

数理逻辑

杨睿之

复旦大学哲学学院

2018 年秋季

前情提要

前情提要

- 命题逻辑的可靠性

$$\Sigma \vdash \tau \Rightarrow \Sigma \vDash \tau$$

- 命题逻辑的完全性

$$\Sigma \vDash \tau \Rightarrow \Sigma \vdash \tau$$

前情提要

- 命题逻辑的可靠性

Σ 可满足 \Rightarrow Σ 一致

- 命题逻辑的完全性

Σ 一致 \Rightarrow Σ 可满足

前情提要

完全性证明:

仍给 Σ , 假设 Σ 一致。扩张 Σ 为极大一致集 $\Delta \supset \Sigma$ 。利用 Δ 提供的信息构造赋值 v 。 ($v(A) = 1 \Leftrightarrow A \in \Delta$) 证明 v 满足 Δ 因而满足 Σ

关于命题逻辑

- 存在一个能行的方式（计算机程序），任给一个命题逻辑公式判断它是否是重言式

真值表

- 根据可靠性与完全性定理，命题逻辑的内定理集是可判定的

关于命题逻辑

- 存在一个能行的方式（计算机程序），任给一个命题逻辑公式判断它是否是重言式

真值表

- 根据可靠性与完全性定理，命题逻辑的内定理集是可判定的

关于命题逻辑

- 存在一个能行的方式（计算机程序），任给一个命题逻辑公式判断它是否是重言式

真值表

- 根据可靠性与完全性定理，命题逻辑的内定理集是可判定的

一阶谓词逻辑

不是命题逻辑的逻辑

- If there is one ring to rule them all, then everyone is ruled by a ring
- If everyone is ruled by a ring, then there is one ring rules everyone

$$\exists x \forall y Rxy \rightarrow \forall y \exists x Rxy$$

$$\forall y \exists x Rxy \rightarrow \exists x \forall y Rxy$$

不是命题逻辑的逻辑

- If there is one ring to rule them all, then everyone is ruled by a ring
- If everyone is ruled by a ring, then there is one ring rules everyone

$$\exists x \forall y Rxy \rightarrow \forall y \exists x Rxy$$

$$\forall y \exists x Rxy \rightarrow \exists x \forall y Rxy$$

不是命题逻辑的逻辑

- If there is one ring to rule them all, then everyone is ruled by a ring
- If everyone is ruled by a ring, then there is one ring rules everyone

$$\exists x \forall y Rxy \rightarrow \forall y \exists x Rxy$$

$$\forall y \exists x Rxy \rightarrow \exists x \forall y Rxy$$

一阶逻辑的语言

符号:

- 逻辑符号
 - 括号: $(,)$
 - 命题联词: \neg, \rightarrow
 - 量词: \forall
 - 变元: v_1, v_2, \dots

一阶逻辑的语言

符号:

- 非逻辑符号 (参数符号)
 - 常数符号: c_1, c_2, \dots (*)
 - 函数符号: f_1, f_2, \dots (*)
 - 谓词符号: P_1, P_2, \dots (*)
 - 等词: \approx (可有可无)
- (*) 可以没有, 也可以有无穷多
- 存在能行的函数 $g, h : \mathbb{N} \rightarrow \mathbb{N}^+$ 告诉我们每个 f_i 是 $g(i)$ -元函数符号, 每个 P_i 是 $h(i)$ -元谓词符号

一阶逻辑的语言

各种一阶逻辑语言

- 集合论语言: $\{\approx, \in\}$
- 初等数论的语言: $\{\approx, 0, <, S, +, \cdot\}$
- 序关系的语言: $\{\approx, R\}$

我们说“给定一个一阶逻辑语言”就是规定各类参数符号的集合。

一阶逻辑的语言

各种一阶逻辑语言

- 集合论语言: $\{\approx, \in\}$
- 初等数论的语言: $\{\approx, 0, <, S, +, \cdot\}$
- 序关系的语言: $\{\approx, R\}$

我们说“给定一个一阶逻辑语言”就是规定各类参数符号的集合。

一阶逻辑的语言

各种一阶逻辑语言

- 集合论语言: $\{\approx, \in\}$
- 初等数论的语言: $\{\approx, 0, <, S, +, \cdot\}$
- 序关系的语言: $\{\approx, R\}$

我们说“给定一个一阶逻辑语言”就是规定各类参数符号的集合。

一阶逻辑的语言

各种一阶逻辑语言

- 集合论语言: $\{\approx, \in\}$
- 初等数论的语言: $\{\approx, 0, <, S, +, \cdot\}$
- 序关系的语言: $\{\approx, R\}$

我们说“给定一个一阶逻辑语言”就是规定各类参数符号的集合。

一阶逻辑的语言

各种一阶逻辑语言

- 集合论语言: $\{\approx, \in\}$
- 初等数论的语言: $\{\approx, 0, <, S, +, \cdot\}$
- 序关系的语言: $\{\approx, R\}$

我们说“给定一个一阶逻辑语言”就是规定各类参数符号的集合。

一阶逻辑的语言

项 (term)

给定一个一阶逻辑语言 \mathcal{L} , 定义 \mathcal{L} 中的项为:

- 每个变元 v_i 是项
令 \mathcal{V} 是所有变元组成的集合
- 每个 \mathcal{L} 中的常数符号是项
- 如果 t_1, \dots, t_n 是项并且 f 是 \mathcal{L} 中 n 元函数符号, 那么 $ft_1 \dots t_n$ 也是项

令 \mathcal{T} 是所有项组成的集合

一阶逻辑的语言

项 (term)

给定一个一阶逻辑语言 \mathcal{L} , 定义 \mathcal{L} 中的项为:

- 每个变元 v_i 是项
 令 \mathcal{V} 是所有变元组成的集合
- 每个 \mathcal{L} 中的常数符号是项
- 如果 t_1, \dots, t_n 是项并且 f 是 \mathcal{L} 中 n 元函数符号, 那么 $ft_1 \dots t_n$ 也是项

令 \mathcal{T} 是所有项组成的集合

一阶逻辑的语言

项 (term)

给定一个一阶逻辑语言 \mathcal{L} , 定义 \mathcal{L} 中的项为:

- 每个变元 v_i 是项
令 \mathcal{V} 是所有变元组成的集合
- 每个 \mathcal{L} 中的常数符号是项
- 如果 t_1, \dots, t_n 是项并且 f 是 \mathcal{L} 中 n 元函数符号, 那么 $ft_1 \dots t_n$ 也是项

令 \mathcal{T} 是所有项组成的集合

一阶逻辑的语言

项 (term)

给定一个一阶逻辑语言 \mathcal{L} , 定义 \mathcal{L} 中的项为:

- 每个变元 v_i 是项
令 \mathcal{V} 是所有变元组成的集合
- 每个 \mathcal{L} 中的常数符号是项
- 如果 t_1, \dots, t_n 是项并且 f 是 \mathcal{L} 中 n 元函数符号, 那么 $ft_1 \dots t_n$ 也是项

令 \mathcal{T} 是所有项组成的集合

一阶逻辑的语言

例:

初等数论语言 $\{\approx, 0, <, S, +, \cdot\}$ 中的项

- v_3
- $S0$
- $+Sv_1SS0$, 常记作 $Sv_1 + SS0$

一阶逻辑的语言

例:

初等数论语言 $\{\approx, 0, <, S, +, \cdot\}$ 中的项

- v_3
- $S0$
- $+Sv_1SS0$, 常记作 $Sv_1 + SS0$

一阶逻辑的语言

合式公式 (well-formed formula)

给定一个一阶逻辑语言 \mathcal{L} , 定义 \mathcal{L} 的合式公式 (\mathcal{F}) 为:

- 如果 t_1, t_2 是项, 那么 $t_1 \approx t_2$ 是公式 (若 \mathcal{L} 中有等词)
- 如果 t_1, \dots, t_n 是项且 P 是一个 n 元谓词符号, 那么 $Pt_1 \dots t_n$ 是公式
称上述公式是 原子公式
- 如果 α, β 是合式公式, 那么 $(\neg \alpha), (\alpha \rightarrow \beta)$ 也是
- 如果 α 是合式公式, x 是变元, 那么 $\forall x \alpha$ 也是

一阶逻辑的语言

合式公式 (well-formed formula)

给定一个一阶逻辑语言 \mathcal{L} , 定义 \mathcal{L} 的 **合式公式** (\mathcal{F}) 为:

- 如果 t_1, t_2 是项, 那么 $t_1 \approx t_2$ 是公式 (若 \mathcal{L} 中有等词)
- 如果 t_1, \dots, t_n 是项且 P 是一个 n 元谓词符号, 那么 $Pt_1 \dots t_n$ 是公式
称上述公式是 **原子公式**
- 如果 α, β 是合式公式, 那么 $(\neg\alpha), (\alpha \rightarrow \beta)$ 也是
- 如果 α 是合式公式, x 是变元, 那么 $\forall x\alpha$ 也是

一阶逻辑的语言

合式公式 (well-formed formula)

给定一个一阶逻辑语言 \mathcal{L} , 定义 \mathcal{L} 的 **合式公式** (\mathcal{F}) 为:

- 如果 t_1, t_2 是项, 那么 $t_1 \approx t_2$ 是公式 (若 \mathcal{L} 中有等词)
- 如果 t_1, \dots, t_n 是项且 P 是一个 n 元谓词符号, 那么 $Pt_1 \dots t_n$ 是公式
称上述公式是 **原子公式**
- 如果 α, β 是合式公式, 那么 $(\neg\alpha), (\alpha \rightarrow \beta)$ 也是
- 如果 α 是合式公式, x 是变元, 那么 $\forall x\alpha$ 也是

一阶逻辑的语言

缩写规定:

$$\blacksquare \alpha \vee \beta =_{\text{abbr}} \neg\alpha \rightarrow \beta$$

$$\blacksquare \alpha \wedge \beta =_{\text{abbr}} \neg(\alpha \rightarrow \neg\beta)$$

$$\blacksquare \alpha \leftrightarrow \beta =_{\text{abbr}} (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$$

$$\blacksquare \exists x\alpha =_{\text{abbr}} \neg\forall x\neg\alpha$$

我们习惯称 $\forall x$ 为 全称量词，称 $\exists x$ 为 存在量词

一阶逻辑的语言

缩写规定:

- $\alpha \vee \beta =_{\text{abbr}} \neg\alpha \rightarrow \beta$
- $\alpha \wedge \beta =_{\text{abbr}} \neg(\alpha \rightarrow \neg\beta)$
- $\alpha \leftrightarrow \beta =_{\text{abbr}} (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$
- $\exists x\alpha =_{\text{abbr}} \neg\forall x\neg\alpha$

我们习惯称 $\forall x$ 为 **全称量词**，称 $\exists x$ 为 **存在量词**

自由出现与约束出现

例

$$b = \sum_{i=1}^k a_i$$

自由出现与约束出现

例

$$b = \sum_{i=1}^k a_i$$

自由出现与约束出现

例

$$b = \sum_{j=1}^k a_j$$

自由出现与约束出现

对公式 α 递归定义 x 在 α 中自由出现：

- 当 α 是原子公式： x 在 α 中出现
- 当 $\alpha = \neg\beta$ ： x 在 β 中自由出现
- 当 $\alpha = \beta \rightarrow \gamma$ ： x 在 β 或 γ 中自由出现
- 当 $\alpha = \forall y\beta$ ： x 在 β 中自由出现且 $x \neq y$

x 在 $\forall x\alpha$ 中的出现称作 约束出现

自由出现与约束出现

对公式 α 递归定义 x 在 α 中自由出现：

- 当 α 是原子公式： x 在 α 中出现
- 当 $\alpha = \neg\beta$ ： x 在 β 中自由出现
- 当 $\alpha = \beta \rightarrow \gamma$ ： x 在 β 或 γ 中自由出现
- 当 $\alpha = \forall y\beta$ ： x 在 β 中自由出现且 $x \neq y$

x 在 $\forall x\alpha$ 中的出现称作 **约束出现**

自由出现与约束出现

我们称一个合式公式 α 是 **语句** (sentence), 当且仅当没有变元在 α 中自由出现

代入

我们定义元语言中的一个表达方式 α_t^x

首先对项 u 递归定义 u_t^x

- 当 $u = y$:

$$u_t^x = \begin{cases} t & \text{若 } x = y \\ y & \text{否则} \end{cases}$$

- 当 $u = ft_1 \dots t_n$: $u_t^x = f(t_1)_t^x \dots (t_n)_t^x$

代入

例

- $(v_1)_{fv_1v_2}^{v_1} = fv_1v_2$
- $(v_2)_c^{v_1} = v_2$
- $(fv_1gv_2)_{gc}^{v_2} = fv_1ggc$

代入

例

- $(v_1)_{fv_1v_2}^{v_1} = fv_1v_2$
- $(v_2)_c^{v_1} = v_2$
- $(fv_1gv_2)_{gc}^{v_2} = fv_1ggc$

代入

例

- $(v_1)_{fv_1v_2}^{v_1} = fv_1v_2$
- $(v_2)_c^{v_1} = v_2$
- $(fv_1gv_2)_{gc}^{v_2} = fv_1ggc$

代入

其次对公式 α 递归定义 α_t^x

- 当 $\alpha = t_1 \approx t_2$: $\alpha_t^x = (t_1)_t^x \approx (t_2)_t^x$
- 当 $\alpha = Pt_1 \dots t_n$: $\alpha_t^x = P(t_1)_t^x \dots (t_n)_t^x$
- 当 $\alpha = \neg\beta$: $\alpha_t^x = (\neg\beta)_t^x = \neg\beta_t^x$
- 当 $\alpha = \beta \rightarrow \gamma$: $\alpha_t^x = (\beta \rightarrow \gamma)_t^x = \beta_t^x \rightarrow \gamma_t^x$
- 当 $\alpha = \forall y\beta$:

$$\alpha_t^x = (\forall y\beta)_t^x = \begin{cases} \alpha & \text{若 } x = y \\ \forall y\beta_t^x & \text{否则} \end{cases}$$

代入

例

- $(Pv_1fv_2 \rightarrow \exists v_2Pv_1v_2)_c^{v_1}$
- $(Pv_1fv_2 \rightarrow \exists v_2Pv_1v_2)_c^{v_2}$
- $((Pv_1fv_2 \rightarrow \exists v_2Pv_1v_2)_{v_2}^{v_1})_{v_1}^{v_2}$

代入

例

- $(Pv_1fv_2 \rightarrow \exists v_2Pv_1v_2)_c^{v_1}$
- $(Pv_1fv_2 \rightarrow \exists v_2Pv_1v_2)_c^{v_2}$
- $((Pv_1fv_2 \rightarrow \exists v_2Pv_1v_2)_{v_2}^{v_1})_{v_1}^{v_2}$

代入

例

- $(Pv_1fv_2 \rightarrow \exists v_2Pv_1v_2)_{c}^{v_1}$
- $(Pv_1fv_2 \rightarrow \exists v_2Pv_1v_2)_{c}^{v_2}$
- $((Pv_1fv_2 \rightarrow \exists v_2Pv_1v_2)_{v_2}^{v_1})_{v_1}^{v_2}$

一阶谓词逻辑的希尔伯特系统

一阶谓词逻辑的希尔伯特系统

定义 (全称概括)

称公式 φ 是公式 ψ 的 **全称概括** , 当且仅当存在自然数 n 和变元 x_1, \dots, x_n 使得

$$\varphi = \forall x_1 \dots \forall x_n \psi$$

一阶谓词逻辑的希尔伯特系统

定义 (素公式)

我们称原子公式或形如 $\forall x\beta$ 的公式为 **素公式**

令 $\langle \beta_1, \beta_2, \dots \rangle$ 是对所有素公式的**枚举**。我们定义一个一阶逻辑公式 α 的命题逻辑公式对应 α^P :

■ 当 $\alpha = \beta_i$ 是一个素公式: $\alpha^P = \beta_i^P = A_i$

■ 当 $\alpha = \neg\beta$: $\alpha^P = \neg\beta^P$

■ 当 $\alpha = \beta \rightarrow \gamma$: $\alpha^P = \beta^P \rightarrow \gamma^P$

一阶谓词逻辑的希尔伯特系统

公理：一阶逻辑希尔伯特系统的公理是由下列公式的全称概括组成的

- 1 对应的命题逻辑公式 α^P 是重言式的一阶逻辑公式 α
- 2 $\forall x\alpha \rightarrow \alpha_t^x$, 其中项 t 可以在 α 中替代 x
- 3 $\forall x(\alpha \rightarrow \beta) \rightarrow (\forall x\alpha \rightarrow \forall x\beta)$
- 4 $\alpha \rightarrow \forall x\alpha$, 其中变元 x 不在 α 中自由出现

一阶谓词逻辑的希尔伯特系统

公理：一阶逻辑希尔伯特系统的公理是由下列公式的全称概括组成的

- 1 对应的命题逻辑公式 α^P 是重言式的一阶逻辑公式 α
- 2 $\forall x\alpha \rightarrow \alpha_t^x$, 其中项 t 可以在 α 中替代 x
- 3 $\forall x(\alpha \rightarrow \beta) \rightarrow (\forall x\alpha \rightarrow \forall x\beta)$
- 4 $\alpha \rightarrow \forall x\alpha$, 其中变元 x 不在 α 中自由出现

一阶谓词逻辑的希尔伯特系统

公理：一阶逻辑希尔伯特系统的公理是由下列公式的全称概括组成的

- 1 对应的命题逻辑公式 α^P 是重言式的一阶逻辑公式 α
- 2 $\forall x\alpha \rightarrow \alpha_t^x$, 其中项 t 可以在 α 中替代 x
- 3 $\forall x(\alpha \rightarrow \beta) \rightarrow (\forall x\alpha \rightarrow \forall x\beta)$
- 4 $\alpha \rightarrow \forall x\alpha$, 其中变元 x 不在 α 中自由出现

一阶谓词逻辑的希尔伯特系统

公理：一阶逻辑希尔伯特系统的公理是由下列公式的全称概括组成的

- 1 对应的命题逻辑公式 α^P 是重言式的一阶逻辑公式 α
- 2 $\forall x\alpha \rightarrow \alpha_t^x$, 其中项 t 可以在 α 中替代 x
- 3 $\forall x(\alpha \rightarrow \beta) \rightarrow (\forall x\alpha \rightarrow \forall x\beta)$
- 4 $\alpha \rightarrow \forall x\alpha$, 其中变元 x 不在 α 中自由出现

一阶谓词逻辑的希尔伯特系统

公理：一阶逻辑希尔伯特系统的公理是由下列公式的全称概括组成的

若语言中含有等词，则还有

5 $x \approx x$

6 $x \approx y \rightarrow \alpha \rightarrow \alpha'$ ，其中 α 为原子公式，且 α' 是将 α 中若干个 x 的出现替换为 y 所得到的公式

一阶谓词逻辑的希尔伯特系统

不可替代的例子

令 $\alpha = \exists y x \neq y$ 。分别考虑

- $\forall x \alpha \rightarrow \alpha_z^x$

- $\forall x \alpha \rightarrow \alpha_y^x$

如果我们希望定义项 t 可以在 α 中替代 x 为“替换后 t 中的变元不会被 α 中已有的量词抓住”，我们该怎样严格地给出定义？

一阶谓词逻辑的希尔伯特系统

不可替代的例子

令 $\alpha = \exists y x \neq y$ 。分别考虑

- $\forall x \alpha \rightarrow \alpha_z^x$

- $\forall x \alpha \rightarrow \alpha_y^x$

如果我们希望定义 **项 t 可以在 α 中替代 x** 为“替换后 t 中的变元不会被 α 中已有的量词抓住”，我们该怎样严格地给出定义？

习题

- 分别写出一阶谓词逻辑 **合式公式** 自上而下与自下而上的定义
- 定义递归函数 $F : \mathcal{F} \times \mathcal{V} \rightarrow P(\mathbb{N})$ 使得
 $F(\alpha, x) = \{i_1, \dots, i_k\}$ 当且仅当在 α 从左到右第 i_1, \dots, i_k 个 x 的出现是自由的出现
- 3.1.2, 3.1.3
- 4.1.1