# In Search of a Consistent World View: Induction as Extension

Panu A. Kalliokoski

March 28, 2008

**Abstract**

In this paper, I develop an account of classificatory induction that gives, for any observation report, a theory that contains all inductive consequences of the observation report. Such a theory is called the *maximal plausible generalisation* of the observation report, and it is shown to be consistent, unique for each observation report, and to capture nicely the intuitive notion of inductive consequence by being the most informative generalisation that is still plausible.

In the course of defining the maximal plausible generalisation, I present the conditions of entailment, consistency and plausibility, which any relation of inductive consequence should observe. These conditions also hold for the maximal plausible generalisation.

Keywords: classificatory induction, confirmation, consistency, constraint, generalisation, induction, induction, inductive consequence, inductive logic, logic, plausibility.

# Contents


**1 The vague concept of induction** **5**
- 1.1 Induction as extension . . . 5
- 1.2 Structure of this thesis . . . 6

**2 Conventions and terminology** **7**
- 2.1 Terminology . . . 7
- 2.2 Notational conventions . . . 8
- 2.3 Logical language . . . 9

**3 Hempel's account of induction** **10**
- 3.1 Hempel's criteria for confirmation . . . 10
- 3.2 Hempel's proposal for the confirmation relation . . . 12
- 3.3 Shortcomings in Hempel's confirmation relation . . . 14

**4 What is induction not?** **17**
- 4.1 Mathematical induction . . . 17
- 4.2 Conceptualisation . . . 17
- 4.3 Validation of hypotheses . . . 18
- 4.4 Abduction . . . 18
- 4.5 Setting of induction . . . 18
- 4.6 Classification of objects . . . 19
- 4.7 Generalisation of clauses . . . 20
- 4.8 Calculation of probabilities . . . 20

**5 Theoretical background** **22**
- 5.1 Generality of concepts and generality of statements . . . 22
- 5.2 Version spaces . . . 24
- 5.3 Conjunctive normal form . . . 26
  - 5.3.1 Rewriting formulae into CNF . . . 27
  - 5.3.2 Significance of CNF . . . 29
- 5.4 $\theta$-subsumption . . . 30
- 5.5 Two types of induction . . . 32

**6 Version spaces applied to logic** **35**
- 6.1 Version spaces of truth . . . 35
- 6.2 Generalisation of truth as a version space problem . . . 35
- 6.3 Logical version spaces and Hempel's conditions . . . 38
- 6.4 Minimal and maximal generalisations . . . 39
- 6.5 Choosing the correct generalisation . . . 40
- 6.6 Search in the generalisation space . . . 41

**7 Plausible clauses** **43**
- 7.1 Plausibility: why and how? . . . 43
- 7.2 Maximal plausible generalisation . . . 45
- 7.3 Conditions for the preference relation . . . 46

**8 Symmetry** **50**
8.1 Implausibility of irrelevant clauses . . . . . 51
8.2 Implausibility of irrelevant weakenings of falsified clauses . . . . . 52
8.3 Conclusion . . . . . 54

**9 Total strength ordering of clauses** **55**
9.1 Definition of the preference relation . . . . . 56
9.2 Properties of the preference relation . . . . . 58
9.3 Examples of plausible clauses . . . . . 59
9.4 Problems with the preference relation . . . . . 61

**10 Possibilities of the maximal plausible generalisation** **62**
10.1 Conclusions . . . . . 62
10.2 Algorithmic induction . . . . . 62
10.3 Developments in the preference relation . . . . . 63
10.4 Conceptualisation . . . . . 64

# Preface

This thesis is the outgrowth of a simple idea about human reasoning I had in fall 2006. This idea was based on the observation that people, upon encountering something surprising (something that does not fit their current conception of the world), tend to search for explanatory conditions in the context of the surprising observation. Could this process not be formalised? Attempts to formalise this account of human reasoning eventually produced my bachelor's thesis [Kal07], which is in practice a description of algorithmic constraint induction. In the course of developing this theory, I also implemented it as a scheme program.

However, working with concrete examples of induction quickly brought into my attention many cases where the seemingly plausible conclusions of inductive inference were inconsistent. Of course, it is a basic tenet of induction that any hypothesis that is inconsistent with our experience is rejected; but in many cases, two (or more) hypotheses are just *jointly* inconsistent with our experience and it is not easy to tell which one is the "culprit". This was problematic, since people certainly strive for a consistent world view — any theory of induction that produced mutually inconsistent results would hardly be a satisfactory description of the inductive process.

I had also come across Hempel's account of confirmation [Hem43], and noticed that Hempel had set the exactly same condition for confirmed hypotheses: that they should all be jointly consistent with our experience. However, I found Hempel's concrete proposal for confirmation relation highly unsatisfactory, and set out to search refinements to Hempel's confirmation relation in later research. Quite surprisingly, there were none.

So, this thesis is an attempt to find the most general relation of inductive consequence that will still only produce mutually consistent conclusions, and to give precise formalisation for that relation of inductive consequence. The result is far from flawless, but still a big improvement over Hempel's account of confirmation, and can be used as a starting point for further refinements.

I would like to thank my friends, who have provided valuable feedback in several discussions and have often had the patience of commenting on my ideas even when the purpose of them has been obscure. Especially, the discussions with Miikka Silfverberg and Lauri Alanko have been helpful. I would also like to thank the people who have invested their time in proofreading the thesis. Naturally, I am also grateful to my instructor Ahti-Veikko Pietarinen for taking the time to go over my work and suggest corrections.

# 1 The vague concept of induction

Given that induction is mentioned on almost every introductory course on logic and it is central in the scientific method, the concept of induction remains astoundingly vaguely defined. Induction is usually described as *generalisation from several instances*, and contrasted with deduction: while deduction produces true conclusions from true premisses, induction only produces plausible conclusions (from true premisses). However, induction is usually pursued no further. It is simply something that generalises from instances and is not deduction.

It is hardly a philosophical attitude to leave central concepts ill-defined; after all, defining concepts clearly is a very important part of all philosophical activity. There are crystal clear criteria for what is and what is not a deductive inference; why are there no similar criteria for inductive inference? There are algorithms to deterministically produce all deductive consequences of a set of clauses; why are there no algorithms to produce all inductive consequences in the same way?

Part of the problem of defining induction is the vast scope of induction. Induction supposedly covers a broad area of human thinking: everything that extracts any kind of rules or lawlike results from experience. It is very difficult to form a definition that applies to everything that people would intuitively call "induction". Many formalisations of induction (such as the ones mentioned below) seem to apply to some cases of inductive thinking, but are not applicable to other cases.

However, much of the difficulty is also certainly due to the unclear initial intuitions about induction. Induction has usually been explained by examples, and the choice of particular examples has tended to guide the work in the area. Thus, those interested in machine learning (e.g. [Ren86]) have mostly studied the problem of *classification*, which means categorisation of instances (examples) into several classes that are either given or invented in the process; logic programmers (e.g. [Mug92a]) have been interested in the synthesis of logic programs, which in practice means the search for a concise definition of the sufficient and necessary conditions of one or several predicates; and philosophers of science (e.g. [Car50]) have extensively studied the process of *verification* (or *confirmation*) and *falsification* of hypotheses. These approaches, while possibly reconcilable, address induction from different angles and within different frameworks. It is also possible that some of the approaches depict totally different kinds of induction.

## 1.1 Induction as extension

The general approach in this thesis is to consider induction as a broadening, or extension, of our conception of the world. Our knowledge of the world is necessarily incomplete, that is, the world is underdetermined by our experience. Induction is the process of extending this world view by statements that are plausible. A statement is plausible if and only if it is *based on* and consistent with our knowledge. This thesis attempts to carefully define the circumstances where a proposition is based on some piece of knowledge.

Induction has a double goal: it should produce as strong claims as possible while only producing claims that are plausible. The first goal is called the

*informativeness* of the induced theory. Scientific theories are expected to be informative: the more situations a theory denies, the more informative it is. The second goal is called the *plausibility* of the theory: a theory is plausible if it is possible in view of what we know. As the informativeness of a theory increases, it becomes more difficult for the theory to remain plausible, because there are more potential situations that would disprove the theory.

The ultimate goal of inductive inference is to find *all* propositions that are plausible. This will give us a theory that is as informative as possible without sacrificing plausibility. I will call such a theory the *maximal plausible generalisation* of our experience.

## 1.2 Structure of this thesis

This thesis consists of two parts. The first part (sections 3–5) is dedicated to defining what is meant by induction in this thesis and excluding different kinds of activities that are also called induction, in other sources. Some of these activities are more obviously different from the conception of induction in this thesis, while some differences are subtle enough to warrant special attention. In the first part, we examine the account of confirmatory induction given by Hempel as a brave but unsatisfactory attempt at defining induction.

The second part (sections 6–9) is an application of modern machinery such as version spaces and $\theta$-subsumption to the problem of induction. In this part, I study the meaning of generalisation in the setting of logic, and develop conditions of plausibility that pertain to inductive generalisations that are intuitively "good". Finally, I present a proposal that meets the requirements of the conditions.

# 2 Conventions and terminology

Some conventions are used throughout this thesis.

## 2.1 Terminology

Most of this thesis deals with formulae in *first-order predicate logic* without identity (**FOL**). A well-formed formula is called a *proposition* if it contains no free (unbound) variables; otherwise, it is called a *matrix* of the variables that are free in it. A proposition is also called *statement* or *claim* to emphasise its semantic side; and *sentence* to emphasise its syntactic side. A proposition is *atomic* when it does not contain *connectives*: conjunctions, disjunctions, implications or equivalences. Note that an atomic proposition may be negated. A set of propositions is also called a *theory*.

When propositions are in conjunctive normal form (CNF, see section 5.3), a *clause* is a sentence that is a universally quantified disjunction of atomic matrices. The atomic matrices in a clause are called the *literals* of the clause. An empty clause is a clause with no literals. Such a clause represents the contradiction (the strongest claim) and is written as $\bot$. An empty theory is a theory with no clauses. Since the theory does not claim anything, it represents the weakest claim and is written as $\top$.

For every binary relation $R$, there is a *converse relation* $R^c$ defined by:

$$\forall x \forall y (R^c xy \equiv Ryx) \tag{1}$$

Different kinds of *generality* are discussed extensively in this thesis. A concept (or relation) is called *more general* than another if its instances (or tuples of individuals satisfying it) include all the instances of the other one. Moreover, a concept is *strictly more general* than another if it is more general than the other concept but the converse does not hold. The converse relations are *more specific* and *strictly more specific*, respectively. On the other hand, a statement is more general than another if it logically entails the other statement. Strict generality, specificity and strict specificity are then defined equivalently for statements. For a detailed discussion about this terminology, see section 5.1.

A sentence $S$ is called *ground* if and only if it contains no variables. This means that the terms of $S$ are elements of a Herbrand universe. A sentence $S$ is a *ground instance* of another sentence $S'$ if and only if $S$ is ground and can be obtained by substituting variables in $S'$ with other terms.

A set of propositions $K$ is *consistent with* or *logically compatible with* another set of propositions $O$ if their combination $K \cup O$ is consistent, and *inconsistent with* or *contradictory with* $O$ if $K \cup O$ is inconsistent.

A set of propositions $H$ is *maximally consistent* if and only if all proper supersets of $H$ are inconsistent but $H$ is consistent. Correspondingly, $H$ is *minimally inconsistent* if and only if $H$ is inconsistent but all proper subsets of $H$ are consistent. Moreover, $H$ is *minimally inconsistent with* another set of propositions $O$ if and only if $H \cup O$ is inconsistent but for all proper subsets $H' \subset H$, the theory $H' \cup O$ is consistent. This is not the same as minimal inconsistency of $H \cup O$, because $O$ may contain clauses that do not affect the consistency.

## 2.2 Notational conventions

The formulae in this thesis refer to many kinds of objects. In the formulae, the letters $S$, $T$, $U$ are used to refer to sentences (propositions) or matrices, while $p$, $q$ and $r$ are used to only refer to propositions. For clauses of CNF, the capital letters $C$ and $D$ are used; these clauses are represented by sets of literals, so that $L \in C$ means that $L$ is a literal (i.e. disjunct) of $C$. In this notation, clauses are implicitly taken to be universally quantified with respect to all their variables.

The capital letters $P$, $Q$, $R$ are used to denote predicate symbols. Other capital letters such as $E$, $H$, $K$ and $O$ are used for sets of propositions, and capital Greek letters $\Gamma$ and $\Delta$ are used for sets of propositions that are generalisations of other sets of propositions. All letters may have subscripts or primes to extend the vocabulary.

In propositions and matrices, names of individuals are written as $a$, $b$, $c$, etc., and names of variables as $x$, $y$, $z$. Occasionally we also need *metavariables* (variables that may have as their value the name of another variable); the lowercase Greek letters $\alpha$ and $\beta$ are used for names of metavariables. Arbitrary terms are referred to by lowercase letters $t$ and $u$.

In the **FOL** object language, the letters $f$, $g$, $h$ are used to denote function symbols, but occasionally more descriptive names are used, such as Stinky for individuals, ContainerOf for functions and Cow for predicates. In the metalanguage, functions are usually written capitalised, such as $D(\dots)$ or $Sk(\dots)$. For both predicates and functions, the arguments follow the predicate or function symbol, parenthesised and separated with commas, as in $P(a, b, c)$ or $f(b_1, g(b_2))$. However, for predicates the shorter notation with simple juxtaposition is sometimes used for brevity, as in $Pabc$. Most binary relations are written in infix form: $S \in O$ means $\in (S, O)$ ($S$ is a member of the set $O$), $O \rightsquigarrow \Gamma$ means $\rightsquigarrow (O, \Gamma)$ etc. See the next section for the precedence rules of infix symbols.

*Substitution* of terms in a sentence or matrix is written in postfix form as $S[t/u]$, which means $S$ with occurrences of the term $t$ replaced by another term $u$. Substitutions may naturally be composed as in $S[t_1/u_1][t_2/u_2]$; such a substitution of multiple terms is denoted by the lowercase Greek letters $\sigma$ and $\theta$, also written postfix as in $S\sigma$.

Various symbols have the following readings:

| symbol | reading |
|---|---|
| $=$ | is |
| $\equiv$ | if and only if |
| $\rightarrow$ | implies |
| $\in$ | belongs to (the set) |
| $\models$ | entails |
| $\longmapsto$ | rewrites to |
| $\mid\!\sim$ | confirms |
| $\rightsquigarrow$ | generalises to |
| $\supseteq$ | is more general than / is a superset of |
| $\supset$ | is strictly more general than / is a proper superset of |
| $\preceq_\theta$ | $\theta$-subsumes |
| $\geq_{\mathrm{pl}}$ | is preferred to |

In addition, we use the expression $\mathrm{Mod}\,\Gamma$ to denote that the set of propositions $\Gamma$ is consistent (has a model).

## 2.3 Logical language

This thesis uses **FOL** for two purposes: as an object language, to describe the propositions, predicates, functions and terms that we refer to; and as a metalanguage, to reason about these propositions, predicates, functions and terms, and sets thereof. The same language is used for both, except that the metalanguage is much richer with relations such as $\in$ ("belongs to"). This means that sometimes parentheses must be used to denote whether a given connective is to be understood to belong to the object language or the metalanguage; for instance, the proposition $p \rightarrow q \in \Gamma$ means that whenever proposition $p$ is true, proposition $q$ belongs to the set of propositions $\Gamma$, whereas the proposition $(p \rightarrow q) \in \Gamma$ means that the proposition $p \rightarrow q$ belongs to the set of propositions $\Gamma$. In the first case, the implication is a part of the metalanguage, and in the second case, of the object language.

The notational conventions of **FOL** are as follows. Logical negation, conjunction, disjunction, implication and equivalence are denoted by the signs $\neg$, $\wedge$, $\vee$, $\rightarrow$ and $\equiv$, respectively; of the connectives, negation binds most tightly, followed by conjunction and disjunction, followed by implication and equivalence. Whenever the structure of an expression must be disambiguated, parentheses are used.

Universal and existential quantification are denoted by the signs $\forall$ and $\exists$, respectively. The sign is followed by a variable and a matrix of that variable, and quantification binds on the same level as negation. Quantifiers and negation naturally associate to the right since they are prefix operators.

The metalanguage has the relation $\in$ (set membership) which binds more tightly than the logical connectives, and the relations $\mathrel{|\!\sim}$, $\models$, $=$, $\preceq_\theta$ and $\geq_{\mathrm{pl}}$ which bind less tightly than the logical connectives. Set comprehensions are also used: $\{F(x) : P(x)\}$ means the set of $F(x)$ for all $x$ for which $P(x)$ is true.

The relation $\models$ (entails) is overloaded with respect to its first operand. In the conventional way, $\Gamma \models p$ means that the proposition $p$ is true in every model of the set of propositions $\Gamma$, that is, $\Gamma$ logically entails $p$. For a single proposition $q$, $q \models p$ means the same as $\{q\} \models p$.

In some places, the inference rule of *resolution* is referred to. The rule is as follows:

$$\frac{p \vee r \qquad \neg r \vee q}{p \vee q} \tag{2}$$

Together with substitution of terms, resolution is a *refutation complete* inference rule. This means that every proposition $p$ that is entailed by a theory $H$ can be proved by resolution by showing that $\neg p$ is inconsistent with $H$. In the application of resolution, it is said that $p \vee r$ is *resolved against* $\neg r \vee q$. The result $p \vee q$ is called the *resolution* of the original clauses, and the literal $r$ is called the *resolvent*.

# 3 Hempel's account of induction

To give a first impression of the kind of induction this thesis describes, let us investigate Hempel's early work on confirmation relations [Hem43, Hem45]. This work was later criticised (and appraised) mostly for its account on what confirmation is, rather than Hempel's concrete proposal for a confirmation relation. Eventually, this discussion evolved to the probabilistic approach to confirmation, which is no longer applicable to Hempel's concrete confirmation relation. However, I hold the view that Hempel's view of confirmation, which is *classificatory* instead of *quantitative*, is far less faulty than Hempel's proposal for the implementation of it.

Hempel's discussion of confirmation is one of the most interesting and ambitious accounts of induction. Confirmation is a relation between evidence and hypothesis: some evidence is said to confirm a hypothesis if the evidence provides support — not necessarily conclusive — that the hypothesis might be true. We use the symbol $\mid\!\sim$ to denote such a *confirmation relation*; $E \mid\!\sim H$ means "evidence $E$ confirms hypothesis $H$". Evidence is represented by an *observation report*, which is a logical description (i.e. conceptualisation) of an observation.

Confirmation is important for induction, because any confirmation relation can be seen as a definition of inductive consequence. Indeed, $E \mid\!\sim H$ can also be read "$H$ is an inductive consequence of $E$", and the inductive closure of $E$ is given by $\{H : E \mid\!\sim H\}$.

## 3.1 Hempel's criteria for confirmation

Hempel defines some necessary conditions that any confirmation relation should satisfy. The article motivates the conditions by appealing to intuition, and Hempel admits that there is no kind of *proof* for them; the conditions are just an attempt to capture the properties of the intuitive notion of confirmation.[1]

**Condition 3.1** (Entailment condition)**.** Any sentence which is entailed by an observation report is confirmed by it.

$$(E \models H) \rightarrow (E \mid\!\sim H) \tag{3}$$

**Condition 3.2** (Consequence condition)**.** If an observation report confirms every one of a class $K$ of sentences, then it also confirms any sentence which is a logical consequence of $K$.

$$(\forall S(S \in K \rightarrow (E \mid\!\sim S))) \wedge (K \models H) \rightarrow (E \mid\!\sim H) \tag{4}$$

**Condition 3.3** (Consistency condition)**.** Every logically consistent observation report is logically compatible with the class of all the hypotheses which it confirms.

$$\operatorname{Mod} E \rightarrow \operatorname{Mod}(E \cup \{H : E \mid\!\sim H\}) \tag{5}$$

[1] Hempel also considers a fourth condition, the converse consequence condition, but rejects it right away: no confirmation relation can satisfy it together with the consequence condition and the consistency condition.

There are several things to notice about the conditions. Firstly, as Hempel elaborately explains, this view of the confirmation relation does not take into account different *degrees* of confirmation. An observation report that entails the hypothesis confirms it certainly (Hempel calls this *conclusive confirmation*), while inductive inference also admits confirmation of a weaker kind: tentative confirmation. Hempel does not stipulate about what kinds of confirmation there are; he believes that it is of a more pragmatic character to assess hypotheses with various differently confirming and disconfirming pieces of evidence, that to give a qualitative account whether a given piece of evidence confirms, disconfirms, or is neutral with respect to a hypothesis [Hem45, 114–115].

Secondly, the confirmation relation is heavily underdetermined by the conditions. Hempel mentions, as an example, that the entailment relation satisfies the criteria. Indeed, we notice that the conditions in no way suggest that observation reports might sometimes confirm hypotheses that make stronger claims than the observation reports. Hempel, however, clearly presupposes this, because his own suggestion for the confirmation relation allows observation reports about individuals to confirm hypotheses that have universal quantification. Hempel justifies this by explaining that in addition to the explicit conditions given above, the confirmation relation should also match our *intuitions* about scientific confirmation. These intuitions, however, are left formally rather unspecified. They are conveyed by Hempel's own proposal for the confirmation relation, which we shall soon study.

So, it should now be obvious that Hempel's conditions allow a plethora of possible confirmation relations, some more general and some more specific.[2] It is probably quite possible to find very counterintuitive confirmation relations that still manage to fulfill Hempel's conditions.

However, conditions 3.1 and 3.2 define nontrivial minimal requirements for the confirmation relation, and condition 3.3 defines a nontrivial maximal requirement for the confirmation relation. By this I mean that confirmation relations cannot be so general as to confirm mutually inconsistent sentences, and not so specific as to fail to confirm the logical consequences of a sentence. Consequently, both the most specific admissible confirmation relations and the most general admissible confirmation relations are nontrivial relations.

It is quite easy to see that the logical entailment relation $\models$ is the most specific confirmation relation possible in the sense that every confirmation relation $\mid\!\sim$ which is more specific fails to satisfy condition 3.1.

*Proof.* If the confirmation relation $\mid\!\sim$ were more specific than logical entailment, then by definition 5.3, there would be some sentences $p$ and $q$ for which

$$(p \models q) \wedge \neg(p \mid\!\sim q) \tag{6}$$

This violates condition 3.1. □

So, the weakest confirmation relation is uninteresting from the point of view of induction: it is the familiar notion of deductive consequence.

However, it is not so clear what would be the most general confirmation relation possible. This turns our attention to condition 3.3, since it is the only condition setting an upper limit for the confirmation relation. This problem is

[2] For definitions of these terms with respect to relations, please refer to section 5.1.

investigated in more detail in section 6, where we restate conditions on confirmation relations, and section 7, where we develop conditions of plausibility that guarantee the consistency of the confirmation relation.

## 3.2 Hempel's proposal for the confirmation relation

After stating his conditions in [Hem43], Hempel proceeds to give a definition of the confirmation relation for **FOL**. This definition is apparently motivated by a certain property of scientific generalisations. Namely, observation reports are always about particular individuals, but scientific theories[3] are general rules. However, the accepted theories are those that hold true for the particular part of universe that the observation report talks about. For instance, if we have a theory $\forall x Px$, then we would probably say that an observation report confirms our theory (tentatively) if it talks about some individuals, e.g. $a$, $b$ and $c$, and shows $Px$ for all of those individuals, i.e. $Pa \wedge Pb \wedge Pc$.

Hempel makes an attempt to formalise this notion of confirmation. The result is, in effect, a formal account of what it means for a rule to hold for the particular part of universe that an observation report talks about. The meaning of a rule for a part of universe is called the *$C$-development* of the rule for a given finite set $C$ of individuals.

**Definition 3.4** ($C$-development)**.** The $C$-development $D_C(C, p)$ of a proposition $p$ for the set of individuals $C$ is defined thus:[4]

$$\begin{aligned}
D_C(C, S) &= S \quad \text{if } S \text{ is atomic} \\
D_C(C, \neg S) &= \neg D(C, S) \\
D_C(C, S_1 \vee S_2) &= D(C, S_1) \vee D(C, S_2) \\
D_C(C, S_1 \wedge S_2) &= D(C, S_1) \wedge D(C, S_2) \\
D_C(\{\}, \forall \alpha S) &= \top \\
D_C(\{x\} \cup C', \forall \alpha S) &= D_C(\{x\} \cup C', (S[\alpha/x] \wedge D(C', \forall \alpha S))) \\
D_C(\{\}, \exists \alpha S) &= \bot \\
D_C(\{x\} \cup C', \exists \alpha S) &= D_C(\{x\} \cup C', (S[\alpha/x] \vee D(C', \exists \alpha S)))
\end{aligned} \tag{7}$$

where $S[x/a]$ means the sentence $S$ with occurrences of the variable $x$ replaced by the constant $a$.

Intuitively, the $C$-development of a universally quantified sentence $S$ for a set $C$ of individuals states that $S$ is true for all those individuals, and the $C$-development of an existentially quantified sentence $S$ states that $S$ is true for at least one of the individuals. For sentences $S$ without quantification, the $C$-development of $S$ is the original sentence.

**Example 3.5.** The $C$-development of a conditional rule $\forall x(Px \rightarrow Qx)$ for

[3] Here, I use the term "scientific theory" in the narrow sense of a confirmable claim.

[4] The definition is adapted for CNF, i.e. it is assumed that abbreviations such as implication connectives are already eliminated, and given a more rigorous definition with respect to $C$ than in the original [Hem43].

individuals $a$ and $b$ is as follows:

$$\begin{aligned}
&D_C(\{a,b\}, \forall x(\neg Px \vee Qx)) \\
&\quad = D_C(\{a,b\}, (\neg Pa \vee Qa) \wedge D_C(\{b\}, \forall x(\neg Px \vee Qx))) \\
&\quad = D_C(\{a,b\}, (\neg Pa \vee Qa) \wedge (\neg Pb \vee Qb) \wedge D_C(\{\}, \forall x(\neg Px \vee Qx))) \\
&\quad = D_C(\{a,b\}, (\neg Pa \vee Qa) \wedge (\neg Pb \vee Qb) \wedge \top) \\
&\quad = D_C(\{a,b\}, (\neg Pa \vee Qa) \wedge (\neg Pb \vee Qb)) \\
&\quad = D_C(\{a,b\}, \neg Pa \vee Qa) \wedge D_C(\{a,b\}, \neg Pb \vee Qb) \\
&\quad = (D_C(\{a,b\}, \neg Pa) \vee D_C(\{a,b\}, Qa)) \wedge (D_C(\{a,b\}, \neg Pb) \vee D_C(\{a,b\}, Qb)) \\
&\quad = (\neg D_C(\{a,b\}, Pa) \vee Qa) \wedge (\neg D_C(\{a,b\}, Pb) \vee Qb) \\
&\quad = (\neg Pa \vee Qa) \wedge (\neg Pb \vee Qb)
\end{aligned} \tag{8}$$

Hempel then proceeds to define confirmation in terms of the $C$-development of the rule to be confirmed. Firstly, an observation report $E$ *directly confirms* a hypothesis $H$, expressed $E \mid\!\sim_d H$, if and only if it *entails* the $C$-development of $H$ for all individuals in the observation report. However, this notion of direct confirmation does not satisfy Hempel's condition 3.2, so Hempel defines confirmation in terms of direct confirmation.

**Example 3.6.** Let us have the observation report $E$ and the hypotheses:

$$\begin{aligned}
E &= \{Pa\} \\
H_1 &= \forall x Px \\
H_2 &= Pc
\end{aligned} \tag{9}$$

Now, $E \mid\sim H_1$ because the $C$-development of $H_1$ for $I(E)$ is $Pa$, which is equivalent with $E$ and thus logically entailed by $E$. However, $E \not\mid\sim H_2$. Since $H_1 \models H_2$, this violates condition 3.2.

**Definition 3.7** ($C$-development confirmation)**.** An observation report $E$ confirms a hypothesis $H$ if $H$ is entailed by a class of sentences $K$, and $E$ directly confirms all sentences in $K$. The definitions can be formally expressed as:[5]

$$\begin{aligned}
(E \mid\!\sim_d H) &\equiv (E \models D_C(I(E), H)) \\
(E \mid\sim H) &\equiv (\{H' : E \mid\!\sim_d H'\} \models H)
\end{aligned} \tag{10}$$

where $I(E)$ means the set of individual constants in $E$.

Hempel further constrains observation reports to *molecules*, by which he means ground propositions. Hempel also requires that the $C$-development $D_C(I(E), H)$ is not analytic unless $H$ also is, for reasons that we will study in the next section.

**Example 3.8** (Red and green objects)**.** Let our hypothesis be "no object is both totally red and totally green". Then, let us have an observation report of

[5] The formalisation here is a simplified but logically equivalent version of Hempel's definitions of $\text{Cfd}_2$ and $\text{Cf}_2$ [Hem43, 138].

three individuals $a$, $b$ and $c$, one of which is green, another red, and the third one is neither. This gives us the following propositions and $C$-development:

$$\begin{aligned} H &= \forall x(\neg Rx \vee \neg Gx) \\ E &= \neg Ra \wedge Ga \wedge Rb \wedge \neg Gb \wedge \neg Rc \wedge \neg Gc \\ D_C(I(E), H) &= D_C(\{a, b, c\}, H) \\ &= (\neg Ra \vee \neg Ga) \wedge (\neg Rb \vee \neg Gb) \wedge (\neg Rc \vee \neg Gc) \end{aligned} \tag{11}$$

The $C$-development is clearly entailed by $E$. So $E$ directly confirms $H$, and consequently, also confirms it.

Hempel's confirmation relation can be shown to satisfy all of Hempel's conditions. Consequently, whatever critique is to be directed against Hempel's definition, it must rest on our intuitions about the nature of confirmation. The confirmation relation manages to follow common conceptions about confirmation in many cases, but now we will study cases where it does not seem to do so.

## 3.3 Shortcomings in Hempel's confirmation relation

Despite its intuitive appeal, Hempel's account of confirmation is not without problems. Some of these problems Hempel notices himself, and makes workarounds for them, such as the requirement of nonanalytical $C$-developments mentioned in the last section.

One critical problem is that $I(E)$, that is, the set of individuals in the observation report, leads to some strange $C$-developments if its cardinality is too small. One example given by Hempel is the hypothesis $H = \forall xPx \vee \forall x\neg Px$. This hypothesis is not analytic, but its $C$-development over a singleton set is:

$$D_C(\{a\}, \forall xPx \vee \forall x\neg Px) = Pa \vee \neg Pa. \tag{12}$$

Thus, the $C$-development is entailed by any observation report whatsoever, and, consequently, the hypothesis is confirmed by any observation report that only mentions one individual, for instance $Qa \wedge Ra$.

We might add that an observation report mentioning no individuals at all, for instance $\top$, confirms all universally quantified hypotheses. Hempel does not encounter this problem, because his language does not include nullary predicates and observation reports are required to be ground.

A difficulty in another direction is that the observation reports may introduce individuals which in no way affect the logical content of the observation report. This problem is cursorily discussed in a very long footnote in [Hem45, 110–111].

**Example 3.9.** Consider the hypothesis $H = \forall xPx$ and the observation reports

$$\begin{aligned} E_1 &= Pa \\ E_2 &= Pa \wedge (Qb \vee \neg Qb). \end{aligned} \tag{13}$$

Now, $E_1$ and $E_2$ are logically equivalent, but the set of individuals in $E_1$ is $I(E_1) = \{a\}$ whereas in $E_2$ it is $I(E_2) = \{a, b\}$. This leads to different $C$-developments of $H$, namely $D_C(I(E_1), \forall xPx) = Pa$ and $D_C(I(E_2), \forall xPx) = Pa \wedge Pb$. It is, then, easily seen that both observation reports entail the former $C$-development, while the latter is entailed by neither. As a consequence, $E_1 \mathrel{|\!\sim} H$ while $E_2 \not\mathrel{|\!\sim} H$, in spite of the logical equivalence.

These problems do not violate any of Hempel's criteria for confirmation relations, since the conditions actually say very little about how two somewhat similar observation reports relate to each other. There is no condition that logically equivalent observation reports confirm the same hypotheses; nor need the conjunction of two observation reports confirm anything that the original observation reports confirm. This is probably sensible given the nonmonotonicity of confirmation: strengthening the observation report $Pa \wedge Pb$ by a new conjunct $\neg Pc$ should definitely drop its confirmation for the hypothesis $\forall x Px$. However, the unintuitive results above call for criteria that explicate the relations between observation reports.

Hempel deals with the problem of extraneous individuals in observation reports by introducing the concept of *essential* individuals. An individual is essential in a proposition if and only if it is present in every logically equivalent proposition. In my opinion, this is hardly a sufficient remedy for the situation. Some individuals may be essential to some hypotheses and others may be essential for others. This calls for a more fine-grained notion of essentiality.

**Example 3.10.** Take, for instance, two totally unrelated observation reports $E_1 = Pa$ and $E_2 = Qb$. For the individual observation reports, we have:

$$\begin{aligned} E_1 &\mid\sim \forall x Px \\ E_2 &\mid\sim \forall x Qx \end{aligned} \tag{14}$$

However, the conjunction of the observation reports $E_1 \wedge E_2$ fails to confirm either. The situation cannot be remedied by an account of essential and inessential variables, since both $a$ and $b$ are obviously essential for their respective predicates, $P$ and $Q$.

Now someone can claim that it is not appropriate for scientific observation reports to combine unrelated data, so $E_1$ and $E_2$ should just be treated separately. However, in the absence of any formal definition of when observation reports can be combined and when they cannot, the exact confirmative content of $E_1 \wedge E_2$ remains unclear. Worse yet, in light of the observation report $Pa \wedge Pb \wedge \neg Pc$, observation reports cannot be split arbitrarily and still produce reliable results: the clause $\forall x Px$ is supported by $Pa$ and $Pb$, but falsified by $Pc$.

It is also possible to come up with examples where the interdependence of observations or lack thereof is not so clear. Take, for instance,

$$E = Pa \wedge Pb \wedge Qb. \tag{15}$$

Now, should $E$, intuitively, confirm the hypothesis $\forall x Qx$? By Hempel's confirmation relation, it does not. However, the only reason here is that it is unknown whether $Qa$ holds. What kind of criterion could there be to exclude $Pa$ from the observation report, in order to drop $a$ from $I(E)$? A similar situation occurs if, in our example 3.8, we add a new individual $d$ for which we only know that it is red ($Rd$) but the observation report fails to include any information about its greenness ($Gd$ or $\neg Gd$).

So, actually there appear to be two problems with Hempel's confirmation relation. The first one is that Hempel's notion of *universe of discourse*, upon which $C$-development is based, is very crude. Simply taking that set on individuals that an observation report somehow mentions does not yield adequate

results in the way that a small set of individuals makes it "too easy" to confirm a rule, and a large set makes it "too hard". Maybe there could be some kind of *local $C$-development* that would give a different set of individuals for rules that concern different parts of the observation report.

But there is the even harder problem that Hempel's confirmation relation is too cautious, i.e. too strong. This is caused by the fact that the observation report is required to *entail* the $C$-development of a rule in order to confirm it. Consequently, any absence of information (such as information about greenness above) tends to thwart otherwise totally sensible rule candidates. This is a depressing result, because our experience of the world is usually quite incomplete. If our ability to scientifically confirm anything depends on picking *exactly* the right data to do so and ignoring the rest of our knowledge, this hardly seems an good account of scientific confirmation.

# 4 What is induction not?

Given the enormous scope and usage of the word "induction", it is necessary to dedicate a whole section for explaining what is *not* meant by induction in this article.

## 4.1 Mathematical induction

The first thing to mention is probably the contrast between mathematical induction and induction in philosophy. Mathematical induction is actually a kind of deduction (necessary reasoning), because it produces irrefutably true conclusions from true premisses. The premisses of mathematical induction are also different from the premisses of philosophical induction: philosophical induction proceeds from particular instances only, while mathematical induction uses at least one generic rule as its premiss.

**Example 4.1.** The typical examples of philosophical and mathematical induction are the following inferences. To emphasise the *similarities* between both kinds of induction, we denote an arbitrary enumeration $\mathcal{E}$ of individuals by a function $f$ from natural numbers to individuals. This is philosophical induction:

$$\frac{P(f(1)) \qquad P(f(2)) \qquad P(f(3))}{\forall n(n \in \mathbb{N} \to P(f(n)))} \tag{16}$$

And this is mathematical induction:

$$\frac{P(f(0)) \qquad \forall n(P(f(n-1)) \to P(f(n)))}{\forall n(n \in \mathbb{N} \to P(f(n)))} \tag{17}$$

The greatest difference between mathematical and philosophical induction is that philosophical induction can produce general rules from finite premisses, that is, premisses that only involve a finite number of individuals. What the two kinds of induction have in common is that they both produce general claims by some kind of *enumeration* of individuals. For mathematical induction, this enumeration starts from $f(0)$ and covers $f$ for all natural numbers, so it is exhaustive. The enumeration in philosophical induction is also exhaustive within the finite domain of the premisses; the result is just raised to a broader domain.

## 4.2 Conceptualisation

Although inductive thinking usually proceeds from experience, it is not necessary to define what experience is and how experience is related to generic thinking. Rather than such a psychological theory, we are developing a syntactic theory that deals with *representations* of experience and *representations* of general rules. The chosen language for representing both is **FOL**.

The process of transforming experience into conceptual representations of that experience is called *conceptualisation*. Conceptualisation is a very interesting problem in its own right. Conceptualisation is also intimately connected with induction, because both give some kind of structure to the world: induction collects random facts into general rules, whereas conceptualisation collects random experience, such as sense data, into concepts, such as individuals and their relations. Both may also involve errors and nonmonotonic thinking: an

inductive conclusion may prove to be false, and a concept may turn out to not to refer to anything at all.

However, the process of conceptualisation is outside the scope of this thesis. For our purposes, we simply assume we have a correct conceptualisation of our experience, called an observation report, and build the rules of producing inductive conclusions from these observation reports.

## 4.3 Validation of hypotheses

Induction produces hypothetical claims which are based on the information available at a given moment. The assessment of these hypotheses is an important part of inductive thinking. If we would not, for instance, abandon a hypothesis when it is refuted, the results of induction could hardly be considered to be based on experience.

However, validation of hypotheses does not constitute *all* of inductive inference. The hypotheses do not come out of the blue. It would seem that it is possible to build logics that depict at least the simplest forms of inductive inference: people have a tendency to form the same conclusions from the same experience, and this process may be given a definite form. It is hardly a philosophical attitude to deny the possibility of such an account without even attempting to formalise it. I feel that a lot of interesting inference has been neglected simply because formation of hypotheses has been mystified, for instance as in [Car50, 192–193].

Some aspects of theory formation may be harder to formalise than others, such as the need for new concepts. However, whatever the process is that people use for such activities, it is possible to at least model the process. It is hard to see why there *could* not be a logic for thinking processes in general and scientific thinking processes in particular. The possibility of such a logic is also defended in [Meh99] and [Fla96]. Such a logic might be complicated, but it can also be tackled in smaller pieces. The piece that this thesis deals with is the generalisation of statements that express our experience.

## 4.4 Abduction

The terms *induction* and *abduction* warrant a terminological note. Both are nowadays used in many meanings, but the most widely accepted definitions are probably these: induction is reasoning that produces a general rule out of many particular instances, while abduction is "inverse deduction", reasoning from conclusions to premisses. But general propositions usually logically entail their particular instances, and premisses logically entail their conclusions. Thus, for many cases, the definitions actually coincide or at least overlap: both include types of reasoning where we take a proposition $p$ and produce from that other propositions that logically entail $p$.

Since of these two, the term abduction seems even less clearly defined, I shall only talk about induction in this thesis.

## 4.5 Setting of induction

In this thesis, induction is treated as an operation in *logic*, more specifically, in first-order predicate logic (**FOL**). While this choice might seem uncontroversial,

it has been challenged by several general frameworks of induction: for instance, in [Sol64], induction is defined as extrapolation (prediction) of symbol sequences, and in [Ren86], induction is defined as partitioning of a set of objects into subsets (classification).

The classical account of inductive inference is that it takes examples and produces propositions that extrapolate the properties of these examples to predict some properties of future examples. The examples are usually observations of some kind, but many problems of induction also use other kinds of input such as background theories that codify assumed knowledge of the world. I doubt that this classic view of induction can be accommodated by the aforementioned frameworks.

As for extrapolation of symbol sequences, the problem is that while symbol sequences can be used to codify all kinds of information, they are inadequate for codifying the unimportance or lack of a certain kind of information, such as independence of two points of data. For the purposes of logical induction, the order of both the learning data and the data to be predicted are irrelevant unless explicitly studied. For instance, if we observe properties of different kinds of trees, induction is not required to predict whether we will see a tree or a crow first in the future, or whether we will ever observe a tree anymore. How could this kind of irrelevance be expressed in symbol sequences?

The problem with classification is that the results of logical induction cannot always be seen as definitions of distinct classes. There are certain subcases of induction that are clearly classification, but not all inductive thinking can be seen as classification (see section 4.6).

In this thesis, induction is simply thought of as an operation that takes logical formulae as input and produces logical formulae as output. Further restrictions on induction are defined in sections 6 and 7. More precisely, induction is treated as inductive inference: as a logical process of deriving correct inductive consequences from a set of clauses that represent known facts. Induction is taken to have a definite logical form: for any given set of facts, we should be able to tell exactly which clauses are its inductive consequences and which are not.

## 4.6 Classification of objects

The problem of classification of objects has similarities with induction, and concept learning, which is a subcase of induction, is basically the same as classification of objects. For instance, if we are told that the numbers 5, 11 and 23 are prime but 8, 15 and 26 are not, we can try to induce some kind of definition for primality. In doing so, we have classified numbers into two kinds: prime and not prime. This is exactly the kind of problem that inductive logic programming (**ILP**, see for example [Mug92c]) researches.

In logic, classes of objects are represented by predicates. If we induce sentences that provide the sufficient and necessary conditions for a predicate to be true, we have effectively defined a class, because for each object, we can use these conditions to determine whether it belongs to the class. This is the similarity between induction and classification.

However, inductive thinking need not produce clear-cut classes. Induction may well leave the truth value of a predicate unknown for some objects; induction doesn't necessarily fix, for every possible claim, some "experience-based"

truth value. Also, classification does not work as well for real-world objects as for mathematical objects. The difference between the two is that mathematical (or logical) objects are totally determined by their name; for instance, all the properties of number 3 can be found out by referring to its definition. Real-world objects, on the other hand, have no definitions, so there is no telling whether an individual called Spark belongs to the class of fish or the class of dogs (or some other class).

For this reason, **ILP** methods usually involve *background theories*, which provide information about the properties of the individuals in the observation reports. The result of these methods, then, is a classification of objects based on the properties mentioned in the background theory.

The induction method in this thesis does not produce rules for determining whether an object belongs to a given class, but rather induces constraints that hold across all the individuals in our observation reports. As a side effect, it does not need any kind of background theory. From another point of view, the induced rules use the facts in the observation report to explain other facts in the observation report.

The difference between classificatory and constraint induction are further discussed in section 5.5. There, both are studied for their goal, which for classificatory induction is usually information compression, and for constraint induction, informativeness.

## 4.7 Generalisation of clauses

There is another aspect in which **ILP** has a narrower setting than our account of induction. In our account of induction, the result of induction is an extension of our experience and so logically implies all of our experience. However, it is not required that individual *clauses* in the inductive consequence theory always imply some individual clauses in the original input theory.

**ILP**, on the other hand, is concerned with construction of clauses that are logically stronger than clauses in the input theory. When combined with the language bias of ordinary **ILP** where no conjunctions are permitted (**ILP** deals with clauses of conjunctive normal form, see section 5.3), this produces fairly different results from our conception of induction.

## 4.8 Calculation of probabilities

There has been, in the philosophy of science, a long discussion whether induction is about the *probabilities* or *degrees of acceptance* of various hypotheses, or of a totally different nature. In [ZZ96], Zwirn and Zwirn show that there are two fundamentally different models of confirmation, which are named *absolute* and *relative* confirmation after terminology that was adopted from Carnap's classic [Car50].[6]

Carnap uses his own probabilistic framework [Car50, 468–482] to criticise the conditions presented by Hempel in [Hem45]. Hempel's conditions pertain to absolute confirmation, whereas Carnap builds a theory that is based on relative confirmation, and proceeds to show that Hempel's conditions fail on Carnap's definition of confirmation. If we take quantitative confirmation to be the most

[6] Actually, [ZZ96] also presents a third kind of confirmation, which is not discussed here.

fundamental kind of confirmation, this shows strong support that the kind of confirmation Hempel had in mind is a fundamentally flawed concept.

However, it is shown by [ZZ96, 218–223] that while Hempel's criteria are mutually consistent, no probabilistic criterion of confirmation can satisfy them. So, the argument that seemed to be unfavourable for absolute confirmation actually turns out to show that probabilistic accounts of induction are simply fundamentally incompatible with the kind of confirmation Hempel studies. This is also the case of the maximal plausible generalisation presented in this thesis.

# 5 Theoretical background

In this section, I discuss theories and concepts that are important for induction, machine learning and concept learning in general. For the problem at hand, the most important of these are the concept of version space presented by [Mit82] and the $\theta$-subsumption relation [Plo71].

## 5.1 Generality of concepts and generality of statements

Let us first examine the concept of generalisation. Generalisation seems to carry two meanings. The first and by far more common one is the generalisation of *concepts*, which we will define first.

A common view is that a concept, such as "dog", is a class; its members (*instances*) are all the concrete dogs that exist. All concepts have zero to numerous instances. Generality of concepts is defined as a inclusion relation of their instances. In logic, concepts are usually represented by predicates: a concept $c$ is represented by a predicate $P_c$ which is true of exactly those individuals that are instances of concept $c$. This allows the following definition of generality.

**Definition 5.1** (Generality of concepts)**.** A concept $c_1$ is more general than another concept $c_2$ if all of the instances of $c_2$ are also instances of $c_1$. Using $\supseteq$ for "more general than", this becomes:

$$c_1 \supseteq c_2 \equiv \forall x(P_{c_2}(x) \rightarrow P_{c_1}(x)) \tag{18}$$

In this situation, we use the same terminology for the predicates, calling $P_{c_1}$ more general than $P_{c_2}$.

**Example 5.2.** The concept of *binary relation* is a generalisation of the concept of *equality relation*. Using $Eq(x)$ to denote "$x$ is an equality relation" and $R^2(x)$ to denote "$x$ is a binary relation", this can readily be expressed as $\forall x(Eq(x) \rightarrow R^2(x))$.

This definition of generality generalises naturally to predicates of an arbitrary valence, i.e. relations.

**Definition 5.3** (Generality of relations)**.** A relation $R_1$ is more general than $R_2$ if and only if all tuples of individuals that satisfy $R_2$ also satisfy $R_1$.

$$R_1 \supseteq R_2 \equiv \forall x \forall y \ldots (R_2(x, y, \ldots) \rightarrow R_1(x, y, \ldots)) \tag{19}$$

**Example 5.4.** The relation "is a parent of" is a generalisation of the relation "is a father of". This can be expressed as

$$\forall x \forall y(\text{Father}(x, y) \rightarrow \text{Parent}(x, y)). \tag{20}$$

However, sometimes the word generalisation is also used to refer to generalisation of *statements*, especially so when discussing inductive reasoning. The definition of generality above, which is based on the inclusion relation of instances, is problematic for statements because the meaning of "instance" is far less obvious for statements than for concepts. Even more problematically, the common usage of "more general" for statements seems contrary to the one for

concepts in the following way. For concepts, the claim that $c_1 \supseteq c_2$ means that $c_1$ is a *weaker* concept, placing less restrictions on its instances than $c_2$, and consequently, has more instances. However, for statements $S_1 \supseteq S_2$ means that $S_1$ is a *stronger* statement, placing more restrictions on the world, and consequently, has less situations where it remains true.

**Example 5.5.** The statement $S_1$ = "every human is mortal" is a generalisation of the statement $S_2$ = "every human is a spatial object". This is because, presuming that all mortal things are spatial objects, $S_2$ is true whenever $S_1$ is true. $S_1$ is a stronger claim than $S_2$ because it claims more.

This suggests that we should really investigate what the "instance" of a statement is. I'll take a look at two alternative approaches. The first one is a model-theoretic approach and the second one is a meaning-based approach that tries to unify the view of generalisation with respect to concepts and statements. The model-theoretic generality of statements has the following definition.

**Definition 5.6** (Model-theoretic generality of statements)**.** An instance of a statement $S$ is a possible world (or situation, context) where $S$ is true. Using $w : S$ to mean that $S$ is true in possible world $w$, the common use of "$S_1$ is more general than $S_2$" can be described thus:

$$S_1 \supseteq S_2 \equiv \forall w((w : S_1) \rightarrow (w : S_2)) \tag{21}$$

Now, this makes clearly visible the discrepancy mentioned above. Contrasting 5.1 with 5.6, we notice that the implication points in the opposite direction, which corresponds to the reversal of "stronger" and "weaker" in our account of generality.

The second approach is to consider as instances of a statement, not the worlds where it is true, but the constraints it places on the world(s) where it is true. This way, a stronger statement which places more constraints on the world, has more instances. The statement is "weaker" in the way that it places less restrictions on its instances, which are constraints on the world.

**Definition 5.7** (Truth-based generality of statements)**.** An instance of a statement $S'$ is another statement $S$ that is true in every model of $S'$, that is, $S$ represents a constraint that is observed in all models of $S'$. Formally put:

$$S_1 \supseteq S_2 \equiv \forall S((S_2 \models S) \rightarrow (S_1 \models S)) \tag{22}$$

Now, we can see that 5.1 and 5.7 are equivalent, with $P_{S_n}(S) = (S_n \models S)$. This might seem counterintuitive because the instances of statements are other statements, but because all constraints on the world are facts and every fact is expressible as a statement, this is not so surprising. Actually, this interpretation of generality lends itself to the interpretation that facts are *examples* of truth and generalisations are extrapolations of truth. This view is used in section 6.

It should be noted that both definitions 5.6 and 5.7 are really equivalent; they just express different views of the common use of the generality relation between statements. In fact, both are equivalent to:

$$S_1 \supseteq S_2 \equiv (S_1 \models S_2) \tag{23}$$

As it happens, this is also the definition of generality given by Stephen Muggleton in [Mug92a]. The value of the restriction-based interpretation is that is allows us to apply the concept of version spaces to generalisation of statements.

The confusion that remains is that for concepts, the relation "more general" is synonymous with "weaker" and the relation "more specific" is synonymous with "stronger", whereas for statements, the situation is the other way around: "more general" is synonymous with "stronger" and "more specific" is synonymous with "weaker". There is little to be done about this terminological problem. Usage with relations follows that of concepts: the weaker a relation is, the more general it is, which means that it holds for at least the same individuals as the more specific (i.e. stronger) relation. E.g. [IA93] follows this convention.

## 5.2 Version spaces

Tom M. Mitchell presented in [Mit82] a generic framework for generalisation of concepts. Generalisation of concepts is a broad area of application indeed, but it turns out Mitchell's framework is applicable to even broader areas, with a slight stretch to what is considered a "concept". We shall now study Mitchell's concept of *version space* in detail.

Mitchell's analysis applies to a class of generalisation problems that can be defined in the following way. A *generalisation algorithm* accepts descriptions of *training instances* as input. The training instances are represented in a language called the *instance language*, and they are accompanied with a classification of whether or not they belong to the *target generalisation* that the generalisation algorithm should find. The algorithm then outputs a *generalisation*, which corresponds to a class of instances and is represented in a language called *generalisation language*. In order to test various possible generalisations, the algorithm also has access to a *matching predicate*, which tells whether a generalisation *matches* an instance, i.e. whether the instance belongs to the class of instances that the generalisation represents. This problem can be formalised in the following way.[7]

**Definition 5.8** (Generalisation algorithms)**.** A generalisation algorithm takes, as input:

1. a matching predicate $M(g, i)$ which is true if and only if the generalisation $g$ matches (includes) the instance $i$;
2. a set of positive instances $I_+$ that belong to the target generalisation, in the instance language; and
3. a set of negative instances $I_-$ that do not belong to the target generalisation, in the instance language.

As output, the algorithm produces a generalisation $g$ that matches all the positive training instances while matching none of the negative training instances. It is also said that $g$ is *consistent* with the training sets.

**Definition 5.9** (Version space)**.** The set of all generalisations that are consistent with the training instances is called the *version space* for those instances

[7] This formalisation is adapted from [Mit82, 204].

$VS(I_+, I_-)$. More formally put:

$$VS(I_+, I_-) = \{g : \forall i(i \in I_+ \rightarrow M(g,i)) \wedge \forall i(i \in I_- \rightarrow \neg M(g,i))\} \tag{24}$$

The training set $I_+$ gives a lower bound for the version space, because it makes inconsistent those generalisations that are too specific to match instances in $I_+$; and correspondingly, the training set $I_-$ gives an upper bound for the version space by making inconsistent those generalisations that are general enough to match instances in $I_-$.

**Example 5.10** (Ranges of real numbers)**.** As an example, consider an instance language where the instances are real-valued numbers and a generalisation language where the generalisations are inclusive ranges on the real scale. A generalisation $[y, z]$ matches an instance $x$ if and only if $x$ belongs to the range, i.e. $y \leq x \leq z$. Then, for the positive training set $\{0, 1\}$ and negative training set $\{-10, 3, 5\}$, the version space is defined as:

$$VS(\{0,1\}, \{-10,3,5\}) = \{[y,z] : -10 < y \leq 0 \wedge 1 \leq z < 3\} \tag{25}$$

It is easy to notice that the set of all generalisations $\mathcal{G}$ forms a kind of *search space*. The goal of a generalisation algorithm is to find a generalisation $g \in VS(I_+, I_-) \subset \mathcal{G}$ which is consistent with the training sets; generally there may be many such generalisations, but this depends on the training sets and the expressiveness of the generalisation language. It is also important to notice that the matching predicate $M$ gives an inherent structure for $\mathcal{G}$. More precisely, $M$ gives rise to a partial ordering of generalisations, defined below.

**Definition 5.11** (Generality of generalisations)**.** A generalisation $g_1$ is *more general than* another generalisation $g_2$ (denoted $g_1 \supseteq g_2$) if and only if $g_1$ matches at least the same instances as $g_2$. The converse relation is that $g_2$ is *more specific than* $g_1$. Likewise, a generalisation $g_1$ is *strictly more general than* $g_2$ (denoted $g_1 \supset g_2$) if $g_1$ matches more instances than $g_2$, and then $g_2$ is *strictly more specific than* $g_1$.

$$\begin{aligned} g_1 \supseteq g_2 &\equiv \forall i(M(g_2,i) \rightarrow M(g_1,i)) \\ g_1 \supset g_2 &\equiv g_1 \supseteq g_2 \wedge g_2 \not\supseteq g_1 \end{aligned} \tag{26}$$

The existence of such a partial ordering means that in every generalisation problem, if we have a way to systematically construct, from a generalisation $g$, the sets of more general generalisations $\{g' : g' \supseteq g\}$ and more specific generalisations $\{g' : g \supseteq g'\}$, then we can search through $\mathcal{G}$ by generalising and specialising until we arrive at a consistent generalisation. Another requisite is that we have some kind of starting point for the search. There is often a natural starting point in $\mathcal{G}$, or the starting point can often be formed from the training instances. In section 6.6, we will examine what this means from the viewpoint of generalisations in logic.

**Example 5.12.** The starting point for real ranges is easily obtained from any positive instance: for example, the instance 1 is minimally matched by the generalisation $[1, 1]$. Ranges can be generalised by lowering the lower bound or raising the upper bound, and specialised by the contrary actions.

The partial ordering of generality and the boundaries of version spaces together give rise to another two concepts: the *minimal* and *maximal* generalisations.

**Definition 5.13** (Minimal and maximal generalisations)**.** A minimal generalisation is a generalisation for which there are no strictly more specific consistent generalisations. Conversely, a maximal generalisation is a generalisation for which there are no strictly more general consistent generalisations.

$$\begin{aligned} \min(g, M, I_+, I_-) &\equiv \forall g'(g \supset g' \rightarrow \exists i(i \in I_+ \wedge \neg M(g', i))) \\ \max(g, M, I_+, I_-) &\equiv \forall g'(g' \supset g \rightarrow \exists i(i \in I_- \wedge M(g', i))) \end{aligned} \tag{27}$$

**Example 5.14.** In example 5.10, there is only one minimal and one maximal generalisation in the version space, the minimal one being $[0, 1]$ and the maximal one being $[\lim^+_{x \to -10} x, \lim^-_{x \to 3} x]$.

One way to look at the minimal and maximal generalisations is to say that a minimal generalisation predicts that all unknown instances (whose classification is left undefined by the training sets) are outside the target generalisation, while a maximal generalisation predicts that all unknown instances do belong to the target generalisation.

One of the most important contributions of Mitchell's article was to note that the sets of minimal and maximal generalisations together totally define the version space. This is because every generalisation in the version space is more general than at least one of the minimal generalisations and more specific than at least one of the maximal generalisations. Moreover, Mitchell introduced a technique for finding generalisations that incrementally tracks the whole version space by updating the list of minimal and maximal generalisations to account for all instances in the training sets.

## 5.3 Conjunctive normal form

For the treatment of logical formulae, it is practical to keep their language as simple as possible. For **FOL**, we can greatly simplify the logical language by transforming all propositions and sets of propositions, i.e. theories, into conjunctive normal form (CNF). Since every **FOL** formula can be transformed into CNF, we can do this without losing any expressivity.

In CNF, all conjunctions of a formula are on the top level of the formula. The conjuncts of the formula, also called *clauses*, are disjunctions of atomic formulae, also called *literals*. Every literal is either a predicate expression or the negation of a predicate expression. All quantifiers enclose a whole clause. The requirement that negations are only used before atomic formulae is also called negative normal form (NNF), and the requirement that quantifiers only occur on top level of clauses is called prenex normal form (PNF). The clauses are usually also *skolemised*, which means that existentially quantified variables are replaced by functions of the universally quantified variables which the existentially quantified variable depends on, that is, the universally quantified variables the scope of which the existential quantifier resides in.

**Example 5.15.** The following proposition is in CNF:

$$\begin{aligned}&\forall x \forall y(\neg P(x) \vee Q(x,y) \vee P(y)) \wedge \\ &\forall x(\neg Q(x,a) \vee R(x)) \wedge \\ &\forall x \forall y(\neg Q(x, f(y)) \vee \neg Q(y,x))\end{aligned} \tag{28}$$

### 5.3.1 Rewriting formulae into CNF

The following method for obtaining the CNF of a formula is adapted from [RN03, 295–297] by stating the rewrite rules more rigorously. The CNF of a formula can be obtained by successive applications of rewrite rules to the formula. First, we eliminate abbreviations (implications and equivalences) by the following rewrite rules:

$$\begin{aligned}(S \equiv T) &\longmapsto (S \to T) \wedge (T \to S) \\ (S \to T) &\longmapsto \neg S \vee T\end{aligned} \tag{29}$$

**Example 5.16.** If we have a claim that a sword is dangerous if and only if its blade is sharp, we eliminate the abbreviations in the following way:

$$\begin{aligned}&\forall x(\text{DangerousSword}(x) \equiv \text{Sword}(x) \wedge \exists y(\text{Blade}(x,y) \wedge \text{Sharp}(y))) \\ \longmapsto\ &\forall x((\text{DangerousSword}(x) \to \text{Sword}(x) \wedge \exists y(\text{Blade}(x,y) \wedge \text{Sharp}(y))) \wedge \\ &\quad (\text{Sword}(x) \wedge \exists y(\text{Blade}(x,y) \wedge \text{Sharp}(y)) \to \text{DangerousSword}(x))) \\ \longmapsto\ &\forall x((\text{DangerousSword}(x) \vee \neg(\text{Sword}(x) \wedge \exists y(\text{Blade}(x,y) \wedge \text{Sharp}(y)))) \wedge \\ &\quad ((\text{Sword}(x) \wedge \exists y(\text{Blade}(x,y) \wedge \text{Sharp}(y))) \vee \neg\text{DangerousSword}(x)))\end{aligned} \tag{30}$$

The next step is to obtain negative normal form by permuting negations deeper into clauses by double negation elimination, De Morgan's laws and mutual definitions of quantifiers. This gives the following rewrite rules:

$$\begin{aligned}\neg\neg S &\longmapsto S \\ \neg(S \vee T) &\longmapsto \neg S \wedge \neg T \\ \neg(S \wedge T) &\longmapsto \neg S \vee \neg T \\ \neg\exists x S &\longmapsto \forall x \neg S \\ \neg\forall x S &\longmapsto \exists x \neg S\end{aligned} \tag{31}$$

**Example 5.17.** The NNF of the definition of dangerous swords looks like this:

$$\begin{aligned}&\forall x((\text{DangerousSword}(x) \vee \neg\text{Sword}(x) \vee \forall y(\neg\text{Blade}(x,y) \vee \neg\text{Sharp}(y))) \wedge \\ &\quad ((\text{Sword}(x) \wedge \exists y(\text{Blade}(x,y) \wedge \text{Sharp}(y))) \vee \neg\text{DangerousSword}(x)))\end{aligned} \tag{32}$$

The process of skolemisation is not so easily expressed as rewrite rules, because the rewriting of existentially quantified variables depends on their context: the universal variables they depend on. Because of this, we define skolemisation as a recursive function where the depended-on variables are passed as a parameter. Moreover, skolemisation is based on substitution of terms, which we also define.

**Definition 5.18** (Substitution of terms)**.** Substitution of a term $t$ by another term $u$ in a formula $S$ is denoted by $S[t/u]$.

$$\begin{aligned}
S[t/u] &\longmapsto S \quad \text{if } t \text{ does not occur in } S \\
(\neg S)[t/u] &\longmapsto \neg S[t/u] \\
(S \vee T)[t/u] &\longmapsto S[t/u] \vee T[t/u] \\
(S \wedge T)[t/u] &\longmapsto S[t/u] \wedge T[t/u] \\
(\forall x S)[t/u] &\longmapsto (\forall z S[x/z])[t/u] \quad \text{if } x \text{ occurs in } t \text{ or } u;\ z \text{ is a new variable} \\
(\forall x S)[t/u] &\longmapsto \forall x S[t/u] \quad \text{otherwise} \\
(\exists x S)[t/u] &\longmapsto (\exists z S[x/z])[t/u] \quad \text{if } x \text{ occurs in } t \text{ or } u;\ z \text{ is a new variable} \\
(\exists x S)[t/u] &\longmapsto \exists x S[t/u] \quad \text{otherwise} \\
P(v_1, v_2, \dots)[t/u] &\longmapsto P(v_1[t/u], v_2[t/u], \dots) \quad \text{where P is a predicate symbol} \\
f(v_1, v_2, \dots)[t/u] &\longmapsto f(v_1[t/u], v_2[t/u], \dots) \quad \text{where f is a function symbol} \\
t[t/u] &\longmapsto u \\
v[t/u] &\longmapsto v \quad \text{if } t \text{ does not occur in } v
\end{aligned} \tag{33}$$

**Definition 5.19** (Skolemisation)**.** The first parameter $D$ which is a set of depended-on variables is initially the empty set.

$$\begin{aligned}
Sk(D, S) &\longmapsto S \quad \text{if } S \text{ does not include existential quantifiers} \\
Sk(D, \neg S) &\longmapsto \neg Sk(D, S) \\
Sk(D, S \vee T) &\longmapsto Sk(D, S) \vee Sk(D, T) \\
Sk(D, S \wedge T) &\longmapsto Sk(D, S) \wedge Sk(D, T) \\
Sk(D, \forall x P) &\longmapsto \forall x Sk(D \cup \{x\}, P) \\
Sk(D, \exists x P) &\longmapsto Sk(D, P[x/f_x(\overline{D})])
\end{aligned} \tag{34}$$

where $f_x(\overline{D})$ is a function term whose parameters are the variables in $D$ and the function symbol $f_x$ is totally new (it does not occur anywhere in the formula before). As a special case, $f_x$ is a constant (a nullary function) if the set $D$ is empty, which means that $x$ does not depend on any universally quantified variables.

**Example 5.20.** Our sword example has only one existential quantifier (for $y$), which is in the scope of one universal quantifier (for $x$). Skipping the intermediate steps of skolemisation, this yields:

$$\begin{aligned}
&\forall x((\text{DangerousSword}(x) \vee \neg\text{Sword}(x) \vee \forall y(\neg\text{Blade}(x, y) \vee \neg\text{Sharp}(y))) \wedge \\
&\quad ((\text{Sword}(x) \wedge \exists y(\text{Blade}(x, y) \wedge \text{Sharp}(y))) \vee \neg\text{DangerousSword}(x))) \\
\longmapsto\ &\forall x((\text{DangerousSword}(x) \vee \neg\text{Sword}(x) \vee \forall y(\neg\text{Blade}(x, y) \vee \neg\text{Sharp}(y))) \wedge \\
&\quad ((\text{Sword}(x) \wedge \text{Blade}(x, f_y(x)) \wedge \text{Sharp}(f_y(x))) \vee \neg\text{DangerousSword}(x)))
\end{aligned} \tag{35}$$

Intuitively, the meaning of $f_y(x)$ is "the blade of $x$".

After skolemisation, the formula is put into CNF by raising all conjunctions above disjunctions via disjunction distribution. We also permute universal quantifiers below conjunctions in order to be able to apply disjunction distribution

to universally quantified formulae.

$$\begin{aligned} \forall x(S \wedge T) &\longmapsto \forall x S \wedge \forall x T \\ (S \wedge T) \vee U &\longmapsto U \vee (S \wedge T) \\ S \vee (T \wedge U) &\longmapsto (S \vee T) \wedge (S \vee U) \end{aligned} \tag{36}$$

**Example 5.21.** The CNF of the sword definition looks like this:

$$\begin{aligned} &\forall x(\text{DangerousSword}(x) \vee \neg\text{Sword}(x) \vee \forall y(\neg\text{Blade}(x,y) \vee \neg\text{Sharp}(y)))\wedge \\ &\forall x(\neg\text{DangerousSword}(x) \vee \text{Sword}(x))\wedge \\ &\forall x(\neg\text{DangerousSword}(x) \vee \text{Blade}(x, f_y(x)))\wedge \\ &\forall x(\neg\text{DangerousSword}(x) \vee \text{Sharp}(f_y(x))) \end{aligned} \tag{37}$$

Finally, we obtain prenex normal form by lifting universal quantifiers over disjunctions. In doing so, the only special consideration is that we must not cause name clashes between variables in different literals.

$$\begin{aligned} S \vee \forall x T &\longmapsto \forall x T \vee S \\ \forall x S \vee T &\longmapsto \forall x(S \vee T) \quad \text{if } x \text{ does not occur in } T \\ \forall x S \vee T &\longmapsto \forall y(S[x/y] \vee T) \quad \text{otherwise} \end{aligned} \tag{38}$$

Here, $y$ is a new variable occurring in neither $S$ nor $T$.

**Example 5.22.** The PNF of the sword definition is as follows.

$$\begin{aligned} &\forall x \forall y(\neg\text{Blade}(x,y) \vee \neg\text{Sharp}(y) \vee \text{DangerousSword}(x) \vee \neg\text{Sword}(x))\wedge \\ &\forall x(\neg\text{DangerousSword}(x) \vee \text{Sword}(x))\wedge \\ &\forall x(\neg\text{DangerousSword}(x) \vee \text{Blade}(x, f_y(x)))\wedge \\ &\forall x(\neg\text{DangerousSword}(x) \vee \text{Sharp}(f_y(x))) \end{aligned} \tag{39}$$

This final form bears little resemblance to our original example and can be deemed rather unintuitive. However, it is readable and intuitive in its own way. Every clause (conjunct) represents a *rule* about the relationship between various concepts, and the literals of a clause are the alternatives that may make the rule hold for given individuals; at least one of the alternatives must be true for every assignment of variables.

### 5.3.2 Significance of CNF

The benefit of transforming a proposition or a set of propositions into conjunctive normal form is twofold. Firstly, CNF makes it easy to algorithmically reason about the proposition or set of propositions. Practically all automated deduction methods beginning with [Rob65] deal with theories in conjunctive normal form.

One of the most obvious benefits of CNF is that it eliminates the distinction between a proposition and a set of propositions that are taken to hold together. Because the conjunctions are all on the top level of the proposition, it does not really matter whether the clauses are combined with conjunctions or simply put in the same set of propositions. For this reason, many automatic theorem provers only deal with sets of clauses and forget about conjunctions altogether.

Also, because clauses are in prenex normal form and skolemised, there are no existential quantifiers and the scopes of the universal quantifiers always include a whole clause, making quantifiers superfluous.

But these considerations are just a sign of a deeper significance of conjunctive normal form. Namely, CNF gives a simple theory of what kind of propositions there actually *are*. Since every proposition can be expressed in CNF, we can study the properties of all propositions by only studying different kinds of sets of clauses. It is also much more simple to algorithmically generate clauses than whatever propositions. Since CNF is equivalent to full **FOL** without identity, this means that generation of clauses suffices: any other proposition is at best an abbreviation for a set of clauses.

## 5.4 $\theta$-subsumption

In order to systematically investigate the space of possible propositions, we need a means to produce, from a given proposition $p$, the propositions that are more general than $p$ and the propositions that are more specific than $p$. This is because, we need a way to strengthen the generalisations that are too weak (to be informative) and weaken the generalisations that are too strong (to be plausible).

If we work in CNF, a proposition $p$ can be factorised into clauses $C_1$, $C_2$, $C_3$, ... Then, generalisation or specialisation of $p$ is reduced to the generalisation or specialisation of these clauses. Unfortunately, the generality relation given by implication is problematic, because implication between clauses (and consequently, propositions) is undecidable. This means that there is no algorithm that would, in finite time, calculate for any two given clauses $C$ and $D$ whether $C \models D$. See [SS88] for the proof.

There is, however, a much simpler relation between clauses that resembles implication. It was developed by Plotkin [Plo71] and was also shown not to be equivalent to implication in the same thesis. This relation is called the *$\theta$-subsumption* of clauses.

**Definition 5.23** ($\theta$-subsumption)**.** A clause $C$ $\theta$-subsumes another clause $D$ (written as $C \preceq_\theta D$) if and only if some subset of the literals of $D$ is a more particular instance of $C$, i.e. it can be obtained by substituting the variables in $C$ by other terms. This can be formalised as:

$$C \preceq_\theta D \equiv \exists\theta(C\theta \subseteq D) \tag{40}$$

where $C$ and $D$ are represented as sets of literals and $\theta$ is a substitution of variables by other terms.

**Example 5.24.**

$$\begin{aligned} &\forall x \forall y \mathrm{Loves}(x,y) \preceq_\theta \forall x(\mathrm{Loves}(\mathrm{Panu},x) \vee \mathrm{Loves}(\mathrm{Igoemon},x)) \\ \equiv &\{\mathrm{Loves}(x,y)\}\theta \subseteq \{\mathrm{Loves}(\mathrm{Panu},x), \mathrm{Loves}(\mathrm{Igoemon},x)\} \end{aligned} \tag{41}$$

These are true since there are two substitutions $\theta$ that make the latter claim true, namely $\theta = [x/\mathrm{Panu}][y/x]$ and $\theta = [x/\mathrm{Igoemon}][y/x]$. Of course, even one substitution would suffice.

How exactly does $\theta$-subsumption "resemble" implication? It can be easily seen that whenever $C \preceq_\theta D$, then $C \models D$. The proof can be found in e.g. [IA93, 21]. However, the converse does not hold. This means that $\theta$-subsumption is a strictly more specific (i.e. stronger) relation than implication. Which are then the cases where a clause $C$ implies another but does not $\theta$-subsume it?

**Example 5.25** (Implication and $\theta$-subsumption)**.** These clauses form a classic example:

$$\begin{aligned} C &= \forall x(P(x) \vee \neg P(f(x))) \\ D &= \forall x(P(x) \vee \neg P(f(f(x)))) \end{aligned} \tag{42}$$

It can now be seen that $C$ implies $D$ but does not $\theta$-subsume it.

The general answer as to which implications $\theta$-subsumption "misses" can be found in [Mug92b]. Basically, $\theta$-subsumption fails to cover those implications whose proof uses a clause *recursively*.

**Definition 5.26** (Recursive use of a clause)**.** In order to define what is a recursive use of a clause, we need to make use of the inference rule of *resolution*, often used in artificial intelligence (see section 2). A clause is used recursively if and only if in a proof, it is resolved against itself, either directly ($p \vee r$ and $\neg r \vee q$ are instances of the same clause) or indirectly (a clause is resolved against a clause that has been produced by resolution against the same clause).

It follows from this property of $\theta$-subsumption that only the specialisations and generalisations of *recursive clauses* are potentially not captured by $\theta$-subsumption. A recursive clause is one that can be resolved against itself.

**Example 5.27.** In example 5.25, the clause $C$ is recursive because it has an instance (namely, $\forall x(P(f(x)) \vee \neg P(f(f(x))))$) that can be resolved against itself. This produces:

$$\frac{\forall x(P(x) \vee \neg P(f(x))) \qquad \forall x(P(f(x)) \vee \neg P(f(f(x))))}{\forall x(P(x) \vee \neg P(f(f(x)))}, \tag{43}$$

that is, $D$, which itself is also recursive.

**Definition 5.28** (Recursive clauses)**.** A clause $C$ is recursive if and only if it contains literals $S \in C$ and $T \in C$ such that $S$ and $\neg T$ are *unifiable*, that is, there are substitutions $\theta_1$ and $\theta_2$ of variables into terms such that

$$S\theta_1 = (\neg T)\theta_2. \tag{44}$$

The good news is that $\theta$-subsumption can be used to reliably identify, for non-recursive clauses, all clauses that are more general or more specific. This is of great help in algorithmic generalisation and specialisation, because of the following theorem.

**Theorem 5.29.** *A clause $C$ may be generalised, under $\theta$-subsumption, in the following ways:*

1. *by substituting some (but not all) occurrences of a variable $\alpha$ in $C$ by a new variable $y$ not previously occurring in $C$;*

2. *by substituting some or all of the occurrences of a non-variable term $t$ in $C$ by a new variable $x$ not previously occurring in $C$; and*

3. *by dropping literals from $C$.*

*Proof.* Let $C$ and $D$ be clauses such that $C \preceq_\theta D$ but $D \not\preceq_\theta C$. The first method of generalisation corresponds to a substitution $\theta = [x/y]$ where the variable $y$ that is already used in the clause $D$. Since $D$ already has occurrences of of $y$, there is no converse substitution, so $C$ strictly $\theta$-subsumes $D$.

The second method of generalisation corresponds to a substitution $\theta = [x/t]$ where $t$ is not a variable. Since the substitution in $\theta$-subsumption may only substitute variables, there is no converse substitution, so $C$ strictly $\theta$-subsumes $D$.

The third method of generalisation simply describes a situation where $\theta$ is a null substitution and $C$ is a proper subset of $D$. Consequently, $D$ cannot be a subset of $C$, so $C$ strictly $\theta$-subsumes $D$. □

It is also interesting to note the following property.

**Theorem 5.30.** *For any clause $C$, there is a finite number of clauses $C'$, modulo variable renaming, which are more general than $C$ by $\theta$-subsumption ($C' \preceq_\theta C$).*

*Proof.* I prove that there is a finite number of ways to generalise a clause $C$ under $\theta$-subsumption (up to variable renaming). The number of literals in a clause is finite, so the number of subsets of $C$ is finite. The number of different terms in a subset $C' \subseteq C$ is finite too, and the number of subsets of occurrences of these different terms in $C'$ is finite. Now, partitioning every set of occurrences of different terms in $C'$ into subsets that are left alone or substituted with new variables, there is a finite number of ways to make these choices, except for picking the names of the new variables. □

Conversely, a clause $C$ can be specialised under $\theta$-subsumption by:

1. substituting the occurrences of a variable $x$ in $C$ by some term $t$ (possibly already occurring in $C$);[8] and

2. by adding literals to $C$.

We return to the use of $\theta$-subsumption in search of generalisations in section 6.6.

## 5.5 Two types of induction

Hempel's conditions in [Hem45] are by no means the only conditions that have been offered for induction (or confirmation). Hempel himself briefly discusses the converse consequence condition that is incompatible with his other conditions, and numerous other conditions have been offered elsewhere. So many, in fact, that it seems unclear what was meant by induction in the first place.

Peter Flach [Fla95, 16–26] made the distinction between two kinds of induction, explanatory and confirmatory induction, and gave sets of conditions for

[8] If $t$ did not already occur in $C$, then $t$ may not be a variable.

both. Zwirn and Zwirn [ZZ96] went even further, proving the mutual relations of thirteen different conditions for confirmation, especially mutual inconsistencies and cases where one condition is a weakened version of another. The result on Zwirns' research was three maximally consistent subsets of conditions which correspond to three different kinds of induction (or confirmation). Two of them correspond to Flach's explanatory and confirmatory induction, called relative and absolute confirmation by Zwirn and Zwirn, and the third one seems almost unresearched.

This distinction between two (or three) kinds of induction is surprising, as lots of work on induction has been made without noticing the difference. The essential component of explanatory induction is the converse consequence condition, while the essential component of confirmatory induction is the consequence condition 3.2. It should be noted that no practical implementation of induction strictly meets the requirements of explanatory induction, because the converse consequence condition requires that if a theory has any inductive consequence at all, then it has also absurdity (and inconsistent clause) as its inductive consequence. The conditions of confirmatory induction, on the other hand, are met by such relations as deductive consequence, so they can hardly be considered sufficient conditions of inductive consequence relations.

However, the converse consequence relation and the consequence relation give different foci for the inductive process, which could be called the *compression-oriented* and *information-oriented* views of induction respectively. We shall now describe them briefly.

In compression-oriented induction, we are concerned with the problem of inverting logical entailment. This means that given a proposition $p$, we search for all propositions $q \models p$. Since the induced proposition $q$ logically entails $p$, it can be said to stand for $p$. Because many propositions $q$ meet this condition, the ones that are deemed "good" are those that express the facts in $p$ most succinctly. This is what I mean by the compression orientation of this kind of induction. Usually this compression also results in some kind of predictive power, since as a side product, $q$ usually covers some cases not covered by $p$.

Information-oriented induction, on the other hand, is concerned with inducing, for a given proposition $p$, just any proposition $q$ that is logically compatible with $p$. Since $q$ does not try to express any facts, the succinctness of $q$ is hardly any criterion of its goodness. Quite on the contrary, $q$ is often expected to be as informative as possible: to make as strong a claim as possible. An upper limit to $q$ is given by some kind of criterion of plausibility, such as the consistency condition 3.3.

Practically all work in inductive logic programming (**ILP**), for instance, has been guided by compression-oriented induction, while the system I presented in [Kal07] and a constraint search system in Flach's [Fla95, 131–154] fall to the category of information-oriented induction. From the information-oriented point of view, the work in **ILP** is further confused by the fact that the usual setting of **ILP** is the Prolog language, which saturates every theory by treating every proposition whose truth value is unknown as false.[9] In such an environment, all theories are implicitly maximally informative, so there is little to be done for information-oriented induction.

This distinction gives perspective why the account of induction presented

[9] This is called *negation as failure*.

here is quite different from other work on algorithmic induction. The goals of compression-oriented induction and information-oriented induction simply differ. A lot of background research, however, is relevant to both kinds of induction; for instance, the problem of inverting logical entailment is relevant to both.

# 6 Version spaces applied to logic

The version space model was originally developed for generalisation of concepts, but logic is not only about concepts. Instead, in the case of induction, we will deal with generalisation of *statements*. It is not immediately obvious exactly what the concepts are that we try to generalise when we think inductively.

## 6.1 Version spaces of truth

The approach taken in this thesis is to construct induction as a system of generalising the *concept of truth* (and falsehood). This corresponds to a world-view where known facts such as observations are *examples of truth*, and their negations, examples of falsehood. From these examples, inductive generalisation produces broader claims about what is true and what is false; optimally, we would have a saturated theory where, for every imaginable claim about the world, we would have a prediction of its truth.

**Example 6.1.** Let us have an instance (i.e. observation) that the sun is bright. This can be conceptualised as Bright(Sun) and states an example of what is true. Then $\forall x\text{Bright}(x)$ is a generalisation that extrapolates our example of truth to other propositions. However, if we had an observation $\neg\text{Bright}(\text{Moon})$, then the generalisation would be overly broad because it extends the concept of truth to a claim that is false, namely Bright(Moon).

This is by no means the only choice for interpretation of the word "generalisation" with respect to logic. As mentioned in section 5.1, one natural interpretation of *instance of a proposition* is a model of the proposition, that is, a possible world where the proposition is true. In this setting, the generalisation problem would mean an algorithm that takes a set of *actual* possible worlds and a set of *counterfactual* possible worlds, and produces a theory which is true in all of the actual possible worlds while being false in all of the counterfactual possible worlds. However, while such an activity seems sensible, it arguably does not describe induction. This is because in induction, we usually have incomplete information about the world, but giving whole models as input already requires us to stipulate the total state of the world. It would, however, probably be worthwhile to study this problem of generalisation and its utility. This is outside the scope of this thesis.

## 6.2 Generalisation of truth as a version space problem

Let us now formalise the problem of generalisation of truth. In this problem, our given knowledge, which is obtained from observations or is treated as unquestionable for some other reason, is represented by propositions, usually ground propositions. This knowledge simultaneously defines both $I_+$ and $I_-$, since the examples of falsehood are the negations of the examples of truth. If we denote our given knowledge by $O$, this gives the following definition.

**Definition 6.2** (Instance sets for generalisation in logic)**.** The set of positive instances is the set of propositions in our given knowledge, and the set of negative instances are their negations.

$$\begin{aligned} I_+ &= O \\ I_- &= \{\neg S : S \in O\} \end{aligned} \tag{45}$$

The generalisation is a theory, that is, a set of clauses $\Gamma$ matching our observations and not matching their negations. Note that in the case of logic, we use the same language for both instances and generalisations.[10] **FOL** allows using the same language for both, because a fragment of **FOL** is a good fit for representing observations: the set of nondisjunctive ground clauses.

**Example 6.3.** Let us have an observation that Tweety is a sparrow, Croaky is a raven, and Croaky is not a sparrow. Then a correct generalisation of our observations is a theory that matches all instances in $I_+$ and does not match any instance in $I_-$, given below.

$$\begin{aligned} I_+ &= \{\text{Sparrow}(\text{Tweety}), \text{Raven}(\text{Croaky}), \neg\text{Sparrow}(\text{Croaky})\} \\ I_- &= \{\neg\text{Sparrow}(\text{Tweety}), \neg\text{Raven}(\text{Croaky}), \text{Sparrow}(\text{Croaky})\} \end{aligned} \tag{46}$$

What, then, is the meaning of *matching* in this setting? There are a couple of choices. One obvious one is that $M(\Gamma, S)$ if and only if $\Gamma$ implies $S$. This choice is motivated by the observation that in these situations, $\Gamma$ can stand for $S$ when we talk about what is true, because $\Gamma$ contains all the information about $S$. But other choices are possible. We could, for instance, make a broader definition of matching, requiring $\Gamma$ only to be consistent with $S$. This would allow

$$M(\text{Sparrow}(\text{Tweety}), \text{Human}(\text{Socrates})) \tag{47}$$

We could also go in the narrower direction, saying that $M(\Gamma, S)$ if and only of $S$ is a ground instance of some proposition in $\Gamma$. This would make

$$\begin{aligned} \neg M(&\text{Human}(\text{Socrates}) \wedge \forall x(\text{Human}(x) \rightarrow \text{Mortal}(x)), \\ &\text{Mortal}(\text{Socrates})) \end{aligned} \tag{48}$$

As the choice of the interpretation of matching is largely a matter of intuition when a proposition "includes" another, I will take the liberty to define matching as logical entailment.

**Definition 6.4** (Matching in logic)**.** If $S$ is a sentence in logic and $\Gamma$ is a theory in logic, then $\Gamma$ matches $S$ if and only if $\Gamma$ logically entails $S$.

$$M(\Gamma, S) \equiv (\Gamma \models S). \tag{49}$$

**Example 6.5.** Let us use the observation in example 6.3. Then,

$$\begin{aligned} \Gamma = \{&\text{Sparrow}(\text{Tweety}), \text{Raven}(\text{Croaky}), \text{Invisible}(\text{Croaky}), \\ &\forall x(\neg\text{Sparrow}(x) \vee \neg\text{Raven}(x))\} \end{aligned} \tag{50}$$

is a generalisation of our observation, because it implies all sentences in $I_+$ while implying none of those in $I_-$.

Let us then look at the whole problem of induction as expressed as a generalisation problem.

**Definition 6.6** (Induction algorithms)**.** An induction algorithm is one that satisfies the following criteria:

[10] This is called the "single representation trick".

**input** The input of the algorithm is a set of propositions $O$.

**output** The output of the algorithm is a set of propositions $\Gamma$ for which the following conditions hold:

1. $\forall S(S \in O \rightarrow (\Gamma \models S))$ and
2. $\forall S(S \in O \rightarrow (\Gamma \not\models \neg S))$.

**Theorem 6.7.** *If we work in classical logic, every generalisation $\Gamma$ is simply a consistent theory implying all propositions in $O$.*

*Proof.* From the first condition on output in definition 6.6, we know that for all sentences $S \in O$, $\Gamma \models S$. Now, if the second condition did not hold, there would be a sentence $S \in O$ for which $\Gamma \models \neg S$. But then $\Gamma$ would entail both the affirmation and negation of a sentence, making $\Gamma$ inconsistent. On the other hand, if $\Gamma$ is inconsistent, it entails all sentences, negations of sentences $S \in O$ included. This completes the proof that the second condition is equivalent with saying that $\operatorname{Mod}\Gamma$. □

This lets us devise the following definition of generalisation which can be decomposed into two conditions that every generalisation must satisfy:

**Definition 6.8** (Generalisations)**.** A generalisation $\Gamma$ of a set of propositions $O$ is a consistent set of propositions that logically entails all propositions in $O$. Using $\rightsquigarrow$ as the sign for "has a generalisation of", we can write this in the following way:

$$(O \rightsquigarrow \Gamma) \equiv \forall S(S \in O \rightarrow (\Gamma \models S)) \wedge \operatorname{Mod}\Gamma \tag{51}$$

**Condition 6.9** (Entailment of generalisations)**.** Every generalisation $\Gamma$ of a set of propositions $O$ logically entails all sentences in $O$.

$$(O \rightsquigarrow \Gamma) \rightarrow \forall S(S \in O \rightarrow (\Gamma \models S)) \tag{52}$$

**Condition 6.10** (Consistency of generalisations)**.** Every generalisation $\Gamma$ of every set of propositions $O$ is consistent.

$$(O \rightsquigarrow \Gamma) \rightarrow \operatorname{Mod}\Gamma \tag{53}$$

Interestingly, even if we did take the more liberal approach of defining matching as mutual consistency, we would get the same result:

**Theorem 6.11.** *If matching is defined as*

$$M(\Gamma, S) \equiv \operatorname{Mod}(\Gamma \cup \{S\}) \tag{54}$$

*then $\Gamma$ is a generalisation of $O$ if and only if it is consistent and implies $O$.*

*Proof.* In order to not match propositions in $I_-$, $\Gamma$ will have to be inconsistent with them. This means that $\Gamma$ must imply their negations. However, the set of negations of $I_-$ is the same as the set of our original observations $O$. Also, to be consistent with $I_+$, $\Gamma$ will have to be consistent itself. Thus, the matching requirements translate to conditions 6.9 and 6.10. □

This provides vague support for the view that conditions 6.9 and 6.10 are fundamental for this kind of generalisation in logic. In the following section, I will compare these conditions with Hempel's criteria on confirmation.

## 6.3 Logical version spaces and Hempel's conditions

If we now compare the version space based notion of generalisation with Hempel's conditions in section 3.1, we notice interesting similarities and dissimilarities. But in order to compare the definitions, we need to first define the relation between the relations "confirms" ($\mid\!\sim$) and "has a generalisation of" ($\rightsquigarrow$).

**Definition 6.12** (Inductive closure)**.** As mentioned in the section 3.1, a confirmation relation gives rise to an inductive closure defined by

$$IC(E) = \{H : E \mid\!\sim H\}. \tag{55}$$

I will take the approach that the inductive closure is a generalisation of $E$. This means that confirmation relations and generalisations of an observation report have a one-to-one correspondence.

For now, I will simply presume that there is a correct generalisation for every consistent set of prior knowledge, and that an observation report confirms every sentence entailed by this correct generalisation. In section 6.4, I show that every consistent set of prior knowledge has at least one generalisation.

**Definition 6.13** (Relation of confirmation and generalisation)**.** If a set of propositions $O$ is consistent, then it has a unique generalisation $\Gamma$ called the correct generalisation of $O$, which logically entails all and only sentences confirmed by $O$.

$$\forall O(\operatorname{Mod} O \rightarrow \exists \Gamma((O \rightsquigarrow \Gamma) \wedge \forall S(O \mid\!\sim S \equiv (\Gamma \models S)))). \tag{56}$$

Having defined the relation between confirmation and generalisation in logic thus, the most obvious similarity between Hempel's conditions and our account of generalisation is the consistency condition.

**Theorem 6.14.** *Condition 6.10 and condition 3.3 are equivalent.*

*Proof.* $\rightarrow$ . If the correct generalisation $\Gamma$ of a theory $O$ has a model, then all sentences implied by $\Gamma$ are mutually consistent. Because all sentences in $O$ are implied by $\Gamma$ by condition 6.9 and all sentences confirmed by $O$ are implied by $\Gamma$ by definition 6.13, it follows that $\operatorname{Mod}(O \cup \{H : O \mid\!\sim H\})$.

$\leftarrow$ . If the inductive closure $IC(O)$ of a theory $O$ is consistent with $O$, then $IC(O)$ is also consistent by itself. Consequently, all sentences implied by the correct generalisation $\Gamma$ of $O$ are mutually consistent by definition 6.13. It follows that $\Gamma$ is consistent. □

**Theorem 6.15.** *The condition 6.9 implies Hempel's conditions 3.1 and 3.2.*

*Proof.* For 3.1, we have to show that $O \models H$ implies $O \mid\!\sim H$. Let $\Gamma$ be a generalisation of $O$. If $O \models H$, then also its generalisation $\Gamma \models H$, because $\Gamma$ implies every sentence in $O$ by condition 6.9. And if $\Gamma \models H$, then $O \mid\!\sim H$ by definition 6.13, because $\Gamma$ is a generalisation of $O$.

For 3.2, we have to show that $\forall S(S \in K \rightarrow (O \mid\!\sim S))$ and $K \models H$ together imply $O \mid\!\sim H$. By the first antecedent and 6.13, $\forall S(S \in K \rightarrow (\Gamma \models S))$, where $\Gamma$ is a generalisation of $O$. Together with $K \models H$, this gives that $\Gamma \models H$ too. Again, by definition 6.13, then $O \mid\!\sim H$. □

## 6.4 Minimal and maximal generalisations

It is easy to note that by this account of generalisation, every consistent theory is its own generalisation. This is because for every consistent set of propositions $O$,

$$\forall S(S \in O \rightarrow (O \models S)) \wedge \operatorname{Mod} O \tag{57}$$

so $O$ itself satisfies the conditions 6.9 and 6.10, and consequently $O \rightsquigarrow O$. This is important because it means that every consistent set of observations has at least one generalisation. On the other hand, no inconsistent theory has a generalisation at all.

**Theorem 6.16.** *If theory $O$ is inconsistent ($\neg \operatorname{Mod} O$), no theory $\Gamma$ is a generalisation of the theory $O$.*

*Proof.* If a theory $O$ is inconsistent, then it claims for some (or equivalently, every) proposition $p$ that $p \wedge \neg p$. This means that its generalisation $\Gamma$ must also imply $p \wedge \neg p$. But then $\neg \operatorname{Mod} \Gamma$, so no $\Gamma$ can satisfy $O \rightsquigarrow \Gamma$. □

For this reason, I presume in this thesis that every theory to be generalised is consistent: generalisation is in any case an operation that is only defined for consistent input data. It would probably be quite possible to define generalisation for inconsistent input by working in paraconsistent logic; this is outside the scope of this thesis.

A related fact worth noticing is that not only is a consistent theory its own generalisation, it is also the unique minimal generalisation for itself. This is also important because it means that the minimal generalisation is uninteresting from induction's point of view: it is the traditional notion of deductive consequence.

**Theorem 6.17.** *Every theory $O$ is the only minimal generalisation of itself.*

$$\forall O \forall \Gamma((O \rightsquigarrow \Gamma) \rightarrow (\Gamma \sqsupseteq O)) \tag{58}$$

*Proof.* This can be demonstrated by showing that all generalisations of a consistent theory $O$ are more general than $O$. We recall from definition 5.11 that $g_1 \sqsupseteq g_2 \equiv \forall i(M(g_2, i) \rightarrow M(g_1, i))$. For our definition of matching, this reads as

$$\Gamma \sqsupseteq O \equiv \forall S((O \models S) \rightarrow (\Gamma \models S)). \tag{59}$$

Now, if a sentence $S$ is implied by $O$ (i.e. $O \models S$) then $\Gamma \models S$ because $S$ is implied by some subset of $O$ and $\Gamma$ implies all propositions in that subset by definition 6.8. It follows that $\Gamma$ is indeed more general than $O$ if $O \rightsquigarrow \Gamma$. □

If the minimal generalisation is relatively uninteresting, then how about the maximal generalisation(s)? It turns out that the situation is not at all so simple there.

**Definition 6.18** (Maximally consistent extension)**.** A theory $\Gamma$ is a maximally consistent extension of another theory $O$ if and only if $\Gamma$ entails all propositions in $O$, $\Gamma$ is consistent and for each proposition $p$, either $p \in \Gamma$ or $\neg \operatorname{Mod}(\Gamma \cup \{p\})$.

$$\begin{aligned} MCE(\Gamma, O) \equiv & \forall p(p \in O \rightarrow (\Gamma \models p)) \wedge \operatorname{Mod} \Gamma \wedge \\ & \forall p(\operatorname{Mod}(\Gamma \cup \{p\}) \rightarrow p \in \Gamma) \end{aligned} \tag{60}$$

**Lemma 6.19.** *The maximal generalisations of $O$ are exactly the maximally consistent extensions of $O$.*

*Proof.* The first two conditions of maximally consistent extensions are equivalent with conditions 6.9 and 6.10. A generalisation $\Gamma'$ is strictly more general than $\Gamma$ if and only if there is a proposition $p$ for which $\Gamma' \models p$ but $\Gamma \not\models p$. Since $p \notin \Gamma$, $\Gamma'$ is inconsistent by definition 6.18, so it is not a generalisation of $O$. Thus, the third condition is equivalent with saying that there are no strictly more general generalisations of $O$. □

**Lemma 6.20.** *If $\Gamma$ is a maximally consistent extension of some theory, then for each proposition $p$, either $p \in \Gamma$ or $(\neg p) \in \Gamma$, but not both.*

*Proof.* $\Gamma$ is consistent by definition 6.18, so it cannot contain both $p$ and $\neg p$ for any proposition $p$. Now, assume $\Gamma$ does not contain either. Then if $p$ is consistent with $\Gamma$, $p \in \Gamma$ so the assumption is false. On the other hand, if $p$ is inconsistent with $\Gamma$, then $\Gamma$ must entail $\neg p$. Since $\Gamma$ is consistent and entails $\neg p$, $\Gamma \cup \{\neg p\}$ is also consistent, and consequently $(\neg p) \in \Gamma$, so the assumption is false. □

**Theorem 6.21.** *A proposition $p$ belongs to every maximally consistent extension $\Gamma$ of a theory $O$ if and only if $O \models p$.*

*Proof.* If $O \not\models p$, then $\mathrm{Mod}(O \cup \{\neg p\})$. Because the maximally consistent extensions of $O \cup \{\neg p\}$ are also maximally consistent extensions of $O$ and cannot contain proposition $p$, not all maximally consistent extensions of $O$ contain proposition $p$ in that case. If, on the other hand, $O \models p$, then $\neg\,\mathrm{Mod}(O \cup \{\neg p\})$, so there are no maximally consistent extensions of $O \cup \{\neg p\}$ as all its extensions are inconsistent. Consequently, $p$ belongs to all of the maximally consistent extensions of $O$ in this case. □

In the view of this, the maximal generalisations of $O$ are also quite irrelevant from the point of view of induction. They are very numerous since every proposition $p$ that is not implied nor contradicted by $O$ divides the maximal generalisations into two sets: those where $p$ holds and those where $\neg p$ holds. The only propositions that are common to all maximal generalisations of $O$, that is, the propositions in the intersection of all maximal generalisations of $O$, are those implied by $O$. This means that maximal generalisations of $O$ have a high degree of arbitrariness.

## 6.5 Choosing the correct generalisation

Since the minimal generalisation of a theory is uninteresting and the maximal generalisations make arbitrary decisions about the truth value of various propositions, it is necessary to develop further criteria that restrict the generalisations to those we would consider *plausible* or *sensible*. In this thesis, I take the stance that the "correct" generalisation of a theory $O$ is the strongest generalisation that is plausible. This corresponds to the view that induction has a double goal: that of producing as *many* predictions as possible (informativeness) while minimising the likelihood that these predictions will turn out to be false (plausibility).

**Example 6.22.** In example 6.5, the clause Invisible(Croaky) is not plausible since it has nothing to support it in our observations.

In section 7, I propose restrictions on the kind of clauses that a generalised theory may contain. These restrictions can also be viewed as restrictions on generalisations, because they outrule those generalisations that contain clauses forbidden by our criteria. Among the allowed generalisations, the "correct" one is then the maximal one. I'll call this generalisation the *maximal plausible generalisation* (MPG). The MPG of a theory $O$ can be formed by extending the theory $O$ with all plausible clauses. The MPG is more formally defined in section 7.2.

## 6.6 Search in the generalisation space

There remain two domain-specific requisites for being able to apply the version space theory to **FOL**. First, we need to be able to generate some initial generalisation(s) to start the search from, and second, we need a means to generalise overly specific generalisations and to specialise overly general generalisations. The first requisite is easily covered since every theory is its own generalisation; we can always start the search for generalisations of a theory $O$ from $O$ itself. But how is the second requisite satisfied?

The answer is $\theta$-subsumption (see section 5.4), which gives a straightforward way to generalise and specialise (nonrecursive) clauses. $\theta$-subsumption can also be used for recursive clauses; the only possible harm in doing so is the risk of missing a more minimal generalisation or specialisation.

Recall from section 5.4 that a clause $C$ can be generalised by substituting a subset of occurrences of some term $t$ in $C$ by a new variable or dropping literals from $C$, and specialised by substituting all occurrences of a variable $x$ in $C$ with some term $t$ or adding literals to $C$. This means that there are, in a sense, two dimensions of generality: the generality of *terms* and the generality of *disjunctions*. The generality of terms increases when more specific terms are replaced by variables and decreases when variables are replaced by more specific terms, whereas the generality of disjunctions increases when disjuncts are discarded and decreases when new disjuncts are added.

**Example 6.23.** Consider the clauses

$$\begin{aligned} C &= \forall x(\text{Pretty}(x) \vee \neg\text{Pretty}(\text{MotherOf}(x))) \\ D_1 &= \forall x \neg\text{Pretty}(\text{MotherOf}(x)) \\ D_2 &= \forall y \forall x(\text{Pretty}(x) \vee \neg\text{Pretty}(y)) \\ D_3 &= \forall x(\text{Pretty}(x) \vee \neg\text{Pretty}(\text{MotherOf}(x)) \vee \neg\text{Female}(x)) \\ D_4 &= \forall x(\text{Pretty}(\text{SpouseOf}(x)) \vee \neg\text{Pretty}(\text{MotherOf}(\text{SpouseOf}(x)))) \end{aligned} \tag{61}$$

Then, $D_1$ and $D_2$ are generalisations of $C$ under $\theta$-subsumption, whereas $D_3$ and $D_4$ are specialisations of $C$ under $\theta$-subsumption ($D_1 \preceq_\theta C$, $D_2 \preceq_\theta C$, $C \preceq_\theta D_3$ and $C \preceq_\theta D_4$).

It is interesting to note that if the initial theory $O$ is a typical observation report, it starts at the most specific extreme in one of these dimensions but at the most general extreme in the other. Observation reports contain no variables, since they only express facts about particular individuals, so their terms are as

specific as possible; but they contain no disjunctions either, because the facts they express are not conditional, so their disjunctions are as general as possible — excluding the empty clause, whose presence would make the observation report inconsistent. This naturally suggests the following strategy for searching the generalisations of a theory $O$: starting from $O$, generalise the clauses of $O$ by generalising their terms; when the clauses become strong enough to be inconsistent with $O$, specialise them by adding disjuncts.

# 7 Plausible clauses

In this section, I will introduce further conditions on generalisations: the conditions of *plausibility*. Intuitively, a generalisation is plausible if and only if we have some reason to believe everything in the generalisation. Since even one clause can render a generalisation implausible, the conditions of plausibility of generalisations translate to conditions on clauses. Accordingly, I shall in this section define the conditions of plausibility of clauses.

**Definition 7.1** (Plausible generalisations)**.** A generalisation is plausible if and only if all clauses in the generalisation are plausible. We will use $Pl(\Gamma, O)$ to denote that $\Gamma$ is plausible with respect to the set of propositions $O$, and $Pl(C, O)$ to denote that a clause $C$ is plausible with respect to $O$.

$$Pl(\Gamma, O) \equiv \forall C(C \in \Gamma \rightarrow Pl(C, O)) \tag{62}$$

**Condition 7.2** (Plausibility)**.** The correct generalisation $\Gamma$ of a theory $O$ is plausible with respect to $O$. So, for all theories $O$ and all propositions $H$,

$$(O \mathrel{|\!\sim} H) \rightarrow Pl(H, O) \tag{63}$$

Plausibility of clauses is defined gradually as we build the necessary conceptual machinery for defining plausibility.

## 7.1 Plausibility: why and how?

In section 6.4, it was proved that maximal generalisations have a high degree of *arbitrariness*: for each proposition which is neither implied by nor inconsistent with our observations, a maximal generalisation will include either the proposition or its negation, but of course not both, since that would make the generalisation inconsistent. Due to this arbitrariness, the maximal generalisations provide few hints about the intuitive meaning of generalisation.

**Example 7.3.** Consider the observational theory $O = \{Pa, Pb\}$. It is universally accepted as an example of induction that the clause $\forall x Px$ generalises $O$, which can be expressed in our notational conventions:

$$\begin{aligned} &\Gamma = O \cup \{\forall x Px\} \\ &O \rightsquigarrow \Gamma \end{aligned} \tag{64}$$

However, with our definition of generalisation, $\Delta = O \cup \{\neg Pc\}$ is also a generalisation of $O$. As $\Gamma$ and $\Delta$ are mutually inconsistent, no maximal generalisation will have both as subsets. If we were to choose $\Gamma$ or $\Delta$ as the generalisation of $O$, we would probably prefer $\Gamma$. But on what grounds?

The conditions of plausibility are an attempt to develop criteria to prefer one generalisation to another. Viewed from another angle, plausibility is a criterion to prefer one *clause* to another, since by definition 7.1, the plausibility of generalisations is defined by the plausibility of clauses. Plausibility of clauses, in turn, can be defined in terms of the preference of clauses to each other. The idea is that in a set of clauses that is minimally inconsistent with our prior knowledge, the least preferred clause(s) are the implausible one(s).

**Definition 7.4** (Preference relations)**.** Let us have a weak order relation *preferred to* on clauses,[11] written as $p \geq_{\text{pl}} q$. The corresponding strict ordering relation *strictly preferred to* is written as $p >_{\text{pl}} q$, and the equivalence relation *equally preferable with* is written $p =_{\text{pl}} q$. The following hold for all $p$, $q$ and $r$:

$$\begin{gathered} p \geq_{\text{pl}} p \\ p \geq_{\text{pl}} q \wedge q \geq_{\text{pl}} r \rightarrow p \geq_{\text{pl}} r \\ p >_{\text{pl}} q \equiv p \geq_{\text{pl}} q \wedge \neg q \geq_{\text{pl}} p \\ p =_{\text{pl}} q \equiv p \geq_{\text{pl}} q \wedge q \geq_{\text{pl}} p \end{gathered} \tag{65}$$

**Definition 7.5** (Plausible clauses)**.** Given the $\geq_{\text{pl}}$ relation, a clause $C$ is implausible with respect to a theory $O$ if and only if there is a set of clauses $K$ for which the following conditions hold:

1. $K \cup O$ is consistent (and consequently, $K$ is consistent).
2. $K \cup O \cup \{C\}$ is inconsistent.
3. Every clause in $K$ is preferred to $C$.

Naturally, a clause $C$ is plausible if and only if $C$ is not implausible. In precise notation:

$$\begin{aligned} Pl(C, O) &\equiv \neg\exists K(\text{Mod}(K \cup O) \wedge \neg\,\text{Mod}(K \cup O \cup \{C\}) \wedge \\ &\quad \forall D(D \in K \rightarrow D \geq_{\text{pl}} C)) \\ &\equiv \forall K(\text{Mod}(K \cup O) \wedge \neg\,\text{Mod}(K \cup O \cup \{C\}) \rightarrow \\ &\quad \exists D(D \in K \wedge \neg D \geq_{\text{pl}} C)) \end{aligned} \tag{66}$$

If such a set $K$ exists, it is called a *witness* of the implausibility of $C$.

It is interesting to note that the preference relation $\geq_{\text{pl}}$ is *not* relative to the theory $O$ being generalised. I believe that there are good reasons to universally prefer certain clauses in generalisations to others; intuitively, the role given to $O$ by definition 7.5 is that $O$ filters away from this preference hierarchy those clauses that are inconsistent with $O$. The effects of different preference relations are discussed in section 7.3.

**Theorem 7.6.** *All clauses are plausible with respect to an inconsistent theory $O$.*

*Proof.* If $\neg\,\text{Mod}\,O$, then no set $K$ is consistent with $O$. Consequently, for any clause $C$, there is no set $K$ that would witness the implausibility of $C$. □

Accordingly, implausibility is interesting only for consistent reference theories, i.e. consistent theories that we are generalising from. From here on, we will presume that the reference theory $O$ is consistent.

**Theorem 7.7.** *Every clause $C$ that is inconsistent with a (consistent) theory $O$ is implausible with respect to $O$.*

*Proof.* If $\neg\,\text{Mod}(\{C\} \cup O)$, then the empty set $\{\}$ is a witness of the implausibility of $C$, since it satisfies the conditions of implausibility in definition 7.5. □

[11] Any reflexive and transitive binary relation is a weak order.

**Theorem 7.8.** *Every clause $C$ that is entailed by a (consistent) theory $O$ is plausible with respect to $O$.*

*Proof.* Since $O \models C$, then if $K \cup \{C\}$ is inconsistent with $O$, $K$ is also inconsistent with $O$ as $C$ is redundant. It follows that no set $K$ can be a witness of the implausibility of $C$. Consequently, $C$ is plausible. □

**Definition 7.9** (Plausible closure)**.** The *plausible closure* of a theory $O$ (denoted $PlC(O)$) is the set of clauses that are plausible with respect to $O$.

$$PlC(O) = \{C : Pl(C, O)\} \tag{67}$$

**Theorem 7.10.** *If the preference relation $\geq_{\text{pl}}$ is a total order, then $PlC(O)$ is consistent with $O$.*[12]

*Proof.* Suppose the plausible closure $PlC(O)$ is inconsistent with $O$. Then there is at least one set $K \subseteq PlC(O)$ which is minimally inconsistent with $O$, meaning that all proper subsets $K' \subset K$ are consistent with $O$. Since $\geq_{\text{pl}}$ is a total order, there is at least one clause $C \in K$ that is *minimal* with respect to $K$, which means $\forall D(D \in K \rightarrow D \geq_{\text{pl}} C)$. This clause $C$ is implausible, because $K$ is inconsistent with $O$, $K \setminus \{C\}$ is consistent with $O$ (as $K$ is minimally inconsistent with $O$) and all clauses in $K \setminus \{C\}$ are preferred to $C$.

Since $C \in K \subseteq PlC(O)$, this is a contradiction with definition 7.9. Consequently, the assumption of the inconsistency of $PlC(O)$ with $O$ is false. □

If the preference relation is not total, then the set of all plausible clauses may well be inconsistent, since the inconsistency may be caused by clauses neither of which is preferred to the other.

**Example 7.11** (Mutually inconsistent clauses)**.** Examine the following background theory $O$ and clauses:

$$\begin{aligned} O &= \{Pa, Qa, Ra, \neg Pb, \neg Qb, \neg Rb, Pc, \neg Rc\} \\ C_1 &= \forall x(\neg Px \vee Qx) \\ C_2 &= \forall x(\neg Qx \vee Rx) \end{aligned} \tag{69}$$

If we use logical entailment as the preference relation, then both $C_1$ and $C_2$ are plausible with respect to $O$, because no clause stronger (more general) than $C_1$ is consistent with $O$ and consequently in any set $K$ that is inconsistent with $O$, no other clause is preferred to $C_1$ (and similarly for $C_2$). Moreover, neither $C_1 \models C_2$ nor $C_2 \models C_1$, so neither clause is preferred to the other one.

However, $O \cup \{C_1, C_2\}$ is inconsistent because of $Pc$ and $\neg Rc$.

## 7.2 Maximal plausible generalisation

The condition 7.2 determines an upper bound for generalisations: generalisations should never be so general as to include implausible clauses. However, induction also has the goal of producing informative theories, that is, as general theories as possible. This gives the basis for the following condition.

[12] The order relation $\geq_{\text{pl}}$ is total if and only if the following axiom holds for all $p$ and $q$:

$$p \geq_{\text{pl}} q \vee q \geq_{\text{pl}} p \tag{68}$$

**Condition 7.12** (Maximal plausibility)**.** The correct generalisation $\Gamma$ of a theory $O$ is *maximally plausible* with respect to $O$: $\Gamma$ is plausible with respect to $O$ and all strictly more general theories are implausible with respect to $O$.

In section 7.1, I proved that if the preference relation $\geq_{\text{pl}}$ is total, then all clauses that are plausible with respect to a given theory $O$ are consistent with $O$ and mutually consistent. This is very beneficial, because it means that we can extend the theory $O$ with *all* clauses that are plausible with respect to $O$ (the whole plausible closure of $O$) and get a unique maximal plausible generalisation. This gives basis for the following conditions.

**Condition 7.13** (Totality)**.** The preference relation $\geq_{\text{pl}}$ is total.

**Condition 7.14** (Completeness)**.** The correct generalisation $\Gamma$ of a consistent theory $O$ includes all clauses that are plausible with respect to $O$. For all theories $O$ and all propositions H,

$$Pl(H, O) \rightarrow (O \mid\sim H) \tag{70}$$

**Theorem 7.15.** *If the preference relation $\geq_{\text{pl}}$ is total, then the theory $\Gamma = PlC(O)$ is a generalisation of $O$.*

*Proof.* Since all clauses entailed by $O$ are plausible with respect to $O$, $\Gamma$ is a superset of $O$, so $\Gamma$ includes and consequently implies every sentence $S \in O$. As proved by theorem 7.10, $PlC(O)$ is consistent, so $\Gamma$ is consistent. It follows from definition 6.8 that $O \rightsquigarrow \Gamma$. □

**Corollary 7.16.** *If the preference relation $\geq_{\text{pl}}$ is total, then for each consistent theory $O$, there is exactly one correct generalisation $\Gamma$ that satisfies conditions of entailment (6.9), consistency (6.10), plausibility (7.2), maximal plausibility (7.12) and completeness (7.14).*

*Proof.* The theory $\Gamma = PlC(O)$ is such a generalisation by theorem 7.15 and because conditions 7.2 and 7.14 together totally determine the clauses in the correct generalisation. □

This theory $\Gamma$ is called the *maximal plausible generalisation* of $O$, or MPG for short. The MPG is the most informative generalisation that is still plausible, and so it is also the generalisation that best captures the two goals of induction, which are informativeness and plausibility. This provides strong support that the MPG of $O$ nicely captures the notion of *inductive consequence* of $O$. This also gives the definition of classificatory confirmation by definition 6.13: $O$ confirms every sentence $S$ that is entailed by the MPG of $O$.

## 7.3 Conditions for the preference relation

Obviously, the choice of the preference relation $\geq_{\text{pl}}$ totally determines the kind of plausible clauses there can be, and consequently, the kind of plausible generalisations we can obtain. However, not every kind of preference relation produces results that look like induction; some properties of preference relations are clearly desirable.

Firstly, from a syntactic point of view (that is, lacking any information about the meanings of atomic predicates), there's never reason to strictly prefer

a clause $C$ to another which is equivalent to $C$ except that one of the atomic matrices of $C$ has been negated. This is naturally because for every predicate $P$, there is another predicate $\bar{P}$ which is true for those and only those individuals for which $P$ is false. Now, if we were to strictly prefer the affirmative or the negative modality, the preference would produce exactly contrary results for $P$ and $\bar{P}$.

Interestingly, systems that make such assumptions nevertheless exist. The logic programming environments (especially Prolog) usually incorporate a feature called *negation as failure*, which means that if the truth value of a clause is unknown, then it is assumed to be false. Similar results can be obtained by strictly preferring all negative ground clauses to all affirmative ones, and by strictly preferring ground clauses to all non-ground clauses. As a result, negation as failure can be considered a very peculiar kind of generalisation technique.[13]

However, I will base all discussion about implausibility on the balanced choice that neither the affirmative nor the negative modality of an atomic predicate is strictly preferred to the other. The preference relation is called *symmetric with respect to negation* if it treats the affirmation and negation of an atomic sentence as equally preferable with each other. Symmetry with respect to negation is more rigorously defined in section 8.

**Condition 7.17** (Symmetry)**.** The preference relation $\geq_{\text{pl}}$ is symmetric with respect to negation.

A preference relation that strictly prefers stronger (more general) clauses to weaker (more specific) ones seems sensible because more informative generalisations are preferable over less informative ones as long as they remain plausible. This choice also seems to be supported by the preference of $\Gamma$ over $\Delta$ in example 7.3.

Another reason why it might be a good idea to strictly prefer more general clauses to less general ones is that for every clause $C$, there is a finite number of clauses that are more general by $\theta$-subsumption (see section 5.4). This guarantees that an algorithm that computes the plausibility of a clause $C$ will terminate: it will only have to check whether any subset of a finite number of clauses will contradict $C$ in order to find out whether $C$ is implausible.

However, if we are to use generality between clauses (as given by implication or $\theta$-subsumption) as the preference relation, there is a problem. Even when generalised by giving equal preference to affirmatives and negatives, the generality of clauses is not a total ordering.

**Example 7.18.** Consider the clauses:

$$\begin{aligned} C_1 &= \forall x Px \\ C_2 &= Pa \\ C_3 &= \neg Pa \\ C_4 &= Qa \end{aligned} \tag{71}$$

[13] To see that negation as failure is a generalisation technique, consider the following proof. If an affirmative clause $C$ is considered for the generalisation of a theory $O$ and $O$ does not entail $C$, there are always strictly preferred negative clauses $D_1$, $D_2$, ... which contradict $C$, making $C$ implausible. Thus, the only clauses that belong to a plausible generalisation of $O$ with this preference relation are affirmative clauses that are logical consequences of $O$ and negative clauses that don't contradict $O$. This is exactly the definition of negation as failure.

The proposed preference relation has $C_1 >_{\text{pl}} C_2$ since $C_1 \preceq_\theta C_2$, $C_2 =_{\text{pl}} C_3$ since both negative and affirmative modality are equally preferable, and $C_1 >_{\text{pl}} C_3$ by transitivity. However, the clause $C_4$ is incomparable with the other clauses.

This suggests that the preference relation should be a total ordering that honors generality of clauses.

**Definition 7.19** (Honoring)**.** A weak total ordering $\geq$ honors a weak partial ordering $\sqsupseteq$ if and only if these conditions hold for all $x$ and $y$:

$$\begin{aligned} x \sqsupseteq y \wedge y \sqsupseteq x \rightarrow x \geq y \wedge y \geq x \\ x \sqsupseteq y \wedge \neg y \sqsupseteq x \rightarrow x \geq y \wedge \neg y \geq x \end{aligned} \tag{72}$$

**Condition 7.20** (Honoring generality)**.** The preference relation $\geq_{\text{pl}}$ honors generality of clauses.

Such a total ordering can be constructed by weakening the generality of clauses by adding $C \geq_{\text{pl}} D$ or $D \geq_{\text{pl}} C$ (or both) for clauses $C$ and $D$ which are incomparable (neither $C \models D$ nor $D \models C$ is true). However, from the point of view of informativeness, it is not good to have too weak a preference relation. To get as informative MPG's as possible, it is good to have as specific a preference relation as possible.

**Theorem 7.21.** *If the preference relation $\geq^1_{\text{pl}}$ is more specific (i.e. stronger) than the preference relation $\geq^2_{\text{pl}}$, then the maximal plausible generalisations generated by $\geq^1_{\text{pl}}$ are supersets of the maximal plausible generalisations generated by $\geq^2_{\text{pl}}$.*

*Proof.* If $\geq^1_{\text{pl}}$ is more specific than $\geq^2_{\text{pl}}$, then for all clauses $C$ and $D$,

$$C \geq^1_{\text{pl}} D \rightarrow C \geq^2_{\text{pl}} D. \tag{73}$$

It follows that

$$\begin{aligned} &\exists K(\text{Mod}(K \cup O) \wedge \neg\, \text{Mod}(K \cup O \cup \{C\}) \wedge \forall C'(C' \in K \rightarrow C' \geq^1_{\text{pl}} C)) \\ \rightarrow\, &\exists K(\text{Mod}(K \cup O) \wedge \neg\, \text{Mod}(K \cup O \cup \{C\}) \wedge \forall C'(C' \in K \rightarrow C' \geq^2_{\text{pl}} C)) \end{aligned} \tag{74}$$

That is, if a clause $C$ is implausible under $\geq^1_{\text{pl}}$, then $C$ is also implausible under $\geq^2_{\text{pl}}$ by the same witness set $K$. □

Intuitively, the clauses that will be plausible by $\geq^1_{\text{pl}}$ but not by $\geq^2_{\text{pl}}$ are those that are equally preferable with some implausible clause in a minimally inconsistent set by $\geq^2_{\text{pl}}$ but not by $\geq^1_{\text{pl}}$. It is easy to see that if the preference relation is to be total, the only way to strengthen it is to treat less clauses as equally preferable with each other.

There's still one more condition to set on the preference relation $\geq_{\text{pl}}$. Usually, inductive thinking is used to produce *rules*, that is, statements that pertain to all individuals. Consequently, it makes sense to prefer universally quantified sentences over particular ones, even in the cases where they are otherwise incomparable in terms of generality.

**Condition 7.22** (Preferring rules)**.** The preference relation $\geq_{\text{pl}}$ prefers universally quantified clauses to particular ones. Less restricted value domains are also preferred to more restricted ones.

**Example 7.23.** Let us have

$$\begin{aligned} C &= \forall x Px \\ D_1 &= \forall x(Px \vee Qx) \\ D_2 &= Pa \end{aligned} \tag{75}$$

Both $D_1$ and $D_2$ are more specific than $C$ by $\theta$-subsumption (and consequently also logical entailment) but mutually incomparable. However, for generalisation, we should prefer the rule-like $D_1$ to the particular $D_2$.

**Example 7.24.** In example 6.23, the clauses $D_3$ and $D_4$ are mutually incomparable. However, the clause $D_3$ is to be preferred to $D_4$, because $D_3$ makes a claim that pertains to all individuals, while $D_4$ only affects individuals that are somebody's spouses.

In section 9, we will return to the problem of defining a preference relation that observes all these conditions and requirements.

# 8 Symmetry

The consistency condition 6.10 or Hempel's equivalent condition 3.3 give interesting results when combined with the symmetry condition 7.17. In this section, we will discuss these results.

**Lemma 8.1** (Implausibility of symmetric clauses)**.** *If two (or more) clauses are equally preferable but contradict each other, then a plausible generalisation may have neither (or none) of those clauses.*

*Proof.* Consider a set of clauses $K = \{C_1, C_2, \dots\}$ that is minimally inconsistent with a theory $O$ and where all clauses are equally preferable with each other. Then, all of those clauses are implausible, that is, $\neg Pl(C_n, O)$ for all $n$, because $K \cup O$ is inconsistent, $(K \cup O) \setminus \{C_n\}$ is consistent and $\forall D(D \in K \to D \geq_{\text{pl}} C_n)$. □

Symmetry with respect to negation is especially important, because mutual negations are inconsistent. But which clauses exactly should be treated as equally preferable? It is now time to give a more precise definition of what is meant by symmetry with respect to negation. Any clause $C$ should be equally preferable with its *inverse*: a clause where one of the atomic predicates of $C$, say $P$, is replaced by its inverse predicate $\bar{P}$, given as

$$\bar{P}(x, y, \dots) \equiv \neg P(x, y, \dots) \tag{76}$$

for all individuals $x$, $y$, ...

**Definition 8.2** (Inverse)**.** The *inverse set* of a clause $C$, denoted $\text{Inv}(C)$, consists of those clauses that can be obtained by negating exactly one literal in $C$. All clauses $D \in \text{Inv}(C)$ are called *inverses* of $C$. Representing a clause by a set of literals, we get the following formalisation.

$$\begin{aligned} \text{Inv}(\{\}) &= \{\} \\ \text{Inv}(\{L\} \cup C') &= \{\{\neg L\} \cup C'\} \cup \{\{L\} \cup D : D \in \text{Inv}(C')\} \end{aligned} \tag{77}$$

**Example 8.3.** Given the clauses,

$$\begin{aligned} C &= \forall x \forall y(\neg Px \vee \neg Rxy \vee Py) \\ D_1 &= \forall x \forall y(Px \vee \neg Rxy \vee Py) \\ D_2 &= \forall x \forall y(\neg Px \vee Rxy \vee Py) \\ D_3 &= \forall x \forall y(\neg Px \vee \neg Rxy \vee \neg Py) \end{aligned} \tag{78}$$

The clauses $D_1$, $D_2$ and $D_3$ are inverses of $C$ (and vice versa), and there are no other inverses of $C$. Note, however, that none of the clauses $D_1$, $D_2$ and $D_3$ are mutual inverses.

**Example 8.4.** The empty clause $\perp$ has no inverses, and the inverse of a singleton ground clause is its negation.

**Definition 8.5** (Symmetry with respect to negation)**.** The preference relation $\geq_{\text{pl}}$ is called symmetric with respect to negation if and only if mutual inverses are equally preferable with each other.

$$C' \in \text{Inv}(C) \to C =_{\text{pl}} C' \tag{79}$$

Having thus precisely defined symmetry with respect to negation, it is time to look at an interesting result that pertains to all preference relations that are symmetric with respect to negation: the implausibility of irrelevant clauses.

## 8.1 Implausibility of irrelevant clauses

If all mutual inverses are symmetric, then there are wide classes of clauses that are implausible. The first one to be studied are the irrelevant clauses, which intuitively mean those clauses that have nothing to do with our observational theory $O$.

**Definition 8.6** (Inverse class)**.** An inverse class is a set of clauses that is closed under inversion.

$$\text{InverseClass}(K) \equiv \forall C(C \in K \rightarrow \exists D(D \in K \wedge C \in \text{Inv}(D))) \tag{80}$$

The inverse class of a clause $C$ is the inverse class to which $C$ belongs. The inverse class of $C$, denoted $IC(C)$, can be constructed by forming a closure over inversions of $C$.

$$IC(C) = \{C\} \cup \bigcup_{C' \in \text{Inv}(C)} IC(C') \tag{81}$$

**Example 8.7.** The following set of clauses is an inverse class:

$$\begin{aligned} O = \{&\forall x(Px \vee Qx), \forall x(Px \vee \neg Qx), \\ &\forall x(\neg Px \vee Qx), \forall x(\neg Px \vee \neg Qx)\} \end{aligned} \tag{82}$$

**Theorem 8.8.** *Every inverse class is minimally inconsistent.*

*Proof.* The inconsistency of inverse classes can be established by induction on the number of literals in the clauses of the class. For clauses of zero literals, the inconsistency of the inverse class $\{\bot\}$ is trivially established. For clauses of more than zero literals, the inverse class $K$ can be formed from a inverse class of shorter clauses $K'$ by constructing all clauses with affirmative and negative versions of a new literal $L$ added:

$$K = \{\{L\} \cup C : C \in K'\} \cup \{\{\neg L\} \cup C : C \in K'\} \tag{83}$$

Now, from this inverse class $K$ we can deduce all clauses in $K'$ by resolving, for every clause $C \in K'$, the clauses $\{L\} \cup C$ and $\{\neg L\} \cup C$, producing the original clause $C$. It follows that the inverse class $K$ is inconsistent if $K'$ is. This completes the proof by induction.

Minimal inconsistency follows from the fact that in every inductive step, every clause in the theory $K$ is needed for resolution to produce the smaller inverse class $K'$. It follows that no proper subset of $K$ is inconsistent. □

**Definition 8.9** (Independence)**.** A set of sentences $K$ is *independent* from a theory $O$ if and only if all proper subsets of $K$ are consistent with $O$.

Intuitively, a set of sentences $K$ is independent from $O$ when $O$ does not affect the consistency of $K$.

**Theorem 8.10.** *All clauses in an inverse class which is independent from a theory $O$ are implausible with respect to $O$.*

*Proof.* Because every inverse class is inconsistent by theorem 8.8, then by definition of independence, every independent inverse class is minimally inconsistent with $O$. In addition, all clauses in an inverse class are symmetric. Since the independent inverse class is minimally inconsistent with $O$ and all clauses are preferred to all clauses in it, its every clause is implausible. □

**Definition 8.11** (Irrelevance)**.** A clause $C$ is *irrelevant* with respect to a theory $O$ if and only if none of the clauses in $\mathrm{Inv}(C)$ resolve against any clause in $O$.

**Theorem 8.12.** *If $C$ is irrelevant with respect to $O$, the inverse class $IC(C)$ is independent from $O$.*

*Proof.* Since $\mathrm{Inv}(C)$ contains all the literals in $IC(C)$, irrelevance of $C$ with respect to $O$ implies that no clauses in $O$ will resolve against a clause in $IC(C)$, and thus $IC(C)$ has no consequences with $O$. It follows that $IC(C)$ is independent from $O$ if and only if all proper subsets of $IC(C)$ are consistent, which is true for every inverse class by theorem 8.8. □

**Corollary 8.13.** *All irrelevant clauses are implausible.*

*Proof.* For every clause $C$ that is irrelevant with respect to a theory $O$, $IC(C)$ is inconsistent by 8.8, independent from $O$ by 8.12, and consequently all clauses in $IC(C)$ are implausible by 8.10. Since $C \in IC(C)$, $C$ is implausible. □

**Example 8.14.** Let us have the following theory $O$ and clauses:

$$\begin{aligned} O &= \{P_1a, P_2b, \neg P_2c, P_3c\} \\ C_1 &= \forall x P_4 x \\ C_2 &= \forall x (P_4x \vee P_5x) \\ C_3 &= \forall x (\neg P_1x \vee P_4x) \\ C_4 &= \forall x (P_2x \vee P_3x) \end{aligned} \tag{84}$$

Of these clauses, the clauses $C_1$ and $C_2$ are irrelevant with respect to $O$, because they only contain predicates not mentioned in $O$ and, consequently, their inverse classes are independent from $O$.

The clauses $C_3$ and $C_4$, on the other hand, are not irrelevant. Accordingly, the clause $C_4$ has the inverse $\forall x(P_2x \vee \neg P_3x)$ which is disproved by the individual $c$ in $O$; and the clause $C_3$ has the following set $K \subset IC(C_3)$ which is inconsistent with the clause $P_1a$ in $O$:

$$K = \{C_3, \forall x(\neg P_1x \vee \neg P_4x)\} \tag{85}$$

## 8.2 Implausibility of irrelevant weakenings of falsified clauses

It turns out that the implausibility of irrelevant clauses is just a special case of a more general phenomenon. For any clause $C$, we find that if $C$ has been falsified by $O$ (i.e. $C$ is inconsistent with $O$), then all clauses $C \cup D$ where $D$ is irrelevant with respect to $O$, are implausible.

**Definition 8.15** (IC-extensions)**.** The class of clauses $K$ is called an *IC-extension* of a clause $C$ (by $K'$) if there is some inverse class of clauses $K'$ such that

$$K = \{C \cup D : D \in K'\} \tag{86}$$

and $C$ is irrelevant with respect to $K'$.

**Theorem 8.16.** *If a clause $C$ is inconsistent, then all IC-extensions of $C$ are minimally inconsistent.*

*Proof.* This can be proved in the similar way as theorem 8.8, except that the empty clause $\bot$ is substituted with the clause $C$. □

**Lemma 8.17.** *All IC-extensions are subsets of some inverse class.*

*Proof.* Given two inverse classes $K_1$ and $K_2$ whose clauses are irrelevant with respect to each other, the *combination* of these inverse classes

$$K = K_1 \times K_2 = \{C \cup D : C \in K_1 \wedge D \in K_2\} \tag{87}$$

is also an inverse class. This is because for each clause $(C \cup D) \in K$, all clauses in $\mathrm{Inv}(C \cup D)$ also belong to $K$, since the clauses where the inverted literal belongs to $D$ can be found in $\{C \cup D' : D' \in K_2\} \subset K$ and the clauses where the inverted literal belongs to $C$ can be found in $\{C' \cup D : C' \in K_1\} \subset K$.

Since an IC-extension of $C$ by an inverse class $K'$ is a subset of $IC(C) \times K'$, every IC-extension is indeed a subset of some inverse class. □

**Theorem 8.18.** *Let $K$ be an IC-extension of an inconsistent clause $C$ and suppose $K$ is independent from a theory $O$. Then all clauses in $K$ are implausible with respect to $O$.*

*Proof.* Since an IC-extension $K$ of a clause $C$ is a subset of an inverse class by theorem 8.17, all clauses in $K$ are symmetric. Since $K$ is minimally inconsistent with respect to $O$ by theorem 8.16 and definition of independence, all clauses in $K$ are implausible. □

**Definition 8.19** (Irrelevant IC-extensions)**.** An IC-extension $K$ of a clause $C$ by the inverse class $K'$ is irrelevant with respect to a theory $O$ if and only if the clauses in $K'$ are irrelevant with respect to $O$.

**Theorem 8.20.** *An irrelevant IC-extension $K$ of a clause $C$ (by $K'$) is independent from theory $O$ if $C$ is inconsistent with $O$.*

*Proof.* If the clause $C$ is inconsistent with $O$, then the IC-extension $K$ of $C$ by the inverse class $K'$ is logically equivalent with $K'$, since it is formed as disjunctions of clauses in $K'$ with $C$, which is already known to be false. We also know by definition of irrelevant IC-extensions that $K'$ is irrelevant with respect to $O$. Then by theorem 8.12, we know that $K'$ is independent from $O$. Since $K$ and $K'$ are logically equivalent, naturally $K$ is also independent from $O$. □

**Corollary 8.21.** *All clauses that are disjunctions of a falsified clause and an irrelevant clause are implausible.*

*Proof.* For every clause $C$ that is inconsistent with a theory $O$ and inverse class $K'$ that is irrelevant with respect to $O$, the set of sentences $K = \{C\} \times K'$ is inconsistent by 8.16, independent from $O$ by 8.20, and consequently, every clause in $K$ is implausible by 8.18. Since every disjunction of $C$ with a clause that is irrelevant with respect to $O$ belongs to some such set $K$, all such clauses are implausible. □

**Example 8.22.** Consider the theory $O$ and clause $C_3$ from example 8.14. The clause $C = \forall x \neg P_1 x$ is disproved by $O$ because of the clause $P_1 a$, and $C_3$ extends $C$ with the irrelevant literal $P_4 x$. $C_3$ is implausible because it belongs to this irrelevant IC-extension of $C$:

$$K = \{C_3, \forall x(\neg P_1 x \vee \neg P_4 x)\} \tag{88}$$

It can be readily verified that $K$ is minimally inconsistent with $O$ and both clauses in $K$ are equally preferable.

## 8.3 Conclusion

In addition to being intuitively attractive, the symmetry condition 7.17 has the satisfactory consequence of making all irrelevant clauses and irrelevant extensions of falsified clauses implausible. This strongly suggests that any account of inductive thinking should observe the condition of symmetry.

# 9 Total strength ordering of clauses

From sections 7.2 and 7.3, we know that the following properties of the preference relation are tempting for various reasons. To summarise:

**Totality** (Condition 7.13.) If the preference relation is a total order, then for every theory $O$, there is a unique maximal plausible generalisation that maximises the measures of informativeness and plausibility which define the utility of any inductive consequence of $O$.

**Symmetry with respect to negation** (Condition 7.17.) If the preference relation treats mutual inverses as equally plausible, which also has some appeal of its own, then the generalisations of any theory $O$ do not include clauses that are irrelevant with respect to $O$.

**Honoring generality of clauses** (Condition 7.20.) If the preference relation is a generalisation of the generality of clauses, then generalisations will consist of the most general clauses that are mutually consistent and consistent with the reference theory $O$. This seems good for reasons of both informativeness and implementation.

**Preferring rules over particularities** (Condition 7.22.) If the preference relation always prefers more rule-like clauses, then the generalisation will have clauses that apply to as many individuals as possible.

**Strength** The stronger (more specific) the preference relation is, the more clauses will be plausible under the preference relation. This will improve the informativeness of maximal plausible generalisations.

It is now time to devise a definition of the preference relation $\geq_{\text{pl}}$ that will meet the conditions above.

The first thing to notice is that, if we simply weaken the generality of clauses by treating all incomparable clauses as equally good, the result will be trivial: all clauses will be equally good. For the proof, consider the following: for all clauses $C_1$ and $C_2$, there is a third clause $D$ that is incomparable with both $C_1$ and $C_2$ — for instance, a clause that is irrelevant with respect to them. If incomparable clauses are equally good, then $C_1 =_{\text{pl}} D$ and $D =_{\text{pl}} C_2$. By transitivity of preference relations, then $C_1 =_{\text{pl}} C_2$.

Such a preference relation does not honor generality of clauses, either.

**Example 9.1.** Consider the following clauses.

$$\begin{aligned} C_1 &= \forall x Px \\ C_2 &= \forall x(Qx \vee Rx) \\ C_3 &= \forall x Qx \end{aligned} \tag{89}$$

Now, $C_3 \models C_2$ and $C_2 \not\models C_3$, so in order for the generality relation to honor generality of clauses, we should have $C_3 >_{\text{pl}} C_2$. However, because $C_1 =_{\text{pl}} C_2$ and $C_1 =_{\text{pl}} C_3$, then by transitivity of preference relations, $C_3 =_{\text{pl}} C_2$.

This suggests that disjunctions should *not* be treated as equally preferable with non-disjunctive clauses, even for totally unrelated predicates. Generalising this idea further, the preference of clauses should always be based on some kind of syntactic generality. This leads to the following definitions, which are also heavily motivated by considering what it takes to honor $\theta$-subsumption.

## 9.1 Definition of the preference relation

In this section, we build a definition of the preference relation that meets the criteria listed at the beginning of this section. The definition of preference is based on a property of clauses called *relative universality*, which in turn is based on a similarly named property of literals.

**Definition 9.2** (Universality of literals)**.** The *universality* of a literal $L$ *on rank* $n$ is the number of universally quantified variables nested within at least $n$ function terms in $L$ — direct arguments of predicates are on rank 0, arguments of direct arguments of predicates are on rank 1 and so on. Ranks with smaller number are called *higher* ranks.

$$\begin{aligned}
\mathrm{Vars}(\neg L, n) &= \mathrm{Vars}(L, n) \\
\mathrm{Vars}(P(\bar{T}), n) &= \bigcup_{t \in T} \mathrm{Vars}(t, n) \\
\mathrm{Vars}(f(\bar{T}), n) &= \bigcup_{t \in T} \mathrm{Vars}(t, n-1) \\
\mathrm{Vars}(x, 0) &= \{x\} \text{ if } x \text{ is a variable} \\
\mathrm{Vars}(x, n) &= \{\} \text{ if } n \neq 0 \\
\mathrm{Vars}(\mathrm{a}) &= \{\} \text{ if a is a constant} \\
\mathrm{VarsDownTo}(L, 0) &= \mathrm{Vars}(L, 0) \\
\mathrm{VarsDownTo}(L, n) &= \mathrm{Vars}(L, n) \cup \mathrm{VarsDownTo}(L, n-1) \\
\mathrm{Univ}(L, n) &= |\mathrm{Vars}(L, n) \setminus \mathrm{VarsDownTo}(L, n-1)|
\end{aligned} \tag{90}$$

where $P(\bar{T})$ means a predicate application whose argument terms form the set $T$, $f(\bar{T})$ means a function application whose argument terms form the set $T$, and $|C|$ is the cardinality of the set $C$.

**Definition 9.3** (Relative universality of literals)**.** A literal $L_1$ is *strictly more universal than* another literal $L_2$ (denoted $L_1 >_{\mathrm{univ}} L_2$) if and only if the universality of $L_1$ is greater than that of $L_2$ on the highest rank where the universalities differ. If the universalities don't differ on any rank, $L_1$ and $L_2$ are *equally universal*.

$$\begin{aligned}
\mathrm{MoreUniv}(L_1, L_2, n) \equiv &\mathrm{Univ}(L_1, n) > \mathrm{Univ}(L_2, n) \vee \\
&(\mathrm{Univ}(L_1, n) = \mathrm{Univ}(L_2, n) \wedge \mathrm{MoreUniv}(L_1, L_2, n+1)) \\
L_1 >_{\mathrm{univ}} L_2 \equiv &\mathrm{MoreUniv}(L_1, L_2, 0)
\end{aligned} \tag{91}$$

**Example 9.4.** Let us have the literals

$$\begin{aligned}
L_1 &= P(x) \\
L_2 &= Q(x, f(y)) \\
L_3 &= Q(x, f(f(x, z))) \\
L_4 &= Q(x, \mathrm{a})
\end{aligned} \tag{92}$$

$L_2$ is strictly more universal than $L_1$ since $\mathrm{Univ}(L_1, 0) = \mathrm{Univ}(L_2, 0) = 1$ and $\mathrm{Univ}(L_1, 1) = 0 < \mathrm{Univ}(L_2, 1) = 1$. $L_2$ is also strictly more universal than

$L_3$, since $\mathrm{Univ}(L_3, 1) = 0$ (the variable $z$ on rank 2 does not affect the situation). $L_3$ is strictly more universal than $L_1$ as they have the same universality on ranks 0 and 1, while $\mathrm{Univ}(L_3, 2) = 1 > \mathrm{Univ}(L_1, 2) = 0$.

$L_4$ is equally universal with $L_1$, because both have universality 1 on rank 0 and 0 on lower ranks.

**Definition 9.5** (Relative universality of clauses)**.** Within a clause $C$, a *minimally universal* literal is a literal $L$ which is not strictly more universal than any other literal $L'$ in $C$. A clause $C_1$ is strictly more universal than another clause $C_2$ if and only if the minimally universal literal $L_1$ of $C_1$ is strictly more universal than the minimally universal literal $L_2$ of $C_2$, or if $L_1$ and $L_2$ are equally universal and $(C_1 \setminus \{L_1\})\sigma_1$ is strictly more universal than $(C_2 \setminus \{L_2\})\sigma_2$, where $\sigma_n$ is a substitution of variables in $L_n$ with new constants.[14]

The empty clause $\bot$ which does not have a minimally universal literal is strictly more universal than any other clause. If the literals of $C_1$ and $C_2$ are all equally universal (as arranged by universality), then $C_1$ and $C_2$ are equally universal.

$$\begin{aligned} \min{}_{\mathrm{univ}}\{L\} &= L \\ \min{}_{\mathrm{univ}}(\{L\} \cup C) &= \begin{cases} L & \text{if } \min{}_{\mathrm{univ}} C >_{\mathrm{univ}} L \\ \min{}_{\mathrm{univ}} C & \text{otherwise} \end{cases} \end{aligned} \tag{93}$$

Relative universality is defined as:

$$\begin{aligned} C_1 >_{\mathrm{univ}} C_2 \equiv{}& \min{}_{\mathrm{univ}} C_1 >_{\mathrm{univ}} \min{}_{\mathrm{univ}} C_2 \vee \\ & (\neg(\min{}_{\mathrm{univ}} C_2 >_{\mathrm{univ}} \min{}_{\mathrm{univ}} C_1) \wedge \\ & (C_1 \setminus \{\min{}_{\mathrm{univ}} C_1\})\sigma_1 >_{\mathrm{univ}} (C_2 \setminus \{\min{}_{\mathrm{univ}} C_2\})\sigma_2) \end{aligned} \tag{94}$$

where $\sigma_n$ is a substitution that substitutes variables in $\min{}_{\mathrm{univ}} C_n$ by new constants.

**Example 9.6.** Suppose we have the clauses,

$$\begin{aligned} C_1 &= \forall x(P(x) \vee \neg P(f(x))) \\ C_2 &= \forall x(P(x) \vee \neg P(f(f(x)))) \\ C_3 &= \forall x \forall y(R(x, y) \vee \neg R(x, f(y))) \\ C_4 &= \forall x \forall y(R(x, x) \vee \neg Q(y, f(x))) \\ C_5 &= \forall x \forall y(R(f(x), f(y)) \vee \neg R(x, f(y))) \end{aligned} \tag{95}$$

The clauses $C_3$ and $C_4$ are strictly more universal than $C_1$ and $C_2$, since their minimally universal literals have one variable on rank 0, while $C_1$ and $C_2$ have none. $C_5$ is also strictly more universal, since its minimally universal literal has two variables on rank 1, while $C_1$ has one and $C_2$ has none. Also consequently, $C_1$ is strictly more universal than $C_2$.

The clauses $C_3$ and $C_4$ are equally universal: their minimally universal literals are equally universal, and "bind" both $x$ and $y$, so the stronger literals become equally universal too. Both are strictly more universal than $C_5$.

[14] These variables are, in a way, "bound" by the literals $L_n$.

**Definition 9.7.** A clause $C$ is preferred to another clause $D$ if and only if $C$ is strictly more universal than or equally universal with $D$.

$$C \geq_{\text{pl}} D \equiv \neg(D >_{\text{univ}} C) \tag{96}$$

## 9.2 Properties of the preference relation

**Theorem 9.8.** *Preference is total.*

*Proof.* Universality on rank $n$ is total, because it is based on the total ordering of natural numbers. Relative universality of literals is total, because it combines the total orders of different ranks in a dictionary order. Relative universality of clauses is total, because it combines the total orders of different literals in a dictionary order. Consequently, preference is total, because it is the corresponding weak order for the strict total order of universality. □

**Theorem 9.9.** *Preference is symmetric with respect to negation.*

*Proof.* Because the affirmation and negation of a literal are equally universal by definition 9.2, mutual inverses are equally universal. It follows that all clauses in an inverse class are equally universal and so equally preferable. □

**Theorem 9.10.** *Preference gives priority to rules over particular claims.*

*Proof.* Since a literal with no (universally quantified) variables has the lowest possible universality and clauses are compared by their minimally universal literals, a clause that involves variables in every literal is always preferred to a clause that does not. In addition to this, variables in function terms are on a lower rank than other variables, ensuring that a clause whose every literal involves variables as direct arguments of predicates is preferred to a clause including a literal that does not. □

**Theorem 9.11.** *If $C \preceq_\theta D$ and $D \preceq_\theta C$, then $C =_{\text{pl}} D$.*

*Proof.* If $C \preceq_\theta D$ and $D \preceq_\theta C$, then there is a substitution $\theta$ for which $C\theta = D$, and there must be an inverse substitution $\theta^{-1}$ such that $D\theta^{-1} = C$. Now, $\theta$ must be a substitution that substitutes distinct variables with other distinct variables, because otherwise the inverse $\theta^{-1}$ would not be a substitution of variables, contradicting the definition of $\theta$-subsumption. Since the substitution $\theta$ does not do anything than rename variables, $C$ and $D$ have equal universality. □

**Theorem 9.12.** *If $C \preceq_\theta D$ but $D \not\preceq_\theta C$, then $C >_{\text{pl}} D$.*

*Proof.* As seen in section 5.4, a clause $C$ may be generalised, under $\theta$-subsumption, in two ways: by substituting some occurrences of one of its terms by a new variable, or by dropping literals from it.

If the clause is generalised by substitution, we have three cases.

1. Some occurrences of a constant are substituted with a new variable. Because of the new variable, the universality of at least some rank of some literal increases, which makes the clause more universal.

2. Some occurrences of a function term are substituted with a new variable. This may decrease the universality of some rank in some literal $L$, but it will increase the universality of a higher rank in $L$, which makes the clause more universal. (Think, for instance, of substituting $f(x)$ with $y$.)

3. Some (but not all) occurrences of a variable are substituted with a new variable. On each rank where the variable occurs, we have either the same universality if all occurrences on that rank were substituted, or greater universality if only some occurrences were substituted (so some occurrences of the old variable remain). Consequently, we have increased the universality of at least some rank of some literal, which makes the clause more universal.

If the clause is generalised by dropping literals, we can prove that a subset $C'$ given by $C = C' \cup \{L\}$ is always more universal by induction on number of literals. If $C'$ is the empty clause, the result follows trivially since the empty clause is more universal than any other clause. Likewise, if all literals in $C'$ are strictly more universal than $L$, then $C'$ is trivially more universal than $C$.

For the inductive step, suppose the above trivial cases do not hold. Then, the minimally universal literal $L' = \min_{\text{univ}} C' = \min_{\text{univ}} C$, and $C'$ is strictly more universal than $C$ if and only if $C' \setminus L'$ is strictly more universal than $C \setminus L'$, which is given by the inductive assumption.

Consequently, in whatever way a clause $C'$ $\theta$-subsumes clause $C$, $C' >_{\text{pl}} C$. Since any clause that $\theta$-subsumes $C$ can be produced by applying some number of such generalisation steps, by transitivity of the preference relation, all those clauses are strictly preferred to $C$. □

**Corollary 9.13.** *Preference honors generality as given by $\theta$-subsumption.*

*Proof.* Follows directly from definition 7.19 together with theorems 9.11 and 9.12. □

However, it is not known whether preference honors logical entailment.

We now see that the presented preference relation meets all requirements listed at the beginning of this section. However, the definition given is not necessarily the strongest one that the other conditions permit. Future work may show improvements in some of these directions:

1. an intuitively acceptable strengthening of the given preference relation

2. a preference relation that provably honors logical entailment

3. a proof that the preference relation presented here honors logical entailment

4. a more straightforward definition of the preference relation.

## 9.3 Examples of plausible clauses

Now that we have a definition of preference, it is possible to examine some examples of plausible clauses. Let us start with the simplest possible example, a generalisation of an unfalsified unary predicate.

**Example 9.14.** The clause $C = \forall x Px$ is plausible with respect to the theory $O = \{Pc\}$.

*Proof.* Of the clauses that are preferred to $C$, only its inverse $\forall x \neg Px$ is relevant with respect to $C$. However, the latter clause is inconsistent with $O$, so no witness set $K$ may contain it. We don't need to consider clauses that are irrelevant with respect to $C$, because they cannot have any consequences when combined with $C$. Consequently, there is no witness of $C$'s implausibility, so $C$ is plausible. □

A merger of two theories with no common predicates will have all and only the plausible clauses of both theories. This is not true for Hempel's confirmation relation (see section 3.2).

**Example 9.15.** The clause $C_1$ is plausible with respect to theory $O_1$, and $C_2$ is plausible with respect to theory $O_2$:

$$\begin{aligned} O_1 &= \{Pa\} \\ C_1 &= \forall x Px \\ O_2 &= \{Qb\} \\ C_2 &= \forall x Qx \\ C_3 &= \forall x(\neg Px \vee \neg Qx) \end{aligned} \tag{97}$$

Both $C_1$ and $C_2$ are also plausible with respect to $O_1 \cup O_2$. $C_3$ is consistent with $O_1 \cup O_2$ but implausible: there is a witness set $K = \{C_1, C_2\}$ of its implausibility, since $K$ is consistent with $O$, $K \cup \{C_3\}$ is inconsistent and both clauses in $K$ are preferred to $C_3$. $C_3$ is also implausible with respect to both $O_1$ and $O_2$, since for both theories, it is a disjunction of a falsified clause with an irrelevant clause.

A disjunctive clause may be plausible if all the inverses of its disjuncts are falsified.

**Example 9.16.** The clause $C = \forall x(Px \vee Qx)$ is plausible with respect to the theory $O = \{Pa, \neg Pb, \neg Qa, Qb\}$.

*Proof.* In the list, we have the clauses which are preferred to and relevant with respect to $C$:

$$\begin{aligned} C_1 &= \forall x Px \\ C_2 &= \forall x \neg Px \\ C_3 &= \forall x Qx \\ C_4 &= \forall x \neg Qx \\ C_5 &= \forall x(Px \vee \neg Qx) \\ C_6 &= \forall x(\neg Px \vee Qx) \\ C_7 &= \forall x(\neg Px \vee \neg Qx) \end{aligned} \tag{98}$$

Of these, $C_1 \ldots C_6$ are falsified by $O$, so they cannot be in any witness set of $C$'s implausibility. As for $C_7$, we notice that the set $O \cup \{C, C_7\}$ is consistent, so the set $\{C_7\}$ cannot witness the implausibility of $C$, either. For similar considerations, $C_7$ is also plausible with respect to $O$. □

## 9.4 Problems with the preference relation

The preference relation presented here possesses some nice properties, but it is by no means perfect. One aspect that is not taken properly into account is the plausibility of clauses with restricted domain. This means a clause which contains an universally quantified variable, but that variable is an argument of a function $f$, so the domain of the quantification is constrained to the codomain of the function $f$.

**Example 9.17.** It would seem intuitive if clause $C$ was plausible with respect to theory $O$:

$$\begin{aligned} O &= \{\neg P(a), P(f(a))\} \\ C &= \forall x P(f(x)) \end{aligned} \tag{99}$$

However, the following clauses prove $C$ to be implausible:

$$\begin{aligned} D_1 &= \forall x \forall y (\neg P(x) \vee Q(x, y)) \\ D_2 &= \forall x \forall y (\neg Q(f(b), y)) \end{aligned} \tag{100}$$

It can be seen that $K = \{D_1, D_2\}$ is a witness set for $C$, because $K$ implies $\neg P(f(b))$ which is consistent with $O$ but inconsistent with $C$, and both $D_1$ and $D_2$ are preferred to $C$.

The problem here seems to be that the variable $y$ on rank 0 protects, so to say, the clause $D_2$ from being less universal than $C$. This would suggest that the problem could be fixed by giving a definition of universality of literals that treats variables differently, but it is not so easy to invent a definition that fixes this problem and still meets the other conditions of the preference relation.

Another problem is that the preference relation potentially disregards recursive clauses, as it is based on $\theta$-subsumption and not logical entailment. This has potentially big consequences. A recursive clause $C$ may, for instance, be implausible by a witness set whose clauses are all more specific than $C$, as preference does not necessarily honor logical entailment for recursive clauses. Dealing with these problems is left for future research.

# 10 Possibilities of the maximal plausible generalisation

## 10.1 Conclusions

This thesis has presented a novel approach to induction. The approach is motivated by three main ideas:

1. the need to find a precise definition for the term *plausible*;
2. the definition of induction as a process that produces consistent extensions of our world view; and
3. an attempt to devise a description of induction that can be implemented algorithmically.

These three goals are nicely met by the definition of maximal plausible generalisation. The definition presented is by no means the only possible one, but for every choice that has been made, some intuitive argumentation has been offered. There are numerous details that need to be sorted out, such as the best definition of the preference relation. However, I believe that the conditions of plausibility and completeness provide a firm foundation for classificatory induction.

The definition of plausibility deserves special attention, as it is quite different from any probability-based definition. Clauses are either totally plausible or totally implausible; and while all conclusively proved statements are always plausible and all conclusively refuted statements are always implausible, the rest of all statements also fall into the two categories. Plausibility of a statement is defined with respect to the validity of *other* statements. Simplifying a little, we could say that a sentence is plausible when it is the broadest hypothesis that is unrefuted by our experience.

The definition of induction in this thesis has the benefit that it has a very simple setting. The result of inductive inference is only dependent on the input theory; the inductive process uses no background theories, no initial probabilities, nor any kind of anterior hypotheses. The results of induction are purely an extension of the facts that we already know. As such, this kind of induction can be used to generalise any kind of theory whatsoever, be it originally based on observation or not.

The induction framework also accounts for a problem that has been seldom addressed in inductive logic. Namely, the maximal plausible generalisation excludes irrelevant sentences, which are usually permitted by refutation-based frameworks for induction: if induction is taken to confirm any kind of unrefuted hypothesis, then all kinds of irrefutable hypotheses become as "plausible" as hypotheses that can actually be tested. Our plausibility framework, combined with symmetry with respect to negation, provides an argument for why sentences are not confirmed by a theory for which they are irrelevant.

## 10.2 Algorithmic induction

Although the matter has not been directly addressed in this thesis, it should be possible to implement this kind of induction as a terminating algorithm. Since it is unnecessary to examine irrelevant clauses, we can find all (nonrecursive) inductive consequences of a theory $O$ by the following steps:

1. From all clauses $C \in O$, form all generalisations and gather them in $H_1 = \{D : C \in O, D \preceq_\theta C\}$.

2. *Connect* the clauses in $H_1$ by forming disjunctions of them so that variables in two clauses may be rewritten to be the same:

$$\begin{aligned} \text{Conn}(\{\}) &= \{\{\}\} \\ \text{Conn}(\{C\} \cup K) &= \text{Conn}(K) \cup \{C\sigma \cup D : D \in \text{Conn}(K), \sigma \in SS(C, D)\} \\ H_2 &= \text{Conn}(H_1) \end{aligned} \tag{101}$$

   where $SS(C, D)$ is the set of all substitutions that substitute variables in $C$ with those in $D$.

3. From the clauses in $H_2$, form all subsets that are minimally inconsistent with $O$, and filter out the clauses that are least preferred in those minimally inconsistent subsets.

4. The remaining clauses in $H_2$ form the inductive closure of $O$.

The efficiency of this method can be improved in many ways. For instance, clauses in $H_2$ that are inconsistent with $O$ can be disregarded, because they are implausible and no witness set can contain them. It is also probably possible to build $H_2$ incrementally by considering each of the clauses $C \in O$ in turn and appropriately weakening clauses that are refuted by $C$. However, such concerns are outside the scope of the current thesis, and left for future research.

Also, future research in inverting implication may give a more fine-grained method of generalisation and specialisation of clauses than $\theta$-subsumption.

## 10.3 Developments in the preference relation

The preference relation, as defined in section 9, manages to meet the conditions postulated in section 7, but is quite probably not the best possible definition. The preference relation could be improved in several areas.

**Strength ordering of terms** The preference relation honors generality of clauses as given by $\theta$-subsumption, but literals with complicated terms are sometimes ordered unintuitively (see section 9.4). As a result, the MPG excludes some clauses that would seem intuitively plausible.

**Simplicity** The definition of the preference relation is complicated and could possibly be simplified. One possibility in this direction would be some application of *flattened* representation of clauses, where function terms are replaced by literals that state their conditions explicitly.

**Honoring logical entailment** In the current work, it is left open whether the preference relation honors generality of clauses as defined by logical entailment.

## 10.4 Conceptualisation

The applicability of the maximal plausible generalisation of a theory rests critically on the correctness of the information (and informativeness) of the input theory. However, real-world induction processes rely on observational data, and the process of producing a precise and correct observation report from observational data — that is, conceptualisation of observational data — is anything but trivial.

Even though conceptualisation is strictly outside the scope of this thesis, the processes of syntactic induction and conceptualisation are clearly related. For example, the formation of the maximal plausible generalisation gives hints about what *kind* of conceptualisation is needed for induction. Also, the definition of plausibility gives a practical heuristic of what should be done when we have two equally strong but mutually inconsistent hypotheses: gather proof against one or the other. Finally, the MPG can potentially be used in verification of conceptualisations: a counterintuitive MPG hints at an error in conceptualisation.

# References


[Car50] Rudolf Carnap. *Logical Foundations of Probability.* The University of Chicago Press, 1950.

[Fla95] Peter A. Flach. *Conjectures – an inquiry concerning the logic of induction.* Institute for Language Technology and Artificial Intelligence, April 1995.

[Fla96] Peter A. Flach. On the logic of induction. 1996. fetched, April 20, 2007, http://www.cs.bris.ac.uk/~flach/Conjectures/PS/flach-LogicOfInduction.pdf.

[Hem43] Carl G. Hempel. A purely syntactical definition of confirmation. *Journal of Symbolic Logic*, 8(4):122–143, 1943.

[Hem45] Carl G. Hempel. Studies in the logic of confirmation. *Mind*, 54(213–214):1–26,97–121, 1945.

[IA93] Peter Idestam-Almquist. *Generalization of clauses.* PhD thesis, Stockholm University, Department of Computer and Systems Sciences, Edsbruk, Sweden, 1993.

[Kal07] Panu A. Kalliokoski. A theory-based logic of inductive generalisation. Bachelor's thesis, University of Helsinki, Department of Theoretical Philosophy, 2007.

[Meh99] Joke Meheus. Deductive and ampliative adaptive logics as tools in the study of creativity. *Foundations of Science*, 4:325–336, 1999.

[Mit82] Tom M. Mitchell. Generalization as search. *Artificial Intelligence*, 18(2):203–226, 1982.

[Mug92a] S.H. Muggleton. Inductive logic programming. In Muggleton [Mug92c], pages 3–27.

[Mug92b] S.H. Muggleton. Inverting implication. In *Proceedings of the Second Inductive Logic Programming Workshop*, pages 19–39, Tokyo, 1992. ICOT (Technical report TM-1182).

[Mug92c] Stephen Muggleton, editor. *Inductive Logic Programming.* Academic Press, London, England, United Kingdom, 1992.

[Plo71] G.D. Plotkin. *Automatic Methods of Inductive Inference.* PhD thesis, Edinburgh University, 1971.

[Ren86] Larry Rendell. A general framework for induction and a study of selective induction. *Machine Learning*, 1:166–226, 1986.

[RN03] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: a Modern Approach.* Prentice Hall, Pearson Education International, 2003.

[Rob65] J. A. Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12(1):23–41, 1965.

[Sol64] R. J. Solomonoff. A formal theory of inductive inference. *Information and Control*, 7:1–22, 1964.

[SS88] Manfred Schmidt-Schauss. Implication of clauses is undecidable. *Theoretical Computer Science*, 59:287–296, 1988.

[ZZ96] Denis Zwirn and Hervé P. Zwirn. Metaconfirmation. *Theory and Decision*, 41(3):195–228, 1996.