

Lógica de primer orden: Repaso y notación

IIC3263

Lógica de primer orden: Vocabulario

Una fórmula en lógica de primer orden está definida sobre algunas constantes y predicados.

Un vocabulario $\mathcal{L}$ es la unión de dos conjuntos:

constantes : $\{c_1, \dots, c_\ell, \dots\}$
relaciones : $\{R_1, \dots, R_n, \dots\}$

Lógica de primer orden: Vocabulario

Una fórmula en lógica de primer orden está definida sobre algunas constantes y predicados.

Un vocabulario $\mathcal{L}$ es la unión de dos conjuntos:

constantes : $\{c_1, \dots, c_\ell, \dots\}$
relaciones : $\{R_1, \dots, R_n, \dots\}$

Notación

La aridad de una relación R es el número de argumentos de R .

► *Cada relación tiene una aridad mayor o igual a 0*

Ejemplo

Para los números naturales $\mathcal{L}$ es la unión de

constantes : $\{0, 1\}$

relaciones : $\{suma, mult, suc, <\}$

suma y *mult* son relaciones ternarias, *suc* y $<$ son relaciones binarias.

Lógica de primer orden: Sintaxis

Las fórmulas de la lógica de primer orden se construyen usando:

- ▶ Conectivos lógicos: $\neg$, $\vee$, $\wedge$, $\rightarrow$ y $\leftrightarrow$
- ▶ Paréntesis: (y)
- ▶ Relación binaria =
- ▶ Variables
- ▶ Cuantificadores: $\forall$ y $\exists$

Veamos algunos ejemplos, antes de introducir formalmente la sintaxis de la lógica de primer orden.

Sintaxis de la lógica de primer orden: Ejemplos

Sea $\mathcal{L} = \{0, 1, \textit{suma}, \textit{mult}, \textit{suc}, <\}$

- ▶ $\textit{suc}(0, 1)$
- ▶ $\forall x \forall y (\textit{suc}(x, y) \rightarrow x < y)$

Usamos notación infija para relaciones comunes.

- ▶ $\forall x \exists y \textit{suma}(y, y, x)$
- ▶ $\forall x \forall y \forall z ((\textit{suc}(x, y) \wedge \textit{suc}(x, z)) \rightarrow y = z)$

Sintaxis de la lógica de primer orden: Términos

Desde ahora en adelante: Suponemos dada una lista infinita de variables.

$\mathcal{L}$ -términos:

- ▶ Cada constante c en $\mathcal{L}$ es un $\mathcal{L}$ -término
- ▶ Cada variable x es un $\mathcal{L}$ -término

Ejemplo

Términos en el vocabulario para los números naturales: 0, 1 y x .

Sintaxis de la lógica de primer orden: Fórmulas

El conjunto de $\mathcal{L}$ -fórmulas es el menor conjunto que satisface las siguientes condiciones:

Sintaxis de la lógica de primer orden: Fórmulas

El conjunto de $\mathcal{L}$ -fórmulas es el menor conjunto que satisface las siguientes condiciones:

- ▶ Si t_1 y t_2 son $\mathcal{L}$ -términos, entonces $t_1 = t_2$ es una $\mathcal{L}$ -fórmula.

Sintaxis de la lógica de primer orden: Fórmulas

El conjunto de $\mathcal{L}$ -fórmulas es el menor conjunto que satisface las siguientes condiciones:

- ▶ Si t_1 y t_2 son $\mathcal{L}$ -términos, entonces $t_1 = t_2$ es una $\mathcal{L}$ -fórmula.
- ▶ Si $t_1, \dots, t_n$ son $\mathcal{L}$ -términos y R es una relación n -aria en $\mathcal{L}$, entonces $R(t_1, \dots, t_n)$ es una $\mathcal{L}$ -fórmula.

Sintaxis de la lógica de primer orden: Fórmulas

El conjunto de $\mathcal{L}$ -fórmulas es el menor conjunto que satisface las siguientes condiciones:

- ▶ Si t_1 y t_2 son $\mathcal{L}$ -términos, entonces $t_1 = t_2$ es una $\mathcal{L}$ -fórmula.
- ▶ Si $t_1, \dots, t_n$ son $\mathcal{L}$ -términos y R es una relación n -aria en $\mathcal{L}$, entonces $R(t_1, \dots, t_n)$ es una $\mathcal{L}$ -fórmula.
- ▶ Si φ y ψ son $\mathcal{L}$ -fórmulas, entonces $(\neg\varphi)$, $(\varphi \vee \psi)$, $(\varphi \wedge \psi)$, $(\varphi \rightarrow \psi)$ y $(\varphi \leftrightarrow \psi)$ son $\mathcal{L}$ -fórmulas.

Sintaxis de la lógica de primer orden: Fórmulas

El conjunto de $\mathcal{L}$ -fórmulas es el menor conjunto que satisface las siguientes condiciones:

- ▶ Si t_1 y t_2 son $\mathcal{L}$ -términos, entonces $t_1 = t_2$ es una $\mathcal{L}$ -fórmula.
- ▶ Si $t_1, \dots, t_n$ son $\mathcal{L}$ -términos y R es una relación n -aria en $\mathcal{L}$, entonces $R(t_1, \dots, t_n)$ es una $\mathcal{L}$ -fórmula.
- ▶ Si φ y ψ son $\mathcal{L}$ -fórmulas, entonces $(\neg\varphi)$, $(\varphi \vee \psi)$, $(\varphi \wedge \psi)$, $(\varphi \rightarrow \psi)$ y $(\varphi \leftrightarrow \psi)$ son $\mathcal{L}$ -fórmulas.
- ▶ Si φ es una $\mathcal{L}$ -fórmula y x es una variable, entonces $(\exists x \varphi)$ y $(\forall x \varphi)$ son $\mathcal{L}$ -fórmulas.

Sintaxis de la lógica de primer orden: Fórmulas

El conjunto de $\mathcal{L}$ -fórmulas es el menor conjunto que satisface las siguientes condiciones:

- ▶ Si t_1 y t_2 son $\mathcal{L}$ -términos, entonces $t_1 = t_2$ es una $\mathcal{L}$ -fórmula.
- ▶ Si $t_1, \dots, t_n$ son $\mathcal{L}$ -términos y R es una relación n -aria en $\mathcal{L}$, entonces $R(t_1, \dots, t_n)$ es una $\mathcal{L}$ -fórmula.
- ▶ Si φ y ψ son $\mathcal{L}$ -fórmulas, entonces $(\neg\varphi)$, $(\varphi \vee \psi)$, $(\varphi \wedge \psi)$, $(\varphi \rightarrow \psi)$ y $(\varphi \leftrightarrow \psi)$ son $\mathcal{L}$ -fórmulas.
- ▶ Si φ es una $\mathcal{L}$ -fórmula y x es una variable, entonces $(\exists x \varphi)$ y $(\forall x \varphi)$ son $\mathcal{L}$ -fórmulas.

Notación

$t_1 = t_2$ y $R(t_1, \dots, t_n)$ son llamadas *fórmulas atómicas*.

Notación

Omitimos paréntesis si no se produce ambigüedad.

¿Es $\forall x \exists y \text{ suma}(y, y, x)$ cierta en $\mathcal{L} = \{0, 1, \text{suma}, \text{mult}, \text{suc}, <\}$?

- ▶ Si pensamos en los números naturales es falsa.
- ▶ Pero $\mathcal{L}$ también puede usarse como vocabulario para los números reales, y en este conjunto la fórmula es cierta.

El valor de verdad de una fórmula depende de la **interpretación que se da a las constante y relaciones.**

- ▶ Tenemos que introducir la noción de estructura.

Semántica de la lógica de primer orden: Estructuras

Una $\mathcal{L}$ -estructura interpreta todos los componentes de $\mathcal{L}$ en un dominio.

Una $\mathcal{L}$ -estructura $\mathfrak{A}$ contiene:

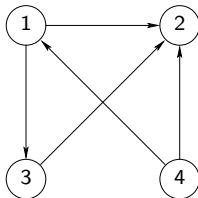
- ▶ Un dominio A no vacío.
- ▶ Para cada constante $c \in \mathcal{L}$, una interpretación $c^{\mathfrak{A}} \in A$ de c .
- ▶ Para cada relación n -aria $R \in \mathcal{L}$, una interpretación $R^{\mathfrak{A}} \subseteq A^n$ de R .

Notación

$$\mathfrak{A} = \langle A, c^{\mathfrak{A}}, \dots, R^{\mathfrak{A}}, \dots \rangle$$

Algunos ejemplos de estructuras

Para representar grafos usamos un vocabulario $\mathcal{L} = \{E\}$. Por ejemplo, el siguiente grafo:



es representado por la estructura $\mathfrak{A} = \langle A, E^{\mathfrak{A}} \rangle$, donde:

$$\begin{aligned} A &= \{1, 2, 3, 4\} \\ E^{\mathfrak{A}} &= \{(1, 2), (1, 3), (3, 2), (4, 1), (4, 2)\} \end{aligned}$$

Algunos ejemplos de estructuras

Los números naturales son representados por la estructura:

$$\mathfrak{N} = \langle \mathbb{N}, 0^{\mathfrak{N}}, 1^{\mathfrak{N}}, \textit{suma}^{\mathfrak{N}}, \textit{mult}^{\mathfrak{N}}, \textit{suc}^{\mathfrak{N}}, <^{\mathfrak{N}} \rangle,$$

Mientras que los números reales son representados por la estructura:

$$\mathfrak{R} = \langle \mathbb{R}, 0^{\mathfrak{R}}, 1^{\mathfrak{R}}, \textit{suma}^{\mathfrak{R}}, \textit{mult}^{\mathfrak{R}}, \textit{suc}^{\mathfrak{R}}, <^{\mathfrak{R}} \rangle.$$

Ahora podemos decir que $\mathfrak{N}$ no satisface $\forall x \exists y \textit{suma}(y, y, x)$ y que $\mathfrak{R}$ si satisface esta fórmula.

Semántica de la lógica de primer orden: Variables libres

Sea $V(\varphi)$ el conjunto de variables de una fórmula φ .

El conjunto de variables libres de una $\mathcal{L}$ -fórmula φ se define como:

- ▶ Si φ es una fórmula atómica, entonces $VL(\varphi) = V(\varphi)$.
- ▶ Si $\varphi = (\neg\psi)$, entonces $VL(\varphi) = VL(\psi)$.
- ▶ Si $\varphi = (\psi \star \theta)$ ($\star \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$), entonces $VL(\varphi) = VL(\psi) \cup VL(\theta)$.
- ▶ Si $\varphi = (\exists x \psi)$ o $\varphi = (\forall x \psi)$, entonces $VL(\varphi) = VL(\psi) \setminus \{x\}$.

Semántica de la lógica de primer orden: Variables libres

Ejemplos

$$VL(P(x) \wedge \exists y Q(x, y)) = \{x\}$$

$$VL(P(z) \wedge \exists z R(z)) = \{z\}$$

Semántica de la lógica de primer orden: Variables libres

Ejemplos

$$VL(P(x) \wedge \exists y Q(x, y)) = \{x\}$$

$$VL(P(z) \wedge \exists z R(z)) = \{z\}$$

Notación

- ▶ Si φ es una fórmula, entonces usamos $\varphi(x_1, \dots, x_k)$ para indicar que $VL(\varphi) = \{x_1, \dots, x_k\}$.
- ▶ Decimos que φ es una **oración** si $VL(\varphi) = \emptyset$.

Semántica de la lógica de primer orden: Definición

Si una fórmula contiene variables libres, entonces no podemos decir directamente que es verdadera o falsa en una estructura.

- ▶ ¿Es $suc(0, x)$ cierta en $\mathfrak{N}$?

El valor de verdad de una fórmula con variables libres depende de los valores dados a estas variables.

- ▶ Si x es 2, entonces $suc(0, x)$ es falsa en $\mathfrak{N}$. Pero si x es 1, entonces es cierta.

Semántica de la lógica de primer orden: Definición

Dada una $\mathcal{L}$ -estructura $\mathfrak{A}$ con dominio A , una asignación σ es una función tal que:

- ▶ $\sigma(x) \in A$ para cada variable x ,
- ▶ $\sigma(c) = c^{\mathfrak{A}}$ para cada constante c en $\mathcal{L}$.

Semántica de la lógica de primer orden: Definición

Dada una $\mathcal{L}$ -estructura $\mathfrak{A}$ con dominio A , una asignación σ es una función tal que:

- ▶ $\sigma(x) \in A$ para cada variable x ,
- ▶ $\sigma(c) = c^{\mathfrak{A}}$ para cada constante c en $\mathcal{L}$.

Definición

Decimos que $(\mathfrak{A}, \sigma)$ satisface una $\mathcal{L}$ -fórmula φ , denotado como $(\mathfrak{A}, \sigma) \models \varphi$, si y sólo si:

- ▶ $\varphi = t_1 = t_2$ y $\sigma(t_1) = \sigma(t_2)$.
- ▶ $\varphi = R(t_1, \dots, t_n)$ y $(\sigma(t_1), \dots, \sigma(t_n)) \in R^{\mathfrak{A}}$.
- ▶ $\varphi = (\neg\psi)$ y $(\mathfrak{A}, \sigma) \not\models \psi$.
- ▶ $\varphi = (\psi \vee \theta)$ y $(\mathfrak{A}, \sigma) \models \psi$ o $(\mathfrak{A}, \sigma) \models \theta$.

Semántica de la lógica de primer orden: Definición

- ▶ $\varphi = (\psi \wedge \theta)$, $(\mathfrak{A}, \sigma) \models \psi$ y $(\mathfrak{A}, \sigma) \models \theta$.
- ▶ $\varphi = (\psi \rightarrow \theta)$ y $(\mathfrak{A}, \sigma) \not\models \psi$ o $(\mathfrak{A}, \sigma) \models \theta$.
- ▶ $\varphi = (\psi \leftrightarrow \theta)$ y ambos $(\mathfrak{A}, \sigma) \models \psi$, $(\mathfrak{A}, \sigma) \models \theta$ o ambos $(\mathfrak{A}, \sigma) \not\models \psi$, $(\mathfrak{A}, \sigma) \not\models \theta$.
- ▶ $\varphi = (\exists x \psi)$ y existe $a \in A$ tal que $(\mathfrak{A}, \sigma[x/a]) \models \psi$, donde

$$\sigma[x/a](y) = \begin{cases} a & y = x \\ \sigma(y) & y \neq x \end{cases}$$

- ▶ $\varphi = (\forall x \psi)$ y para todo $a \in A$ se tiene que $(\mathfrak{A}, \sigma[x/a]) \models \psi$.

Nota: Si φ es una oración, podemos decir que $\mathfrak{A} \models \varphi$.

Ejemplo

Sea $\mathfrak{A} = \langle A, E^{\mathfrak{A}} \rangle$, donde $A = \{1, 2, 3, 4\}$ y $E^{\mathfrak{A}} = \{(1, 2), (1, 3), (3, 2), (4, 1), (4, 2)\}$.

- ¿Cuáles de las siguientes fórmulas son ciertas en $\mathfrak{A}$:
 $\exists x \forall y E(x, y)$, $\forall x \exists y E(x, y)$, $\exists x \forall y \neg E(x, y)$, $\forall x \exists y \neg E(x, y)$?

Dos nociones útiles

Decimos que una $\mathcal{L}$ -fórmula φ es **satisfacible** si existe una $\mathcal{L}$ -estructura $\mathfrak{A}$ y una asignación σ para $\mathfrak{A}$ tal que $(\mathfrak{A}, \sigma) \models \varphi$.

- ▶ Si φ es oración, entonces φ es satisfacible si existe $\mathfrak{A}$ tal que $\mathfrak{A} \models \varphi$

Decimos que una $\mathcal{L}$ -fórmula φ es **válida** si para toda $\mathcal{L}$ -estructura $\mathfrak{A}$ y toda asignación σ para $\mathfrak{A}$ se tiene que $(\mathfrak{A}, \sigma) \models \varphi$.

- ▶ Si φ es oración, entonces φ es válida si para todo $\mathfrak{A}$ se tiene que $\mathfrak{A} \models \varphi$

Ejercicio

Construya una fórmula válida.

Dos nociones útiles

Al igual que en la lógica proposicional, la lógica de primer orden tiene asociados algunos problemas de decisión:

$SAT = \{\varphi \mid \varphi \text{ es una oración satisfacible}\}$

$VAL = \{\varphi \mid \varphi \text{ es una oración válida}\}$

Dos nociones útiles

Al igual que en la lógica proposicional, la lógica de primer orden tiene asociados algunos problemas de decisión:

$SAT = \{\varphi \mid \varphi \text{ es una oración satisfacible}\}$

$VAL = \{\varphi \mid \varphi \text{ es una oración válida}\}$

Teorema (Church)

VAL y SAT son indecidibles.

La noción de consecuencia lógica

Dado: Conjunto de $\mathcal{L}$ -oraciones $\Sigma \cup \{\varphi\}$.

Decimos que φ es consecuencia lógica de Σ si para cada $\mathcal{L}$ -estructura $\mathfrak{A}$ tal que $\mathfrak{A} \models \Sigma$, se tiene que $\mathfrak{A} \models \varphi$

Notación

$$\Sigma \models \varphi$$

La noción de consecuencia lógica

Dado: Conjunto de $\mathcal{L}$ -oraciones $\Sigma \cup \{\varphi\}$.

Decimos que φ es consecuencia lógica de Σ si para cada $\mathcal{L}$ -estructura $\mathfrak{A}$ tal que $\mathfrak{A} \models \Sigma$, se tiene que $\mathfrak{A} \models \varphi$

Notación

$$\Sigma \models \varphi$$

Ejemplo

$$\{\forall x R(x, x)\} \models \forall x \exists y R(x, y)$$

El sistema de Hilbert: Lógica de Primer Orden

El sistema de deducción de Hilbert para la lógica de primer orden está formado por los siguientes elementos:

- ▶ Esquemas para generar fórmulas válidas:
 - ▶ $\varphi \rightarrow (\psi \rightarrow \varphi)$
 - ▶ $(\varphi \rightarrow (\psi \rightarrow \theta)) \rightarrow ((\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow \theta))$
 - ▶ $(\neg\varphi \rightarrow \neg\psi) \rightarrow ((\neg\varphi \rightarrow \psi) \rightarrow \varphi)$
 - ▶ $(\forall x \varphi(x)) \rightarrow \varphi(t)$, donde t es un término cualquiera
 - ▶ $\varphi(t) \rightarrow (\exists x \varphi(x))$, donde t es un término cualquiera
 - ▶ $(\exists x \varphi) \leftrightarrow (\neg\forall x \neg\varphi)$

► Axiomas para la igualdad:

- $\forall x (x = x)$
- $\forall x \forall y (x = y \rightarrow y = x)$
- $\forall x \forall y \forall z ((x = y \wedge y = z) \rightarrow x = z)$
- Para todo predicado m -ario P :

$$\forall x_1 \cdots \forall x_m \forall y_1 \cdots \forall y_m ((P(x_1, \dots, x_m) \wedge \\ x_1 = y_1 \wedge \cdots \wedge x_m = y_m) \rightarrow P(y_1, \dots, y_m))$$

El sistema de Hilbert: Lógica de Primer Orden

- ▶ Reglas de inferencia:

- ▶ Modus Ponens:

$$\frac{\varphi \rightarrow \psi \quad \varphi}{\psi}$$

- ▶ Generalización: Si y no aparece libre en φ , entonces

$$\frac{\varphi \rightarrow \psi(y)}{\varphi \rightarrow \forall x \psi(x)}$$

El sistema de Hilbert: Lógica de Primer Orden

Dado un conjunto de fórmulas $\Sigma \cup \{\varphi\}$, una deducción formal de φ desde Σ es una secuencia de fórmulas $\varphi_1, \varphi_2, \dots, \varphi_n$ tal que:

- ▶ Para cada $i \leq n$:
 - ▶ $\varphi_i \in \Sigma$ o
 - ▶ φ_i es un axioma lógico o
 - ▶ existen $j, k < i$ tales que φ_i es obtenido desde φ_j y φ_k usando modus ponens o
 - ▶ existe $j < i$ tal que φ_i es obtenido desde φ_j usando la regla de generalización.
- ▶ $\varphi_n = \varphi$

El sistema de Hilbert: Lógica de Primer Orden

Dado un conjunto de fórmulas $\Sigma \cup \{\varphi\}$, una deducción formal de φ desde Σ es una secuencia de fórmulas $\varphi_1, \varphi_2, \dots, \varphi_n$ tal que:

- ▶ Para cada $i \leq n$:
 - ▶ $\varphi_i \in \Sigma$ o
 - ▶ φ_i es un axioma lógico o
 - ▶ existen $j, k < i$ tales que φ_i es obtenido desde φ_j y φ_k usando modus ponens o
 - ▶ existe $j < i$ tal que φ_i es obtenido desde φ_j usando la regla de generalización.
- ▶ $\varphi_n = \varphi$

Notación

$\Sigma \vdash \varphi$

El sistema de Hilbert: Propiedades

Teorema (Corrección)

Si $\Sigma \vdash \varphi$, entonces $\Sigma \models \varphi$.

El sistema de Hilbert: Propiedades

Teorema (Corrección)

Si $\Sigma \vdash \varphi$, entonces $\Sigma \models \varphi$.

Teorema (Compleitud de Gödel)

Si $\Sigma \models \varphi$, entonces $\Sigma \vdash \varphi$.

El teorema de Compacidad

Notación

Decimos que un conjunto Σ de fórmulas es finitamente satisfacible si cada subconjunto finito de Σ es satisfacible.

El teorema de Compacidad

Notación

Decimos que un conjunto Σ de fórmulas es finitamente satisfacible si cada subconjunto finito de Σ es satisfacible.

Teorema (Compacidad)

Un conjunto de fórmulas Σ es satisfacible si y sólo si Σ es finitamente satisfacible.

El teorema de Compacidad

Notación

Decimos que un conjunto Σ de fórmulas es finitamente satisfacible si cada subconjunto finito de Σ es satisfacible.

Teorema (Compacidad)

Un conjunto de fórmulas Σ es satisfacible si y sólo si Σ es finitamente satisfacible.

Ejercicio

Demuestre el corolario.

Poder expresivo de una lógica

Notación

Dado un vocabulario $\mathcal{L}$, $\text{ALLSTRUCT}[\mathcal{L}]$ es el conjunto de todas las $\mathcal{L}$ -estructuras.

Una propiedad $\mathcal{P}$ de las $\mathcal{L}$ -estructuras es un subconjunto de $\text{ALLSTRUCT}[\mathcal{L}]$.

Ejemplo

El conjunto de las $\mathcal{L}$ -estructuras con dos elementos en el dominio.

Poder expresivo de una lógica

Decimos que una propiedad $\mathcal{P}$ es **expresable en lógica de primer orden** si existe una $\mathcal{L}$ -oración φ tal que para toda $\mathfrak{A} \in \text{ALLSTRUCT}[\mathcal{L}]$:

$$\mathfrak{A} \in \mathcal{P} \quad \text{si y sólo si} \quad \mathfrak{A} \models \varphi$$

Poder expresivo de una lógica

Decimos que una propiedad $\mathcal{P}$ es **expresable en lógica de primer orden** si existe una $\mathcal{L}$ -oración φ tal que para toda $\mathfrak{A} \in \text{ALLSTRUCT}[\mathcal{L}]$:

$$\mathfrak{A} \in \mathcal{P} \quad \text{si y sólo si} \quad \mathfrak{A} \models \varphi$$

Ejemplo

El conjunto de las $\mathcal{L}$ -estructuras con dos elementos en el dominio es definible en lógica de primer orden.

Ejercicios

Usando el teorema de compacidad, demuestre que las siguientes propiedades no son expresables en lógica de primer orden.

1. Sea $\mathcal{L}_1$ un vocabulario cualquiera y $\mathcal{P}_1$ el conjunto de todas las $\mathcal{L}_1$ -estructuras con dominio finito.
2. Sea $\mathcal{L}_2 = \{E(\cdot, \cdot), a, b\}$, donde a y b son constantes, y $\mathcal{P}_2$ el conjunto de todas las $\mathcal{L}_2$ -estructuras que tienen un camino de largo finito entre a y b .