Equivalence and Reduction

Two FOL conjunctions or FOL clauses h, h' **are equivalent** if $h \subseteq_\theta h'$ and $h' \subseteq_\theta h$. We write $h \approx_\theta h'$.

Let h be a FOL conjunction or clause. We say that h **is reduced** if for no FOL conjunction (clause) h' such that $\text{Lits}(h') \subset \text{Lits}(h)$ it holds $h' \approx_\theta h$. We say that h' **is a reduction of** h if $h \approx_\theta h'$ and h' is reduced.

Algorithm returning a reduction of h:

- If there is a literal $\mathcal{L} \in h$ such that $h \subseteq_\theta h \setminus \mathcal{L}$ for some θ, set $h := h\theta$ and repeat, else return h.

$h \setminus \mathcal{L}$ denotes the result of removing literal $\mathcal{L}$ from h.

Example: Reduction

$$h := \mathrm{vertex}(x) \wedge \mathrm{vertex}(y) \wedge \mathrm{vertex}(z) \wedge \neg\mathrm{edge}(x, y) \wedge \neg\mathrm{edge}(x, z)$$
$$\mathcal{L} = \mathrm{vertex}(z),\ \ h \subseteq_\theta h \setminus \mathcal{L} \text{ with } \theta = \{\, z \mapsto y \,\},$$
$$h := h\theta = \mathrm{vertex}(x) \wedge \mathrm{vertex}(y) \wedge \neg\mathrm{edge}(x, y)$$

Note that *two* literals were removed by setting $h := h\theta$ above.

On the other hand,

$$h := \mathrm{vertex}(x) \wedge \mathrm{vertex}(y) \wedge \mathrm{vertex}(z) \wedge \neg\mathrm{edge}(x, y) \wedge \neg\mathrm{edge}(y, z)$$

is already reduced (cannot reduce further).

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Decision Policy for FOL Clauses and Disjunctios

In generalization, we used policy (??) because the assumption of *contingent* propositional examples implies the assumptions of Theorem (??) so (??) is equivalent to (??) ((??), respectively) for clausal (conjunctive) examples.

The advantage of (??) is that it is decidable and enables us to use the least general generalization, and specifically the lgg operator for learning.

Similarly, when generalizing from FOL clausal (FOL conjunctive, respectively) hypotheses and examples, we use the policy

$$h(x) = 1 \text{ if } h \subseteq_\theta x \tag{1}$$

on the assumption that examples x are *not self-resolving*, which implies the assumptions of Theorem (??) making (1) equivalent to (??) ((??)).
(*exercise*)

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Example: lgg of Clauses

lgg of

$$\text{male}(x) \wedge \text{female}(y) \wedge \text{parent}(x, y) \rightarrow \text{daughter}(y, x)$$

and

$$\text{female}(x) \wedge \text{parent}(\mathsf{ann}, x) \rightarrow \text{daughter}(x, \mathsf{ann})$$

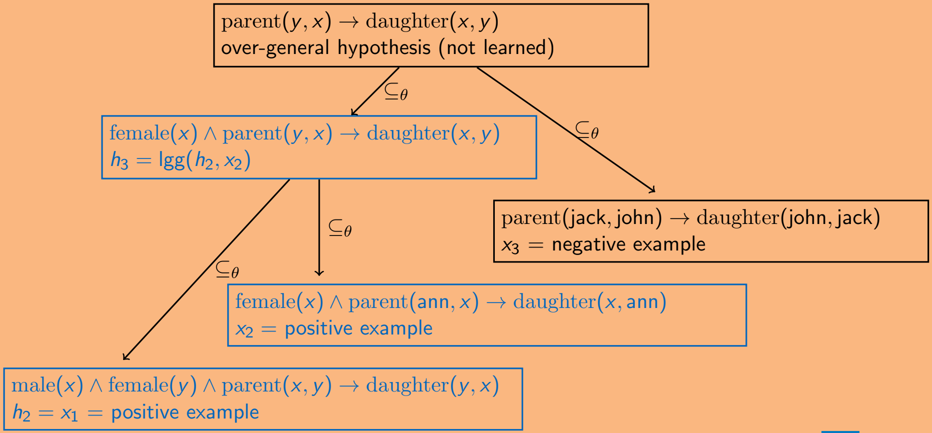
	$\neg\text{female}(x)$	$\neg\text{parent}(\mathsf{ann}, x)$	$\text{daughter}(x, \mathsf{ann})$
$\neg\text{male}(x)$			
$\neg\text{female}(y)$	$\neg\text{female}(v_1)$		
$\neg\text{parent}(x, y)$		$\neg\text{parent}(v_2, v_1)$	
$\text{daughter}(y, x)$			$\text{daughter}(v_1, v_2)$

θ	σ	
y	x	v_1
x	ann	v_2

is

$$\text{female}(v_1) \wedge \text{parent}(v_2, v_1) \rightarrow \text{daughter}(v_1, v_2)$$

The Generalization Algorithm with X = FOL clauses



Generalizing FOL Conjunctions or Clauses

Because the lgg operator was extended to FOL, the generalization algorithm can be used with policy (1), assuming

$$X = \text{non-self-resolving FOL conjunctions} \quad (2)$$

or

$$X = \text{non-self-resolving FOL clauses} \quad (3)$$

producing the lgg of all positive examples. *(exercise problem)*

However, in the FOL case, we *cannot* prove a polynomial mistake bound analogical to the propositional case. In other words, Theorem (??) (where $n = |\mathcal{P}|$) does not hold if its assumption on X is replaced by (2) or (3), even if $|\mathcal{F}| = \emptyset$. *(Showing this is an exercise problem.)*

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Background Knowledge: Motivation

Following the definition, $\text{lgg}(x_1, x_2)$ of

$$x_1 = \mathrm{female}(x) \wedge \mathrm{father}(y, x) \rightarrow \mathrm{daughter}(x, y) \tag{4}$$

and

$$x_2 = \mathrm{female}(x) \wedge \mathrm{mother}(y, x) \rightarrow \mathrm{daughter}(x, y) \tag{5}$$

is

$$\mathrm{female}(x) \rightarrow \mathrm{daughter}(x, y)$$

This does not seem satisfactory since the 'natural' generalization would be

$$\mathrm{female}(x) \wedge \mathrm{parent}(y, x) \rightarrow \mathrm{daughter}(x, y) \tag{6}$$

The learning agent would need to 'know' beforehand that father and mother are special cases of parent.

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Background Knowledge: Motivation (cont'd)

We aim at generalization with respect to **background knowledge**, i.e., formal knowledge available prior to learning. Schematically:

$$\text{female}(x) \wedge \text{parent}(y, x) \rightarrow \text{daughter}(x, y)$$

$$\text{father}(x, y) \rightarrow \text{parent}(x, y)$$
$$\text{mother}(x, y) \rightarrow \text{parent}(x, y)$$

Background knowledge

$$\text{female}(x) \wedge \text{mother}(y, x) \rightarrow \text{daughter}(x, y)$$

$$\text{female}(x) \wedge \text{father}(y, x) \rightarrow \text{daughter}(x, y)$$

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

For clausal hypotheses h, clausal observations x, *and background knowledge B*, we extend the consequence-based policy (??) to **relative consequence** policy

$$h(x) = 1 \text{ iff } B \wedge h \models x \tag{7}$$

Note that with $B =$

$$(\mathrm{father}(x, y) \rightarrow \mathrm{parent}(x, y)) \wedge (\mathrm{mother}(x, y) \rightarrow \mathrm{parent}(x, y))$$

we have $B \wedge h \models x_1$, $B \wedge h \models x_2$ for x_1 (4), x_2 (5), h (6) as expected.

Just like we reduced (??) to (1) under the assumption on non-self-resolving examples, we investigate on which assumptions on x and B we reduce the relation in (7) to one based on $\subseteq_\theta$.

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Relative Subsumption

A formula is **ground** if it contains no variables.

Let h, h' be FOL clauses and B a ground FOL conjunction. We say that

- h **subsumes** h' **relative to** B (written $h \subseteq_\theta^B h'$) if

$$h \subseteq_\theta (B \rightarrow h') \quad (8)$$

- h **strictly subsumes** h' **relative to** B, if $h \subseteq_\theta^B h'$ but $h' \not\subseteq_\theta^B h$.

Theorem 1

Let h, h' be FOL clauses and B a ground FOL conjunction. If $h \subseteq_\theta h'$ then $h \subseteq_\theta^B h'$.

The proof is an exercise.

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Relative Subsumption vs. Relative Consequence

Theorem 2

Let h, h' be FOL clauses. If $h \subseteq_\theta^B h'$ then $B \wedge h \models h'$. Let furthermore h not be self-resolving. Then $h \subseteq_\theta^B h'$ if and only if $B \wedge h \models h'$.

Proof: $h \models h'$ iff $\models h \rightarrow h'$ (i.e. iff the implication is tautologically true), thus $B \wedge h \models h'$ iff $h \models B \rightarrow h'$. Since B is a conjunction and h' a clause, $B \rightarrow h'$ is a clause. The asserted relationships between $B \wedge h \models h'$ (i.e., $h \models B \rightarrow h'$) and $h \subseteq_\theta^B h'$ (i.e., $h \subseteq_\theta B \rightarrow h'$) thus follow from Theorem (??).

Relative Subsumption: Example

$$B = \mathrm{male}(\mathsf{john}) \wedge \mathrm{parent}(\mathsf{ann}, \mathsf{john})$$
$$h = \mathrm{male}(x) \wedge \mathrm{parent}(y, x) \rightarrow \mathrm{son}(x, y)$$

We have

$$B \wedge h \models \mathrm{son}(\mathsf{john}, \mathsf{ann})$$

which can be shown through a resolution proof. But also

$$h \subseteq_\theta^B \mathrm{son}(\mathsf{john}, \mathsf{ann})$$

because

$$h \subseteq_\theta (\mathrm{male}(\mathsf{john}) \wedge \mathrm{parent}(\mathsf{ann}, \mathsf{john}) \rightarrow \mathrm{son}(\mathsf{john}, \mathsf{ann}))$$

Verify with $\theta = \{ x \mapsto \mathsf{john}, y \mapsto \mathsf{ann} \}$.

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Relative Equivalence and Relative Reduction

For FOL clauses, we extend the equivalence and reduction definitions straightforwardly to account for background knowledge.

Let h, h' be FOL clauses and B a ground FOL conjunction. We say that

- **h, h' are equivalent relative to B** (written $h \approx_\theta^B h'$) if $h \subseteq_\theta^B h'$ and $h' \subseteq_\theta^B h$.
- **h is reduced relative to B** if for no FOL clause g such that $\text{Lits}(g) \subset \text{Lits}(h)$ it holds $g \approx_\theta^B h$.
- **h' is a reduction of h relative to B** if $h \approx_\theta^B h'$ and h' is reduced relative to B.

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Relative Equivalence and Relative Reduction (cont'd)

To obtain a reduction of

$$h = h^1 \vee h^2 \vee \dots h^{n_h}$$

relative to

$$B = b^1 \wedge b^2 \wedge \dots b^{n_B}$$

first realize that any literal of h that is also in B with the opposite sign can be removed from h (obtaining h'). To see this for $h^i = \neg b^j$,

$$B \to h = b^1 \wedge \dots \wedge b^j \wedge \dots b^{n_B} \to h^1 \vee \dots \vee h^i \vee \dots h^{n_h}$$
$$\models\!\mid \neg b^1 \vee \dots \vee \neg b^j \vee \dots \neg b^{n_B} \vee h^1 \vee \dots \vee h^i \vee \dots h^{n_h}$$

so the literal $h^i = \neg b^j$ occurs twice in $B \to h$. Thus $B \to h \approx_\theta B \to h'$ and so by definition, also $h \approx_\theta^B h'$. After the removal of all such literals, use the reduction algorithm on h'.

Relative Least General Generalization

Let h, h' be FOL clauses and B a ground FOL conjunction. g is a **least general generalization of h and h' relative to B** if $g \subseteq_\theta^B h$, $g \subseteq_\theta^B h'$, and there is no FOL clause g' such that $g \subset_\theta^B g'$, $g' \subseteq_\theta^B h$, $g' \subseteq_\theta^B h'$.

In other words, g is a least general generalization of h and h' relative to B iff it is a least general generalization of $B \to h$ and $B \to h'$. *To see this, check the definitions of $\subseteq_\theta^B$, $\subset_\theta^B$.*

Define $\mathrm{rlgg}_B(h, h')$ as $\mathrm{lgg}(B \to h, B \to h')$. Corollary of Theorem (??): $\mathrm{rlgg}_B(h, h')$ is a least general generalization of h and h' relative to B.

(Implementing rlgg with a subsequent clause reduction is the subject of Project 2.)

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Example: Relative Least General Generalization

$$B = \mathrm{female}(\mathsf{a}) \wedge \mathrm{parent}(\mathsf{a}, \mathsf{b}) \wedge \mathrm{male}(\mathsf{b}) \wedge \mathrm{parent}(\mathsf{b}, \mathsf{c}) \wedge \mathrm{male}(\mathsf{c}) \tag{9}$$

$$x_1 = \mathrm{son}(\mathsf{b}, \mathsf{a})$$
$$x_2 = \mathrm{son}(\mathsf{c}, \mathsf{b})$$
$$\mathrm{rlgg}_B(x_1, x_2) = \mathsf{lgg}(B \rightarrow x_1, B \rightarrow x_2)$$

Rewrite $B \rightarrow x_1$ as

$$\mathrm{son}(\mathsf{b}, \mathsf{a}) \vee \neg \mathrm{female}(\mathsf{a}) \vee \neg \mathrm{parent}(\mathsf{a}, \mathsf{b}) \vee \neg \mathrm{male}(\mathsf{b}) \vee \neg \mathrm{parent}(\mathsf{b}, \mathsf{c}) \vee \neg \mathrm{male}(\mathsf{c})$$

Rewrite $B \rightarrow x_2$ as

$$\mathrm{son}(\mathsf{c}, \mathsf{b}) \vee \neg \mathrm{female}(\mathsf{a}) \vee \neg \mathrm{parent}(\mathsf{a}, \mathsf{b}) \vee \neg \mathrm{male}(\mathsf{b}) \vee \neg \mathrm{parent}(\mathsf{b}, \mathsf{c}) \vee \neg \mathrm{male}(\mathsf{c})$$

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Example: Relative Least General Generalization (cont'd)

Now compute the lgg as in the older example. Below, only the first letters of predicate symbols are shown.

	$\mathrm{s}(\mathsf{c}, \mathsf{b})$	$\neg\mathrm{f}(\mathsf{a})$	$\neg\mathrm{p}(\mathsf{a}, \mathsf{b})$	$\neg\mathrm{m}(\mathsf{b})$	$\neg\mathrm{p}(\mathsf{b}, \mathsf{c})$	$\neg\mathrm{m}(\mathsf{c})$
$\mathrm{s}(\mathsf{b}, \mathsf{a})$	$\mathrm{s}(v_1, v_2)$					
$\neg\mathrm{f}(\mathsf{a})$		$\neg\mathrm{f}(\mathsf{a})$				
$\neg\mathrm{p}(\mathsf{a}, \mathsf{b})$			$\neg\mathrm{p}(\mathsf{a}, \mathsf{b})$		$\neg\mathrm{p}(v_2, v_1)$	
$\neg\mathrm{m}(\mathsf{b})$				$\neg\mathrm{m}(\mathsf{b})$		$\neg\mathrm{m}(v_1)$
$\neg\mathrm{p}(\mathsf{b}, \mathsf{c})$			$\neg\mathrm{p}(v_3, v_4)$		$\neg\mathrm{p}(\mathsf{b}, \mathsf{c})$	
$\neg\mathrm{m}(\mathsf{c})$				$\neg\mathrm{m}(v_4)$		$\neg\mathrm{m}(\mathsf{c})$

θ	σ	new variable
b	c	v_1
a	b	v_2
b	a	v_3
c	b	v_4

So $\mathrm{rlgg}_B(x_1, x_2) =$

$$\mathrm{s}(v_1, v_2) \vee \neg\mathrm{f}(\mathsf{a}) \vee \neg\mathrm{p}(\mathsf{a}, \mathsf{b}) \vee \neg\mathrm{p}(v_2, v_1) \vee \neg\mathrm{m}(\mathsf{b}) \vee$$
$$\neg\mathrm{m}(v_1) \vee \neg\mathrm{p}(v_3, v_4) \vee \neg\mathrm{p}(\mathsf{b}, \mathsf{c}) \vee \neg\mathrm{m}(v_4) \vee \neg\mathrm{m}(\mathsf{c}) \qquad (10)$$

Now reduce the clause.

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Example: Relative Clause Reduction

Reduce $h =$ (10) relative to $B =$ (9).

First remove literals of h that are in B with the opposite sign, obtaining

$$h := \mathrm{s}(v_1, v_2) \vee \neg\mathrm{p}(v_2, v_1) \vee \neg\mathrm{m}(v_1) \vee \neg\mathrm{p}(v_3, v_4) \vee \neg\mathrm{m}(v_4)$$

Now follow the reduction algorithm

$$h \subseteq_\theta (h \setminus \neg\mathrm{p}(v_3, v_4)) \text{ with } \theta = \{ v_3 \mapsto v_1, v_4 \mapsto v_2 \}$$
$$h := h\theta = \mathrm{s}(v_1, v_2) \vee \neg\mathrm{p}(v_2, v_1) \vee \neg\mathrm{m}(v_1)$$

Rewriting h into the implication form and plugging in the full predicate symbols, we get

$$\mathrm{parent}(v_2, v_1) \wedge \mathrm{male}(v_1) \to \mathrm{son}(v_1, v_2)$$

which is the expected generalization. (*exercise problem*)

Herbrand Semantics

For a finite set $\mathcal{P}$ of predicates and a finite set $\mathcal{F}$ of functions (including constants)

- the **Herbrand base** HB is the set of all ground atoms made using $\mathcal{P}$ and $\mathcal{F}$.
- a **Herbrand interpretation** HI is a subset of the HB.

A HI is a special case of FOL interpretation so we write $HI \models \phi$ whenever HI is a model of FOL formula ϕ.

In propositional logic as a special case of FOL, a HI is just a truth assignment to all the unary atoms, more precisely, the set of all atoms assigned the True value.

st-CNF

A positive learnability result for a FOL hypothesis class can be established if we impose *size bounds* on the target hypothesis, like we did for *s*-DNF, *s*-CNF, *s*-term DNF, and *s*-clause CNF.

An ***st*-clause** is a FOL clause with at most s literals and at most t term occurrences in each literal. So e.g.

$$\mathrm{born}(x) \rightarrow \mathrm{reproduced}(\mathsf{mother}(x), \mathsf{father}(x))$$

has 2 literals with 1 term occurrence in the LHS literal and 4 term occurrences in the RHS literal. So it is a 2,4-clause but not e.g. a 2,3-clause.

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

Online Learnability of *st*-CNF

Consider learning from Herbrand interpretations, i.e. FOL analogies of truth assignments $X = \{0, 1\}^n$ to propositional variables.

Theorem 3

Let X contain Herbrand interpretations for a finite set of $\mathcal{P}$ predicates and a finite set $\mathcal{F}$ of functions, and the observation complexity n_X be the tuple $(|\mathcal{P}|, |\mathcal{F}|)$. The hypothesis class st-CNF is learnable online from X.

The proof is an exercise.

As observations have to be finite, Herbrand interpretations are strictly *less expressive* than clausal or conjunctive observations. For example, with $\mathcal{P} = \{\mathrm{p}/1\}$ and $\mathcal{F} = \{\mathsf{a}/0, \mathsf{f}/1\}$, $\mathrm{p}(x)$ has exactly one Herbrand model, which is the *infinite* Herbrand interpretation $\{\mathrm{p}(\mathsf{a}), \mathrm{p}(\mathsf{f}(\mathsf{a})), \mathrm{p}(\mathsf{f}(\mathsf{f}(\mathsf{a}))), \ldots\}$.

FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE