Truth and Consequence

Satisfaction Example

- We take a structure $\mathfrak{M}$ with domain *D = {a, b, c}.*
- Suppose our language contains the binary predicate **Likes**
- Let the extension of this predicate be the following set of pairs:

  $\text{Likes}^{\mathfrak{M}} = \{\langle a, a\rangle, \langle a, b\rangle, \langle c, a\rangle\}$
- That is, a likes itself and b, c likes a, and b likes no one.
- Consider the wff: $\exists y\,(\text{Likes}(x, y) \land \neg\text{Likes}(y, y))$
- Notice that x is a free variable.
- If the definition of satisfaction is doing its stuff, it should turn out that an assignment g satisfies this wff just in case *g* assigns *a* to the variable x.
- After all, *a* is the only individual who likes someone who does not like himself.

Satisfaction Example

- Note that *g* has to assign *some* value to x, since it has to be appropriate for the formula.
- Call this value *e*; *e* is one of *a, b*, or *c*.
- Next, we see from the clause (8) for ∃ that *g* satisfies our wff just in case there is some object d ∈ D such that *g[y/d]* satisfies the wff Likes(x, y)∧ ¬Likes(y, y)
- *g[y/d]* satisfies this wff if and only if it satisfies Likes(x, y) but does not satisfy Likes(y, y) by the clauses for conjunction and negation.

Satisfaction Example

- Looking at the atomic case, we see that this is true just in case the pair *<e, d>* is in the extension of Likes, while the pair <d, d> is not.
- But this can only happen if e = a and d = b.
- Thus the only way our original g can satisfy our wff is if it assigns a to the variable x, as we anticipated.

Truth

- **Truth**. Let *L* be some first-order language and let $\mathfrak{M}$ be a structure for *L*. A sentence **P** of *L* is *true* in $\mathfrak{M}$ if and only if the empty variable assignment g_0 satisfies **P** in $\mathfrak{M}$. Otherwise **P** is false in $\mathfrak{M}$.
- Just as we write M $\models$ **Q**[*g*] if *g* satisfies a wff **Q** in $\mathfrak{M}$, so too we write:

  $$\mathfrak{M} \models \mathbf{P}$$

  if the sentence **P** is true in $\mathfrak{M}$.

Recall Satisfaction

7. Universal quantification. Suppose P is ∀v **Q**. Then *g* satisfies **P** in $\mathfrak{M}$ if and only if for <u>every</u> *d* ∈ $D^{\mathfrak{M}}$, *g*[**v**/d] satisfies **Q**.

8. **Existential quantification.** Suppose P is ∃v **Q**. Then *g* satisfies **P** in $\mathfrak{M}$ if and only if for <u>some</u> d ∈ $D^{\mathfrak{M}}$, *g*[**v**/d] satisfies **Q**.

First-Order Consequence

- Definition. A sentence **Q** is a *first-order consequence* of a set *T* = {P_1, …} of sentences if and only if every structure that makes all the sentences in *T* true also makes **Q** true.

First-Order Validity

- Definition. A sentence **P** is a *first-order validity* if and only if every structure makes **P** true.

Truth Example

- Let's look back at the structure given just above and see if the sentence $\exists x\ \exists y\ (Likes(x, y) \wedge \neg Likes(y, y))$ comes out as it should under this definition.
- First, notice that it is a sentence, i.e., it has no free variables.
- Thus, the empty assignment is appropriate for it.
- Does the empty assignment satisfy it?

Truth Example

- According to the definition of satisfaction, it does if and only if there is an object that we can assign to the variable x so that the resulting assignment satisfies

  $\exists y\ (\text{Likes}(x, y) \land \neg \text{Likes}(y, y))$

- But we have seen that there is such an object, namely, *a*.
- So the sentence is true in $\mathfrak{M}$; in symbols, $\mathfrak{M} \models \exists x\ \exists y\ (\text{Likes}(x, y) \land \neg \text{Likes}(y, y))$.

Another Truth Example

- Consider the sentence
  $\forall x\ \exists y\ (Likes(x, y) \wedge \neg Likes(y, y))$
- Does the empty assignment satisfy this?
- It does if and only if for every object *e* in the domain, if we assign *e* to *x,* the resulting assignment *g* satisfies
  $\exists y\ (Likes(x, y) \wedge \neg Likes(y, y))$
- But, as we showed earlier, *g* satisfies this only if *g* assigns *a* to x.
- If it assigns, say, b to x, then it does not satisfy the wff.
- Hence, the empty assignment does not satisfy our sentence, i.e., the sentence is not true in M.
- So its negation is; in symbols,
  $\mathfrak{M} \models \neg\exists x\ \exists y\ (Likes(x, y) \wedge \neg Likes(y, y))$.

Proposition 1

- Let $\mathfrak{M}_1$ and $\mathfrak{M}_2$ be structures which have the same domain and assign the same interpretations to the predicates and constant symbols in a wff **P**. Let g_1 and g_2 be variable assignments that assign the same objects to the free variables in **P**. Then $\mathfrak{M}_1 \models P[g_1]$ iff $\mathfrak{M}_2 \models P[g_2]$.

Remember

- First-order structures are mathematical models of the domains about which we make claims using FOL.
- Variable assignments are functions mapping variables into the domain of some first-order structure.
- A variable assignment satisfies a wff in a structure if, intuitively, the objects assigned to the variables make the wff true in the structure.
- Using the notion of satisfaction, we can define what it means for a sentence to be true in a structure.
- Finally, once we have the notion of truth in a structure, we can model the notions of logical truth, and logical consequence.

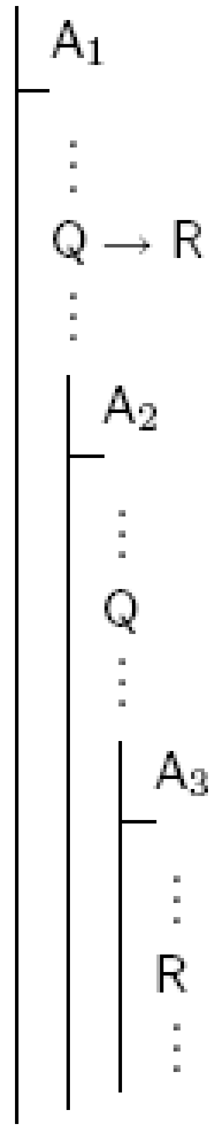
Soundness of FOL

- **Theorem:** If $T \vdash \mathbf{S}$, then **S** is a first-order consequence of set *T*.
- The proof is very similar to the proof of soundness for propositional logic.

Proof of → Elim Case

- Suppose the n^{th} step derives the sentence **R** from an application of → Elim to sentences **Q → R** and **Q** appearing earlier in the proof.
- Let $A_1, \ldots, A_k$ be a list of all the assumptions in force at step *n*.
- By our induction hypothesis we know that **Q → R** and **Q** are both established at valid steps.
- In other words, they are first-order consequences of the assumptions in force at those steps.

→ Elimination



Proof of → Elim Case

- *F* only allows us to cite sentences in the main proof or in subproofs whose assumptions are still in force.
- Hence, we know that the assumptions in force at steps **Q → R** and **Q** are also in force at **R**.
- Hence, the assumptions for these steps are among $A_1, \ldots, A_k$.
- Thus, both Q → R and Q are first-order consequences of $A_1, \ldots, A_k$.
- We now show that R is a first-order consequence of $A_1, \ldots, A_k$.

Proof of → Elim Case

- Suppose $\mathfrak{M}$ is a first-order structure in which all of $A_1, \ldots, A_k$ are true.
- Then we know that $\mathfrak{M} \models$ **Q → R** and $\mathfrak{M} \models$ **Q**, since these sentences are first-order consequences of $A_1, \ldots, A_k$.
- In that case, by the definition of truth in a structure we see that $\mathfrak{M} \models$ R as well.
- So **R** is a first-order consequence of $A_1, \ldots, A_k$.
- Hence, step *n* is a valid step.

- Notice that the only difference in this case from the corresponding case in the proof of soundness of FT is our appeal to first-order structures rather than rows of a truth table.

Proof of ∃ Elim Case

- Suppose the n^{th} step derives the sentence **R** from an application of ∃ Elim to the sentence ∃x**P**(x) and a subproof containing R at its main level, say at step *m*.
- Let **c** be the new constant introduced in the subproof.
- In other words, **P**(c) is the assumption of the subproof containing R:

Proof of ∃ Elim Case

$\vdots$

$j.\ \exists x\, P(x)$

$\vdots$

$\boxed{c}\ P(c)$

$\vdots$

$m.\ R$

$\vdots$

$\vdots$

$n.\ R$

Proof of ∃ Elim Case

- Let A_{1}, . . . , A_{k} be the assumptions in force at step *n*.
- Our inductive hypothesis assures us that steps *j* and *m* are valid steps.
- Hence ∃x**P**(x) is a first-order consequence of the assumptions in force at step j.
- Those assumptions are a subset of A_{1}, . . . , A_{k}
- **R** is a first-order consequence of the assumptions in force at step *m.*
- They are a subset of A_{1}, . . . , A_{k}, *plus the sentence* **P**(c), the assumption of the subproof in which m occurs.

Proof of ∃ Elim Case

- We need to show that **R** is a first-order consequence of A_{1}, . . . , A_{k} alone.
- To this end, assume that $\mathfrak{M}$ is a first-order structure in which each of A_{1}, . . . , A_{k} is true.
- We need to show that **R** is true in $\mathfrak{M}$ as well.
- Since ∃x**P**(x) is a consequence of A_{1}, . . . , A_{k}, we know that this sentence is also true in $\mathfrak{M}$.
- Notice that the constant **c** cannot occur in any of the sentences A_{1}, . . . , A_{k}, ∃xP(x), or R, by the restriction on the choice of temporary names imposed by the ∃ Elim rule**.**
- Since $\mathfrak{M}$ |= ∃x**P**(x), we know that there is an object, say *b*, in the domain of $\mathfrak{M}$ that satisfies **P**(x).

Proof of ∃ Elim Case

- Let $\mathfrak{M}$' be exactly like $\mathfrak{M}$, except that it assigns the object *b* to the individual constant **c**.
- Clearly, $\mathfrak{M}$' |= **P**(c), by our choice of interpretation of **c**.
- By Proposition 1, $\mathfrak{M}$' also makes each of the assumptions A_{1}, . . . , A_{k} true.
- But then $\mathfrak{M}$' |= **R**, because **R** is a first-order consequence of these sentences.
- Since **c** does not occur in **R**, **R** is also true in the original structure $\mathfrak{M}$, again by Proposition 1.