Example

Consider the following sentences:

1. If it rains, the aquaphobes won’t vote
2. John will win only if the aquaphobes and vegetarians vote
3. Either John or Peter will win, but not both
4. If it rains, Peter will win.

How would these look in propositional logic?
Representation

• R (It rains); A (Aquaphobes Vote); J (John Wins); V (Vegetarians Vote); P (Peter Wins)

1. If it rains, the aquaphobes won’t vote:
2. John will win only if the aquaphobes and vegetarians vote:
3. Either John or Peter will win, but not both:
4. If it rains, Peter will win:
Representation

- R (It rains); A (Aquaphobes Vote); J (John Wins); V (Vegetarians Vote); P (Peter Wins)

1. If it rains, the aquaphobes won’t vote: $R \rightarrow \neg A$
2. John will win only if the aquaphobes and vegetarians vote:
3. Either John or Peter will win, but not both:
4. If it rains, Peter will win:
Representation

- R (It rains); A (Aquaphobes Vote); J (John Wins); V (Vegetarians Vote); P (Peter Wins)

1. If it rains, the aquaphobes won’t vote: \( R \rightarrow \neg A \)
2. John will win only if the aquaphobes and vegetarians vote: \( A \land V \rightarrow J \)
3. Either John or Peter will win, but not both:
4. If it rains, Peter will win:
Representation

- R (It rains); A (Aquaphobes Vote); J (John Wins); V (Vegetarians Vote); P (Peter Wins)

1. If it rains, the aquaphobes won’t vote: \( R \rightarrow \neg A \)
2. John will win only if the aquaphobes and vegetarians vote: \( A \land V \rightarrow J \)
3. Either John or Peter will win, but not both: \( (J \lor P) \land \neg (J \land P) \)
4. If it rains, Peter will win:
Representation

- R (It rains); A (Aquaphobes Vote); J (John Wins); V (Vegetarians Vote); P (Peter Wins)

1. If it rains, the aquaphobes won’t vote: \( R \rightarrow \neg A \)
2. John will win only if the aquaphobes and vegetarians vote: \( A \land V \rightarrow J \)
3. Either John or Peter will win, but not both: \( (J \lor P) \land \neg (J \land P) \)
4. If it rains, Peter will win: \( R \rightarrow P \)
More Examples

Consider the following sentences:

- Tweety has feathers.
- Donald has feathers.
- John is the son of Jack.
- If it rains, aquaphobes won’t vote.
- If it is cloudy, some aquaphobes will vote.

Convert them into propositional logic.
Motivation

Problems with Propositional Logic

- Propositional logic does not allow us to formulate general rules or represent relations and is declarative
- Meaning in propositional logic is context-independent
- Propositional logic has very limited expressive power
- Reasons:
  - No “domain” variables
  - No quantification
- Solution:
  - Introduce domain variables and quantification
  - FOL
First Order Logic and Propositional Logic

Propositional logic assumes the world contains facts.

First-order logic assumes the world contains:

- **Objects**: people, houses, numbers, colors, baseball games, wars, . . .
- **Relations**: red, round, siblingof, biggerthan, partof, comesbetween, . . .
- **Functions**: onemorethan, plus, . . .
First Order Logic and Propositional Logic . . .

Ontological commitment

*what a representation assumes about existence.*
Temporal logic assumes that facts hold at specific times. Higher-order logic takes relations and functions of FOL to be objects themselves.
First Order Logic and Propositional Logic . . .

Ontological commitment

*what a representation assumes about existence.* Temporal logic assumes that facts hold at specific times. Higher-order logic takes relations and functions of FOL to be objects themselves.

Epistemological commitment

*what can be known about what exists.* In PL and FOL, sentences are believed to be true or false (or to have an unknown value).
Parts of Formulas

Formulas have:

- **Variables:**
  - $x, y, z$
  - Can be instantiated

- **Functions:**
  - $f, g, h$
  - Mappings

- **Constants:**
  - $3, \text{John}$
  - Individuals

- **Predicates:**
  - $P(x, y)$
  - Functions whose range is $\{\text{True, False}\}$

- **Quantifiers:**
  - $\forall$ for all
  - $\exists$ there exists
FOL: WFFs

A term is

- A constant, or
- A variable, or
- \( f(t_1, t_2, \ldots, t_n) \) where \( f \) is an n-ary function symbol, and \( t_i \) are terms
FOL: WFFs

Atom

If \( P \) is an \( n \)-place predicate symbol, and \( t_1, t_2, \ldots, t_n \) are terms, then

\[
P(t_1, \ldots, t_n)
\]

is an atom. It is sometimes referred to as an atomic formula.

Examples

- Reigns(Pope-Innocent-III, Europe, 1200)
- Reigns(You, World, 2013)
FOL: WFFs...

A sentence in FOL is either an atomic sentence or complex sentence.

A complex sentence is defined as:

**Complex sentence:**

- Sentence, or
- \( \neg \) Sentence, or
- Sentence \( \wedge \) Sentence, or
- Sentence \( \lor \) Sentence, or
- Quantifier Variable Sentence
Quantification

Universal

\[ \forall \langle \text{Variables} \rangle \langle \text{Sentence} \rangle \]

Everyone at NCSU is smart: \[ \forall x \text{At}(x, \text{NCSU}) \rightarrow \text{Smart}(x) \]

\[ \forall x \text{P} \text{ is true in a model } m \text{ iff } P \text{ is true where } x \text{ is each possible object in the model.} \]
Quantification

Universal

\( \forall \langle \text{Variables} \rangle \langle \text{Sentence} \rangle \)

Everyone at NCSU is smart: \( \forall x \text{At}(x, NCSU) \rightarrow \text{Smart}(x) \)

\( \forall x P \) is true in a model \( m \) iff \( P \) is true where \( x \) is each possible object in the model.

Existential

\( \exists \langle \text{Variables} \rangle \langle \text{Sentence} \rangle \)

Someone at NCSU is smart: \( \exists x \text{At}(x, NCSU) \land \text{Smart}(x) \)

\( \exists x P \) is true in a model \( m \) iff \( P \) is true where \( x \) is some possible object in the model.
Quantification

**Question:** What do the following mean?

- $\forall x \text{At}(x, NCSU) \land \text{Smart}(x)$
- $\exists x \text{At}(x, NCSU) \rightarrow \text{Smart}(x)$
Quantification . . .

∃x∃y is the same as ∃y∃x. ∀x∀y is the same as ∀y∀x.

But... ∀x∃y is not the same as ∃x∀y.

Consider:
∃x∀y\text{Loves}(x, y)
∀y∃x\text{Loves}(x, y)
∀x∃y\text{Loves}(x, y)
∀x∀y\text{Loves}(x, y)
Quantification . . .

\(\exists x \exists y\) is the same as \(\exists y \exists x\). \(\forall x \forall y\) is the same as \(\forall y \forall x\).

But... \(\forall x \exists y\) is not the same as \(\exists x \forall y\).

Consider:

\(\exists x \forall y \text{Loves}(x, y)\)
“There is a person who loves everyone in the world”

\(\forall y \exists x \text{Loves}(x, y)\)
\(\forall x \exists y \text{Loves}(x, y)\)
Quantification . . .

∃x∃y is the same as ∃y∃x. ∀x∀y is the same as ∀y∀x.

But... ∀x∃y is not the same as ∃x∀y.

Consider:
∃x∀yLoves(x, y)  
“There is a person who loves everyone in the world”

∀y∃xLoves(x, y)  
“Everyone in the world is loved by at least one person”

∀x∃yLoves(x, y)
Quantification . . .

\[ \exists x \exists y \text{ is the same as } \exists y \exists x. \quad \forall x \forall y \text{ is the same as } \forall y \forall x. \]

But... \[ \forall x \exists y \text{ is not the same as } \exists x \forall y. \]

Consider:
\[ \exists x \forall y \text{Loves}(x, y) \]
“There is a person who loves everyone in the world”

\[ \forall y \exists x \text{Loves}(x, y) \]
“Everyone in the world is loved by at least one person”

\[ \forall x \exists y \text{Loves}(x, y) \]
“Everyone in the world loves at least one person”
Quantification . . .

Quantifier duality: each can be expressed using the other.

\[ \forall x \text{Likes}(x, \text{IceCream}) \]
\[ \neg \exists x \neg \text{Likes}(x, \text{IceCream}) \]

And...

\[ \exists x \text{Likes}(x, \text{Broccoli}) \]
\[ \neg \forall x \neg \text{Likes}(x, \text{Broccoli}) \]
Free vs Bound Variables

Definition

An occurrence of a variable in a formula is bound iff the occurrence is in the scope of a quantifier employing the variable; otherwise it is free.

Examples

- $\forall x. P(x, y)$
- $x$ is bound
- $y$ is free

Predicate calculus requires all variables to be bound.
More Examples

Consider the sample sentences again:

- Tweety has feathers.
- Donald has feathers.
- John is the son of Jack.
- If it rains, aquaphobes won’t vote.
- If it is cloudy, some aquaphobes will vote.

Convert them into Predicate Logic.
A few conventions for FOL

It is important to standardize a few things to start with:

- Variables will always be lower case letters
- Constants will always be upper case letters
- Try not to use the same name for a function and a predicate
- \textit{predicateName/n} represents a predicate that can take \( n \) arguments
- A predicate may never be nested within another!
Example - Affine Spaces Theorem

Lexicon:
Constants: none
Variables: x, y
Functions: none
Predicates:

affine(x) – x is an affine space
perequiv(x,y) – spaces x and y are perceptually equivalent
euclidean(x) – x is a Euclidean space
dim(x) – we live in space x
trans(x,y) – space y is a transformation of space x
Example - Affine Spaces Theorem

Consider the sentences:

1. All affine spaces are perceptually equivalent.
2. All Euclidean spaces are affine.
3. Not all affine spaces are Euclidean.
4. The space we live in is Euclidean.
5. Not all transformations produce affine spaces.

Prove that:
all affine transformations of the space we live in are perceptually equivalent to it.
Affine Spaces . . .

1. All affine spaces are perceptually equivalent.
2. All Euclidean spaces are affine.
3. Not all affine spaces are Euclidean.
4. The space we live in is Euclidean.
5. Not all transformations produce affine spaces.
Affine Spaces . . .

1. All affine spaces are perceptually equivalent.
   \( \forall x (\forall y (\text{affine}(x) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \)

2. All Euclidean spaces are affine.

3. Not all affine spaces are Euclidean.

4. The space we live in is Euclidean.

5. Not all transformations produce affine spaces.
**Affine Spaces . . .**

1. All affine spaces are perceptually equivalent.
   \[ \forall x (\forall y (\text{affine}(x) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \]

2. All Euclidean spaces are affine.
   \[ \forall x (\text{euclidean}(x) \rightarrow \text{affine}(x)) \]

3. Not all affine spaces are Euclidean.

4. The space we live in is Euclidean.

5. Not all transformations produce affine spaces.
Affine Spaces . . .

1. All affine spaces are perceptually equivalent.
\[ \forall x (\forall y (\text{affine}(x) \land \text{affine}(y) \to \text{perequiv}(x, y))) \]

2. All Euclidean spaces are affine.
\[ \forall x (\text{euclidean}(x) \to \text{affine}(x)) \]

3. Not all affine spaces are Euclidean.
\[ \exists x (\text{affine}(x) \land \neg \text{euclidean}(x)) \]

4. The space we live in is Euclidean.

5. Not all transformations produce affine spaces.
Affine Spaces . . .

1. All affine spaces are perceptually equivalent.
\[ \forall x (\forall y (\text{affine}(x) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \]

2. All Euclidean spaces are affine.
\[ \forall x (\text{euclidean}(x) \rightarrow \text{affine}(x)) \]

3. Not all affine spaces are Euclidean.
\[ \exists x (\text{affine}(x) \land \neg \text{euclidean}(x)) \]

4. The space we live in is Euclidean.
\[ \forall x (\text{live}(x) \rightarrow \text{euclidean}(x)) \]

5. Not all transformations produce affine spaces.
Affine Spaces . . .

1. All affine spaces are perceptually equivalent.
   \[ \forall x (\forall y (\text{affine}(x) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \]

2. All Euclidean spaces are affine.
   \[ \forall x (\text{euclidean}(x) \rightarrow \text{affine}(x)) \]

3. Not all affine spaces are Euclidean.
   \[ \exists x (\text{affine}(x) \land \neg \text{euclidean}(x)) \]

4. The space we live in is Euclidean.
   \[ \forall x (\text{live}(x) \rightarrow \text{euclidean}(x)) \]

5. Not all transformations produce affine spaces.
   \[ \exists x (\exists y (\text{trans}(x, y) \land \neg \text{affine}(y))) \]
Affine Spaces . . .

**Conclusion:** all affine transformations of the space we live in are perceptually equivalent to it.
Affine Spaces . . .

**Conclusion:** all affine transformations of the space we live in are perceptually equivalent to it.

$$
\forall x (\forall y (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y)))
$$
So, how do we prove a theorem?

In order to prove a theorem, we must:

• represent the given sentences in CNF
• negate the conclusion – since we are proving by contradiction
• iteratively find a pair of clauses that can be resolved till we reach a contradiction
So, how do we prove a theorem?

In order to prove a theorem, we must:

- represent the given sentences in CNF
- negate the conclusion – since we are proving by contradiction
- iteratively find a pair of clauses that can be resolved till we reach a contradiction
CNF in predicate Logic

Consider the following clauses and identify which of these are in CNF (textbook, p. 346):

- $\forall x \ p(x) \rightarrow q(x)$
- $\neg \forall x \ P(x) \lor Q(x)$
- $\forall x \ P(x) \lor \exists x \ Q(x)$
- $\forall x \exists y \ P(x) \lor Q(y)$
CNF in predicate Logic

Consider the following clauses and identify which of these are in CNF (textbook, p. 346):

- \( \forall x \, p(x) \rightarrow q(x) \)
  - Remove ‘\( \rightarrow \)’: \( \forall x \, \neg p(x) \lor q(x) \)
- \( \neg \forall x \, P(x) \lor Q(x) \)
- \( \forall x \, P(x) \lor \exists x \, Q(x) \)
- \( \forall x \exists y \, P(x) \lor Q(y) \)
CNF in predicate Logic

Consider the following clauses and identify which of these are in CNF (textbook, p. 346):

- $\forall x \, p(x) \rightarrow q(x)$
  - Remove ‘→’: $\forall x \, \neg p(x) \vee q(x)$

- $\neg \forall x \, P(x) \vee Q(x)$
  - Move the negation inwards: $\exists x \, \neg P(x) \land \neg Q(x)$

- $\forall x \, P(x) \lor \exists x \, Q(x)$
- $\forall x \exists y \, P(x) \lor Q(y)$
CNF in predicate Logic

Consider the following clauses and identify which of these are in CNF (textbook, p. 346):

- \( \forall x \ p(x) \rightarrow q(x) \)
  Remove ‘\( \rightarrow \)’: \( \forall x \ \neg p(x) \lor q(x) \)
- \( \neg \forall x \ P(x) \lor Q(x) \)
  Move the negation inwards: \( \exists x \ \neg P(x) \land \neg Q(x) \)
- \( \forall x \ P(x) \lor \exists x \ Q(x) \)
  Standardize the variables: \( \forall x_1 \ P(x_1) \lor \exists x_2 \ Q(x_2) \)
- \( \forall x \exists y \ P(x) \lor Q(y) \)
CNF in predicate Logic

Consider the following clauses and identify which of these are in CNF (textbook, p. 346):

- $\forall x \ p(x) \rightarrow q(x)$
  Remove ‘→’: $\forall x \neg p(x) \lor q(x)$

- $\neg \forall x P(x) \lor Q(x)$
  Move the negation inwards: $\exists x \neg P(x) \land \neg Q(x)$

- $\forall x P(x) \lor \exists x Q(x)$
  Standardize the variables: $\forall x_1 P(x_1) \lor \exists x_2 Q(x_2)$

- $\forall x \exists y P(x) \lor Q(y)$
  Interesting...
CNF in predicate Logic

Consider the following clauses and identify which of these are in CNF (textbook, p. 346):

- $\forall x \ p(x) \rightarrow q(x)$
  Remove ‘$\rightarrow$’: $\forall x \neg p(x) \lor q(x)$
- $\neg \forall x \ P(x) \lor Q(x)$
  Move the negation inwards: $\exists x \neg P(x) \land \neg Q(x)$
- $\forall x \ P(x) \lor \exists x \ Q(x)$
  Standardize the variables: $\forall x_1 \ P(x_1) \lor \exists x_2 \ Q(x_2)$
- $\forall x \exists y \ P(x) \lor Q(y)$
  Interesting...

By the way, none of these are in CNF yet!!
Dealing with Quantifiers

We can handle ‘∃’ as follows:

- We eliminate the ‘∃’ by a process called Skolemization
- We will use special terms called Skolem Constants and Skolem Functions
- Consider:
  - $\forall x \exists y \, P(x) \lor Q(y)$
  - $\exists x \forall y \, P(x) \lor Q(y)$

Once this is complete, we can drop the ‘∀’ quantifier without any change to the corresponding variables.
Dealing with Quantifiers

We can handle ‘∃’ as follows:

• We eliminate the ‘∃’ by a process called Skolemization

• We will use special terms called Skolem Constants and Skolem Functions

• Consider:
  • \( \forall x \exists y P(x) \lor Q(y) \)
  Replace \( y \) with Skolem function ‘\( f(x) \)’: \( \forall x P(x) \lor Q(f(x)) \)

• \( \exists x \forall y P(x) \lor Q(y) \)

Once this is complete, we can drop the ‘∀’ quantifier without any change to the corresponding variables.
Dealing with Quantifiers

We can handle ‘∃’ as follows:

- We eliminate the ‘∃’ by a process called **Skolemization**
- We will use special terms called **Skolem Constants** and **Skolem Functions**
- Consider:
  - \( \forall x \exists y P(x) \lor Q(y) \)
    Replace \( y \) with Skolem function ‘\( f(x) \)’: \( \forall x P(x) \lor Q(f(x)) \)
  - \( \exists x \forall y P(x) \lor Q(y) \)
    Replace \( x \) with Skolem constant ‘\( X_1 \)’: \( \forall y P(X_1) \lor Q(y) \)

Once this is complete, we can drop the ‘∀’ quantifier without any change to the corresponding variables.
Example

- $\forall x \exists y (\text{man}(x) \rightarrow \text{woman}(y) \land \text{loves}(x, y))$
- $\exists x \forall y \forall z \exists u \forall v \exists w P(x, y, z, u, v, w)$
Example

- $\forall x \exists y (\text{man}(x) \rightarrow \text{woman}(y) \land \text{loves}(x, y))$
- $\forall x (\text{man}(x) \rightarrow \text{woman}(f(x)) \land \text{loves}(x, f(x)))$
- $\exists x \forall y \exists z \exists u \forall v \exists w \ P(x, y, z, u, v, w)$
Example

- $\forall x \exists y (man(x) \rightarrow woman(y) \land loves(x, y))$
- $\forall x (man(x) \rightarrow woman(f(x)) \land loves(x, f(x)))$
- $\exists x \forall y \forall z \exists u \forall v \exists w P(x, y, z, u, v, w)$
  - $\exists x \rightarrow A$
Example

• $\forall x \exists y (\text{man}(x) \rightarrow \text{woman}(y) \land \text{loves}(x, y))$
• $\forall x (\text{man}(x) \rightarrow \text{woman}(f(x)) \land \text{loves}(x, f(x)))$
• $\exists x \forall y \forall z \exists u \forall v \exists w P(x, y, z, u, v, w)$
  • $\exists x \rightarrow A$
  • $\exists u \rightarrow f(y, z)$
Example

- $\forall x \exists y (\text{man}(x) \rightarrow \text{woman}(y) \land \text{loves}(x, y))$
- $\forall x (\text{man}(x) \rightarrow \text{woman}(f(x)) \land \text{loves}(x, f(x)))$
- $\exists x \forall y \forall z \exists u \exists v \exists w \, P(x, y, z, u, v, w)$
  - $\exists x \rightarrow A$
  - $\exists u \rightarrow f(y, z)$
  - $\exists w \rightarrow g(y, z, v)$
Example

- $\forall x \exists y (\text{man}(x) \rightarrow \text{woman}(y) \land \text{loves}(x, y))$
- $\forall x (\text{man}(x) \rightarrow \text{woman}(f(x)) \land \text{loves}(x, f(x)))$
- $\exists x \forall y \forall z \exists u \forall v \exists w \quad P(x, y, z, u, v, w)$
  - $\exists x \rightarrow A$
  - $\exists u \rightarrow f(y, z)$
  - $\exists w \rightarrow g(y, z, v)$
- $\forall y \forall z \forall v \quad P(A, y, z, f(y, z), v, g(y, z, v))$
Back to Affine Spaces . . .

Convert the sentences into CNF:

1. All affine spaces are perceptually equivalent.
   \( \forall x (\forall y (\text{affine}(x) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \)

2. All Euclidean spaces are affine.
   \( \forall x (\text{euclidean}(x) \rightarrow \text{affine}(x)) \)

3. Not all affine spaces are Euclidean.
   \( \exists x (\text{affine}(x) \land \neg \text{euclidean}(x)) \)
Back to Affine Spaces . . .

Convert the sentences into CNF:

1. All affine spaces are perceptually equivalent.
   \[ \forall x (\forall y (\text{affine}(x) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \]
   \[ \neg \text{affine}(x) \lor \neg \text{affine}(y) \lor \text{perequiv}(x, y) \]

2. All Euclidean spaces are affine.
   \[ \forall x (\text{euclidean}(x) \rightarrow \text{affine}(x)) \]

3. Not all affine spaces are Euclidean.
   \[ \exists x (\text{affine}(x) \land \neg \text{euclidean}(x)) \]
Back to Affine Spaces . . .

Convert the sentences into CNF:

1. All affine spaces are perceptually equivalent.
   \[
   \forall x (\forall y (\text{affine}(x) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y)))
   \]
   \[
   \neg \text{affine}(x) \lor \neg \text{affine}(y) \lor \text{perequiv}(x, y)
   \]

2. All Euclidean spaces are affine.
   \[
   \forall x (\text{euclidean}(x) \rightarrow \text{affine}(x))
   \]
   \[
   \neg \text{euclidean}(x) \lor \text{affine}(x)
   \]

3. Not all affine spaces are Euclidean.
   \[
   \exists x (\text{affine}(x) \land \neg \text{euclidean}(x))
   \]
Back to Affine Spaces . . .

Convert the sentences into CNF:

1. All affine spaces are perceptually equivalent.
   \[ \forall x (\forall y (\text{affine}(x) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \]
   \[ \neg \text{affine}(x) \lor \neg \text{affine}(y) \lor \text{perequiv}(x, y) \]

2. All Euclidean spaces are affine.
   \[ \forall x (\text{euclidean}(x) \rightarrow \text{affine}(x)) \]
   \[ \neg \text{euclidean}(x) \lor \text{affine}(x) \]

3. Not all affine spaces are Euclidean.
   \[ \exists x (\text{affine}(x) \land \neg \text{euclidean}(x)) \]
   \[ \text{affine}(A) \land \neg \text{euclidean}(A) \]
Back to Affine Spaces . . .

Convert the sentences into CNF:

1. The space we live in is Euclidean.
   \[ \forall x (\text{live}(x) \rightarrow \text{euclidean}(x)) \]

2. Not all transformations produce affine spaces.
   \[ \exists x (\exists y (\text{trans}(x, y) \land \neg \text{affine}(y))) \]
Back to Affine Spaces . . .

Convert the sentences into CNF:

1. The space we live in is Euclidean.
   \[ \forall x (\text{live}(x) \rightarrow \text{euclidean}(x)) \]
   \[ \neg \text{live}(x) \lor \text{euclidean}(x) \]

2. Not all transformations produce affine spaces.
   \[ \exists x (\exists y (\text{trans}(x, y) \land \neg \text{affine}(y))) \]
Back to Affine Spaces . . .

Convert the sentences into CNF:

1. The space we live in is Euclidean.
   \( \forall x (\text{live}(x) \rightarrow \text{euclidean}(x)) \)
   \( \neg \text{live}(x) \lor \text{euclidean}(x) \)

2. Not all transformations produce affine spaces.
   \( \exists x (\exists y (\text{trans}(x, y) \land \neg \text{affine}(y))) \)
   \( \text{trans}(B, C) \land \neg \text{affine}(C) \)
Negated Conclusion for Affine Spaces

Conclusion:
All affine transformations of the space we live in are perceptually equivalent to it.

\[ \forall x(\forall y(\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \]
Negated Conclusion for Affine Spaces

Negated Conclusion:

\[
\neg \forall x (\forall y (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \\
\Rightarrow \neg \forall x (\forall y (\neg (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y))) \lor \neg \text{perequiv}(x, y))) \\
\Rightarrow \exists x (\exists y (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y) \land \neg \text{perequiv}(x, y))) \\
\Rightarrow \text{live}(D) \land \text{trans}(D, E) \land \text{affine}(E) \land \neg \text{perequiv}(D, E)
\]
Negated Conclusion for Affine Spaces

Negated Conclusion:

$$\neg \forall x (\forall y (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y)))$$
Negated Conclusion for Affine Spaces

Negated Conclusion:

\[-\forall x (\forall y (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y)))\]
\[\Rightarrow -\forall x (\forall y (\neg (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y)) \lor \text{perequiv}(x, y)))\]
Negated Conclusion for Affine Spaces

Negated Conclusion:

\[ \neg \forall x (\forall y (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \]

\[ \Rightarrow \neg \forall x (\forall y (\neg (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y))) \lor \text{perequiv}(x, y)) \]

\[ \Rightarrow \exists x (\exists y (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y) \land \neg \text{perequiv}(x, y))) \]
Negated Conclusion for Affine Spaces

Negated Conclusion:

\[
\neg \forall x (\forall y (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y) \rightarrow \text{perequiv}(x, y))) \\
\Rightarrow \neg \forall x (\forall y (\neg (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y))) \lor \text{perequiv}(x, y))) \\
\Rightarrow \exists x (\exists y (\text{live}(x) \land \text{trans}(x, y) \land \text{affine}(y) \land \neg \text{perequiv}(x, y))) \\
\Rightarrow \text{live}(D) \land \text{trans}(D, E) \land \text{affine}(E) \land \neg \text{perequiv}(D, E)
\]
Resolution

Consider:

\[ \neg euclidean(x_2) \lor affine(x_2) \]
\[ \neg affine(c) \]

These have a pair of complementary predicates. But, can we resolve them? If so, how?

Now assume:

\[ \neg euclidean(x_2) \lor affine(f(x_2)) \]
\[ \neg affine(c) \]
Substitutions

Definition

A substitution is a set of pairs \( \{t_1/v_1, t_2/v_2, \ldots, t_n/v_n\} \), where the \( t_i \) are terms, the \( v_i \) are variables, and no two \( v_i \) are the same.

Substitutions always replace variables with terms. Legal substitutions are

- constant-for-var
- function-for-var
- var-for-var

The vars substituted for may occur in either literal.
Substitutions

Definition

A substitution is a set of pairs \( \{t_1/v_1, t_2/v_2, \ldots, t_n/v_n\} \), where the \( t_i \) are terms, the \( v_i \) are variables, and no two \( v_i \) are the same.

Substitutions always replace variables with terms. Legal substitutions are

- constant-for-var
- function-for-var
- var-for-var

The vars substituted for may occur in either literal.
Unification

Definition

The process of finding a substitution that makes two literals complementary is called unification.

Two literals for which a unifying substitution exists are called unifiable.

Importance of Unification

- It is the basis for FOL resolution.
- It is the main way rule-based systems determine which rules apply in a situation.
- It is the way variables are treated in logic.