Resolution: brief summary

Full first-order version:

$$\frac{\ell_1 \vee \cdots \vee \ell_k, \qquad m_1 \vee \cdots \vee m_n}{(\ell_1 \vee \cdots \vee \ell_{i-1} \vee \ell_{i+1} \vee \cdots \vee \ell_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n)\theta}$$

where $\textsc{Unify}(\ell_i, \neg m_j) = \theta$.

For example, $\frac{\neg Rich(x) \vee Unhappy(x) \qquad Rich(Ken)}{Unhappy(Ken)}$ with $\theta = \{x/Ken\}$

Apply resolution steps to $CNF(KB \wedge \neg\alpha)$; complete for FOL

(1) $\neg P(w) \vee Q(w)$ $\qquad$ $\neg Q(y) \vee S(y)$ (2)

$\{y/w\}$

+ standardize variables apart

$\neg P(w_1) \vee S(w_1)$ (3)

Note: before the resolution, the set of clauses is composed of clauses (1) and (2)

after the resolution, the resolvent is added to the set of clauses — it is then composed of clauses (1), (2), and (3)
all with different variable names

1. $P(a, x, f(g(y)))$ Premise

Prove $Q(g(u), u)$

2. $\neg P(z, f(z), f(w)) \vee Q(w, z)$ Premise

3. $\neg Q(g(u), u)$ Negation of what is to be proved

4. $Q(g(y_1), a)$ $1, 2, \{z/a, x/f(a), w/g(y)\}$

+ standardize apart variables

5. $\square$ $3, 4, \{y_1/a, u/a\}$

Some further preliminaries

$$\forall x\, \alpha \equiv \neg \exists x \neg \alpha$$

$$\exists x\, \alpha \equiv \neg \forall x \neg \alpha$$

So, when you move a $\neg$ across a quantifier, flip the quantifier and negate the formula that it scopes (eg $\neg \exists x \alpha \Rightarrow \forall x \neg \alpha$)

Conversion to CNF

Everyone who loves all animals is loved by someone:

$\forall x \Big([\forall y \Big(Animal(y) \implies Loves(x,y) \Big)] \implies [\exists y Loves(y,x)] \Big)$

1. Eliminate biconditionals and implications

$$\forall x \Big([\neg \forall y \Big(\neg Animal(y) \vee Loves(x,y) \Big)] \vee [\exists y Loves(y,x)] \Big)$$

2. Move $\neg$ inwards: $\neg \forall x, p \equiv \exists x \neg p, \quad \neg \exists x, p \equiv \forall x \neg p$:

$\forall x \Big([\exists y \Big(\neg(\neg Animal(y) \vee Loves(x,y)) \Big)] \vee [\exists y Loves(y,x)] \Big)$

$\forall x \Big([\exists y \neg\neg Animal(y) \wedge \neg Loves(x,y) \Big)] \vee [\exists y Loves(y,x)]$

$\forall x \Big([\exists y Animal(y) \wedge \neg Loves(x,y) \Big)] \vee [\exists y Loves(y,x)]$

3. Standardize variables: each quantifier should use a different one

$$\forall x \Big([\exists y \Big(Animal(y) \wedge \neg Loves(x,y) \Big)] \vee [\exists z Loves(z,x)] \Big)$$

Conversion to CNF contd.

4. Skolemize: a more general form of existential instantiation. Each existential variable is replaced by a **Skolem function** of the enclosing universally quantified variables:

$$\forall x \Big([Animal(F(x)) \land \neg Loves(x, F(x))] \lor Loves(G(x), x) \Big)$$

5. Drop universal quantifiers:

$$[Animal(F(x)) \land \neg Loves(x, F(x))] \lor Loves(G(x), x)$$

6. Distribute $\land$ over $\lor$:

$$[Animal(F(x)) \lor Loves(G(x), x)] \land [\neg Loves(x, F(x)) \lor Loves(G(x), x)]$$

7. Break into clauses and Rename variables:

$$[Animal(F(x)) \lor Loves(G(x), x)] \quad [\neg Loves(x_1, F(x_1)) \lor Loves(G(x_1), x_1)]$$

Resolution proof — Did curiosity kill the cat?

1. Jack owns a dog.
2. Every dog owner is an animal lover.
3. No animal lover kills an animal.
5. Either Jack or Curiosity killed the cat,
4. who is named Tuna.
6. Cats are animals.

① $\exists x \, (Dog(x) \wedge Owns(Jack, x))$

② $\forall x \, ((\exists y \, Dog(y) \wedge Owns(x, y)) \rightarrow AnimalLover(x))$

③ $\forall x \, (AnimalLover(x) \rightarrow \forall y \, (Animal(y) \rightarrow \neg Kills(x, y)))$

④ $Cat(Tuna)$

⑤ $Kills(Jack, Tuna) \vee Kills(Curiosity, Tuna)$

⑥ $\forall x \, (Cat(x) \rightarrow Animal(x))$

CNF of ①

$\underbrace{Dog(Rover)}_{\text{clause}} \wedge \underbrace{Owns(Jack, Rover)}_{\text{clause}}$

CNF of ④

$\underbrace{Cat(Tuna)}_{\text{clause}}$

CNF of ⑤

$\underbrace{Kill(Jack, Tuna) \vee Kill(Curosity, Tuna)}_{\text{clause}}$

CNF of 6

$$\underbrace{\neg Cat(x) \lor Animal(x)}_{\text{clause}}$$

CNF of 3

$$\forall x\ (\neg AnimalLover(x) \lor \forall y(\neg Animal(y) \lor \neg Kills(x,y)))$$

$$\underbrace{(\neg AnimalLover(x) \lor \neg Animal(y) \lor \neg Kills(x,y))}_{\text{clause}}$$

CNF of ②

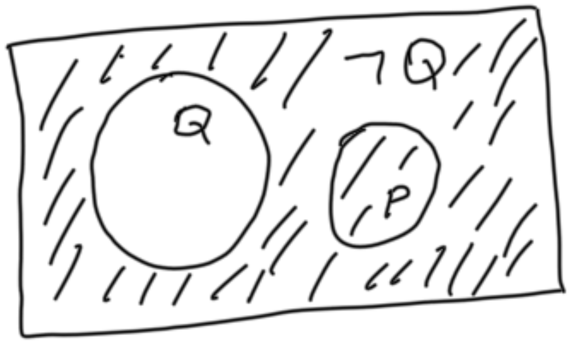
$$\forall x \, (\neg \exists y \, (Dog(y) \wedge Owns(x,y)) \vee AnimalLover(x))$$

$$\forall x \, (\forall y \, \neg (Dog(y) \wedge Owns(x,y)) \vee AnimalLover(x))$$

$$\forall x \, (\forall y \, (\neg Dog(y) \vee \neg Owns(x,y)) \vee AnimalLover(x))$$

$$\underbrace{\neg Dog(y) \vee \neg Owns(x,y) \vee AnimalLover(x)}_{\text{clause}}$$

No Ps are Qs

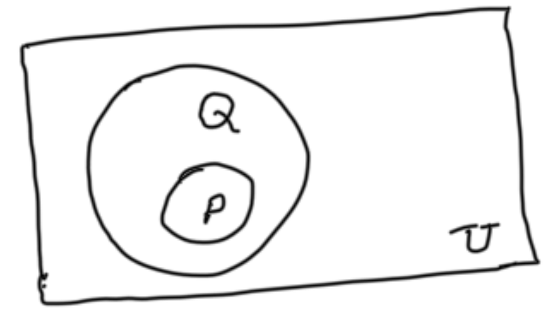


All Ps are not Qs

$$\forall x\,(P(x) \rightarrow \neg Q(x))$$

All Ps are Qs

$$\forall x\,(P(x) \rightarrow Q(x))$$



To obtain the correct representation for the other statement, cannot simply put ¬ in front of this statement

(remember the discussion at the beginning of FOL)

$$\forall x\,(P(x) \rightarrow Q(x))$$

$$\neg\,\forall x\,(\underbrace{\neg P(x) \vee Q(x)}_{\rightarrow})$$

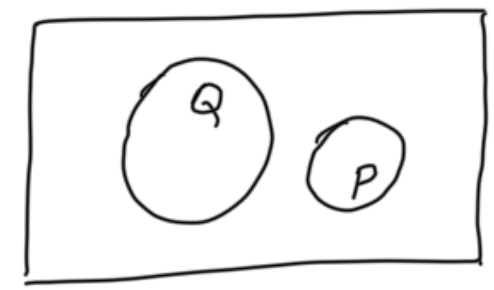
negate the previous sentence

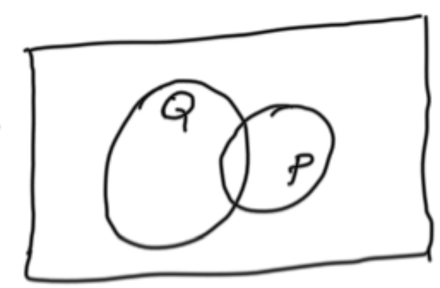
$\forall x \equiv \neg \exists x \neg$

$$\exists x\,\neg(\neg P(x) \vee Q(x))$$

$$\exists x\,(P(x) \wedge \neg Q(x))$$

No Ps are Qs





Some Ps are Qs and some Ps are not Qs

(Not all Ps are Qs)

Prenex Normal Form

quantifiers (matrix)

matrix is quantifier-free

To convert to PNF:

biconditionals must be converted to implications

standardize the variables apart

move negations inwards (to become literal negations)

now, going from inside out, move quantifiers to the front of the sentence using the following rules

Prenex rules

1) $(Q\phi) * \psi \equiv Q(\phi * \psi)$

where Q can be $\forall$ or $\exists$ and $*$ can be $\wedge$ or $\vee$

2) $\rightarrow$ when moving quantifier from antecedent: flip the quantifier

when moving quantifier from consequent: no change

Example:

$(P(a) \vee \exists x\, Q(x)) \rightarrow \forall z\, R(z)$ ← variables are standardized apart

$(\exists x\, (P(a) \vee Q(x))) \rightarrow \forall z\, R(z)$ (1)

$\forall x\, ((P(a) \vee Q(x)) \rightarrow \forall z\, R(z))$ (2)

$\forall x\, \forall z\, ((P(a) \vee Q(x)) \rightarrow R(z))$ (2)

$$\forall x\,(\text{AnimalLover}(x) \rightarrow \underbrace{\forall y\,(\text{Animal}(y) \rightarrow \neg \text{Kills}(x,y))}_{\text{consequent}})$$

$$\forall x\,\forall y\,(\text{AnimalLover}(x) \rightarrow (\text{Animal}(y) \rightarrow \neg \text{Kills}(x,y)))$$

$$P \rightarrow (Q \rightarrow R)$$

$$\equiv (P \wedge Q) \rightarrow R$$

$$\forall x\,\forall y\,((\text{AnimalLover}(x) \wedge \text{Animal}(y)) \rightarrow \neg \text{Kills}(x,y)))$$

Prove $Kills(Curiosity, Tuna)$

1. $Dog(Rover)$
2. $Owns(Jack, Rover)$
3. $Cat(Tuna)$
4. $Kills(Jack, Tuna) \vee Kills(Curiosity, Tuna)$
5. $\neg Dog(y) \vee \neg Owns(x, y) \vee AnimalLover(x)$
6. $\neg Cat(z) \vee Animal(z)$
7. $\neg AnimalLover(x') \vee \neg Animal(y') \vee \neg Kills(x', y')$
8. $\neg Kills(Curiosity, Tuna)$

9.	$Kills(Jack, Tuna)$	4, 8, $\{\}$
10.	$\neg Owns(x'', Rover) \vee AnimalLover(x'')$	1, 5, $\{y/Rover\}$
11.	$AnimalLover(Jack)$	2, 10, $\{x''/Jack\}$
12.	$Animal(Tuna)$	3, 6, $\{z/Tuna\}$
13.	$\neg Animal(y'') \vee \neg Kills(Jack, y'')$	7, 11, $\{x'/Jack\}$
14.	$\neg Kills(Jack, Tuna)$	12, 13, $\{y''/Tuna\}$
15.	$\square$	9, 14, $\{\}$

Prove $\neg L(a,b)$

1. $L(f(x_2, y_2), g(z_2)) \lor \neg L(y_2, z_2)$

2. $\neg L(f(x_3, f(c, f(d,a))), w_3)$

3. $L(a,b)$

4. $L(f(x_4, a), g(b))$ 1, 3, $\{y_2/a,\ z_2/b\}$ + standardize apart variables

5. $L(f(x_5, f(x_6, a)), g(g(b)))$ 1, 4, $\{y_2/f(x_4, a),\ z_2/g(b)\}$ + standardize apart variables

6. $L(f(x_9, f(x_7, f(x_8, a), g(g(g(b))))))$ 1, 5, $\{y_2/f(x_5, f(x_6, a)),\ z_2/g(g(b))\}$ + standardize apart variables

7. $\square$ 2, 6, $\{x_9/x_3,\ x_7/c,\ x_8/d,\ w_3/g(g(g(b)))\}$

1. $L(a,b)$
2. $L(f(x,y),g(z)) \vee \neg L(y,z)$
3. $\neg L(f(x_2,f(c,f(d,a))),w)$

to begin, variables are standardized apart

4. $L(f(x,a),g(b))$ — $1,2, \{y/a, z/b\}$
5. $L(f(x,f(x,a)),g(g(b)))$ — $2,4, \{y/f(x,a), z/g(b)\}$
6. $L(f(x,f(x,f(x,a))),g(g(g(b))))$ — $2,5, \{y/f(x,f(x,a)), z/g(g(b))\}$

Now we have the shape of the arguments to allow the resolution of 3 and 6.

$L(f(x,f(x,f(x,a))),g(g(g(b))))$ $\quad$ $\neg L(f(x_2,f(c,f(d,a))),w)$

but now x has to unify with x_2, c, and d $\quad$ Impossible!

Restricted Language

Definite clauses in the FOL setting

→ called Datalog

$h \leftarrow a_1 \wedge a_2 \wedge \ldots \wedge a_n$

each of h and a_i are first order atoms

if $n=0$ these are called facts and it has an empty body

if $n>0$ these are called rules

Restricted language (continued)

quantification is not used in the Datalog language but all variables are implicitly universally quantified in the scope of the clause

i.e.,
$$P(x) \leftarrow Q(x, y)$$
is to be interpreted as
$$\forall x \forall y \left(P(x) \leftarrow Q(x, y) \right)$$

Bottom-up proof procedure

$KB \vdash g$ if there is g' added to C in this procedure where $g = g'\theta$:

$C := \{\}$;
repeat
    select clause "$h \leftarrow b_1 \wedge \ldots \wedge b_m$" in KB such that
        there is a substitution θ such that
        for all i, there exists $b'_i \in C$ where $b_i\theta = b'_i\theta$ and
        there is no $h' \in C$ such that h' is more general than $h\theta$
    $C := C \cup \{h\theta\}$
until no more clauses can be selected.

Example

* remember in this and the slides to the end of this lecture, predicates and constants are in lower case and variables are in uppercase

$$live(Y) \leftarrow connected\_to(Y, Z) \wedge live(Z). \quad live(outside).$$

$$connected\_to(w_6, w_5). \quad connected\_to(w_5, outside).$$

For bottom up proof procedure we want to know everything that is provable from these 4 definite clauses.

Example

(1) (2)

$live(Y) \leftarrow connected\_to(Y, Z) \wedge live(Z).$ $live(outside).$

(3) $connected\_to(w_6, w_5).$ $connected\_to(w_5, outside).$ (4)

$C = \{live(outside),$ (2)

$connected\_to(w_6, w_5),$ (3)

$connected\_to(w_5, outside),$ (4)

(5) $live(w_5),$ (1)(2)(4) b_2 b_1 $\theta = \{Z/outside, Y/w5\}$

$live(w_6)\}$ (1)(3)(5) b_1 b_2 $\theta = \{Z/w5, Y/w_6\}$

Top-down Proof procedure

- A generalized answer clause is of the form

  $$yes(t_1, \ldots, t_k) \leftarrow a_1 \wedge a_2 \wedge \ldots \wedge a_m,$$

  where $t_1, \ldots, t_k$ are terms and $a_1, \ldots, a_m$ are atoms.
- The SLD resolution of this generalized answer clause on a_i with the clause

  $$a \leftarrow b_1 \wedge \ldots \wedge b_p,$$

  where a_i and a have most general unifier θ, is

  $$(yes(t_1, \ldots, t_k) \leftarrow$$
  $$a_1 \wedge \ldots \wedge a_{i-1} \wedge b_1 \wedge \ldots \wedge b_p \wedge a_{i+1} \wedge \ldots \wedge a_m)\theta.$$

Top-down Proof Procedure

To solve query $?B$ with variables $V_1, \ldots, V_k$:

Set *ac* to generalized answer clause $yes(V_1, \ldots, V_k) \leftarrow B$
while *ac* is not an answer **do**
    Suppose *ac* is $yes(t_1, \ldots, t_k) \leftarrow a_1 \wedge a_2 \wedge \ldots \wedge a_m$
    select atom a_i in the body of *ac*
    choose clause $a \leftarrow b_1 \wedge \ldots \wedge b_p$ in KB
    Rename all variables in $a \leftarrow b_1 \wedge \ldots \wedge b_p$
    Let θ be the most general unifier of a_i and a.
        Fail if they don't unify
    Set *ac* to $(yes(t_1, \ldots, t_k) \leftarrow a_1 \wedge \ldots \wedge a_{i-1} \wedge$
        $b_1 \wedge \ldots \wedge b_p \wedge a_{i+1} \wedge \ldots \wedge a_m)\theta$
end while.

Answer is $V_1 = t_1, \ldots, V_k = t_k$
where *ac* is $yes(t_1, \ldots, t_k) \leftarrow$

Example

$live(Y) \leftarrow connected\_to(Y, Z) \land live(Z).$ $live(outside).$
$connected\_to(w_6, w_5).$ $connected\_to(w_5, outside).$
$?live(A).$

this is the query (i.e., the answer clause)

it has a variable, so the query is asking "what is live?"

$$live(Y) \leftarrow connected\_to(Y, Z) \wedge live(Z). \quad live(outside).$$

$$connected\_to(w_6, w_5). \quad connected\_to(w_5, outside).$$

$$?live(A).$$

$$yes(A) \leftarrow live(A).$$

Rename Y and Z

$$yes(A) \leftarrow connected\_to(A, Z_1) \wedge live(Z_1).$$

$\theta = \{Y_1/A\}$

$$yes(w_6) \leftarrow live(w_5).$$

$\theta = \{A/w_6, Z$

Rename Y and

$$yes(w_6) \leftarrow connected\_to(w_5, Z_2) \wedge live(Z_2).$$

$\theta = \{Y_2/w_5$

$$yes(w_6) \leftarrow live(outside).$$

$\theta = \{Z_2/outside\}$

$$yes(w_6) \leftarrow .$$

$A = w_6$

Could have also chosen:

$yes(A) \leftarrow live(A)$

$\theta = \{A/outside\}$

$yes(outside) \leftarrow$

$A = outside$

Could have also chosen:

Rename Y and Z

$yes(A) \leftarrow live(A)$

$\theta = \{Y_1/A\}$

$yes(A) \leftarrow connected\_to(A, Z_1) \wedge live(Z_1)$

$\theta = \{A/w_5, Z_1/outside\}$

$yes(w_5) \leftarrow live(outside)$

$yes(w_5) \leftarrow$

$A = w_5$