First-order logic:
Satisfiability, validity, logical consequence

Valentin Goranko

DTU Informatics

September 2010

Satisfiability and validity of sentences

A sentence A is:

- satisfiable if $\mathcal{S} \models A$ for some structure $\mathcal{S}$;
- (logically) valid, denoted $\models A$, if $\mathcal{S} \models A$ for every structure $\mathcal{S}$;
- falsifiable, if it is not logically valid, i.e. if it has a counter-model.

Satisfiability and validity of any first-order formulae

A first-order formula A is:

- A is satisfiable if $\mathcal{S}, v \models A$ for some structure $\mathcal{S}$ and some variable assignment v in $\mathcal{S}$.
- (logically) valid, denoted $\models A$, if $\mathcal{S}, v \models A$ for every structure $\mathcal{S}$ and every variable assignment v in $\mathcal{S}$.
- falsifiable, if it is not logically valid.

Let $A = A(x_1, \ldots, x_n)$ be any first-order formula all free variables in which are amongst $x_1, \ldots, x_n$.

The sentence $\exists x_1 \ldots \exists x_n A(x_1, \ldots, x_n)$ is a existential closure of A; the sentence $\forall x_1 \ldots \forall x_n A(x_1, \ldots, x_n)$ is a universal closure of A.

Claim:

- $A(x_1, \ldots, x_n)$ is satisfiable iff $\exists x_1 \ldots \exists x_n A(x_1, \ldots, x_n)$ is satisfiable.
- $\models A(x_1, \ldots, x_n)$ iff $\models \forall x_1 \ldots \forall x_n A(x_1, \ldots, x_n)$.

First-order instances of propositional formulae

Any uniform substitution of first-order formulae for the propositional variables in a propositional formula A produces a first-order formula, called a first-order instance of A.

Example: take the propositional formula

$$A = (p \wedge \neg q) \rightarrow (q \vee p).$$

The uniform substitution of $(\mathbf{5} < x)$ for p and $\exists y(x = y^2)$ for q in A results in the first-order instance

$$((\mathbf{5} < x) \wedge \neg \exists y(x = y^2)) \rightarrow (\exists y(x = y^2) \vee (\mathbf{5} < x)).$$

Note, that every first-order instance of a tautology is logically valid. Thus, for instance,

$$\models \neg\neg(x > \mathbf{0}) \rightarrow (x > \mathbf{0})$$

and

$$\models P(x) \vee \neg P(x).$$

Satisfiability and validity of sentences: examples

- $\exists x P(x)$ is satisfiable: a model is, for instance, the structure of integers $\mathcal{Z}$, where $P(x)$ is interpreted as $x + x = x$.
- However, that sentence is not valid: a counter-model is, any structure $\mathcal{A}$, where $P(x)$ is interpreted as the empty set.
- The sentence $\forall x(P(x) \vee \neg P(x))$ is valid.
- The sentence $\forall x P(x) \vee \forall x \neg P(x)$ is not valid, but is satisfiable. Find a *model* and a *countermodel*!
- The sentence $\exists x(P(x) \wedge \neg P(x))$ is not satisfiable. Why?
- The sentence $\exists x \forall y P(x, y) \rightarrow \forall y \exists x P(x, y)$ is valid.
- However, the sentence $\forall y \exists x P(x, y) \rightarrow \exists x \forall y P(x, y)$ is not valid. Find a *countermodel*!

Logical consequence in first order logic

We fix an arbitrary first-order language $\mathcal{L}$.

Given a set of $\mathcal{L}$-formulae Γ, an $\mathcal{L}$-structure $\mathcal{S}$, and a variable assignment v in $\mathcal{S}$, we write

$$\mathcal{S}, v \models \Gamma$$

to say that $\mathcal{S}, v \models A$ for every $A \in \Gamma$.

A formula A follows logically from a set of formulae Γ, denoted

$$\Gamma \models A,$$

if for every structure $\mathcal{S}$ and a variable assignment $v : \mathrm{VAR} \rightarrow \mathcal{S}$:

$$\mathcal{S}, v \models \Gamma \text{ implies } \mathcal{S}, v \models A.$$

Note that $\emptyset \models A$ iff $\models A$.

Logical consequence: examples

- If $A_1, \ldots, A_n, B$ are prop. formulae such that $A_1, \ldots, A_n \models B$, and $A'_1, \ldots, A'_n, B'$ are first-order instances of $A_1, \ldots, A_n, B$ obtained by the same substitution, then $A'_1, \ldots, A'_n \models B'$.

  For example: $\exists x A, \exists x A \to \forall y B \models \forall y B$.

- $\forall x P(x), \forall x (P(x) \to Q(x)) \models \forall x Q(x)$.

  Note that this is *not* an instance of a propositional logical consequence.

- $\exists x P(x) \land \exists x Q(x) \not\models \exists x (P(x) \land Q(x))$.

  Indeed, the structure $\mathcal{N}'$ obtained from $\mathcal{N}$ where $P(x)$ is interpreted as '*x is even*' and $Q(x)$ is interpreted as '*x is odd*' is a counter-model:

  $\mathcal{N}' \models \exists x P(x) \land \exists x Q(x)$, while $\mathcal{N}' \not\models \exists x (P(x) \land Q(x))$.

Logical consequence: some basic properties

Logical equivalence in first-order logic satisfies all basic properties of propositional logical consequence.

In particular, the following are equivalent:

1. $A_1, \dots, A_n \models B$.
2. $A_1 \wedge \dots \wedge A_n \models B$.
3. $\models A_1 \wedge \dots \wedge A_n \rightarrow B$.
4. $\models A_1 \rightarrow (A_2 \rightarrow \cdots (A_n \rightarrow B) \dots)$.

Furthermore, for any first-order formula A and a term t that is free for substitution for x in A:

1. $\forall x A \models A[t/x]$.
2. $A[t/x] \models \exists x A$.

First-order logical consequence: more basic properties

1. If $A_1, \ldots, A_n \models B$ then $\forall x A_1, \ldots, \forall x A_n \models \forall x B$.
2. If $A_1, \ldots, A_n \models B$ and x does not occur free in $A_1, \ldots, A_n$ then $A_1, \ldots, A_n \models \forall x B$.
3. If $A_1, \ldots, A_n \models B$ and $A_1, \ldots, A_n$ are sentences, then $A_1, \ldots, A_n \models \forall x B$, and hence $A_1, \ldots, A_n \models \overline{B}$, where $\overline{B}$ is any universal closure of B.
4. If $A_1, \ldots, A_n \models B[c/x]$, where c is a constant symbol not occurring in $A_1, \ldots, A_n$, then $A_1, \ldots, A_n \models \forall x B(x)$.
5. If $A_1, \ldots, A_n, A[c/x] \models B$, where c is a constant symbol not occurring in $A_1, \ldots, A_n, A$, or B, then $A_1, \ldots, A_n, \exists x A \models B$.

Testing logical consequence with deductive systems

First-order logical consequence can be established using deductive systems for first-order logic.

In particular, extensions of the Propositional Semantic Tableau and Natural Deduction, with additional rules for the quantifiers, can be constructed that are sound and complete for first-order logic. Likewise, the method of Resolution can be extended to a sound and complete deduction system for first-order logic.

Unlike the propositional case, none of these methods is guaranteed to terminate its search for a derivation, even if such a derivation exists. This happens, for instance, when a first-order logical consequence fails, but the countermodel must be infinite.

In fact, it was proved by Alonso Church in 1936 that the problem whether a given first-order sentence is valid (and consequently, if a given logical consequence holds) is not algorithmically solvable.

Therefore, no sound, complete, and always terminating deductive system for first order logic can be designed.