Sentences and Strings

An Introduction to Model Theory

Gabriel Ong[1]

[1]Columbia University in the City of New York

UMS Meeting, 11th July 2021

Table of Contents

1. What is Model Theory?
2. Introducing Polish Notation
3. Syntax
4. Semantics
5. Conclusions and Thanks

Table of Contents

1. What is Model Theory?
2. Introducing Polish Notation
3. Syntax
4. Semantics
5. Conclusions and Thanks

Correspondence Between Structure and Language

A "Cold" Take

In doing mathematics, we hope to make true statements and prove them.

To make these proofs, we often have to make links between mathematical structures and logic.

Correspondence Between Structure and Language

Recall that in logic we have *terms* that correspond to mathematical structures.

Take for example $x \cdot x$, a term constructed from the binary operation of multiplication while $x \cdot x \geq 0$ is a formula that we evaluate to be true for $x \in \mathbb{R}$.

First and Second Order Structures

Let us now define first and second order structures.

First Order Structures

First order structures are defined by quantification only over the elements of a structure (eg. a group).

Second Order Structures

Second order structures are defined by quantification of elements outside a structure (eg. cyclic groups are defined with the normal group structure and $\mathbb{N}$).

Desideratum

Let Σ be the set of mathematical axioms in ordinary logic. We would like to say

$$\Sigma \vdash \varphi$$

for all true mathematical statements φ.

- Gödel's incompleteness theorems show that there exist true mathematical statements that are unprovable.

But Why?

But why do we study model theory and proof theory?

- By examining the basis of our mathematical arguments, we can assert they have more truth.
- For machine readability and computational theorem proving – do check out @XenaProject on Twitter doing work in the `Lean` system.

Table of Contents

1 What is Model Theory?

2 Introducing Polish Notation

3 Syntax

4 Semantics

5 Conclusions and Thanks

The Complexity of Mathematical Language

Mathematical language, as we use it, is surprisingly complex. Consider the following equation:

$$x + yz = 0$$

To fully understand this equation, we do need some previous mathematical context. While it may be clear to us that yz in the above means $y \times z$, it is possible that someone without knowledge of PEMDAS may very well interpret this $(x + y)z$.

Writing Unambiguously

Given the complexities of even a seemingly simple equation, we would like to find a way to write mathematical statements unambiguously. One way to do this is with Polish Notation, writing operations uniformly in prefix.

Ordinary Notation	Polish Notation
$x + y$	$+xy$
$x + yz$	$+x \cdot yz$
$(x + y)z$	$\cdot + xyz$

A Lexicon for Polish Notation

We can now define a lexicon for Polish Notation.

Lexicon for Polish Notation

Let $\mathcal{W}$ be a set of symbols and α a map $\alpha : \mathcal{W} \to \omega$. The lexicon is given by tuples $(\mathcal{W}, \alpha)$.

1. α is a map to the *arity* of the symbol so $\mathcal{W}_n = \{s \in \mathcal{W} | \alpha(s) = n\}$.
2. Well formed expressions are constructed such that if $s \in \mathcal{W}_n$ and τ_i is an expression for each $i < n$ then $s\tau_0 \dots \tau_{n-1}$ is well formed.

Reading Unambiguously

We would like to interpret sentences of our lexicon such that for any given expression, there is only one interpretation.

Lemma (Unique Readability)

Let σ be an expression of the lexicon $(\mathcal{W}, \alpha)$ then

- (i) No proper initial segment of σ is an expression.
- (ii) If σ has first symbol s of arity n, then there exist *unique* expressions $\tau_0, \ldots, \tau_{n-1}$ such that σ is $s\tau_0 \ldots \tau_{n-1}$.

Reading Unambiguously

Let us prove both statements simultaneously by induction on $|\sigma|$. Existence of an expression is immediate from the definition of an expression.

Let σ' be any expression that is an initial (possibly not proper) segment of σ. Since the empty string is not an expression, we have $\sigma' = s\tau_0' \dots \tau_{n-1}'$ where τ_i' are expressions. Then τ_0 must be the same as τ_0' as otherwise we would have a proper initial segment contradicting (i).

Reading Unambiguously

It remains to show $\tau_i = \tau'_i$. We can do this by induction on i.

If $\tau_j = \tau'_j$ for all $j < i$ then τ_i, τ'_i begin at the same place in σ, so $\tau_i = \tau'_i$ because otherwise one would be a proper initial segment of the other contradicting (i). But now we know $\sigma = \sigma'$ so (i) and (ii) follow. □

Reading Unambiguously

Now let us finish up with some facts about subexpressions.

Lemma (Subexpression)

If σ is an expression of the lexicon $(\mathcal{W}, \alpha)$, then a subexpression of σ is a consecutive sequence from σ that is also an expression.

For the expression $+x \cdot yz$, $\cdot yz$ is a subexpression.

Reading Unambigiously

Definition (Scope)

If σ is an expression of the lexicon $(\mathcal{W}, \alpha)$ then the scope of an occurrence of a symbol in σ is the unique subexpression which it begins.

Let us consider the example $++xy+zu = (x+y)+(z+u)$. The scope of the first $+$, for example, is σ itself.

Table of Contents

1 What is Model Theory?

2 Introducing Polish Notation

3 Syntax

4 Semantics

5 Conclusions and Thanks

Developing a Logical Syntax

Lexicon for Predicate Logic

Let $\mathcal{L}$ be a set of non-logical symbols partitioned into disjoint sets $\mathcal{L} = \mathcal{F} \cup \mathcal{P}$ of function and predicate symbols. We can further partition $\mathcal{F}, \mathcal{P}$ by arity. Symbols in $\mathcal{F}_0$ are constant symbols and symbols in $\mathcal{P}_0$ are proposition letters.

Let's Look at a Group

Group Axioms

$\gamma_1 : \forall xyz[x \cdot (y \cdot z) = (x \cdot y) \cdot z]$
$\gamma_{2,1} : \forall x[x \cdot e = e \cdot x = x]$
$\gamma_{2,2} : \forall x[x \cdot i(x) = i(x) \cdot x = e]$

Here our logic $\mathcal{L} = \{\cdot, i, e\}$ with $\mathcal{F}_2 = \{\cdot\}, \mathcal{F}_1 = \{i\}$, and $\mathcal{F}_0 = \{e\}$.

Formulas

Formulas of $\mathcal{L}$

Let $\mathcal{L}$ be a lexicon. We define atomic formulas of $\mathcal{L}$ as sequences of symbols of the form $p\tau_1 \ldots \tau_n$ where $n \geq 0$ and $\tau_1, \ldots, \tau_n$ are terms of $\mathcal{L}$ and either $p \in \mathcal{P}_n$ or p is the symbol $=$ and $n = 2$. We can write formulas by the rules

- All atomic formulas are formulas.
- If φ is a formula then so is $\forall x \varphi$ and $\exists x \varphi$.
- Applying logical operators to atomic formulas.

Sentences and Free Variables

Occurrences of Variables

An occurrence of a variable y in a formula φ is bound if and only if it is inside the scope of a $\forall$ or $\exists$ acting on it.

Free Occurrences

An occurrence of a variable y is free if and only if it is not bound.

Sentences

The formula φ is a sentence if and only if no variable is free in φ.

Table of Contents

1 What is Model Theory?

2 Introducing Polish Notation

3 Syntax

4 Semantics

5 Conclusions and Thanks

On Squares

Existence of Square Roots

$\forall x(x > 0 \implies \exists y(x = y \cdot y))$

The above is true in $\mathbb{R}$ but false in $\mathbb{Q}$. More generally, we would like to evaluate the truth or falsity of the above in an arbitrary abstract structure $\mathfrak{A}$.

Here the structure $\mathfrak{A} = (A, \cdot_{\mathfrak{A}}, >_{\mathfrak{A}}, 0_{\mathfrak{A}})$ where A is a set containing variables x.

Evaluating Truth Values

Evaluation

Let $\mathfrak{A}$ be a structure for $\mathcal{L}$, then we define $\text{val}_{\mathfrak{A}}(\varphi)[\sigma] \in \{T, F\}$ whenever φ is an atomic formula of $\mathcal{L}$ and σ is an assignment for φ in A as follows:

- $\text{val}_{\mathfrak{A}}(p)[\sigma] = p_{\mathfrak{A}}$ when $p \in \mathcal{P}_0$.
- $\text{val}_{\mathfrak{A}}(p\tau_1 \ldots \tau_n)[\sigma] = T$ if and only if $(\text{val}_{\mathfrak{A}}(\tau_1)[\sigma], \ldots, \text{val}_{\mathfrak{A}}(\tau_n)[\sigma]) \in p_{\mathfrak{A}}$ when $p \in \mathcal{P}_n$ and $n > 0$.
- $\text{val}_{\mathfrak{A}}(= \tau_1\tau_2)[\sigma] = T$ if and only if $\text{val}_{\mathfrak{A}}(\tau_1)[\sigma] = \text{val}_{\mathfrak{A}}(\tau_2)[\sigma]$.

Evaluating Truth Values

Let us evaluate the truth value of our squares example in $\mathbb{R}$. Let us choose $2 \in \mathbb{R}$ so

$$\mathrm{val}_{\mathfrak{A}}(x > 0 \implies \exists y(x = y \cdot y))[2] = T$$

where the [2] denotes interpreting x as the real number 2. Breaking this up

$$\mathrm{val}_{\mathfrak{A}}(x > 0)[2] = T \text{ and } \mathrm{val}_{\mathfrak{A}}(\exists y(x = y \cdot y))[2] = T$$

and the original statement follows as $(T \implies T) = T$.

Models and Consequences

$\mathfrak{A}$ is a model for Σ

Let $\mathfrak{A}$ be a structure for $\mathcal{L}$ and Σ a set of sentences of $\mathcal{L}$, then $\mathfrak{A} \models \Sigma$ if and only if $\mathfrak{A} \models \varphi$ for each $\varphi \in \Sigma$.

ψ is a logical consequence of Σ

Let Σ be a set of sentences of $\mathcal{L}$ and ψ is a sentence of $\mathcal{L}$, then $\Sigma \models \psi$ holds if and only if $\mathfrak{A} \models \psi$ for all $\mathcal{L}$-structures $\mathfrak{A}$ such that $\mathfrak{A} \models \Sigma$.

Proof by Contradiction

Reductio ad Absurdum

Let Σ be a set of sentences of $\mathcal{L}$ and ψ a sentence of $\mathcal{L}$, then

- $\Sigma \models \psi$ if and only if $\Sigma \cup \{\neg\psi\}$ is semantically inconsistent.
- $\Sigma \models \neg\psi$ if and only if $\Sigma \cup \{\psi\}$ is semantically inconsistent

Compactness Theorem

Compactness Theorem

Let Σ be a set of sentences of $\mathcal{L}$ then

- If every finite subset of Σ is semantically consistent, then Σ is semantically consistent.
- If $\Sigma \vDash \psi$ then there is a finite $\Delta \subseteq \Sigma$ such that $\Delta \vDash \psi$.

The Lowenheim-Skolem Theorem

Lowenheim-Skolem Theorem

Let Σ be the set of sentences of $\mathcal{L}$ such that for finite n, Σ has a model of size greater than n. Then for all $\kappa \geq \max(\mathcal{L}, \aleph_0)$, Σ has a model of size κ.

Logical Validity

Logical Validity

If ψ is a formula of $\mathcal{L}$, then ψ is logically valid if and only if $\mathfrak{A} \models \psi[\sigma]$ for all $\mathcal{L}$ structures $\mathfrak{A}$ and all assignments σ for ψ in $\mathfrak{A}$.

Logical Equivalence

Logical Equivalence

If ϕ and ψ are formulas of $\mathcal{L}$, then ϕ, ψ are logically equivalent if and only if the formula $\phi \iff \psi$ is logically valid.

Completeness

Completeness of Sentences

Let Σ be a set of sentences in $\mathcal{L}$ then Σ is complete with respect to $\mathcal{L}$ if and only if Σ is semantically consistent and for all sentences φ of $\mathcal{L}$ we have either $\Sigma \models \varphi$ or $\Sigma \models \neg\varphi$.

Developing a Theory of a Structure

Theories of Structures

Let $\mathfrak{A}$ be a structure for $\mathcal{L}$, then the theory of $\mathfrak{A}$ is the set of all $\mathcal{L}$-sentences φ that $\mathfrak{A} \models \varphi$.

Theories are Complete

Theories of $\mathfrak{A}$ are Complete

$\text{Th}(\mathfrak{A})$ is complete with respect to $\mathcal{L}$.

$\text{Th}(\mathfrak{A})$ is semantically consistent since $\mathfrak{A} \models \text{Th}(\mathfrak{A})$ and for all sentences φ of $\mathcal{L}$, either $\varphi \in \Sigma$ or $(\neg\varphi) \in \Sigma$. $\square$

An Example from Algebra

Using our example of a group, we have $\mathcal{L} = \{\cdot, i, e\}$. This is a complete theory since all sentences we can write using our logical symbols are either satisfied by the axioms, or their negation is satisfied by the axioms.

Table of Contents

1 What is Model Theory?

2 Introducing Polish Notation

3 Syntax

4 Semantics

5 Conclusions and Thanks

References

1. Keneth Kunen *The Foundations of Mathematics*

Thanks and Acknowledgements

A special thanks to Ben Andrews for many a conversation on logic and the foundations of mathematics.