First Order Logic

"Models for first-order logic are much more interesting. First, they have objects in them!"

— AIMA

① Let's try to use propositional logic to solve a simple logic puzzle. Three Scandinavians (a Dane, a Norwegian and a Swede) live in a row of three houses:

Each house is painted a single color, and no two houses are the same color. Nobody lives in the same house. Also:

- The Dane lives directly to the right of the red house.
- The Norwegian lives in the blue house.

What color is the Swede's house?

② Not much of a puzzle (it's red), but surprisingly annoying to express in propositional logic:

"No two houses are the same color." $\rightarrow$

$$\left.\begin{array}{l} House1Blue \Rightarrow \neg House2Blue \land \neg House3Blue \\ House1Red \Rightarrow \neg House2Red \land \neg House3Red \\ \vdots \\ House3Yellow \Rightarrow \neg House1Yellow \land \neg House2Yellow \end{array}\right\} \text{mn for m colors and n houses}$$

and yet, surprisingly compact in English

FIRST ORDER LOGIC

(3) While propositional logic was designed to formalize reasoning about facts pertaining to a single object:

If <u>you</u> are a penguin, then <u>you</u> are a bird.

If <u>you</u> are in the Southern hemisphere, then <u>your</u> summer solstice is in December.

So-called <u>first-order logic</u> is designed to formalize reasoning about relationships between multiple objects:

If <u>I</u> am to <u>your</u> right, then <u>you</u> are to <u>my</u> left.

If <u>Sydney</u> is in <u>Australia</u>, and <u>Australia</u> is in the Southern hemisphere, then <u>Sydney</u> is in the Southern hemisphere.

Also, having acknowledged the existence of multiple objects in the universe, we can also generalize:

If <u>anyone</u> is to <u>somebody</u>'s right, then <u>that somebody</u> is to <u>that anyone</u>'s left.

If <u>a city</u> is in <u>a country</u>, and <u>that country</u> is in <u>a particular hemisphere</u>, then <u>the city</u> is also in <u>that hemisphere</u>.

(4) What might such a language look like? Well, first let's try to make a more canonical version of our English statements:

"No two houses are the same color." → "For any two different houses, the color of the first is not the color of the second."

"Everyone lives in some house." → "For any person, there exists some house such that the person lives in the house."

(5) Next we just make it more math-looking:

"No two houses are the same color." → $\forall x \forall y \; House(x) \wedge House(y) \wedge \neg(x=y) \Rightarrow \neg(Color(x) = Color(y))$

"Everyone lives in some house." → $\forall x \; Person(x) \Rightarrow \exists y (House(y) \wedge Lives(x,y))$

(6) Observe that, even if we go with this "more canonical" language, there are still plenty of ways to express the same thing:

"No two houses are the same color." → "For any color and two different houses, it is not true that the first house is that color and the second house is that color."

→ $\forall z \forall x \forall y \; Color(z) \wedge House(x) \wedge House(y) \Rightarrow \neg(HasColor(x,z) \wedge HasColor(y,z))$

FIRST ORDER LOGIC

⑦ What did we introduce?

FOL is short for First Order Logic

$\boxed{\text{PL}}$

$$\times 9 \left[\begin{array}{l} H1Blue \Rightarrow \neg H2Blue \land \neg H3Blue \\ H1Red \Rightarrow \neg H2Red \land \neg H3Red \\ \vdots \end{array} \right.$$

$\boxed{\text{FOL}}$

$$\forall x\, \forall y\ House(x) \land House(y) \land \neg(x = y) \Rightarrow \neg(Color(x) = Color(y))$$

quantifier variable predicate equality function

⑧ Before we go any further down the road of making a language, let's try to construct a mathematical meaning of the concepts we want to express.

In first order logic, we want to explicitly model objects in the world, like "the Dane", or "the middle house". We call this the <u>domain of discourse</u>:

D = {△, ○, ★, ᗣ, □, @, ◇}

First Order Logic

(9) A <u>signature</u> for first order logic gives names to both:

- objects in the domain of discourse
- properties of these objects

Instead of just a set of symbols, the signature is a set of <u>typed</u> symbols, e.g.

$$\Sigma = \{ \text{House1} : F_0, \ \text{ColorOf} : F_1, \ \text{RightOf} : P_2 \}$$

---

(10) A <u>model</u> (or possible world) will map the symbols of a signature based on each symbol's type.

If symbol σ has type F_k, then a model will map σ to a function $D \times \ldots \times D \to D$

e.g.
$$\text{ColorOf} \xrightarrow{m} \left\{ \begin{array}{lll} \triangle \mapsto \square, & \text{- - - - - -} & \text{"blue"} \\ \circ \mapsto @, & \text{- - - - - -} & \text{"red"} \\ \star \mapsto \diamond, & \text{- - - - - -} & \text{"green"} \\ \vdots & & \end{array} \right\}$$

If symbol σ has type P_k, then a model will map σ to a subset of $\underbrace{D \times \ldots \times D}$

e.g. $\text{RightOf} \xrightarrow{m} \{ \langle \star, \circ \rangle, \langle \star, \triangle \rangle, \langle \circ, \triangle \rangle \}$

We call symbols of type F_k <u>k-argument functions</u>, and of type P_k <u>k-arg predicates</u>

First Order Logic

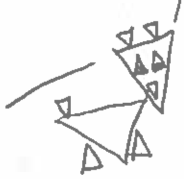
(1) Let's contrast a model of FOL with PL:

PL

$$\boxed{\text{signature } \Sigma = \{Bird, Penguin, Fly\}}$$

model $\in M_{PL}(\Sigma)$
$$\left\{\begin{array}{l} Bird \mapsto 1 \\ Penguin \mapsto 0 \\ Fly \mapsto 1 \end{array}\right.$$

the ontological commitment of FOL is greater



from onto-, meaning "being"

FOL

signature $\Sigma = \{Alice: F_0, Bob: F_0, Debbie: F_0, Bird: P_1, Penguin: P_1, Fly: P_1, ChildOF: P_2\}$

domain of discourse:

$D = \{\Delta, \text{O}, \star, \text{ᗣ}, \square\}$

model $\in M_{FOL}(\Sigma, D)$
$$\left\{\begin{array}{l}
Alice \overset{m}{\mapsto} \{\mapsto \Delta\} \\
Bob \overset{m}{\mapsto} \{\mapsto \text{O}\} \\
Debbie \overset{m}{\mapsto} \{\mapsto \star\} \\
Bird \overset{m}{\mapsto} \{\langle \Delta \rangle, \langle \text{O} \rangle, \langle \star \rangle, \langle \text{ᗣ} \rangle\} \\
Penguin \overset{m}{\mapsto} \{\langle \Delta \rangle, \langle \text{O} \rangle, \langle \text{ᗣ} \rangle\} \\
Fly \overset{m}{\mapsto} \{\langle \star \rangle\} \\
ChildOF \overset{m}{\mapsto} \{\langle \text{O}, \Delta \rangle, \langle \text{ᗣ}, \Delta \rangle\}
\end{array}\right.$$

A model $m \in M_{FOL}(\Sigma, D)$ for signature Σ is a function m s.t.

- $m(\sigma)$ is a function $\underbrace{D \times \ldots \times D}_{k} \mapsto D$ for $\sigma \in F_k(\Sigma)$
- $m(\sigma) \subseteq \underbrace{D \times \ldots \times D}_{k}$ for $\sigma \in P_k(\Sigma)$

we'll use $F_k(\Sigma)$ to refer to the symbols of Σ of type F_k and $P_k(\Sigma)$ to refer to the symbols of Σ of type P_k

First Order Logic

(12) In our example model, we can easily see that:

- every object that is a penguin is a bird
- no penguin can fly
- no child of a penguin can fly

(13) So let's build a formal language for reasoning about these models in a way that corresponds to our intuition.

First, let's assume we have access to a (countably) infinite set X of variables.

Then, for signature Σ, define the set $\mathcal{T}(\Sigma)$ of <u>terms</u> as the smallest set of (finite) strings such that:

- $X \subseteq \mathcal{T}(\Sigma)$
- if $t_1, \ldots, t_k \in \mathcal{T}(\Sigma)$ and $f \in F_k(\Sigma)$, then $f(t_1, \ldots, t_k) \in \mathcal{T}(\Sigma)$

Now we can (finally) define our language $\mathcal{L}_{FOL}(\Sigma)$ as the smallest set of (finite) strings s.t.:

- $\text{True} \in \mathcal{L}_{FOL}(\Sigma)$
- $\text{False} \in \mathcal{L}_{FOL}(\Sigma)$
- if $t_1, t_2 \in \mathcal{T}(\Sigma)$, then $t_1 = t_2 \in \mathcal{L}_{FOL}(\Sigma)$
- if $t_1, \ldots, t_k \in \mathcal{T}(\Sigma)$ and $p \in P_k(\Sigma)$, then $p(t_1, \ldots, t_k) \in \mathcal{L}_{FOL}(\Sigma)$
- if $\alpha \in \mathcal{L}_{FOL}(\Sigma)$, then $\neg\alpha \in \mathcal{L}_{FOL}(\Sigma)$
- if $\alpha, \beta \in \mathcal{L}_{FOL}(\Sigma)$, then:
  - $(\alpha \wedge \beta) \in \mathcal{L}_{FOL}(\Sigma)$
  - $(\alpha \vee \beta) \in \mathcal{L}_{FOL}(\Sigma)$
  - $(\alpha \Rightarrow \beta) \in \mathcal{L}_{FOL}(\Sigma)$
  - $(\alpha \Leftrightarrow \beta) \in \mathcal{L}_{FOL}(\Sigma)$
- if $\alpha \in \mathcal{L}_{FOL}(\Sigma)$ and $x \in X$, then:
  - $\forall x\, \alpha \in \mathcal{L}_{FOL}(\Sigma)$
  - $\exists x\, \alpha \in \mathcal{L}_{FOL}(\Sigma)$

FIRST ORDER LOGIC

(14) Which of the following are sentences in $\mathcal{L}_{FOL}(\Sigma)$, for the FOL signature $\Sigma \equiv \{ Alice : F_0, Bob : F_0, Debbie : F_0,$
$BiologicalMother : F_1, Bird : P_1, Penguin : P_1,$
$Fly : P_1, ChildOf : P_2 \}$

- $\forall x \exists y \; x = BiologicalMother(y)$ ✓
- $\forall x \exists y \; (x \Rightarrow y)$ ✗
- $\forall x \; BiologicalMother(x)$ ✗
- $ChildOf(x, Alice)$ ✓
- $\forall x \; Bird(Alice)$ ✓
- $\exists x \; Bird(Penguin(Bob))$ ✗
- Bob ✗
- x ✗
- $\forall x \; Bird(x) = Fly(x)$ ✗
- $\forall x \; (Bird(x) \Leftrightarrow Fly(x))$ ✓

First Order Logic

(15) At this point we have our language and the meanings we'd like to encode. Just as with propositional logic, we need an interpretation function I that interprets each sentence $\alpha \in \mathcal{L}_{FOL}(\Sigma)$ as a set $I(\alpha) \subseteq M_{FOL}(\Sigma, D)$ of models. Let's first define a helper function $\tau_{m,v}: \mathfrak{T}(\Sigma) \mapsto D$ to interpret terms, where <u>partial</u> function $v: X \to D$ assigns objects to variables; and m is a model $\in M_{FOL}(\Sigma, D)$.

- $\tau_{mv}(x) = v(x)$ for all $x \in dom(v)$
- $\tau_{mv}(f(t_1, \ldots, t_k)) = m(f)(\tau_{mv}(t_1), \ldots, \tau_{mv}(t_k))$ for all $f \in F_k(\Sigma)$, $t_i \in \mathfrak{T}(\Sigma)$

---

(16) Define the <u>interpretation</u> of a sentence in $\mathcal{L}_{FOL}(\Sigma)$, given <u>partial</u> variable assignment $v: X \to D$:

- $I_v(True) = M_{FOL}(\Sigma, D)$
- $I_v(False) = \emptyset$
- if $t_1, t_2 \in \mathfrak{T}(\Sigma)$, then $I_v(t_1 = t_2) = \{m \in M_{FOL}(\Sigma, D) \mid \tau_{mv}(t_1) = \tau_{mv}(t_2)\}$
- if $t_1, \ldots, t_k \in \mathfrak{T}(\Sigma)$ and $p \in P_k(\Sigma)$, then:
  $$I_v(p(t_1, \ldots, t_k)) = \{m \in M_{FOL}(\Sigma, D) \mid \langle \tau_{mv}(t_1), \ldots, \tau_{mv}(t_k) \rangle \in m(p)\}$$
- if $\alpha \in \mathcal{L}_{FOL}(\Sigma)$, then $I_v(\neg\alpha) = M_{FOL}(\Sigma, D) - I_v(\alpha)$
- if $\alpha, \beta \in \mathcal{L}_{FOL}(\Sigma)$, then:
  - $I_v(\alpha \wedge \beta) = I_v(\alpha) \cap I_v(\beta)$
  - $I_v(\alpha \vee \beta) = I_v(\alpha) \cup I_v(\beta)$
  - $I_v(\alpha \Rightarrow \beta) = I_v(\neg\alpha \vee \beta)$
  - $I_v(\alpha \Leftrightarrow \beta) = I_v(\alpha \Rightarrow \beta) \cap I_v(\beta \Rightarrow \alpha)$
- if $\alpha \in \mathcal{L}_{FOL}(\Sigma)$ and $x \in X$, then:
  - $I_v(\exists x\, \alpha) = \bigcup_{d \in D} I_{v[x \mapsto d]}(\alpha)$
  - $I_v(\forall x\, \alpha) = \bigcap_{d \in D} I_{v[x \mapsto d]}(\alpha)$

here, $v[x \mapsto d]$ means we "overwrite" the variable assignment s.t. x now maps to d

First Order Logic

(17) Let's see the interpretation of "all penguins are birds."

$$I\left[\forall x\,(\text{Penguin}(x) \Rightarrow \text{Bird}(x))\right]$$

Note: we let $I(\alpha) = I_{\{\}}(\alpha)$

trivial partial function

$$= \bigcap_{d\in D} I_{\{x\mapsto d\}}\left[P(x) \Rightarrow B(x)\right]$$

$$= \bigcap_{d\in D} I_{\{x\mapsto d\}}\left[\neg P(x) \vee B(x)\right]$$

$$= \bigcap_{d\in D} \left(I_{\{x\mapsto d\}}\left[\neg P(x)\right] \cup I_{\{x\mapsto d\}}\left[B(x)\right]\right)$$

$$= \bigcap_{d\in D} \left(\left(M_{FOL}(\Sigma, D) - I_{\{x\mapsto d\}}\left[P(x)\right]\right) \cup \{m \in M_{FOL}(\Sigma, D) \mid <d> \in m(B)\}\right)$$

$$= \bigcap_{d\in D} \left(\left(M_{FOL}(\Sigma, D) - \{m \mid <d> \in m(P)\}\right) \cup \{m \mid <d> \in m(B)\}\right)$$

$$= \bigcap_{d\in D} \left(\{m \mid <d> \notin m(P)\} \cup \{m \mid <d> \in m(B)\}\right)$$

$$= \bigcap_{d\in D} \left(M_{d\notin P} \cup M_{d\in B}\right)$$

Suppose our domain of discourse consists only of two objects, e.g. $D = \{\square, \Delta\}$. Then the interpretation is:

$$\left(M_{\square\notin P} \cup M_{\square\in B}\right) \cap \left(M_{\Delta\notin P} \cup M_{\Delta\in B}\right)$$

What models belong to this interpretation?

(1) $m_1(P) = \{<\square>\}$
$m_1(B) = \{<\square, \Delta>\}$

(2) $m_2(P) = \{<\square, \Delta>\}$
$m_2(B) = \{<\square>\}$

✓ $m_1 \in I(\forall x(P(x) \Rightarrow B(x)))$

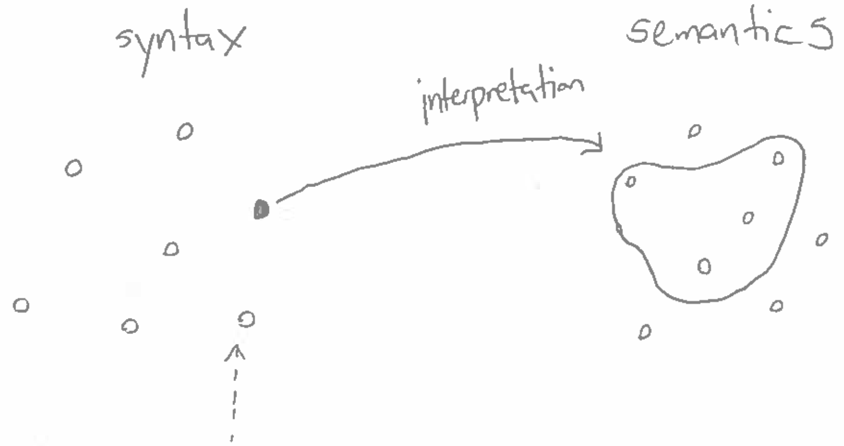
since $m_1 \in M_{\square\in B}$ and $m_1 \in M_{\Delta\notin P}$

X $m_2 \notin I(\forall x(P(x) \Rightarrow B(x)))$

since $m_2 \notin M_{\Delta\notin P}$ and $m_2 \notin M_{\Delta\in B}$

FIRST ORDER LOGIC

18 It's helpful to keep the big picture in mind. In both PL and FOL, we have a mathematical semantic model based on a set of possible worlds (called models). On the other side (syntax), we have a set of sentences that each can be interpreted as a subset of possible worlds:



each of these is a sentence like:

$(\text{Bird} \land \neg \text{Fly}) \Rightarrow \text{Penguin}$ [PL]

$\forall x\, (\text{Penguin}(x) \Rightarrow \text{Bird}(x))$ [FOL]

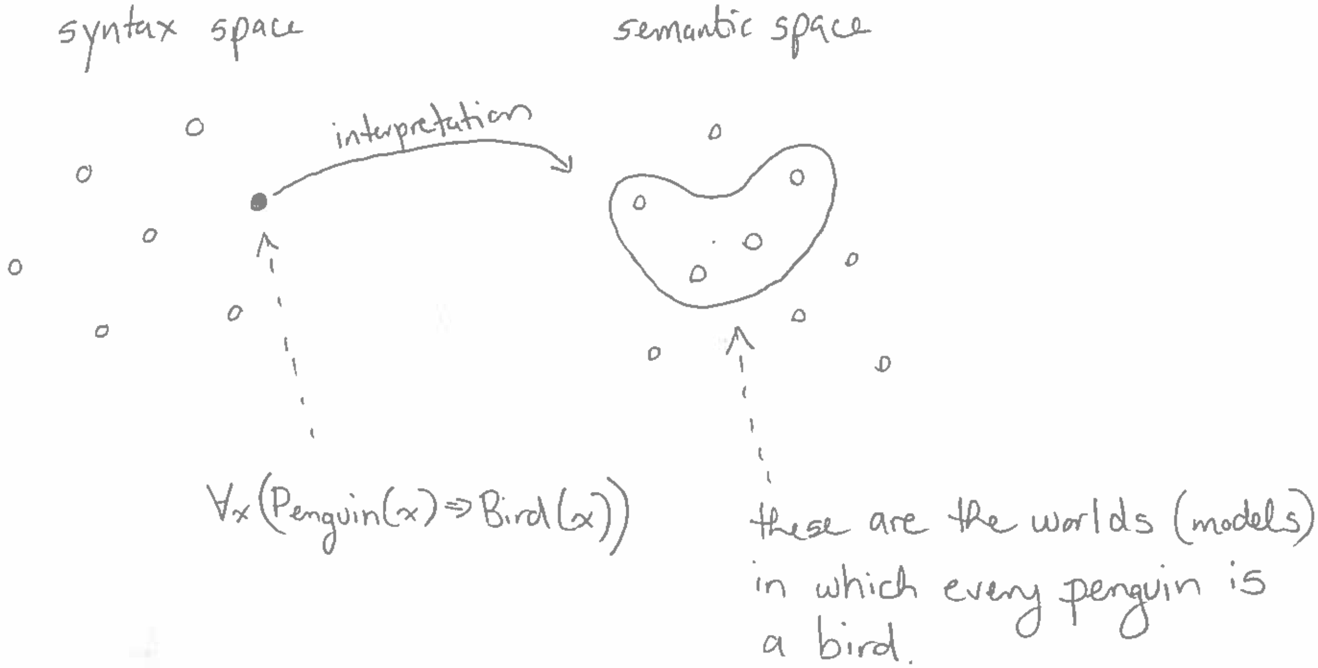
each of these is a model like:

$\{\text{Bird} \mapsto 1, \text{Penguin} \mapsto 0, \text{Fly} \mapsto 0\}$ [PL]

$\text{Penguin} \mapsto \{\langle \square \rangle, \langle \triangle \rangle\}$
$\text{ChildOf} \mapsto \{\langle \square, \triangle \rangle\}$
$\text{Bird} \mapsto \{\langle \square \rangle, \langle \triangle \rangle, \langle \diamond \rangle\}$
[FOL]

First Order Logic

(19) The interpretation corresponds to the worlds in which a sentence is true:



(20) The notion of logical entailment is thus the same for PL and FOL. We have our premise, e.g.

α: Every penguin is a bird and every bird can fly.

And a consequent:

β: Every penguin can fly.

To determine whether α entails β ($\alpha \models \beta$), we want to know whether β holds in every world that α holds. In other words, whether $I(\alpha) \subseteq I(\beta)$: