.1. Parsimony

Most ICL algorithms have a bias towards finding shorter hypotheses (i.e. Parsimony or Occam's Razor), which typically translates to more general hypotheses. We can formalize both notions using two preference relations.

Given a function $size(T)$, which returns the number of symbols required to express the inductive theory T in a given logical language, we can define the preference $T_1 \geq_{\Delta,K} T_2 \Leftrightarrow size(T_1) \leq size(T_2)$, which effectively captures the bias towards shorter hypotheses.

A bias towards more general hypotheses is easier to express as a preference relation between rules. We can define the preference relation $\alpha \rightarrow C \geq_{\Delta,K} \beta \rightarrow C$ iff $\beta \wedge K \vdash \alpha$, i.e. the rule $\alpha \rightarrow C$ is preferred to $\beta \rightarrow C$ if it is more general. Then, using Definition 8, a preference over inductive theories can be established, as well as preferred (Definition 9) and ideal (Definition 10) inductive theories.

4.2. Margin Maximization

In machine learning, *margin* is commonly defined as the distance from the examples to the decision boundary [16]. A classifier which maximizes the margin has the decision boundary far away from every example; this ensures that small variations in the training set do not result in misclassifications.

Margin maximization is usually employed in machine learning and pattern recognition techniques where instances are represented in metric spaces. We will show now that an analogous principle can be applied for logic-based instance representation.

First, in order to use the notion of margin, we need to define some measure of distance, or similarity, between examples. To formalize this notion, we assume for simplicity that all predicates in the language are unary and that examples $\varphi(a) \wedge C(a)$ are such that $\varphi(a)$ is a conjunction of literals, i.e. $\varphi(a)$ is of the form $p_1(a) \wedge \ldots \wedge p_k(a) \wedge \neg p_{k+1}(a) \wedge \ldots \wedge \neg p_n(a)$. In that case, the only variable in a predicate stands for an example identifier, and hence for our purposes here we can actually consider these unary predicates as atomic propositions. Simplifying, we will denote by φ the propositionalized version of $\varphi(a)$, i.e. $\varphi = p_1 \wedge \ldots \wedge p_k \wedge \neg p_{k+1} \wedge \ldots \wedge \neg p_n$. This is indeed the case in the example described in Section 5. We will further assume the set $\mathcal{P}$ of unary predicates (now propositional variables) we work with is finite.

Let $\Omega = \{w : \mathcal{P} \to \{0, 1\}\}$ be the space of possible worlds. Given a propositional formula φ, we will denote by $[\varphi]$ the set of possible worlds satisfying the formula φ (according to classical propositional logic). We assume there is a distance function $\delta : \Omega \times \Omega \to \mathbb{R}^+$. The intuition is that $\delta(w, w')$ evaluates how far or different two worlds w and w' are: $\delta(w, w') = 0$ means that $w = w'$, $0 < \delta(w, w') < 1$ means that w resembles to w' to some degree. A usual choice for δ, among others (see e.g. [17, 18]), is the Hamming distance, that counts the number of elements of $\mathcal{P}$ over which two worlds differ.

Given such a distance function δ on the set of possible worlds Ω, the distance between two propositional formulas built form $\mathcal{P}$ can be measured by the distance between the corresponding sets of possible worlds, using the well-known Hausdorff distance derived from δ: $\delta_H(\varphi, \psi) = \max(I_\delta(\varphi \mid \psi), I_\delta(\psi \mid \varphi))$, where

$$I_\delta(\varphi \mid \psi) = \max_{w \in [\psi]} \min_{w' \in [\varphi]} \delta(w, w')$$

Now, given a set of examples Δ, a distance δ and a threshold $\tau \in \mathbb{R}^+$, we can consider an expanded set of examples Δ^*_τ where for each $\varphi(a) \wedge C(a) \in \Delta$ (resp. $\varphi(b) \wedge \neg C(b) \in \Delta$) we include all those additional fictitious examples $\psi(a') \wedge C(a')$ (resp. $\psi(b') \wedge \neg C(b')$), such that the distance between ψ and φ is at most τ, i.e. such that $\delta_H(\varphi, \psi) \leq \tau$.

Given an inductive theory $T \subseteq IR_K(\Delta)$, we say that T is *valid to the level* τ whenever T is also an inductive theory of $IR_K(\Delta^*_\tau)$ (hence, in particular,

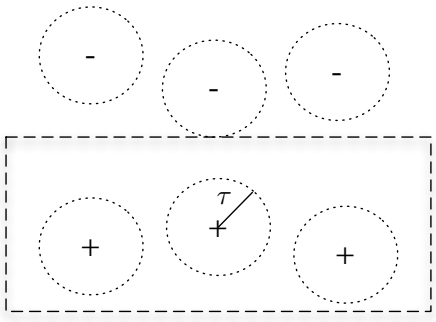

Figure 1: Margin based on similarity measure δ_H.

Δ^*_τ must be consistent). We assign a *preference degree* τ to an inductive theory T, noted $Pref(T) = \tau$, when τ is the maximum for which T is still an inductive theory of $IR_K(\Delta^*_\tau)$ (i.e. $Pref(T)$ is the maximum degree to which T is valid). This induces a natural preference over inductive theories: $T \geq_{\Delta,K} T' \Leftrightarrow Pref(T) \geq Pref(T')$. Moreover, according to Definition 9, a preferred inductive theory T is one such that there is no other inductive theory $T' \subseteq IR_K(\Delta)$ such that $T' >_{\Delta,K} T$.

Notice that, in the present setting, a preferred inductive theory T maximizes the margin according to the distance δ_H. As shown in Fig.1, the reason is that Δ^*_τ is expanding as much as possible around all positive examples $\varphi(a) \wedge C(a)$ and negative examples $\varphi(b) \wedge \neg C(b)$ without T covering any fictitious example of the opposite sign.

4.3. Exemplification

Let us now illustrate the use of preferences by continuing the exemplification started in Section 3.3.

Let us consider again the inductive theories used before: $T_1 = \{r_1\} \subseteq IR_K(\Delta)$, $T_2 = \{r_4\} \subseteq IR_K(\Delta)$ and $T_3 = \{r_1, r_4\} \subseteq IR_K(\Delta)$, and consider a new inductive theory $T_4 = \{r_1, r_5, r_6\}$, where:

$$\begin{aligned} r_5 :=&\quad (\forall x)(hair(x) \wedge fourlegged(x) \rightarrow m(x)) \\ r_6 :=&\quad (\forall x)(milk(x) \wedge fourlegged(x) \rightarrow m(x)) \end{aligned}$$

Given a function $size(\cdot)$, counting the symbols in a formula (ignoring parenthesis), the size of an inductive theory is simply the sum of the sizes of its rules. Thus we have: $size(r_1) = 10$, $size(r_4) = 7$, $size(r_5) = 10$, $size(r_6) = 10$, and therefore: $size(T_1) = 10$, $size(T_2) = 7$, $size(T_3) = 17$, and

$size(T_4) = 30$. Using the parsimony preference we have: $T_2 \geq_{\Delta,K} T_1 \geq_{\Delta,K} T_3 \geq_{\Delta,K} T_4$. In fact, there is no other inductive theory with size smaller than 7, and thus T_2 is a preferred inductive theory. Since T_2 is also minimal, it is actually an ideal inductive theory.

Notice, however, that if we were to use margin maximization as the preference criterion, with the Hamming distance, T_2 would not be preferred, since it is only valid to the level 0. In fact, the margin preference degrees of these inductive theories are $Pref(T_1) = 0$, $Pref(T_2) = 0$, $Pref(T_3) = 0$, $Pref(T_4) = 1$ and, thus, T_4 would be preferred to the others.

5. Induction and Argumentation

One of the main advantages of having a logical model of induction is that it allows an easy integration of inductive reasoning with other forms of logical reasoning. In order to illustrate its benefits, this section presents a model of multiagent ICL obtained by directly integrating our inductive consequence relation with computational argumentation. In this integration, argumentation is used to model the communication between agents, and ICL models their internal learning processes.

For the sake of simplicity of presentation, we will consider a multiagent system scenario with two agents Ag_1 and Ag_2 having a same target concept C. However, as shown in A, our main theoretical result applies to the more general case of an arbitrary number of agents. We make the following assumptions:

1. The background knowledge K of both agents is the same[2],
2. The set of rules $\mathcal{L}_r$ and the set of examples $\mathcal{L}_e$ are defined as before (see Section 3).
3. Each agent has a set of examples $\Delta_1, \Delta_2 \subseteq \mathcal{L}_e$ such that $\Delta_1 \cup \Delta_2$ is consistent.

The goal of each agent Ag_i is to induce an inductive theory T_i such that $T_i \subseteq IR(\Delta_1 \cup \Delta_2)$ and that constitutes an inductive theory w.r.t. $\Delta_1 \cup \Delta_2$. We will call this problem *multiagent ICL*.

[2]For simplicity, since both agents share K and C, in the rest of this paper we will write $IR(\Delta)$ rather than $IR_K(\Delta)$, and just say *inductive theory*, instead of saying *inductive theory of* C.

For this purpose, a naïve approach would be to have both agents sharing their complete sets of examples; however, that might not be always feasible for a number of reasons, like cost or privacy. In this section, we will show that by sharing some of the rules they have induced from examples (rather than all of their examples), two agents can also solve the multiagent ICL problem. Let us start presenting our computational argumentation framework.

5.1. Computational Argumentation

We will follow Dung's abstract argumentation formalization [19] and define an *argumentation framework* as a pair $\mathcal{A} = (\Gamma, \twoheadrightarrow)$, where Γ is a finite set of arguments, and $\twoheadrightarrow$ is an attack relation.

Given two arguments, r and r', we write $r \twoheadrightarrow r'$ to represent that r attacks r'. Moreover, if both $r \twoheadrightarrow r'$ and $r' \twoheadrightarrow r$ we say that r *blocks* r'.

As in any argumentation system, the goal is to determine whether a given argument is defeated or not according to a given semantics. In our case we will adopt the semantics based on *dialectical trees* [20, 21] explained below:

Definition 11. (Argumentation Line) *Given an argumentation framework $\mathcal{A} = (\Gamma, \twoheadrightarrow)$ and $r_0 \in \Gamma$, an* argumentation line *in $\mathcal{A}$ rooted in r_0 is a sequence: $\lambda = \langle r_0, r_1, r_2, \ldots, r_k \rangle$ such that:*

1. $r_{i+1} \twoheadrightarrow r_i$ (f*or* $i \leq k$),
2. *if* $r_{i+1} \twoheadrightarrow r_i$ *and* r_i *blocks* r_{i-1} *then* $r_i \not\twoheadrightarrow r_{i+1}$.

The argument r_k is called the leaf node of λ.

Additionally, for the purposes of ICL, we will assume that the attack relation has no cycles (which is the case for the definition of attack we will introduce later in this paper, Definition 12), and hence there are no repeated arguments in an argumentation line. Consequently, argumentation lines are always finite by construction. The set $\Lambda(r_0)$ of *maximal* argumentation lines rooted in r_0 are those that are not subsequences of other argumentation lines rooted in r_0. Clearly, $\Lambda(r_0)$ can be arranged in the form of a tree, where all paths from the root to the leaf nodes exactly correspond to all the possible maximal argumentation lines rooted in r_0 that can be constructed in the given argumentation framework. In order to decide whether r_0 is defeated in $\mathcal{A}$, the nodes of this tree are marked U (undefeated) or D (defeated) according to the following (cautious) rules:

1. Every leaf node is marked U,

2. Each inner node is marked U iff all of its children are marked D, otherwise it is marked D.

Therefore, the arguments in the argumentation framework $\mathcal{A}$ will be either undefeated or defeated according to their marking, as follows:

Undefeated: $\mathbf{U}(\mathcal{A}) = \{r \in \Gamma \mid r \text{ is marked U in the tree } \Lambda(r)\}$

Defeated: $\mathbf{D}(\mathcal{A}) = \{r \in \Gamma \mid r \text{ is marked D in the tree } \Lambda(r)\}$.

5.2. Argumentation-based Induction

Induction and argumentation can be integrated through the notion of *argumentation-consistent* induction. While induction was defined with respect to a set of observations Δ, argumentation-consistent induction will be defined with respect to a set of observations Δ, and a set of arguments Θ. The essential idea is that we consider arguments to be rules, i.e. $\Theta \subseteq \mathcal{L}_r$ (an example can also be used as an argument through its corresponding lifted rule, see the reflexivity property in Proposition 1). Therefore, in the rest of this paper, we will use the terms "rule" and "argument" interchangeably.

Given that arguments will be rules, we can now define the attack relation $\twoheadrightarrow$ between rules as follows.

Definition 12. (Attack) *Given two rules $r, r' \in \Gamma$, an attack relation $r \twoheadrightarrow r'$ holds whenever:*

1. $r = (\forall x)(\alpha(x) \rightarrow \ell(x))$,
2. $r' = (\forall x)(\beta(x) \rightarrow \neg\ell(x))$, *and*
3. $K \vdash (\forall x)(\alpha(x) \rightarrow \beta(x))$.

where $\neg\ell = \neg C$ when $\ell = C$ and $\neg\ell = C$ when $\ell = \neg C$.

Argumentation-consistent induction consists of inducing rules that agree with both Δ (i.e. not covering negative examples present in Δ) and Θ (i.e. not being defeated by the arguments in Θ).

Definition 13. (Argumentation-consistent Inducible Rule)
A rule $r \in IR(\Delta)$ is argumentation-consistent *with respect to a set of arguments Θ if $r \in \mathbf{U}(\mathcal{A})$, where $\mathcal{A} = (\Theta \cup IR(\Delta), \twoheadrightarrow)$.*

The set of all the argumentation-consistent rules induced is $AIR(\Delta, \Theta) = IR(\Delta) \cap \mathbf{U}(\mathcal{A})$.

Now we can define argumentation-consistent inductive theories.

Definition 14. (Argumentation-consistent Inductive Theory) *An argumentation-consistent inductive theory T, with respect to Δ and a set of arguments Θ, is an inductive theory of Δ such that $T \subseteq AIR(\Delta, \Theta)$.*

In the multiagent context starting in the next section, the goal of an agent is to build an argumentation-consistent inductive theory, where such theory will be composed by rules that have not been defeated by a set of arguments Θ coming from another agent.

6. Argumentation-consistent Induction in Multiagent Systems

Let us now show how the notion of argumentation-consistent induction can be used to model induction in a scenario with two agents. The main idea is that agents induce rules from the examples they know, and then they share them with the other agent. Rules received from the other agent are added into the own agent's argumentation framework to update her argumentation-consistent induced rules. Thus, in addition to the set of examples Δ_i, each agent Ag_i has an individual argumentation framework $\mathcal{A}_i$, containing both (1) the set of inducible rules $IR(\Delta)$ inducted by Ag_i and (2) the set of arguments Θ received from another agent.

Let us now prove that two agents communicating their induced rules and performing argumentation using the kind of attack in Definition 12 would obtain the exact same set of inducible rules as a single agent knowing the examples known to both agents.

Since the attack relation between rules is always the same, in the following we will simply write $\mathbf{U}(\Gamma)$ instead of $\mathbf{U}(\mathcal{A})$ to denote the set of undefeated rules of the argumentation system $\mathcal{A} = (\Gamma, \twoheadrightarrow)$.

Theorem 1. (Argumentation-consistent Induction)
$\mathbf{U}(IR(\Delta_1) \cup IR(\Delta_2)) = IR(\Delta_1 \cup \Delta_2)$.

Proof. Notice that by definition $\mathbf{U}(IR(\Delta)) = IR(\Delta)$; consequently, we have $AIR(\Delta, IR(\Delta)) = IR(\Delta)$.

First, we prove that $IR(\Delta_1 \cup \Delta_2) \subseteq \mathbf{U}(IR(\Delta_1) \cup IR(\Delta_2))$. Let $r \in IR(\Delta_1 \cup \Delta_2)$ then $r = \alpha \rightarrow C$ covers a positive example of $\Delta_1 \cup \Delta_2$ and does not cover any negative example of $\Delta_1 \cup \Delta_2$. W.l.o.g., assume the covered positive example is from Δ_1. Then $r \in IR(\Delta_1)$. Suppose there exists a rule

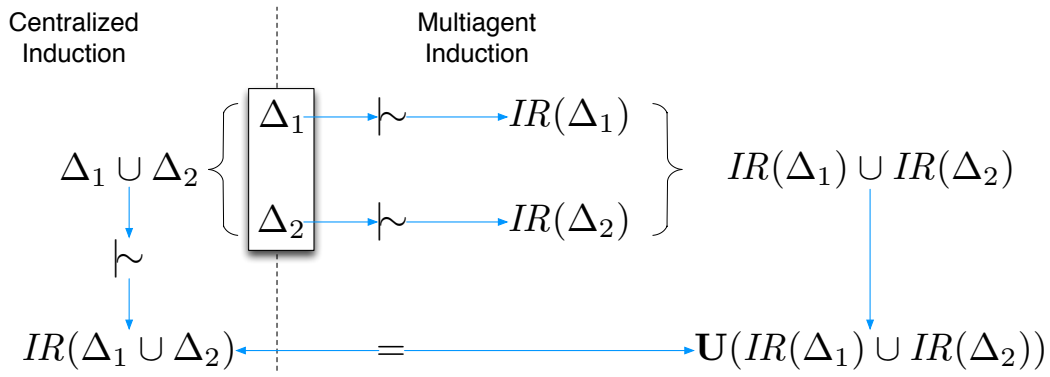

Figure 2: Achieving multiagent induction by combining inductive reasoning and computational argumentation.

$r' = \beta \rightarrow \neg C \in IR(\Delta_1) \cup IR(\Delta_2)$ such that $r' \twoheadrightarrow r$, i.e. such that $K \vdash \beta \rightarrow \alpha$. It is clear that $r' \notin IR(\Delta_1)$, hence assume that $r' \in IR(\Delta_2)$. This means r' covers a negative example $\delta^- \in \Delta_2$, but if r' covers it, r must cover δ^- as well, contradiction.

Second, we prove that $IR(\Delta_1 \cup \Delta_2) \supseteq \mathbf{U}(IR(\Delta_1) \cup IR(\Delta_2))$. Let $r \in \mathbf{U}(IR(\Delta_1) \cup IR(\Delta_2))$. W.l.o.g., assume $r \in IR(\Delta_1)$. Then $r = \alpha \rightarrow C$ covers a positive example of Δ_1 and does not cover any negative example of Δ_1. Assume also, looking for a contradiction, that $r \notin IR(\Delta_1 \cup \Delta_2)$. Since we have assumed that $r \in IR(\Delta_1)$, this means that r covers a negative example of Δ_2. This negative example can be specialized to a rule $r' = \beta \rightarrow \neg C \in IR(\Delta_2)$ such that $K \vdash \beta \rightarrow \alpha$. Since r' is the specialization of an example in Δ_2 and $\Delta_1 \cup \Delta_2$ is consistent, the rule r' is undefeated. Consequently, $r \notin \mathbf{U}(IR(\Delta_1) \cup IR(\Delta_2))$, which contradicts our original assumption. Therefore we can conclude $IR(\Delta_1 \cup \Delta_2) \supseteq \mathbf{U}(IR(\Delta_1) \cup IR(\Delta_2))$. □

The previous theorem shows that, given two agents, Ag_1 and Ag_2, each one with sets of examples Δ_1 and Δ_2 respectively, they can induce the same set of rules either by sharing their induced rules $IR(\Delta_1)$ and $IR(\Delta_2)$ and then using argumentation, or by exchanging all of their examples and then using induction. This equivalence is illustrated in Figure 2, that shows two equivalent approaches to obtain an inductive theory w.r.t $\Delta_1 \cup \Delta_2$: centralized induction (on the left hand side), and argumentation-consistent induction (on the right hand side). In A of this paper, we show how this result applies to the more general case of an arbitrary number of agents.

Clearly, sharing the complete $IR(\Delta_i)$'s is not a practical solution either,

since a) they can be very large, and b) given the reflexivity property, $IR(\Delta_i)$ contains Δ_i. Nevertheless, Theorem 1 shows that theoretically, the problem of multiagent ICL can be modeled using individual induction plus argumentation. In fact, if the purpose is finding inductive theories, not all arguments in the $IR(\Delta_i)$'s need to be exchanged. Section 7 presents a dialogue game that finds an inductive theory w.r.t. $\Delta_1 \cup \Delta_2$ using the same theoretical idea as used in Theorem 1, but focusing on exchanging a smaller subset of rules.

However, let us first illustrate the concepts of argumentation-consistent induction described in this section with an exemplification.

6.1. Exemplification

Consider two agents, Ag_1 and Ag_2, knowing a set of examples $\Delta_1 = \{e_1, e_2, e_4\}$ and $\Delta_2 = \{e_5, e_6, e_7\}$. Here, e_1, e_2 and e_3 are the three examples used in Section 3.3, and the new four examples (e_4 is a sealion, e_5 is a seasnake, e_6 is a platypus, and e_7 is a chicken) are defined as:

$$
\begin{aligned}
e_4 := \; & hair(a_4) \wedge milk(a_4) \wedge aquatic(a_4) \wedge predator(a_4) \wedge toothed(a_4) \\
& \wedge backbone(a_4) \wedge breathes(a_4) \wedge fins(a_4) \wedge twolegged(a_1) \\
& \wedge tail(a_4) \wedge catsize(a_4) \wedge m(a_4) \\
& = \varphi_4(a_4) \wedge m(a_4) \\
e_5 := \; & aquatic(a_5) \wedge predator(a_5) \wedge toothed(a_5) \wedge backbone(a_5) \\
& \wedge venomous(a_5) \wedge fins(a_5) \\
& \wedge tail(a_5) \wedge \neg m(a_5) \\
& = \varphi_5(a_5) \wedge \neg m(a_5) \\
e_6 := \; & hair(a_6) \wedge eggs(a_6) \wedge milk(a_6) \wedge aquatic(a_6) \wedge predator(a_6) \\
& \wedge backbone(a_6) \wedge breathes(a_6) \wedge fourlegged(a_6) \wedge tail(a_6) \\
& \wedge catsize(a_6) \wedge m(a_6) \\
& = \varphi_6(a_6) \wedge m(a_6) \\
e_7 := \; & feathers(a_7) \wedge eggs(a_7) \wedge airborne(a_7) \wedge backbone(a_7) \wedge \\
& breathes(a_7) \wedge twolegged(a_7) \wedge tail(a_7) \wedge \neg m(a_7) \\
& = \varphi_7(a_7) \wedge \neg m(a_7)
\end{aligned}
$$

Thus, Δ_1 contains three positive examples (e_1, e_2 and e_4) and no negative example, and Δ_2 contains two negative examples (e_5 and e_7) and one positive example (e_6). Let us now consider some of the rules that the agents can induce from those examples. For instance, two of the rules that Ag_1 can induce are $r_1, r_3 \in IR(\Delta_1)$ below:

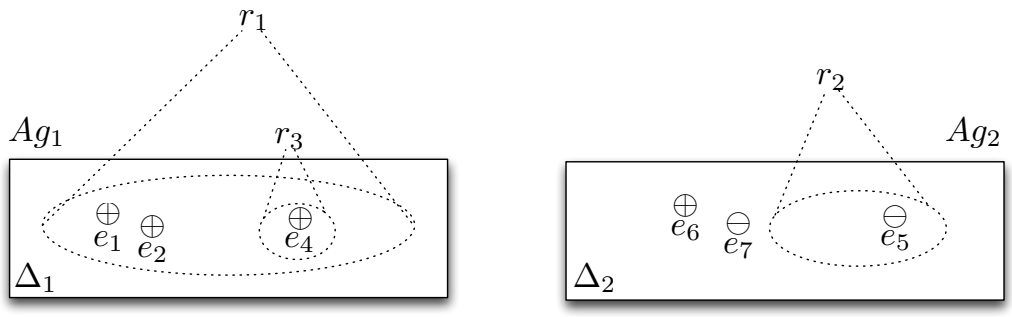

Figure 3: Two agents, Ag_1 and Ag_2, knowing different sets of examples, and some sample rules that can be induced from them.

$$r_1 := \quad (\forall x)(backbone(x) \rightarrow m(x))$$
$$r_3 := \quad (\forall x)(backbone(x) \wedge toothed(x) \wedge twolegged(x) \rightarrow m(x))$$

Agent Ag_2 can induce the rule $r_2 \in IR(\Delta_2)$:
$$r_2 := \quad (\forall x)(backbone(x) \wedge toothed(x) \rightarrow \neg m(x))$$

When the two agents perform induction in isolation, no issues are found with those three rules, as shown in Figure 3. However, let us consider now the situation where agent Ag_1 communicates r_1 and r_3 to Ag_2, and Ag_2 communicates r_2 to Ag_1. In this situation, according to Definition 12, the following attacks hold: $r_2 \twoheadrightarrow r_1$ and $r_3 \twoheadrightarrow r_2$.

Let us first consider Ag_1, who, in addition to its inducible rules $IR(\Delta_1)$, now has access to the set of rules $\Theta_{2\rightarrow 1} = \{r_2\}$. Now, to perform argumentation-consistent induction, Ag_1 assesses which are the rules that are both inducible from Δ_1 and consistent with $\Theta_{2\rightarrow 1}$. For that purpose, Ag_1 constructs the argumentation framework $\mathcal{A}_1 = (IR(\Delta_1) \cup \{r_2\}, \twoheadrightarrow)$. It is easy to verify that, since r_2 is attacked by r_3, and r_3 is not attacked by any other rule, r_2 is defeated. Thus, both r_1 and r_3 are argumentation-consistent inductions and belong to $AIR(\Delta_1, \Theta_{2\rightarrow 1})$. Therefore, knowing r_2 does not change the set of inducible rules of Ag_1, even if r_2 attacks r_1 (see Figure 4).

Now, considering agent Ag_2, in addition to its inducible rules $IR(\Delta_2)$, now Ag_2 has access to the set of rules $\Theta_{1\rightarrow 2} = \{r_1, r_3\}$. Similarly as before, to perform argumentation-consistent induction, Ag_2 assesses which are the rules that are both inducible from Δ_2 and consistent with $\Theta_{1\rightarrow 2}$. Ag_2 constructs the argumentation framework $\mathcal{A}_2 = (IR(\Delta_2) \cup \{r_1, r_3\}, \twoheadrightarrow)$. In this case, the rule r_2, induced by Ag_2 is defeated (because it is attacked by r_3, which is

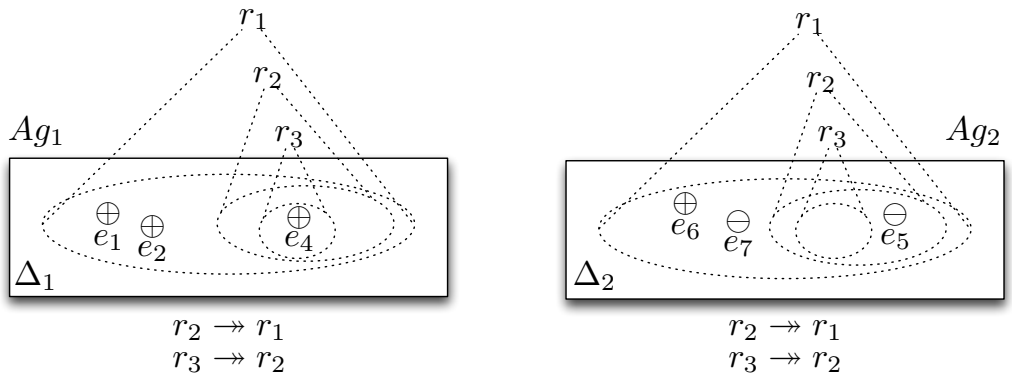

Figure 4: The same two agents from Figure 3, after they communicate some rules.

not attacked by any other rule), and thus $r_2 \notin AIR(\Delta_2, \Theta_{1\to 2})$. Thus, in this case, knowing r_1 and r_3 changes the the set of inducible rules of Ag_2.

7. Reaching Inductive Theories in Multiagent Concept Learning

While Theorem 1 shows that it is possible to solve the problem of multiagent ICL using individual induction plus argumentation, this section shows that when agents want to just agree on a single inductive theory, it is not necessary, in general, to exchange all of their induced rules. This section presents a dialogue game [22] through which two agents can solve the multiagent ICL problem by communication, specifically by exchanging some of the rules they induced from examples. To define the dialogue game, we need to define an interaction protocol, including the types of messages that agents are allowed to use, and the conditions under which types of messages can be exchanged. The dialogue game is defined for two agents Ag_1 and Ag_2, each of which has an individual set of examples Δ_1, Δ_2, and consists of a series of rounds. At each round t of the dialogue game, each agent Ag_i holds a current inductive theory, T_i^t, that is revised after each round. When the game terminates, both agents reach a common inductive theory with respect to $\Delta_1 \cup \Delta_2$.

During the dialogue game, agents communicate to each other rules induced from their examples. Through this rule exchange, an agent Ag_i may attack the inductive theory T_j^t of the other agent Ag_j if it is not consistent with Δ_i.

At the end of each round t, an agent Ag_i knows the following six pieces of information, namely $(\Delta_i, T_i^t, T_j^t, \Theta_{i\to j}^t, \Theta_{j\to i}^t, \mathcal{A}_i^t)$, where:

1. Δ_i is the set of examples known to Ag_i.
2. T_i^t is the current inductive theory w.r.t Δ_i that agent Ag_i is holding.
3. T_j^t is the current inductive theory w.r.t Δ_j that the other agent is holding.
4. $\Theta_{i\to j}^t$ is the set of arguments (rules) that agent Ag_i has sent to Ag_j up to the round t. Notice that $\Theta_{i\to j}^t \subseteq IR(\Delta_i)$.
5. $\Theta_{j\to i}^t$ is the set of arguments (rules) that agent Ag_i has received from Ag_j up to the round t.
6. $\mathcal{A}_i^t = (IR(\Delta_i) \cup \Theta_{j\to i}^t, \twoheadrightarrow)$ is the argumentation framework for Ag_i; notice that the set of arguments is composed of the rules inducible by Ag_i plus the arguments sent by the other agent Ag_j.

Let us now provide some auxiliary definitions, before we introduce the dialogue game interaction protocol.

Definition 15. (Defeaters of a rule)
Given an argumentation framework $\mathcal{A} = (\Gamma, \twoheadrightarrow)$, and a defeated argument $r \in \mathbf{D}(\mathcal{A})$, the set of defeaters of r is:

$$Defeaters(r, \mathcal{A}) = \{r' \in \Gamma \mid r' \twoheadrightarrow r \text{ and } r' \in \mathbf{U}(\mathcal{A})\}$$

That is to say, the set of undefeated arguments that attack r.

Definition 16. (Defeated Arguments From Communication)
Given the set of arguments $\Theta_{j\to i}^t$ communicated by Ag_j to Ag_i, the set of those received arguments that are defeated according to Ag_i is $\mathbf{D}_{j\to i}^t = \mathbf{D}(\mathcal{A}_i^t) \cap \Theta_{j\to i}^t$.

Using the previous definitions, we can now present the dialogue game through which two agents Ag_1 and Ag_2 can find an inductive theory w.r.t. $\Delta_1 \cup \Delta_2$.

Before the first round, at $t = 0$, the two agents are assumed to hold initial inductive theories $T_1^0 \subseteq IR(\Delta_1)$ and $T_2^0 \subseteq IR(\Delta_2)$ w.r.t. Δ_1 and Δ_2 respectively. Moreover, we assume each agent has communicated its own inductive theory to the other agent, and thus:

$$\Theta_{1\to 2}^0 = T_1^0 \text{ and } \Theta_{2\to 1}^0 = T_2^0,$$

Consequently, the initial argumentation systems of the agents are set to:

$$\mathcal{A}_1^0 = (IR(\Delta_1) \cup \Theta_{2\to1}^0, \twoheadrightarrow) \text{ and } \mathcal{A}_2^0 = (IR(\Delta_2) \cup \Theta_{1\to2}^0, \twoheadrightarrow).$$

Then, at each round t, starting at $t = 1$, each agent Ag_i computes the new values of the tuple $(T_i^t, \Theta_{i\to j}^t, \mathcal{A}_i^t)$, based on the values at the previous round $(T_i^{t-1}, \Theta_{i\to j}^{t-1}, \mathcal{A}_i^{t-1})$. Notice that Δ_i does not change and T_j^t and $\Theta_{j\to i}^t$ are computed by the other agent.

Actually, each round $t \geq 1$ of the protocol is divided in five simple steps: generate attacks, send attacks, update inductive theories, send updated inductive theories, and update state. The process ends when no agent generates new attacks. In more detail, a round t of the protocol is as follows:

1. **Generate Attacks**: Ag_i generates a set of attacks $\mathcal{R}_i^t$ by selecting a single argument (whichever) $r' \in \mathit{Defeaters}(r, \mathcal{A}_i^{t-1})$ for each $r \in \mathbf{D}_{j\to i}^{t-1}$ i.e. Ag_i selects one attack for each argument r sent by the other agent that is defeated according to Ag_i.
2. **Send Attacks**: Each agent Ag_i sends $\mathcal{R}_i^t$ to the other agent.

   If $\mathcal{R}_i^t = \mathcal{R}_j^t = \emptyset$, then the process terminates, since this means that the current theories held by each agent (T_i^{t-1} and T_j^{t-1}) are acceptable for the other agent (no attack can be found). Otherwise the protocol proceeds to the next step.

3. **Update Inductive Theories**: Each agent Ag_i generates a new argumentation-consistent inductive theory $T_i^t \subseteq AIR(\Delta_i, \Theta_{j\to i}^{t-1} \cup \mathcal{R}_j^t)$ such that $(T_i^{t-1} \cap \mathbf{U}(\mathcal{B}_i^{t-1})) \subseteq T_i^t$, where $\mathcal{B}_i^{t-1} = (IR(\Delta_i) \cup \Theta_{j\to i}^{t-1} \cup \mathcal{R}_j^t, \twoheadrightarrow)$ —i.e. the new theory T_i^t contains all the undefeated rules from T_i^{t-1} taking into account the attacks received, and replaces the rules that were defeated in T_i^{t-1} by new rules.
4. **Send Updated Inductive Theories**: Each agent Ag_i sends T_i^t to the other agent.
5. **Update State**: the set of arguments received by each agent is increased accordingly:
   - $\Theta_{1\to2}^t = \Theta_{1\to2}^{t-1} \cup \mathcal{R}_1^t \cup T_1^t$
   - $\Theta_{2\to1}^t = \Theta_{2\to1}^{t-1} \cup \mathcal{R}_2^t \cup T_2^t$

   both agents update their argumentation frameworks:
   - $\mathcal{A}_1^t = (IR(\Delta_1) \cup \Theta_{2\to1}^t, \twoheadrightarrow)$

- $\mathcal{A}_2^t = (IR(\Delta_2) \cup \Theta_{1\to 2}^t, \twoheadrightarrow)$

And new round $t+1$ starts by going back to the first step.

When the process terminates, both agents have a common and agreed argumentation-consistent inductive theory, namely $T^* = T_1^t \cup T_2^t$.

The reason is that, when the process terminates, if the set $\Delta_1 \cup \Delta_2$ is consistent, then each agent Ag_i has an argumentation-consistent inductive theory T_i^t w.r.t. Δ_i that is also consistent with the examples in Δ_j. Nevertheless, T_i^t might not be an inductive theory w.r.t. Δ_j, since there might be examples in Δ_j not covered by T_i^t. However, their union $T^* = T_1^t \cup T_2^t$ is an inductive theory w.r.t. the examples in $\Delta_1 \cup \Delta_2$ and, since both agents know T_1^t and T_2^t, both agents can have T^* as a common and agreed argumentation-consistent inductive theory w.r.t. $\Delta_1 \cup \Delta_2$, as the following theorem proves.

Theorem 2. *If the set $\Delta_1 \cup \Delta_2$ is consistent, the previous process always ends in a finite number of rounds t, and when it ends, $T_1^t \cup T_2^t$ is an inductive theory w.r.t. $\Delta_1 \cup \Delta_2$.*

Proof. First, let us prove that the final theories (T_1^t and T_2^t) are consistent with $\Delta_1 \cup \Delta_2$. For this purpose we will show that the termination condition ($\Theta_{1\to 2}^t = \Theta_{1\to 2}^{t-1}$ and $\Theta_{2\to 1}^t = \Theta_{2\to 1}^{t-1}$) implies that the argumentation-consistent inductive theory T_i^t found by agent Ag_i at the final round t has no counterexamples in either Δ_1 nor in Δ_2.

Let us assume that there is an example $a_k \in \Delta_1$ which is a counterexample of a rule $r \in T_2^t$. Because of the reflexivity property, there is a rule $r_k \in IR(\Delta_1)$ which corresponds to that example. Since $\Delta_1 \cup \Delta_2$ is consistent, there is no counterexample of r_k, and thus r_k is undefeated. Since $r_k \twoheadrightarrow r$ by assumption, r would have been defeated, and therefore rule r could not be part of any argumentation-consistent inductive theory generated by Ag_2. The same reasoning can prove that there are no counterexamples of T_1^t in $\Delta_1 \cup \Delta_2$.

Since T_1^t and T_2 are inductive theories w.r.t. Δ_1 and Δ_2 respectively, it follows from the above that $T_1^t \cup T_2^t$ is an inductive theory w.r.t. $\Delta_1 \cup \Delta_2$ because it has no counterexamples in $\Delta_1 \cup \Delta_2$, and every example in $\Delta_1 \cup \Delta_2$ is explained at least by one rule in T_1^t or in T_2^t.

Finally, the process has to terminate in a finite number of steps, since, by assumption, $IR(\Delta_1)$ and $IR(\Delta_2)$ are finite sets, and the sets $\Theta_{1\to 2}^t$ and $\Theta_{2\to 1}^t$ grow at least with one new argument at each round; however, since

$\Theta^t_{i\to j} \subseteq IR(\Delta_i)$, there is only a finite number of new arguments that can be added to $\Theta^t_{i\to j}$ before the termination condition holds. □

Thus, we have shown that the inductive theories achieved by argumentation-consistent induction are sound. Theorem 1 has shown that the set of inductive theories that can be reached through sharing examples is the same as the set of inductive theories that can be reached by sharing induced rules and then performing argumentation-consistent induction. Furthermore, Theorem 2 shows that it is possible to reach one of those inductive theories by using a simple dialogue game that does not require in general the exchange of all the induced rules made by an agent. As a consequence, centralizing all examples into a single inductive process is no longer imperative, at least in ICL, since induction followed by argumentation is a viable option.

The process to find a multiagent inductive theory can be seen as composed of three mechanisms: induction, argumentation and belief revision. Agents use induction to generate general rules from concrete examples, they use argumentation to decide which of the rules sent by another agent cannot be defeated, and finally they perform belief revision when they change their inductive theories in light of the arguments sent by another agent. The belief revision process is embodied in the way the set of undefeated rules $\mathbf{U}(\mathcal{A}^t_i)$ changes from round to round, which also determines how an agent's inductive theory changes in light of the arguments shared by the other agent.

A particular implementation of this integration model is the A-MAIL framework [9], where two agents perform induction on separate example sets and engage in argumentation until they reach individual inductive theories that are consistent with their example sets. The A-MAIL framework offers a particular realization of three mechanisms of induction, argumentation and belief revision. The need of having an argumentation-consistent inductive process is met by ABUI (Argumentation-based Bottom-Up Induction), a new inductive method that finds inductive rules consistent with the set of undefeated rules at any step of the argumentation process.

7.1. Exemplification

Let us assume we have two agents, Ag_1 and Ag_2 and let $\Delta_1 = \{e_1, e_2, e_3\}$ (containing the three examples used in Section 3.3, an aardvark, an antelope and a bass), and $\Delta_2 = \{e_4, e_6, e_7\}$ (containing some of the examples used in Section 6.1, a sealion, a platypus, and a chicken). Now, the two agents want

to find a common inductive theory of the concept *mammal*, represented by the unary predicate m. Let us explain the process.

Before the protocol starts, at $t = 0$, each agent has individually found an inductive theory:

$T_1^0 = \{(\forall x)(breathes(x) \rightarrow m(x))\}$, and
$T_2^0 = \{(\forall x)(aquatic(x) \rightarrow m(x))\}$.

Intuitively, since all the positive examples of mammal known to Ag_1 are land animals, and all the negative ones are not, Ag_1 has induced that *breathing* is enough to characterize a mammal. A similar situation has occurred with Ag_2, who has find by induction that being *aquatic* is enough to characterize a mammal, since it happens that the only two examples of mammals Ag_2 knows are aquatic.

Moreover, at $t = 0$, each agent has communicated to the other agent their individually found inductive theories and build their initial argumentation systems, and thus:

$\Theta_{1\rightarrow 2}^0 = T_1^0$, and $\Theta_{2\rightarrow 1}^0 = T_2^0$;
$\mathcal{A}_1^0 = (IR(\Delta_1) \cup \Theta_{2\rightarrow 1}^0, \twoheadrightarrow)$ and $\mathcal{A}_2^0 = (IR(\Delta_2) \cup \Theta_{1\rightarrow 2}^0, \twoheadrightarrow)$.

The protocol then proceeds as follows.

Round $t = 1$.

1. Agents proceed by generating attacks against the rules they have received they believe are defeated.
   - Since the rule $(\forall x)(aquatic(x) \rightarrow m(x))$ generated by Ag_2 is defeated according to Ag_1, Ag_1 selects one attack to defeat it: $\mathcal{R}_1^1 = \{(\forall x)(aquatic(x) \wedge \neg hair(x) \rightarrow \neg m(x))\}$;
   - Since the rule $(\forall x)(breathes(x) \rightarrow m(x))$ generated by Ag_1 is defeated according to Ag_2, Ag_2 selects one attack to defeat it: $\mathcal{R}_2^1 = \{(\forall x)(breathes(x) \wedge feathers(x) \rightarrow \neg m(x))\}$;
2. These attacks are sent to each other.
3. Agents update their theories:

- Due to the attacks recieved, Ag_1 updates its inductive theory by removing all the defeated arguments, and replacing them by new undefeated arguments, and generates: $T_1^1 = \{(\forall x)(hair(x) \rightarrow m(x))\}$.
- Analogously, Ag_2 updates its inductive theory by removing all the defeated arguments, and replacing them by new undefeated arguments, and generates: $T_2^1 = \{(\forall x)(milk(x) \rightarrow m(x))\}$.

4. These theories are sent to each other.
5. Agents update their states:
   - $\Theta_{1\rightarrow 2}^1 = \Theta_{1\rightarrow 2}^0 \cup \mathcal{R}_1^1 \cup T_1^1$; $\Theta_{2\rightarrow 1}^1 = \Theta_{2\rightarrow 1}^0 \cup \mathcal{R}_2^1 \cup T_2^1$;
   - $\mathcal{A}_1^1 = (IR(\Delta_1) \cup \Theta_{2\rightarrow 1}^1, \twoheadrightarrow), \mathcal{A}_1^2 = (IR(\Delta_2) \cup \Theta_{1\rightarrow 2}^1, \twoheadrightarrow)$

Round $t = 2$.

1. Agents should try now to generate attacks, but since the arguments sent in the previous round $\mathcal{R}_1^1$ and $\mathcal{R}_2^1$ are undefeated in the argumentation systems $\mathcal{A}_2^1$ and $\mathcal{A}_1^1$ respectively, no new attacks can be generated and the protocol ends.

As a result, both agents have reached inductive theories T_1^1 and T_2^1 that are consistent with the whole set of examples of both agents $\Delta_1 \cup \Delta_2$ (i.e. each theory has any counterexample neither in Δ_1 nor in Δ_2). Theorem 2 guarantees that

$$T^* = T_1^1 \cup T_2^1 = \{(\forall x)(hair(x) \rightarrow m(x)), (\forall x)(milk(x) \rightarrow m(x))\}$$

is a common and agreed argumentation-consistent inductive theory. Notice that this result is reached without exchanging any example, and exchanging a small amount of inducible rules.

8. Related Work

Peter Flach [1] introduced a logical analysis of induction, focusing on hypothesis generation. In Flach's analysis, induction was studied on the meta-level of consequence relations and focused on different properties that may be desirable for different kinds of induction. In this paper we cover both hypothesis generation and hypothesis selection, but we focus in a limited form of induction, namely inductive concept learning, extensively studied in

machine learning. A direct difference between Flach's work and the research presented in this paper is that we impose strong syntactical constraints on our inductive consequence relation (from sets of examples to rules), in order to focus on the specific machine learning problem of inductive concept learning, whereas the work of Peter Flach, no restrictions were applied, in order to study the soundness and completeness of sets of meta-level properties of inductive consequence relations. Appendix B offers an in-depth comparison of some properties of our consequence relation with some of Flach's meta-level properties.

A refinement of Flach's consistency-based confirmation using Hempel's direct confirmation was studied in [4]. The authors proposed that inductive generalization can be modeled as a deductive process given a *completion technique*, which captures inductive assumptions, such as "every unknown individual is similar to the known ones." The difference with our work is that, albeit restricted to the particular task of ICL, we propose a specific non-monotonic logic consequence relation, instead of resorting to a completion technique.

Related to the work of Flach is that of DelGrande [3], where he studied the algebra of hypotheses that can be formed by induction from sets of examples. In the same way as Flach, DelGrande limited his study to hypothesis generation, and considered that his model is a restriction with respect to the general problem of induction, where induction as such plays the limited role of proposing an initial set of hypotheses, which is later refined using deductive techniques.

Also related is the work of Datteri et al. [2], where induction (in machine learning) was understood as a deductive process; Dateri et al. modeled a typical process of a machine learning inductive algorithm in several steps, and provided a logical model for each step (that they call "deductive"). The final argument was that machine learning inductive algorithms are then "inductionless," as every step in the process is a logical inference. Our approach, a non-monotonic logical model of the whole process of an inductive algorithm, clarifies the nature of inductive concept learning: it is a form of defeasible (i.e. non-deductive) reasoning, similar (albeit not identical) to other forms of defeasible reasoning modeled by non-monotonic logic.

Concerning the integration of inductive reasoning with other forms of logical reasoning, Michalski [23], in his Inferential Theory of Learning, started a unified characterization of all forms of inference (deduction, analogy, induction, etc.) and defined *knowledge transmutation* operators. However, those

operators were only illustrated with examples, and never completely formalized. In this paper, we have taken on a smaller task: instead of trying to formalize all types of inference, we have focused on a very specific form of inference (inductive generalization), and, in this way, we have managed to completely characterize it in the form of a consequence relation.

Our approach to model multiagent induction is related to that of merging argumentation systems, which has been studied by Coste-Marquis et al. [24], where a group of agents, each one having a different argumentation framework (with potentially inconsistent attack relations) want to merge them. Coste-Marquis et al. proposed to do so by sharing all the arguments and then letting each agent construct a *partial argumentation system* where one argument attacks another when the majority of agents in the group that know both arguments consider there is an attack. After that, agents can merge their opinions on which arguments are defeated. Notice, however, that in our setting, since we are not dealing with an abstract argumentation framework and our arguments are actually logical formulas, all agents agree on the attack relation, and thus, we don't require such merging procedure.

Arguments and argumentation have been used in a few approaches of machine learning. For instance, arguments are used in the argument-based machine learning framework [25]; this approach did not employ an argumentation process, instead it assumed that arguments are given as part of the *input* of the inductive process, and are exploited by the inductive algorithm.

Argumentation has been used in the context of multiagent learning in [26]; however, this approach used argumentation and machine learning as black-boxes that are not integrated, while our logical model of inductive generalization allows for a deep integration of inductive reasoning and argumentation. Amgoud and Serrurier [27] proposed the use of argumentation as a framework to formalize the classification process, and in particular binary classification in the context of concept learning. The main difference between the work of Amgoud and Serrurier and ours is that they focus on classification, i.e. given an unclassified example, a set of examples and a set of hypotheses, find the classification of the new instance together with an explanation of why such classification is provided. Argumentation, in their framework, is used to determine which possible classifications (understood as arguments coming from examples or hypotheses) are acceptable, given all the other hypotheses and examples, and thus determine a classification for the new example. They also considered a preference relation on the set of hypothesis for guiding the search in the hypothesis space and to define the

attack relation between them. In contrast, in our work, we are interested on a logical modeling of the concept learning process itself: the process through which hypotheses (rules) are generated from a given set of examples. We also use a preference relation, but we used it to rank the induced rules and the set of inductive theories, rather than to define the attack relation. In our proposal, argumentation is only used as a communication framework when multiple agents are involved in the learning process.

Our previous work focused first on case-based learning from argumentation-based communication processes [28], where arguments in the form of both rules and cases were interchanged, but no inductive theory was reached: the agents used case-based learning plus argumentation to classify unknown examples. Later, as mentioned before, the A-MAIL framework was the first realization of an argumentation-based approach to multiagent induction [9]. The main difference between [9] and the work presented in this paper is that A-MAIL was a particular implementation that was experimentally validated to work, in the sense that agents achieved mutually consistent inductive definitions of a concept by exchanging arguments and attacks[3]. However, there was no formal proof, in [9], that achieving mutually consistent inductive definitions was always possible, as we have done in this paper. On the other hand, in this paper we focus on providing theoretical results that explain why an approach like A-MAIL may achieve coordinated induction using argumentation.

9. Conclusions and Future Work

This paper presents two main contributions, one being an inductive consequence relation in the framework of non-monotonic reasoning for inductive concept learning, and the other argumentation-consistent induction, integrating learning from examples by inductive generalization with learning from argumentation-based communication.

The standard model of non-monotonic reasoning could not be directly applied to our inductive consequence relation. We needed to relax and rein-

[3]Specifically, in [9] we focused on developing and evaluating an inductive algorithm that take into account argument attacks; this algorithm, called ABUI for argumentation-based bottom-up induction, performs a bottom up search in the space of generalizations to find an induced rule from examples such that is not defeated by the set of known arguments attacking previously induced rules.

terpret some of the properties of this model, taking into account that our inductive consequence relation is defined between two different sets of formulas (examples and rules). Specifically, Cautious Monotonicity and Cautious Right Weakening properties maintain the spirit of the standard model properties by reinterpreting them into a context in which we have two separate sets of formulae.

Furthermore, Proposition 2 presented six additional properties that characterize our inductive consequence relation which, as we have shown, are the properties specific to, and anticipated for, inductive concept learning.

The notion of inductive theory, introduced here, is a formalization of the intuitive notion of the output resulting from an ICL algorithm: a set of formulas that, as a whole, cover and explain all positive examples of the target concept. This notion allows us to deal with hypothesis selection modeled as preferences over inductive theories, modeling well established inductive biases such as parsimony and error margin maximization.

Moreover, the notion of inductive theory has allowed us, in the second part of this paper, to integrate the non-monotonic reasoning process of inductive generalization with another non-monotonic reasoning process, namely argumentation. Argumentation-consistent induction is the key notion in articulating inductive generalization with argumentation: the rules derived by induction are required to be acceptable inside the argumentation framework. Conceptually, the rules induced by an agent are learnt not only from examples but from the arguments that are the result of communicating with another agent.

Finally, argumentation-consistent induction allowed us to prove that a group of agents communicating their induced rules and performing argumentation would obtain the exact same set of inducible rules as a single agent knowing the examples known to all agents. Thus, learning directly from examples is equivalent (modulo inductive theory equivalence) to learning from communication from another agent that also learns from examples. In other words, for two agents or more, first communicating all their examples and then learning by induction is equivalent to first learning by induction individually and then communicating the generalizations they have learnt using argumentation.

In this paper we have centered our analysis on a setting where we assume no noise in the examples, and where we do not allow induced rules to have any counterexamples. ICL techniques usually accept generalizations that are not 100% consistent with the set of examples. Our future work will focus on

moving from a purely Boolean approach to a graded (or weighted) approach, where generalizations that are not 100% consistent with the examples can have a degree of acceptability. This broader framework would be closer to implemented systems such as A-MAIL [9] that accept induced rules with less than 100% consistency as long as they are above a given confidence threshold.

Acknowledgements. We are grateful to Peter Flach for helpful comments and suggestions on an earlier version of this manuscript and to Frances Esteva for valuable discussions on an earlier manuscript. Research partially funded by the projects Agreement Technologies (CSD2007-0022), ARINF (TIN2009-14704-C03-03), Next-CBR (TIN2009-13692-C03-01), LoMoReVI (FFI2008-03126-E/FILO), and by the grants 2009-SGR-1433 and 2009-SGR-1434 of the Generalitat de Catalunya.

Appendix A. Argumentation-consistent Induction for n Agents

The main theoretical result of this paper concerning inductive concept learning in multiagent systems is captured in Theorem 1. Such result states that learning directly from examples is equivalent to learning from communication from another agent that also learns from examples. In this appendix, we generalize this result for multiagent systems with more than two agents.

Theorem 3. (Argumentation-consistent Induction for n Agents) $\mathbf{U}(\bigcup_{i=1\ldots n} IR(\Delta_i)) = IR(\bigcup_{i=1\ldots n} \Delta_i)$.

Proof. Notice that by definition $\mathbf{U}(IR(\Delta)) = IR(\Delta)$; consequently, we have $AIR(\Delta, IR(\Delta)) = IR(\Delta)$.

First, we prove that $IR(\bigcup_{i=1\ldots n} \Delta_i) \subseteq \mathbf{U}(\bigcup_{i=1\ldots n} IR(\Delta_i))$. Let $r = \alpha \rightarrow C$ be such that $r \in IR(\bigcup_{i=1\ldots n} \Delta_i)$, then r covers a positive example of $\bigcup_{i=1\ldots n} \Delta_i$ and does not cover any negative example of $\bigcup_{i=1\ldots n} \Delta_i$. W.l.o.g., assume the covered positive example is from Δ_k. Then $r \in IR(\Delta_k)$. Suppose there exists a rule $r' = \beta \rightarrow \neg C \in \bigcup_{i=1\ldots n} IR(\Delta_i)$ such that $r' \twoheadrightarrow r$, i.e. such that $K \vdash \beta \rightarrow \alpha$. It is clear that $r' \notin IR(\Delta_k)$, hence assume $r' \in IR(\Delta_j)$ for some Δ_j, such that $j \neq k$. This means r' covers a negative example $\delta^- \in \Delta_j$, but if r' covers it, r must cover δ^- as well, contradiction.

Second, we prove that $IR(\bigcup_{i=1\ldots n} \Delta_i) \supseteq \mathbf{U}(\bigcup_{i=1\ldots n} IR(\Delta_i))$. Let $r = \alpha \rightarrow C$ be such that $r \in \mathbf{U}(\bigcup_{i=1\ldots n} IR(\Delta_i))$. W.l.o.g., assume $r \in IR(\Delta_k)$. Then r covers a positive example of Δ_k and does not cover any negative example of Δ_k. Assume also, looking for a contradiction, that $r \notin IR(\bigcup_{i=1\ldots n} \Delta_i)$. Since we have assumed that $r \in IR(\Delta_k)$, this means that r covers a negative example of some Δ_j. This negative example can be specialized to a rule $r' = \beta \rightarrow \neg C \in IR(\Delta_j)$ such that $K \vdash \beta \rightarrow \alpha$. Since r' is the specialization of an example in Δ_j and $\bigcup_{i=1\ldots n} \Delta_i$ is

consistent, the rule r' is undefeated. Consequently, $r \notin \mathbf{U}(\bigcup_{i=1\ldots n} IR(\Delta_i))$, which contradicts our original assumption. Therefore we can conclude $IR(\bigcup_{i=1\ldots n} \Delta_i) \supseteq \mathbf{U}(\bigcup_{i=1\ldots n} IR(\Delta_i))$. □

Appendix B. Flach's general approach to inductive consequence relations

In their seminal paper [7] Kraus, Lehmann and Magidor (KLM) study "general patterns of non-monotonic reasoning and try to isolate properties that could help us map the field of non-monotonic reasoning by reference to positive properties". Following Gabbay [14], KLM focus their study at the level of consequence relations and choose a Gentzen-style notation of axiom schemata and inference rules to express structural properties of a consequence relation that could adequately represent a non-monotonic logic.

Based on the KLM framework, Flach [1, 5] studies the process of inductive hypothesis formation from two perspectives: finding general rules that explain given specific evidence (*explanatory induction*), and finding general rules that are confirmed by the evidence (*confirmatory induction*). Both forms of hypothesis formation are axiomatised also at the level of consequence relations, providing a set of rationality postulates for various forms of induction.

For Flach, an inductive consequence relation $\mid\!\sim$ is a set of pairs of formulae, $\alpha \mid\!\sim \beta$ meaning that "β is a possible inductive hypothesis given evidence α". Inductive consequence relations are intended to model the behaviour of inductive agents. Flach does not fix a particular definition of $\mid\!\sim$, he studies rationality postulates limiting different possible definitions. He starts with a set of general principles for induction and then presents specific sets of principles for each type of induction (explanatory and confirmatory).

Since our consequence relation $\mid\!\sim_K$ is defined between two different sets of formulas (examples and rules), most of these properties do not directly apply to our setting. Nevertheless, it is interesting to check whether the Flach's general principles (listed below) underlying these properties hold for $\mid\!\sim_K$.

1. Verification (a predicted observation verifies the hypothesis)

$$\frac{\vdash \alpha \wedge \beta \rightarrow \gamma,\ \alpha \mid\!\sim \beta}{\alpha \wedge \gamma \mid\!\sim \beta}$$

2. Falsification (an observation, the negation of which was predicted, falsifies the hypothesis)

$$\frac{\vdash \alpha \wedge \beta \rightarrow \gamma,\ \alpha \mid\!\sim \beta}{\alpha \wedge \neg\gamma \not\mid\!\sim \beta}$$

3. Left Logical Equivalence (the logical form of the evidence is immaterial)

$$\frac{\vdash \alpha \leftrightarrow \beta,\ \alpha \mathrel{|\!\sim} \gamma}{\beta \mathrel{|\!\sim} \gamma}$$

4. Right Logical Equivalence (the logical form of the hypothesis is immaterial)

$$\frac{\vdash \beta \leftrightarrow \gamma,\ \alpha \mathrel{|\!\sim} \beta}{\alpha \mathrel{|\!\sim} \gamma}$$

5. Left Reflexivity (evidence allowing some hypothesis is admissible)

$$\frac{\alpha \mathrel{|\!\sim} \beta}{\alpha \mathrel{|\!\sim} \alpha}$$

6. Right Reflexivity (any hypothesis allowed by some evidence is admissible)

$$\frac{\alpha \mathrel{|\!\sim} \beta}{\beta \mathrel{|\!\sim} \beta}$$

7. Right Extension (any hypothesis can be extended with a prediction)

$$\frac{\vdash \alpha \wedge \beta \rightarrow \gamma,\ \alpha \mathrel{|\!\sim} \beta}{\alpha \mathrel{|\!\sim} \beta \wedge \gamma}$$

In order to check the validity of these general principles in our ICL framework, we need first to set out how to interpret Flach's consequence relation $\mathrel{|\!\sim}$ in terms of our inductive consequence relation $\mathrel{|\!\sim}_K$, taking into account our restricted language of rules and examples. Indeed, in an expression $\alpha \mathrel{|\!\sim} \beta$ we interpret the evidence α as a set of (both positive and negative) examples Δ for a concept C, and the hypothesis β as a rule $(\forall x)(\varphi(x) \rightarrow C(x))$.

In this setting, we provide the following justifications and propose an adapted form of these principles to our framework:

1. Verification: interpreting a predicted observation as a new positive example $\gamma(a) \wedge C(a)$ already covered by an induced rule $\beta \rightarrow C$ from a set of examples Δ, the principle holds by property 3 of Proposition 2 (Positive monotonicity).

$$\frac{K \vdash \gamma \rightarrow \beta,\ \Delta \mathrel{|\!\sim}_K \beta \rightarrow C}{\Delta \cup \{\gamma(a) \wedge C(a)\} \mathrel{|\!\sim}_K \beta \rightarrow C}$$

2. Falsification: with the same interpretation as in the previous item, a new negative example $\gamma(a) \wedge \neg C(a)$ is not covered by an induced rule $\beta \rightarrow C$ from Δ when $\gamma(a) \wedge C(a)$ was already covered by $\beta \rightarrow C$. That is,

$$\frac{K \vdash \gamma \rightarrow \beta,\ \Delta \mathrel{|\!\sim}_K \beta \rightarrow C}{\Delta \cup \{\gamma(a) \wedge \neg C(a)\} \not\mathrel{|\!\sim}_K \beta \rightarrow C}$$

   This follows by the very definition of the inductive consequence relation $\mathrel{|\!\sim}_K$.

3. Left Logical Equivalence: if $\Delta \mid\!\sim_K \alpha \to C$ and $\Delta \equiv_K \Delta'$, then $\Delta' \mid\!\sim_K \alpha \to C$. This directly follows from property 2 in Proposition 1.

$$\frac{\Delta \mid\!\sim_K \alpha \to C,\ \Delta \equiv_K \Delta'}{\Delta' \mid\!\sim_K \alpha \to C}$$

4. Right Logical Equivalence: if $K \vdash \beta \leftrightarrow \alpha$ and $\Delta \mid\!\sim_K \alpha \to C$, then $\Delta \mid\!\sim_K \beta \to C$. This directly follows from property 3 in Proposition 1.

$$\frac{K \vdash \beta \leftrightarrow \alpha,\ \Delta \mid\!\sim_K \alpha \to C}{\Delta \mid\!\sim_K \beta \to C}$$

5. Left Reflexivity: if $\Delta \mid\!\sim_K \beta \to C$ for some rule $\beta \to C$, this means that Δ is consistent, and hence, for every $\alpha(a) \wedge C(a) \in \Delta$, we have $\{\alpha(a) \wedge C(a)\} \mid\!\sim_K \alpha \to C$. This follows from property 1 of Proposition 1.

$$\frac{\Delta \mid\!\sim_K \beta \to C,\ \alpha(a) \wedge C(a) \in \Delta}{\{\alpha(a) \wedge C(a)\} \mid\!\sim_K \alpha \to C}$$

6. Right Reflexivity: if $\Delta \mid\!\sim_K \beta \to C$ for some set of examples Δ, for every example $\beta(a) \wedge C(a)$, we have $\{\beta(a) \wedge C(a)\} \mid\!\sim_K \beta \to C$. This follows from property 1 of Proposition 1.

$$\frac{\Delta \mid\!\sim_K \beta \to C}{\{\beta(a) \wedge C(a)\} \mid\!\sim_K \beta \to C}$$

7. Right Extension: if $\Delta \mid\!\sim_K \beta \to C$, by definition of covering, there must exist a positive example $\alpha(a) \wedge C(a) \in \Delta$ such that $\vdash \alpha \to \beta$. Assuming $\vdash \alpha \wedge \beta \to \gamma$, we have that $\vdash \alpha \to \beta \wedge \gamma$. Since Δ is assumed to be consistent, $\beta \wedge \gamma$ cannot cover any negative example, and consequently $\Delta \mid\!\sim_K \beta \wedge \gamma \to C$.

$$\frac{\Delta \mid\!\sim_K \beta \to C, \{\alpha(a) \wedge C(a)\} \mid\!\sim_K \beta \to C,\ \vdash \alpha \wedge \beta \to \gamma}{\Delta \mid\!\sim_K \beta \wedge \gamma \to C}$$

References

[1] P. A. Flach, Logical characterisations of inductive learning, in: Handbook of defeasible reasoning and uncertainty management systems: Volume 4 Abductive reasoning and learning, Kluwer Academic Publishers, Norwell, MA, USA, 2000, pp. 155–196.

[2] E. Datteri, H. Hosni, G. Tamburrini, An inductionless, default based account of machine learning, in: L. Magnani (Ed.), Model-Based Reasoning in Science and Engineering, College Publications, 2006, pp. 379–399.

[3] J. P. Delgrande, A formal approach to learning from examples, International Journal of Man-Machine Studies 26 (1987) 123–141.

[4] N. Lachiche, P. Marquis, A model for generalization based on confirmatory induction, in: Proceedings of the 9th European Conference on Machine Learning (ECML), 1997, pp. 154–161.

[5] P. A. Flach, Rationality postulates for induction, in: Proc. 6th Int. Conf. on Theoretical Aspects of Rationality and Knowledge, Yoav Shoham, Morgan Kaufmann, 1996, pp. 267–281.

[6] R. Pino-Pérez, C. Uzcátegui, Jumping to explanations versus jumping to conclusions, Artificial Intelligence 111 (1-2) (1999) 131 – 169. doi:10.1016/S0004-3702(99)00038-7.

[7] S. Kraus, D. Lehmann, M. Magidor, Nonmonotonic reasoning, preferential models and cumulative logics, Artificial Intelligence 44 (1-2) (1990) 167–207.

[8] S. Ontañón, P. Dellunde, L. Godo, E. Plaza, Towards a logical model of induction from examples and communication, in: Proceedings of the 13th International Conference of the Catalan Association for Artificial Intelligence (CCIA), IOS Press, 2010, pp. 259–268.

[9] S. Ontañón, E. Plaza, Multiagent inductive learning: an argumentation-based approach, in: Procedings of the Twenty Seventh International Conference on Machine Learning (ICML), Omnipress, 2010, pp. 839–846.

[10] T. Mitchell, Machine Learning, McGraw-Hill, 1997.

[11] N. Lavrač, S. Džeroski, Inductive Logic Programming. Techniques and Applications, Ellis Horwood, 1994.

[12] J. R. Quinlan, Learning logical definitions from relations, Machine Learning 5 (1990) 239–266.

[13] H. Enderton, A mathematical introduction to logic. Second Edition, Harcourt/Academic Press, 2001.

[14] D. Gabbay, Theoretical foundations for non-monotonic reasoning in expert systems, in: Logics and models of concurrent systems, Springer-Verlag, New York, NY, USA, 1985, pp. 439–457.

[15] A. Avron, Simple consequence relations, Information and Computation 92 (1) (1991) 105–140.

[16] V. Vapnik, Estimation of Dependences Based on Empirical Data (Information Science and Statistics), Springer, 2006.

[17] C. Lafage, J. Lang, Propositional distances and preference representation, in: S. Benferhat, P. Besnard (Eds.), Proceedings of ECSQARU 2001, Vol. 2143 of LNAI, Springer, 2001, pp. 48–59.

[18] S.-H. Nienhuys-Cheng, Distance between herbrand interpretations: A measure for approximations to a target concept, in: Proceedings of the 7th International Workshop on Inductive Logic Programming, Springer-Verlag, London, UK, 1997, pp. 213–226.

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

[20] C. Chesñevar, G. Simari, A lattice-based approach to computing warranted beliefs in skeptical argumentation frameworks, in: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI), 2007, pp. 280–285.

[21] N. Rotstein, M. Moguillansky, G. Simari, Dialectical abstract argumentation: a characterization of the marking criterion, in: Proceedings of the Twenty First International Joint Conference on Artificial Intelligence (IJCAI), 2009, pp. 898–903.

[22] H. Prakken, Coherence and flexibility in dialogue games for argumentation, Journal of Logic and Computation 15 (2005) 1009–1040.

[23] R. Michalski, Inferential theory of learning as a conceptual basis for multistrategy learning, Machine Learning 11 (2–3) (1993) 111–152.

[24] S. Coste-Marquis, C. Devred, S. Konieczny, M.-C. Lagasquie-Schiex, P. Marquis, On the merging of Dung's argumentation systems, Artificial Intelligence 171 (2007) 730–753.

[25] M. Možina, J. Zabkar, I. Bratko, Argument based machine learning, Artificial Intelligence 171 (10-15) (2007) 922–937.

[26] M. Wardeh, T. J. M. Bench-Capon, F. Coenen, Padua: a protocol for argumentation dialogue using association rules, Artificial Intelligence in Law 17 (3) (2009) 183–215.

[27] L. Amgoud, M. Serrurier, Arguing and explaining classifications, in: Proceedings of the Sixth International Conference on Agents and Multiagent Systems (AAMAS), ACM, New York, NY, USA, 2007, pp. 1–7.

[28] S. Ontañón, E. Plaza, Learning and joint deliberation through argumentation in multiagent systems, in: Proceedings of the Sixth International Conference on Agents and Multiagent Systems (AAMAS), 2007, pp. 971–978.