TEMA 8: LÓGICA DE PRIMER ORDEN

María Inés Fernández Camacho

MATEMÁTICA DISCRETA Y LÓGICA MATEMÁTICA
(GRUPOS E y F)
UCM Curso 18/19

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Introducción al lenguaje

INSUFICIENCIA DE LA LÓGICA PROPOSICIONAL

- x es par
- $$\begin{array}{c}\text{Algunos mamíferos leen}\\ \text{Todos los que leen disfrutan}\\ \hline \therefore\ \text{Algunos mamíferos disfrutan}\end{array}$$

Con frecuencia nuestros razonamientos cotidianos aluden a elementos de un colectivo no como individuos, sino precisamente como elementos de dicho colectivo, pero la lógica proposicional no recoge propiedades de individuos ni generalidades ni relaciones entre individuos, ni trata la frecuencia con la que ocurre algo.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Introducción al lenguaje

EXTENSIÓN DE LA LÓGICA PROPOSICIONAL

- **Dominio o universo de discurso:** colectivo de individuos sobre los que razonamos.
  Ej: x es par **Dominio:** $\mathbb{N}$
- **Constantes:** nombres propios que hacen referencia a individuos concretos
  Ej: 8, Paco, María
- **Variables:** denotan valores cualesquiera del universo. Representan individuos anónimos, generales.
  Notación x, y, ...

EXTENSIÓN DE LA LÓGICA PROPOSICIONAL(2)

- **Predicados:** Enunciados sobre individuos.
  - Monádicos : Propiedades de un individuo. Forma general de un enunciado atributivo, i.e. que atribuye una propiedad a un sujeto.

    (P(x) es un predicado con respecto al dominio D, si para cada x en el dominio, P(x) es una proposición).

    Notación P(x),Q(y), ...

    Ej: P(x): x es par , M(x): x es mamífero,...

    Un ejemplo de P(x) es un valor para el que P(x) es cierto . Un contraejemplo de P(x) es un valor para el que P(x) es falso. 2 es un ejemplo de P(x): x es par, porque P(2) es cierta y 3 es un contraejemplo porque P(3) es falsa.
  - Poliádicos : relaciones entre individuos.

    Ej: H(x,y): x e y son hermanos,...

    "Paco y María son hermanos" se formalizaría H(Paco, María)
- **Funciones**: Descripción de un individuo en función de otro(s)

  Ej: x + y , abs(-5), 3 + 2, pred(suc(x)),...

EXTENSIÓN DE LA LÓGICA PROPOSICIONAL(3)

- **Cuantificadores** : Definen nuevos predicados indicando la frecuencia con que ocurren otros.

  * Cuantificador universal : Indica que algo es cierto para todos los individuos del universo de discurso. Símbolo $\forall$

DEF:

Dado un predicado $P(x)$ sobre un dominio D , $\forall\ x\ P(x)$ es una proposición que afirma que $P(x)$ es cierta para todos los posibles valores de x en el dominio D.

Se lee " Para todo x (en D) se cumple P(x) ", "Todo x (de D) cumple P(x)", "Cada x (de D) cumple P(x)", "Para cada x (en D) se cumple P(x)"

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Introducción al lenguaje

EXTENSIÓN DE LA LÓGICA PROPOSICIONAL(4)

- **Ej:** Dado P(x): x es par , $D = \mathbb{N}$
  $\forall$ x P(x) es una proposición que afirma que todos los números naturales son pares,
  y por lo tanto es falsa (P(3) es un contraejemplo)

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Introducción al lenguaje

EXTENSIÓN DE LA LÓGICA PROPOSICIONAL(5)

* Cuantificador existencial : Indica que algo es cierto para algún(os) individuos del universo de discurso. Símbolo $\exists$

DEF:

Dado un predicado $P(x)$ sobre un dominio D , $\exists$ x $P(x)$ es una proposición que afirma que $P(x)$ es cierta para al menos un valor de la variable x en el dominio D.

Se lee "Existe un x en D tal que P(x)" ," Existe un x en D tal que se cumple P(x)", "Para algún x, P"

EXTENSIÓN DE LA LÓGICA PROPOSICIONAL(6)

Ej:

- Dado $P(x)$: $x + 2 = 7$, $D = \mathbb{Z}$
  $\exists\, x\, P(x)$ es una proposición que es cierta ya que $x = 5$ es un ejemplo de $P(x)$
- Dado $Q(x)$: $2x = 7$, $D = \mathbb{Z}$
  $\exists\, x\, Q(x)$ es una proposición que es falsa ya que no hay ningún entero que cumpla $Q(x)$
- Dado $Q(x)$: $2x = 7$, $D = \mathbb{Q}$
  $\exists\, x\, Q(x)$ es una proposición que es cierta ya que $x = \frac{7}{2}$ es un ejemplo de $Q(x)$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Sintaxis

- **Símbolos primitivos:**
  - **Símbolos lógicos**:
    - Conectivas proposicionales: $\{\bot, \top, \neg, \wedge, \vee \rightarrow, \leftrightarrow\}$
    - Los cuantificadores: $\forall$ (universal) , $\exists$ (existencial).
    - El signo de igualdad: $=$

      (Permitirá formalizar enunciados atómicos que afirman la igualdad entre dos términos, p.e. $2 + 3 = 5$)
  - **Símbolos auxiliares**: (,) y ,
  - **Un conjunto infinito numerable de variables**: $V = \{x, y, z, v, \cdots\}$
  - **Signatura:** $\Sigma = F_\Sigma \cup P_\Sigma$ con $F_\Sigma \cap P_\Sigma = \emptyset$, $\Sigma \cap V = \emptyset$

    $F_\Sigma = \{f/ar(f),\ g/ar(g),\ \cdots\}$ (conjunto de símbolos de función),

    $P_\Sigma = \{P/ar(P), Q/ar(Q), \cdots\}$ (conjunto de símbolos de predicado)

    y ar(nombre de símbolo) denota la aridad del símbolo , es decir el número de argumentos de la función o el predicado.

    Constantes: símbolos de función de aridad 0.

    Proposiciones: símbolos de predicado de aridad 0.

    Σ y V tampoco incluyen los símbolos lógicos ni los auxiliares.
- **Alfabeto de símbolos primitivos:**

$$A_\Sigma = \Sigma \cup \{(,),,\} \cup \{\bot, \top, \neg, \wedge, \vee, \rightarrow, \leftrightarrow, \forall, \exists, =\} \cup V$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Sintaxis (2)

- Dado $n \in \mathbb{N}$,

$$F_\Sigma^n = \{f \in F_\Sigma / ar(f) = n\}\ ,\ P_\Sigma^n = \{Q \in P_\Sigma / ar(Q) = n\}$$

- Se pueden construir dos tipos de expresiones:

  Términos: designan individuos del universo de discurso.

  Fórmulas: representan enunciados. Su construcción usa términos.

Reglas de formación de términos:

T_Σ : conjunto de términos sobre la signatura Σ

Llamamos **términos** a aquellas palabras sobre A_Σ que se construyen aplicando un número finito de veces las siguientes reglas:

(TAt): **(V)** Cada variable $x \in V$ es un término atómico

(c) Cada constante $c \in F_\Sigma^0$ es un término atómico

(TCp): **(F)** Si $t_1, \cdots, t_n \in T_\Sigma$ y $f \in F_\Sigma^n$ con $n > 0$, entonces $f(t_1, \cdots, t_n)$ es un término compuesto.

Reglas de formación de fórmulas:

L_Σ : conjunto de fórmulas de primer orden sobre la signatura Σ
Llamamos **fórmulas** a aquellas palabras sobre A_Σ que se construyen aplicando un número finito de veces las siguientes reglas:

(FAt) (At): $\bot, \top$ y $\mathsf{p} \in P^0_\Sigma$ son fórmulas atómicas

(=): Si $s, t \in T_\Sigma$ entonces $(s = t)$ es una fórmula atómica

(P): Si $t_1, \cdots, t_n \in T_\Sigma$ y $P \in P^n_\Sigma$ con $n > 0$ entonces $P(t_1, \cdots, t_n)$ es una fórmula atómica

(FCp) (¬): si $\varphi \in L_\Sigma$, entonces $\neg\varphi$ es una fórmula compuesta (negaciones)

(□): si $\varphi_1, \varphi_2 \in L_\Sigma$ y $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$ entonces $(\varphi_1 \square \varphi_2)$ es una fórmula compuesta

(conjunción, disyunción, condicional, bicondicional, según la conectiva)

(FK) (∀): si $\varphi \in L_\Sigma$ y $x \in V$, entonces $\forall x\ \varphi$ es una fórmula compuesta llamada **cuantificación universal** de φ

(∃): si $\varphi \in L_\Sigma$ y $x \in V$, entonces $\exists x\ \varphi$ es una fórmula compuesta llamada **cuantificación existencial** de φ

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

DEF:

Dados $s, t \in T_\Sigma$, s *es un* **subtérmino de** t *si una parte de* t *formada por símbolos consecutivos es idéntica a* s (s *aparece en* t).

DEF:

Dadas $\varphi, \psi \in L_\Sigma$, ψ *es una* **subfórmula de** φ *si una parte de* φ *formada por símbolos consecutivos es idéntica a* ψ (ψ *aparece en* φ).

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Ej.: $\Sigma = \{\underbrace{c/0, f/1, g/2, h/2}_{F_\Sigma}, \underbrace{R/2, p/0, q/0}_{P_\Sigma}\}$ $V = \{x, y, z, \cdots\}$

$$x,\ c,\ f(x),\ g(x,y),\ h(f(y),g(x,z)) \ \in T_\Sigma$$

x,$f(y)$,$g(x,z)$,z,y,$h(f(y),g(x,z))$ son los subtérminos de $h(f(y),g(x,z))$

$$R(x,y),\ \ (g(x,y) = h(f(y),z)) \ \ \in L_\Sigma$$

$$\forall z\, \neg(f(z) = h(z,z)),\ \ (p \wedge q) \ \ \in L_\Sigma$$

$$(R(x,z) \vee \exists\, y\, (f(y) = c)) \in L_\Sigma$$

$R(x,z)$, $\exists\, y\, (f(y) = c)$, $(f(y) = c)$, $(R(x,z) \vee \exists\, y\, (f(y) = c))$ son las subfórmulas de $(R(x,z) \vee \exists\, y\, (f(y) = c))$

$x, f(y), c \notin L_\Sigma$ y por lo tanto no son subfórmulas de $(R(x,z) \vee \exists\, y\, (f(y) = c))$

El lenguaje de la lógica de primer orden

Def:

- *Denominamos ámbito de la cuantificación Kx (donde K es $\forall$ o $\exists$) de una fórmula $Kx\,\varphi$ a la subfórmula φ en la que todas las apariciones de la variable x aparecen cuantificadas por K. Una aparición de una variable dentro de una fórmula se llama ligada si cae dentro del ámbito de una cuantificación y libre en caso contrario.*
- *Una fórmula abierta es la que contiene variables libres, en caso contrario se llama fórmula cerrada.*

$$\forall\varphi \in L_\Sigma, \quad lig(\varphi) = \{x \in V / x \text{ aparece ligada en } \varphi\}$$

$$lib(\varphi) = \{x \in V / x \text{ aparece libre en } \varphi \text{ al menos una vez }\}$$

Def:

$t \in T_\Sigma$ se llama término cerrado si no contiene ninguna variable y término abierto en otro caso.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Ej. Dadas $\Sigma = \{\underbrace{c/0, f/1, g/2, h/2}_{F_\Sigma}, \underbrace{R/2}_{P_\Sigma}\}$ $\quad V = \{x, y, z, \cdots\}$

$$\varphi \equiv \forall y \ \underbrace{\forall z \ \underbrace{(\ \neg(f(y) = z) \ \lor \ \exists x \ \overbrace{R(f(x), y)}^{\text{ámbito de } \exists x} \ )}_{\text{ámbito de } \forall z}}_{\text{ámbito de } \forall y}$$

$$lig(\varphi) = \{x, y, z\}$$

$$lib(\varphi) = \emptyset$$

φ es una fórmula cerrada

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Obs.:

1. $\exists \varphi \in L_\Sigma, \quad lig(\varphi) \cap lib(\varphi) \neq \emptyset$

Dem: $\Sigma = \{\underbrace{c/0, f/1, g/2, h/2}_{F_\Sigma}, \underbrace{R/2}_{P_\Sigma}\} \qquad V = \{x, y, z, \cdots\}$

$$\varphi \equiv (\ \exists x \ \overbrace{R(f(\underset{\substack{\uparrow \\ \text{aparición ligada de } x}}{x}), \underset{\substack{\uparrow \\ \text{aparición libre de } y}}{y})}^{\text{ámbito de } \exists x} \ \wedge \ \forall y \ \overbrace{\neg(f(\underset{\substack{\uparrow \\ \text{aparición ligada de } y}}{y}) = \underset{\substack{\uparrow \\ \text{aparición libre de } x}}{x})}^{\text{ámbito de } \forall y} \ )$$

$$lig(\varphi) = \{x, y\} = lib(\varphi)$$

φ es una fórmula abierta

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

2. Un predicado puede estar parcialmente cuantificado.

   Ej: $\forall x\ R(x,y)$ es una fórmula abierta sobre la signatura $\Sigma = \{\underbrace{c/0, f/1, g/2, h/2}_{F_\Sigma}, \underbrace{R/2}_{P_\Sigma}\}$.
   Un predicado parcialmente cuantificado no es una proposición.

3. Una fórmula abierta puede cerrarse reemplazando variables por términos cerrados o cuantificando variables.

   Ej: $\forall x\ R(x,c)$ y $R(f(c),c)$ son fórmulas cerradas sobre la signatura $\Sigma = \{\underbrace{c/0, f/1, g/2, h/2}_{F_\Sigma}, \underbrace{R/2}_{P_\Sigma}\}$.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

- Las variables ligadas de una fórmula expresan un recorrido (existencial o universal) del universo de discurso, sin hacer referencia a un individuo particular del mismo, por eso se dice que las variables ligadas son mudas. En cambio, una variable libre sí hace referencia a un individuo particular (singular), aunque indeterminado (a la espera de ser determinado por una interpretación)

DEF:

Dada $\varphi \in L_\Sigma$, *se llama* **variante** *de* φ *a cualquier otra fórmula* φ' *resultante de reemplazar una o varias subfórmulas* ψ *de la forma* $Kx\ \eta$ *(donde* $x \in V$ *y* $K \in \{\forall, \exists\}$*) por* $Ky\ \eta[x/y]$ *siendo* y *una variable tal que* $y \neq x$ *e* $y \notin lib(\varphi) \cup lig(\varphi)$ *(*y *es nueva en* φ*) y dónde* $Ky\ \eta[x/y]$ *se obtiene sustituyendo todas las apariciones de* x *en* $Kx\ \eta$ *por* y*.*

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Ej: $\Sigma = \{\underbrace{c/0, f/1, g/2, h/2}_{F_\Sigma}, \underbrace{R/2}_{P_\Sigma}\}$ $\quad V = \{x, y, z, \cdots\}$

$$\varphi \equiv (\ \exists x\ \ R(f(x), y) \quad \wedge \quad \forall y\ \ \neg(f(y) = x)\ \ )$$

Variante:

$$\varphi' \equiv (\ \exists u\ \ R(f(u), y) \quad \wedge \quad \forall v\ \ \neg(f(v) = x)\ \ )$$

Las cuantificaciones que φ expresa con ayuda de x e y, se expresan en φ' con ayuda de u y v.

$$lig(\varphi) = \{x, y\} = lib(\varphi)$$

$$lig(\varphi') = \{u, v\}, \quad lib(\varphi') = \{x, y\}, \quad lig(\varphi') \cap lib(\varphi') = \emptyset$$

Principio de inducción estructural sobre términos

PRINCIPIO DE INDUCCIÓN ESTRUCTURAL SOBRE TÉRMINOS

Podemos concluir que todo término $t \in T_\Sigma$ tiene la propiedad P siempre que demostremos:

- **Casos base:**

  (At): Todo término atómico verifica la propiedad P

- **Paso inductivo:**

  (Cp): si $t_1, \cdots t_n \in T_\Sigma$ tienen la propiedad P (hipótesis de inducción) y $f \in F^n_\Sigma$ con $n > 0$, entonces el término compuesto $f(t_1, \cdots, t_n)$ también tiene la propiedad P.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Principio de unicidad estructural para términos

La construcción de cualquier término determina unívocamente su estructura sintáctica.

PRINCIPIO DE UNICIDAD ESTRUCTURAL PARA TÉRMINOS

Todo término $t \in T_\Sigma$ cae dentro de uno y sólo uno de los casos siguientes:

(At): t es atómico.

(Cp): t es de la forma $f(t_1, \cdots, t_n)$ para cierta $f \in F_\Sigma^n$ con $n > 0$ y ciertos $t_1, \cdots, t_n \in T_\Sigma$ unívocamente determinados.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Principio de recursión estructural sobre términos

PRINCIPIO DE RECURSIÓN ESTRUCTURAL SOBRE TÉRMINOS

Dado cualquier conjunto C, para definir una función $h : T_\Sigma \to C$ es válido utilizar el siguiente esquema recursivo:

- **Casos base:**

  (At): Para t atómico
  $h(t) = \cdots$ valor explícito $\cdots$

- **Caso recursivo:**

  (Cp): Si t es de la forma $f(t_1, \cdots, t_n)$ para cierta $f \in F^n_\Sigma$ con $n > 0$
  y ciertos $t_1, \cdots, t_n \in T_\Sigma$
  $h(t) =$ valor dependiendo de f y de $h(t_1), \cdots h(t_n)$.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Principio de inducción estructural sobre fórmulas de primer orden.

PRINCIPIO DE INDUCCIÓN ESTRUCTURAL SOBRE FÓRMULAS DE PRIMER ORDEN

Podemos concluir que toda fórmula $\varphi \in L_\Sigma$ tiene la propiedad P siempre que demostremos:

- **Casos base:**

  (At): Toda fórmula atómica tiene la propiedad P

- **Pasos inductivos:**

  ($\neg$): si φ tiene la propiedad P (hipótesis de inducción) , entonces $\neg\varphi$ también tiene la propiedad P.

  ($\square$): si φ_1 y φ_2 tienen la propiedad P (hipótesis de inducción) y $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$, entonces $(\varphi_1 \square \varphi_2)$ también tiene la propiedad P.

  (K): si φ tiene la propiedad P (hipótesis de inducción) y $x \in V$, entonces $Kx\ \varphi$ también tiene la propiedad P, donde $K \in \{\forall, \exists\}$

La construcción de cualquier fórmula determina unívocamente su estructura sintáctica.

PRINCIPIO DE UNICIDAD ESTRUCTURAL PARA FÓRMULAS DE PRIMER ORDEN

Toda fórmula $\varphi \in L_\Sigma$ cae dentro de uno y sólo uno de los casos siguientes:

(FAt) (At): φ es $\perp$, $\top$ o $p \in P^0_\Sigma$ (símbolo de proposición).

(=): φ es de la forma $(s = t)$ para ciertos $s, t \in T_\Sigma$ unívocamente determinados.

(P): φ es de la forma $P(t_1, \cdots, t_n)$ para cierto $P \in P^n_\Sigma$ con $n > 0$ y ciertos $t_1, \cdots, t_n \in T_\Sigma$ unívocamente determinados.

(FCp) (¬): φ es de la forma $\neg\varphi_1$ para cierta fórmula φ_1 unívocamente determinada.

(□): φ es de la forma $(\varphi_1 \,\square\, \varphi_2)$ para cierta conectiva $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$ y ciertas fórmulas φ_1 y φ_2 unívocamente determinadas.

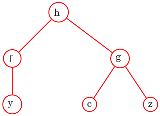
(K): φ es de la forma $Kx\,\varphi_1$ para cierto cuantificador $K \in \{\forall, \exists\}$, cierta variable $x \in V$ y cierta φ_1 unívocamente determinados.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

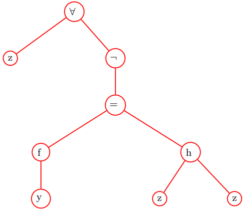
Árboles estructurales

A cada $\varphi \in L_\Sigma$ (respectivamente $t \in T_\Sigma$) se le puede asociar un árbol unívocamente determinado por φ (respectivamente $t \in T_\Sigma$) que representa su estructura de construcción y que se denomina **árbol estructural** de φ (respectivamente $t \in T_\Sigma$).

Ej: 1. Árbol estructural de $h(f(y), g(c, z)) \in T_\Sigma$:



2. Árbol estructural de $\forall z\, \neg(f(y) = h(z, z)) \in L_\Sigma$:



EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Principio de recursión estructural para fórmulas de primer orden

PRINCIPIO DE RECURSIÓN ESTRUCTURAL PARA FÓRMULAS DE PRIMER ORDEN

Dado cualquier conjunto C, para definir una función $h : L_\Sigma \to C$ es válido utilizar el siguiente esquema recursivo:

- **Casos base:**

  (At): Para φ atómica:
  $h(\varphi) = \cdots$ valor explícito $\cdots$

- **Casos recursivos:**

  ($\neg$): $h(\neg\varphi) =$ valor dependiendo de $h(\varphi)$.

  ($\Box$): $h((\varphi_1 \Box \varphi_2)) =$ valor dependiendo de $h(\varphi_1), h(\varphi_2)$ y $\Box$.
  (K): $h(Kx\ \varphi) =$ valor dependiendo de K y de $h(\varphi)$.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

DEF:

El **vocabulario** *de un término $t \in T_\Sigma$ es el conjunto finito,$voc(t)$,formado por todos los símbolos de función que aparecen en t.*

DEF:

El **vocabulario** *de una fórmula $\varphi \in L_\Sigma$es el conjunto finito,$voc(\varphi)$,formado por todos los símbolos de función y predicado que aparecen en φ.*

Ej. Dados $\Sigma = \{\underbrace{c/0, f/1, g/2, h/2,}_{F_\Sigma} \underbrace{R/2}_{P_\Sigma}\}$ $\quad V = \{x, y, z, \cdots\}$

$$voc(h(f(y), g(c, z))) = \{f, g, h, c\}$$

$$voc(\forall x\, \forall y\, (R(x, f(y)) \wedge \forall z\, \neg(h(z, z) = f(y)))) = \{R, f, h\}$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Escritura abreviada de fórmulas de primer orden.

ESCRITURA ABREVIADA DE FÓRMULAS DE PRIMER ORDEN

• Una fórmula está **correctamente** escrita en **forma abreviada** si cumple los siguientes convenios:

- Omite los paréntesis externos.
- Las conectivas tienen el siguiente orden de prioridad:

$$\neg > \wedge > \vee > \rightarrow > \leftrightarrow$$

- Las conectivas $\wedge, \vee, \rightarrow$ asocian por la derecha.
- Los cuantificadores tienen prioridad sobre las conectivas.

Ej.:

$\forall x\, R(y, f(x)) \leftrightarrow (y = c)$ abrevia $(\forall x\, R(y, f(x)) \leftrightarrow (y = c))$ y **no abrevia** $\forall x\, (R(y, f(x)) \leftrightarrow (y = c))$

El lenguaje de la lógica de primer orden

Semántica

- Ha de atribuirse significado a

  Los términos: de manera que representen individuos del universo de discurso.
  Las fórmulas: de manera que representen afirmaciones (enunciados) verdaderas o falsas.

- ¿Cómo?: Sustituyendo valoraciones por interpretaciones

$$\text{Interpretación} = \begin{cases} \text{Estructura: significado para los símbolos de función y predicado de } \Sigma \\ \quad + \\ \text{Estado: significado para las variables libres} \end{cases}$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Semántica (2)

DEF:

Una Σ−estructura es una terna

$$\mathcal{A} = (A, \{f^{\mathcal{A}} / f \in F_\Sigma\}, \{P^{\mathcal{A}} / P \in P_\Sigma\})$$

donde

- $A \neq \emptyset$ *(dominio o universo de discurso)*
- *Para cada* $f \in F_\Sigma^n$, $n > 0$, $f^{\mathcal{A}} : A^n \to A$: *qué elemento de* A *designa la función* f *según sus argumentos.*
- *Para cada* $c \in F_\Sigma^0$, $c^{\mathcal{A}} \in A$: *qué elemento de* A *designa la constante* c*.*
- *Para cada* $P \in P_\Sigma^n$, $n > 0$, $P^{\mathcal{A}} \subseteq A^n$: *qué individuos cumplen el predicado* P*.* ($P^{\mathcal{A}} : A^n \to \{0,1\}$, $P^{\mathcal{A}}(a_1, \cdots, a_n) = 1 \Longleftrightarrow (a_1, \cdots, a_n)$ *cumple* P*. En este caso se escribe simplemente* $P^{\mathcal{A}}(a_1, \cdots, a_n)$).
- *Para cada* $p \in P_\Sigma^0$, $p^{\mathcal{A}} \in \{0,1\}$: *determina si la proposición* p *es verdadera o falsa.*

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Semántica (3)

DEF:

Un **estado** *de las variables sobre una $\Sigma-$estructura $\mathcal{A}$ es cualquier $\sigma : V \to A$ que hace corresponder a cada variable $x \in V$ un individuo $\sigma(x) \in A$, al que llamamos valor de la variable x en el estado σ*

DEF:

Dados $\sigma : V \to A$, $x \in V$ y un individuo $a \in A$, $\sigma[x/a] : V \to A$ es un nuevo estado tal que

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

DEF:

Una **Σ-interpretación** *es cualquier par $(\mathcal{A}, \sigma)$ donde $\mathcal{A}$ es una Σ-estructura y σ un estado de las variables.*

Int_Σ denota el conjunto de todas las Σ-interpretaciones.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Semántica.

Interpretación de términos (4)

INTERPRETACIÓN DE TÉRMINOS

DEF:

Una Σ-interpretación $(\mathcal{A}, \sigma)$ asigna a cada $t \in T_\Sigma$ un valor $[\![t]\!]_\sigma^\mathcal{A} \in A$ definido recursivamente sobre la estructura de t, así:

- **Casos base:**

  (TAt) (V): $[\![x]\!]_\sigma^\mathcal{A} = \sigma(x)$ $\quad x \in V$ (el valor lo da el estado σ)

  (c): $[\![c]\!]_\sigma^\mathcal{A} = c^\mathcal{A}$ $\quad c \in F_\Sigma^0$ (el valor lo da la estructura $\mathcal{A}$)

- **Caso recursivo:**

  (TCp) (F): $[\![f(t_1, \cdots, t_n)]\!]_\sigma^\mathcal{A} = f^\mathcal{A}([\![t_1]\!]_\sigma^\mathcal{A}, \cdots, [\![t_n]\!]_\sigma^\mathcal{A})$ $\quad f \in F_\Sigma^n, \ n > 0$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Semántica. (5)

Interpretación de fórmulas

INTERPRETACIÓN DE FÓRMULAS

DEF:

Una Σ-interpretación $(\mathcal{A}, \sigma)$ asigna a cada $\varphi \in L_\Sigma$ un valor veritativo $[\![\varphi]\!]^{\mathcal{A}}_{\sigma} \in \{0, 1\}$ definido recursivamente sobre la estructura de φ, así:

- **Casos base:**

(⊤): $[\![\top]\!]^{\mathcal{A}}_{\sigma} = 1$ **(⊥)**: $[\![\bot]\!]^{\mathcal{A}}_{\sigma} = 0$

(p): $[\![p]\!]^{\mathcal{A}}_{\sigma} = p^{\mathcal{A}} \qquad p \in P^0_\Sigma$

(=): $$[\![(s = t)]\!]^{\mathcal{A}}_{\sigma} = \begin{cases} 1 & [\![s]\!]^{\mathcal{A}}_{\sigma} = [\![t]\!]^{\mathcal{A}}_{\sigma} \\ 0 & [\![s]\!]^{\mathcal{A}}_{\sigma} \neq [\![t]\!]^{\mathcal{A}}_{\sigma} \end{cases} \qquad s, t \in T_\Sigma$$

(P): $$[\![P(t_1, \cdots, t_n)]\!]^{\mathcal{A}}_{\sigma} = \begin{cases} 1 & P^{\mathcal{A}}([\![t_1]\!]^{\mathcal{A}}_{\sigma}, \cdots, [\![t_n]\!]^{\mathcal{A}}_{\sigma}) \\ 0 & \textit{otro caso} \end{cases} \qquad P \in P^n_\Sigma, \quad n > 0$$

El lenguaje de la lógica de primer orden

Semántica. (6)

Interpretación de fórmulas (2)

- **Casos recursivos:**

($\neg$): $[\![\neg\varphi]\!]^{\mathcal{A}}_{\sigma} = v_{\neg}([\![\varphi]\!]^{\mathcal{A}}_{\sigma})$

$v_{\neg}$ definido como en lógica proposicional

($\square$): $[\![(\varphi_1 \square \varphi_2)]\!]^{\mathcal{A}}_{\sigma} = v_{\square}([\![\varphi_1]\!]^{\mathcal{A}}_{\sigma}, [\![\varphi_2]\!]^{\mathcal{A}}_{\sigma})$ para $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$

$v_{\square}$ definido como en lógica proposicional

($\forall$): $$[\![\forall x\ \varphi]\!]^{\mathcal{A}}_{\sigma} = \begin{cases} 1 & [\![\varphi]\!]^{\mathcal{A}}_{\sigma[x/a]} = 1 \text{ para todo } a \in A \\ 0 & \text{otro caso} \end{cases}$$

($\exists$): $$[\![\exists x\ \varphi]\!]^{\mathcal{A}}_{\sigma} = \begin{cases} 1 & [\![\varphi]\!]^{\mathcal{A}}_{\sigma[x/a]} = 1 \text{ para algún } a \in A \\ 0 & \text{otro caso} \end{cases}$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Sastifactibilidad

DEF:

- *Dados* $\varphi \in L_\Sigma$ *y una* *Σ-interpretación* $(\mathcal{A}, \sigma)$
  - *Si* $[\![\varphi]\!]_\sigma^\mathcal{A} = 1$ *decimos que* $(\mathcal{A}, \sigma)$ **satisface** φ, *que* $(\mathcal{A}, \sigma)$ *es modelo de* φ *y escribimos* $\mathcal{A} \models \varphi\sigma$.
  - *Si* $[\![\varphi]\!]_\sigma^\mathcal{A} = 0$ *decimos que* $(\mathcal{A}, \sigma)$ **no satisface** φ, *que* $(\mathcal{A}, \sigma)$ *no es modelo de* φ *y escribimos* $\mathcal{A} \not\models \varphi\sigma$.
  - **Mod(**φ**)** $= \{(\mathcal{A}, \sigma) / \mathcal{A} \models \varphi\sigma\}$
  - *Si* **Mod(**φ**)** $\neq \emptyset$, *entonces decimos que* φ *es* **satisfactible** *y escribimos* *Sat*(φ) **(predicado de satisfactibilidad)**
  - *Si* **Mod(**φ**)** $= \emptyset$, *entonces decimos que* φ *es* **insatisfactible** *y escribimos* *Insat*(φ)

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Sastifactibilidad (2)

DEF:

- *Dados* $\Phi \subseteq L_\Sigma$ *y una* Σ*-interpretación* $(\mathcal{A}, \sigma)$
  - $(\mathcal{A}, \sigma)$ **satisface** Φ, $(\mathcal{A}, \sigma)$ *es modelo de* Φ *y escribimos* $\mathcal{A} \models \Phi\sigma$ *si* $\mathcal{A} \models \varphi\sigma$ *para* **cualquier** *fórmula* $\varphi \in \Phi$.
  - $(\mathcal{A}, \sigma)$ **no satisface** Φ, $(\mathcal{A}, \sigma)$*no es modelo de* Φ *y escribimos* $\mathcal{A} \not\models \Phi\sigma$ *si* $\mathcal{A} \not\models \varphi\sigma$ *para* **alguna** *fórmula* $\varphi \in \Phi$.
  - **Mod(Φ)** $= \{(\mathcal{A}, \sigma)/\mathcal{A} \models \Phi\sigma\}$
  - *Si* **Mod(Φ)** $\neq \emptyset$, *entonces decimos que* Φ *es* **satisfactible** *y escribimos* $Sat(\Phi)$ **(predicado de satisfactibilidad)**
  - *Si* **Mod(Φ)** $= \emptyset$, *entonces decimos que* Φ *es* **insatisfactible** *y escribimos* $Insat(\Phi)$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Semántica.

Ejemplo

Ej.: La signatura $\Sigma = \{\underbrace{c/0, f/1, g/2, h/2}_{F_\Sigma}, \underbrace{R/2}_{P_\Sigma}\}$ permite representar la aritmética básica de los números naturales por medio de la $\Sigma-$estructura:

$$\mathcal{N} = (\mathbb{N}, \{c^{\mathcal{N}}/0, f^{\mathcal{N}}/1, g^{\mathcal{N}}/2, h^{\mathcal{N}}/2\}, \{R^{\mathcal{N}}/2\})$$

que interpreta los símbolos de F_Σ y P_Σ así:

- $c^{\mathcal{N}}$ es el 0
- $f^{\mathcal{N}}$ es la función $suc : \mathbb{N} \to \mathbb{N}$
- $g^{\mathcal{N}}$ es la función suma $+ : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$
- $h^{\mathcal{N}}$ es la función producto $\cdot : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$
- $R^{\mathcal{N}}$ es la relación de orden $\leq$ usual sobre $\mathbb{N}$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Semántica.

Ejemplo (2)

Dados $V = \{x, y, z, \cdots\}$

$\sigma, \sigma' : V \to \mathbb{N}$, tales que $\sigma(x) = 4$, $\sigma(y) = 3$, $\sigma'(x) = 1$, $\sigma'(y) = 3$

$g(f(x), h(x, y)) \in T_\Sigma$

$\underbrace{\exists y\ (x = g(h(f(f(c)), y), f(c)))}_{\varphi} \in L_\Sigma$ y $(\mathcal{N}, \sigma), (\mathcal{N}, \sigma') \in Int_\Sigma$, se tiene:

$$\begin{aligned} [\![g(f(x), h(x, y))]\!]^{\mathcal{N}}_{\sigma} &= g^{\mathcal{N}}(f^{\mathcal{N}}(\sigma(x)), h^{\mathcal{N}}(\sigma(x), \sigma(y))) \\ &= g^{\mathcal{N}}(f^{\mathcal{N}}(4), h^{\mathcal{N}}(4, 3)) = 5 + 12 = 17 \end{aligned}$$

$$\begin{aligned} [\![g(f(x), h(x, y))]\!]^{\mathcal{N}}_{\sigma'} &= g^{\mathcal{N}}(f^{\mathcal{N}}(\sigma'(x)), h^{\mathcal{N}}(\sigma'(x), \sigma'(y))) \\ &= g^{\mathcal{N}}(f^{\mathcal{N}}(1), h^{\mathcal{N}}(1, 3)) = 2 + 3 = 5 \end{aligned}$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Semántica.

Ejemplo **(3)**

$$\mathcal{N} \not\models \varphi\sigma$$

$$[\![\varphi]\!]^{\mathcal{N}}_{\sigma} = [\![\exists y\ (x = g(h(f(f(c)),y),f(c)))]\!]^{\mathcal{N}}_{\sigma}$$

$[\![\varphi]\!]^{\mathcal{N}}_{\sigma} = 1 \Longleftrightarrow [\![(x = g(h(f(f(c)),y),f(c)))]\!]^{\mathcal{N}}_{\sigma[y/n]} = 1$ para algún $n \in \mathbb{N}$

$\Longleftrightarrow \sigma[y/n](x) = g^{\mathcal{N}}(h^{\mathcal{N}}(f^{\mathcal{N}}(f^{\mathcal{N}}(c^{\mathcal{N}})),\sigma[y/n](y)),f^{\mathcal{N}}(c^{\mathcal{N}}))$para algún $n \in \mathbb{N}$

$\Longleftrightarrow \sigma(x) = g^{\mathcal{N}}(h^{\mathcal{N}}(suc(suc(0)),n),suc(0))$ para algún$n \in \mathbb{N}$

$\Longleftrightarrow 4 = g^{\mathcal{N}}(h^{\mathcal{N}}(2,n),1)$ para algún $n \in \mathbb{N}$

$\Longleftrightarrow 4 = 2n + 1$ para algún $n \in \mathbb{N}$,

lo cual es siempre falso pues 4 es par y 2n +1 impar para cualquier $n \in \mathbb{N}$

El lenguaje de la lógica de primer orden

Semántica.

Ejemplo (4)

$$\mathcal{N} \models \varphi\sigma'$$

$[\![\varphi]\!]^{\mathcal{N}}_{\sigma'}{=}1 \Longleftrightarrow [\![(x = g(h(f(f(c)), y), f(c)))]\!]^{\mathcal{N}}_{\sigma'[y/n]} = 1$ para algún $n \in \mathbb{N}$

$\Longleftrightarrow \sigma'[y/n](x){=}g^{\mathcal{N}}(h^{\mathcal{N}}(f^{\mathcal{N}}(f^{\mathcal{N}}(c^{\mathcal{N}})), \sigma'[y/n](y)), f^{\mathcal{N}}(c^{\mathcal{N}}))$ para algún $n \in \mathbb{N}$

$\Longleftrightarrow 1 = 2n + 1$ para algún $n \in \mathbb{N}$,

lo cual es cierto sin más que tomar $n = 0$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Lema de coincidencia

Cuando interpretamos un término o una fórmula sólo importan los símbolos y las variables libres que aparezcan en ellos.

LEMA DE COINCIDENCIA

Dados $(\mathcal{A}, \sigma), (\mathcal{A}', \sigma') \in Int_\Sigma,\ t \in T_\Sigma,\ \varphi \in L_\Sigma$

- Si $(\mathcal{A}, \sigma)$ y $(\mathcal{A}', \sigma')$ interpretan del mismo modo los símbolos y las variables del término t **(coinciden sobre t)**, entonces $[\![t]\!]^{\mathcal{A}}_{\sigma} = [\![t]\!]^{\mathcal{A}'}_{\sigma'}$
- Si $(\mathcal{A}, \sigma)$ y $(\mathcal{A}', \sigma')$ interpretan del mismo modo los símbolos y las variables libres de la fórmula φ **(coinciden sobre φ)**, entonces $[\![\varphi]\!]^{\mathcal{A}}_{\sigma} = [\![\varphi]\!]^{\mathcal{A}'}_{\sigma'}$ (es decir $\mathcal{A} \models \varphi\sigma \iff \mathcal{A}' \models \varphi\sigma'$).

Dem: Por inducción estructural...

CLASIFICACIÓN SEMÁNTICA DE LAS FÓRMULAS DE PRIMER ORDEN

DEF:

$\varphi \in L_\Sigma$

- *es* **lógicamente válida** *si* $Mod(\varphi) = Int_\Sigma$, *es decir si* $\mathcal{A} \models \varphi\sigma$ *para toda* $(\mathcal{A}, \sigma) \in Int_\Sigma$ **(toda interpretación satisface a φ)**.
- *es* **contradictoria** *si* $Mod(\varphi) = \emptyset$, *es decir si* $\mathcal{A} \not\models \varphi\sigma$ *cualquiera que sea* $(\mathcal{A}, \sigma) \in Int_\Sigma$ **(ninguna interpretación satisface a φ. Es decir, contradictoria e insatisfactible son lo mismo)**.
- *es* **contingente** *si* $Mod(\varphi) \neq \emptyset$ *y* $Mod(\neg\varphi) \neq \emptyset$, *es decir si existen al menos dos* $(\mathcal{A}, \sigma), (\mathcal{A}', \sigma') \in Int_\Sigma$ *tales que* $\mathcal{A} \models \varphi\sigma$, $\mathcal{A}' \not\models \varphi\sigma'$.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Clasificación semántica de las fórmulas de primer orden (2)

Ej.: Dada la signatura $\Sigma = \{\underbrace{\emptyset}_{F_\Sigma}, \underbrace{R/1}_{P_\Sigma}\}$

- $\varphi \equiv (\exists x \;\; R(x)) \vee (\neg \exists x \;\; R(x))$ es **lógicamente válida**

  Sea $(\mathcal{A}, \sigma) \in Int_\Sigma$

  $$[\![\varphi]\!]^{\mathcal{A}}_{\sigma} = v_\vee([\![\exists x \;\; R(x)]\!]^{\mathcal{A}}_{\sigma}, v_\neg([\![\exists x \;\; R(x)]\!]^{\mathcal{A}}_{\sigma}))$$

  $$= v_\vee([\![\exists x \;\; R(x)]\!]^{\mathcal{A}}_{\sigma}, 1 - [\![\exists x \;\; R(x)]\!]^{\mathcal{A}}_{\sigma}) = 1$$

- $\varphi' \equiv \exists x \;\; (R(x) \wedge \neg \, R(x))$ es **contradictoria**

  Sea $(\mathcal{A}, \sigma) \in Int_\Sigma$

  $$[\![\varphi']\!]^{\mathcal{A}}_{\sigma} = 0 \Longleftrightarrow [\![R(x) \wedge \neg R(x)]\!]^{\mathcal{A}}_{\sigma[x/a]} = 0 \text{ para cada } a \in A$$

  $$\Longleftrightarrow v_\wedge([\![R(x)]\!]^{\mathcal{A}}_{\sigma[x/a]}, 1 - [\![R(x)]\!]^{\mathcal{A}}_{\sigma[x/a]}) = 0 \text{ para cada } a \in A$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Clasificación semántica de las fórmulas de primer orden (3)

Ej.: Dada la signatura $\Sigma = \{\underbrace{c/0, f/1, g/2, h/2}_{F_\Sigma}, \underbrace{R/2}_{P_\Sigma}\}$

$\varphi \equiv \exists y\ (x = g(h(f(f(c)), y), f(c)))$ es **contingente**

$\mathcal{N} \not\models \varphi\sigma$

$\mathcal{N} \models \varphi\sigma'$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

DEF:

Dadas las fórmulas $\varphi_0, \varphi_1, \cdots, \varphi_n \in L_\Sigma$ *y* n *símbolos de proposición diferentes* $p_1, \cdots, p_n \in P^0_\Sigma$ *escribimos* $\varphi_0[p_1/\varphi_1, \cdots, p_n/\varphi_n]$ *para designar a la fórmula* φ *resultante de* *sustituir* *simultáneamente todas las apariciones* $p_1, \cdots, p_n$ *en* φ_0 *por* $\varphi_1, \cdots, \varphi_n$*. Decimos que* φ *es un* **caso particular** *de* φ_0

Ej: $\varphi \equiv (\exists x \;\; R(x)) \vee (\neg \exists x \;\; R(x))$ es un caso de $p \vee \neg p$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

TEOREMA

- Dada una tautología φ_0 de la lógica proposicional, cualquier φ de la lógica de primer orden que sea un caso de φ_0 es lógicamente válida. Sin embargo, no toda fórmula de primer orden lógicamente válida es un caso de una tautología.
- Dada una contradicción φ_0 de la lógica proposicional, cualquier φ de la lógica de primer orden que sea un caso de φ_0 es contradictoria. Sin embargo, no toda fórmula de primer orden contradictoria es un caso de una contradicción.
- Dadas φ_0, φ fórmulas de la lógica de primer orden tales que φ es un caso de φ_0, entonces:
  - Si φ_0 es lógicamente válida entonces φ es lógicamente válida.
  - Si φ_0 es contradictoria entonces φ es contradictoria.

Dem: ...

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Ejs:

- $\varphi \equiv (\exists x \;\; R(x)) \vee (\neg\exists x \;\; R(x))$ es lógicamente válida y es un caso de la tautología $p \vee \neg p$.
- $\varphi \equiv \exists x \;\; (R(x) \vee \;\; \neg R(x))$ es lógicamente válida pero no puede obtenerse como un caso de una tautología ($\exists x \; (p \vee \neg p)$ **no** es una fórmula proposicional).
- $\varphi \equiv (\exists x \;\; R(x)) \wedge (\neg\exists x \;\; R(x))$ es contradictoria y es un caso de la contradicción $p \wedge \neg p$.
- $\varphi \equiv \exists x \;\; (R(x) \wedge \neg \;\; R(x))$ es contradictoria pero no puede obtenerse como un caso de una contradicción

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Ej:

Refuta la siguiente afirmación:

Dadas φ_0, φ fórmulas de la lógica de primer orden tales que φ es un caso de φ_0, entonces si φ_0 es contingente entonces φ es contingente.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Sustituciones en términos y fórmulas.

DEF:

Sean $s, t \in T_\Sigma$ *y* $x \in V$. *El término* $s[x/t]$ *resultante de sustituir las apariciones de la variable* x *dentro del término* s *por el término* t *, se define recursivamente sobre la estructura de* s *así:*

- **Casos base:**

$$\begin{array}{ll} (V) & x[x/t] = t \\ (V) & y[x/t] = y \quad si\ y \in V \setminus \{x\} \\ (c) & c[x/t] = c \quad si\ c \in F^0_\Sigma \end{array}$$

- **Caso recursivo:**

$$(F){:}\ f(s_1, \cdots, s_n)[x/t] = f(s_1[x/t], \cdots, s_n[x/t]) \quad f \in F^n_\Sigma, n > 0$$
$$s_i \in T_\Sigma,\ i \in \{1, \cdots, n\}$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Sustituciones en términos y fórmulas. (2)

DEF:

Sean $\varphi \in L_\Sigma, t \in T_\Sigma$ *y* $x \in V$. *La fórmula* $\varphi[x/t]$ *resultante de sustituir las apariciones* **libres** *de la variable* x *dentro de* φ *por el término* t *, se define recursivamente sobre la estructura de* φ *así:*

- **Casos base:**

$$\begin{array}{ll}
(\bot) & \bot[x/t] = \bot \\
(\top) & \top[x/t] = \top \\
(p) & p[x/t] = p \quad si\ p \in P^0_\Sigma \\
(=) & (s_1 = s_2)[x/t] = (s_1[x/t] = s_2[x/t]) \quad si\ s_1, s_2 \in T_\Sigma \\
(P) & P(s_1, \cdots, s_n)[x/t] = P(s_1[x/t], \cdots, s_n[x/t]) \quad si\ \ P \in P^n_\Sigma, n > 0 \\
 & \hfill s_i \in T_\Sigma,\ i \in \{1, \cdots, n\}
\end{array}$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Sustituciones en términos y fórmulas. (3)

- **Casos recursivos:**

$(\neg)$: $\varphi[x/t] = \neg\varphi_1[x/t]$ si $\varphi \equiv \neg\varphi_1$

$(\square)$: $(\varphi_1 \square \varphi_2)[x/t] = (\varphi_1[x/t] \square \varphi_2[x/t])$ para cualquier $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$

(K): Para cualquier $K \in \{\exists, \forall\}$ si $\varphi \equiv Ky\ \varphi_1$

- $\varphi[x/t] = \varphi$ si $y = x$ o $x \notin lib(\varphi)$
- $\varphi[x/t] = Ky\ \varphi_1[x/t]$ si $y \neq x,\ x \in lib(\varphi),\ y \notin var(t)$
  siendo $var(t)$ el conjunto de todas las
  variables del término t
- $\varphi[x/t] = Kz\ \varphi_1[y/z][x/t]$ si $y \neq x,\ x \in lib(\varphi),\ y \in var(t)$
  y $z \neq x$ tal que $z \notin var(t)$ y $z \notin lib(\varphi)$

(es decir, cuidando de renombrar las variables ligadas de φ para evitar que las variables de t resulten afectadas por cuantificadores)

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Lema de sustitución (4)

LEMA DE SUSTITUCIÓN

Dado $t \in T_\Sigma$

- Para todo $s \in T_\Sigma$ y toda $(\mathcal{A}, \sigma) \in Int_\Sigma$

$$[\![s[x/t]]\!]^{\mathcal{A}}_{\sigma} = [\![s]\!]^{\mathcal{A}}_{\sigma[x/[\![t]\!]^{\mathcal{A}}_{\sigma}]}$$

- Para todo $\varphi \in L_\Sigma$ y toda $(\mathcal{A}, \sigma) \in Int_\Sigma$

$$[\![\varphi[x/t]]\!]^{\mathcal{A}}_{\sigma} = [\![\varphi]\!]^{\mathcal{A}}_{\sigma[x/[\![t]\!]^{\mathcal{A}}_{\sigma}]}$$

Dem: Por inducción estructural...

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Lema de sustitución (5)

Ej.:

- La fórmula $\varphi \equiv \exists y(g(y,y) = x)$ interpretada en $(\mathcal{N}, \sigma)$ afirma que $\sigma(x)$ es par

$$\mathcal{N} \models \varphi\sigma \iff \sigma(x) \text{ es par}$$

$$\begin{aligned}
[\![\varphi]\!]^{\mathcal{N}}_{\sigma} = 1 \quad &\iff [\![g(y,y)]\!]^{\mathcal{N}}_{\sigma[y/n]} = \sigma(x) \text{ para algún } n \in \mathbb{N} \\
&\iff n + n = \sigma(x) \quad \text{para algún } n \in \mathbb{N} \\
&\iff 2n = \sigma(x) \quad \text{para algún } n \in \mathbb{N} \\
&\iff \sigma(x) \text{ es par}
\end{aligned}$$

- $\varphi[x/h(y,y)]$ interpretada en $(\mathcal{N}, \sigma)$ afirma que $\sigma^2(y)$ es par

$$\mathcal{N} \models \varphi\sigma \iff \sigma^2(y) \text{ es par}$$

$$\varphi[x/h(y,y)] \equiv \exists y(g(y,y) = x)[y/z][x/h(y,y)] \equiv \exists z(g(z,z) = x)[x/h(y,y)] \equiv \exists z(g(z,z) = h(y,y))$$

Por el lema de sustitución $\quad [\![\varphi[x/h(y,y)]]\!]^{\mathcal{N}}_{\sigma} = [\![\varphi]\!]^{\mathcal{N}}_{\sigma[x/[\![h(y,y)]\!]^{\mathcal{N}}_{\sigma}]}$

$$[\![h(y,y)]\!]^{\mathcal{N}}_{\sigma} = \sigma^2(y)$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Lema de sustitución (6)

- $\varphi[x/f(y)]$ interpretada en $(\mathcal{N},\sigma)$ afirma que $suc(\sigma(y))$ es par

$$\mathcal{N} \models \varphi\sigma \iff suc(\sigma(y)) \text{ es par}$$

$$\boldsymbol{\varphi[x/f(y)] \equiv \exists y(g(y,y)=x)[y/z][x/f(y)] \equiv \exists z(g(z,z)=f(y))}$$

Por el lema de sustitución $\quad [\![\varphi[x/f(y)]]\!]^{\mathcal{N}}_{\sigma} = [\![\varphi]\!]^{\mathcal{N}}_{\sigma[x/[\![f(y)]\!]^{\mathcal{N}}_{\sigma}]}$

$$[\![f(y)]\!]^{\mathcal{N}}_{\sigma} = suc(\sigma(y))$$

- $\varphi' \equiv \exists y(g(y,y)=f(y))$ (mera sustitución sintáctica de x por f(y)) interpretada en $(\mathcal{N},\sigma)$ no afirma que $suc(\sigma(y))$ es par sino que, independientemente del valor que σ asigna a y, existe un número natural n tal que $2n = suc(n)$ lo cual es cierto para $n=1$.

$$\begin{aligned} [\![\varphi']\!]^{\mathcal{N}}_{\sigma} = 1 \quad &\iff [\![g(y,y)]\!]^{\mathcal{N}}_{\sigma[y/n]} = [\![suc(y)]\!]^{\mathcal{N}}_{\sigma[y/n]} \text{para algún } n \in \mathbb{N} \\ &\iff n+n = suc(n) \quad \text{para algún } n \in \mathbb{N} \\ &\iff 2n = suc(n) \quad \text{para algún } n \in \mathbb{N} \end{aligned}$$

lo que es cierto para $n=1$

$\varphi[x/f(y)]$ y φ' interpretadas en $(\mathcal{N},\sigma)$ no dicen lo mismo.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Consecuencia lógica

DEF:

Dados $\Phi \subseteq L_\Sigma$ (conjunto de premisas o hipótesis) y $\psi \in L_\Sigma$ (conclusión o tesis), decimos que ψ es **consecuencia lógica** *de Φ, lo que escribiremos $\Phi \models \psi$, si todo modelo de Φ lo es de ψ (i.e. dada $(\mathcal{A}, \sigma) \in Int_\Sigma$ si $\mathcal{A} \models \Phi\sigma$, entonces $\mathcal{A} \models \psi\sigma$)*

- $\Phi \not\models \psi$ denota que ψ no es consecuencia lógica de Φ (i.e. hay al menos un modelo de Φ que no es modelo de ψ. A dicho modelo se le llama **interpretación contraejemplo**)

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Relación entre consecuencia lógica, insastisfactibilidad y validez lógica.

PROP.:

- ψ es lógicamente válida si y sólo si $\models \psi$.
- ψ es contradictoria si y sólo si $\models \neg\psi$.

TEOREMA DE LA DEDUCCIÓN: Sean $\Phi \subseteq L_\Sigma$ y $\varphi, \psi, \varphi_1, \cdots \varphi_n \in L_\Sigma$. Entonces

1) $\Phi \models (\varphi \to \psi)$ si y sólo si $\Phi \cup \{\varphi\} \models \psi$

2) $\Phi \models \varphi_1 \wedge \cdots \wedge \varphi_n \to \psi$ si y sólo si $\Phi \cup \{\varphi_1, \cdots, \varphi_n\} \models \psi$

3) $\varphi_1, \cdots, \varphi_n \models \psi \Leftrightarrow \; \models \varphi_1 \wedge \cdots \wedge \varphi_n \to \psi$

$\Leftrightarrow \varphi_1 \wedge \cdots \wedge \varphi_n \to \psi$ es lógicamente válida.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Relación entre consecuencia lógica, insastisfactibilidad y validez lógica. (2)

TEOREMA DE LA REDUCCIÓN AL ABSURDO: Sean $\Phi \subseteq L_\Sigma$ y $\varphi, \psi, \varphi_1, \cdots \varphi_n \in L_\Sigma$. Entonces

1) $\Phi \models \psi$ si y sólo si $\Phi \cup \{\neg\psi\}$ es insatisfactible

2) $\varphi_1, \cdots, \varphi_n \models \psi \Leftrightarrow \ \varphi_1 \wedge \cdots \wedge \varphi_n \wedge \neg\psi$ es contradictoria.

COROLARIO: Si existe $(\mathcal{A}, \sigma) \in Int_\Sigma$ tal que $\mathcal{A} \models \varphi_1 \wedge \cdots \wedge \varphi_n \wedge \neg\psi \ \sigma$ entonces ψ no es consecuencia lógica de $\{\varphi_1, \cdots, \varphi_n\}$ y a dicha $(\mathcal{A}, \sigma) \in Int_\Sigma$ se la llama interpretación contraejemplo

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Validez de inferencias y sustitución.

La validez de una inferencia se preserva al realizar la misma sustitución de símbolos de proposición por fórmulas tanto en las premisas como en la conclusión.

TEOREMA: Si $\Phi \models \psi$ entonces $\Phi[p_1/\varphi_1, \cdots, p_n/\varphi_n] \models \psi[p_1/\varphi_1, \cdots, p_n/\varphi_n]$

TEOREMA: $t, t', t'' \in T_\Sigma, \ \ \varphi \in L_\Sigma, \ \ x \in V.$

(RF) $\models (t = t)$ reflexividad de la igualdad

(SM) $(t = t') \models (t' = t)$ simetría de la igualdad

(TR) $(t = t'), (t' = t'') \models (t = t'')$ transitividad de la igualdad

(ST) $\varphi[x/t], (t = t') \models \varphi[x/t']$ sustitutividad de la igualdad

($\forall$ HP) $\forall x\varphi \models \varphi[x/t]$ hipótesis universal

($\exists$ TS) $\varphi[x/t] \models \exists x\varphi$ Tesis existencial

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Validez de inferencias y sustitución. (2)

Dem de (ST):

Sea $(\mathcal{A}, \sigma) \in \mathit{Int}_\Sigma$ tal que $(\mathcal{A}, \sigma) \in \mathit{Mod}(\{\varphi[x/t], (t = t')\})$, entonces
$\mathcal{A} \models \varphi[x/t]\,\sigma, \;\; \mathcal{A} \models (t = t')\,\sigma$.

Por el lema de sustitución:
$\mathcal{A} \models \varphi\,\sigma[x/[\![t]\!]^{\mathcal{A}}_{\sigma}], \;\; [\![t]\!]^{\mathcal{A}}_{\sigma} = [\![t']\!]^{\mathcal{A}}_{\sigma}$

Luego $\mathcal{A} \models \varphi\,\sigma[x/[\![t']\!]^{\mathcal{A}}_{\sigma}]$ y aplicando el lema de sustitución:
$\mathcal{A} \models \varphi[x/t']\,\sigma \;\; (\Longleftrightarrow (\mathcal{A}, \sigma) \in \mathit{Mod}(\varphi[x/t']))$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Formalización y validez de razonamientos.

Al formalizar un enunciado en lógica de primer orden: Se construye una fórmula que en una interpretación adecuada expresa lo afirmado en el enunciado. Para ello ha de elegirse una signatura Σ y basarse en una interpretación particular que sirva de referencia.

Al estudiar la validez de un razonamiento: No se fija ninguna interpretación particular. Se consideran **todas** las posibles interpretaciones.

Al estudiar la invalidez de un razonamiento: Se construye una interpretación contraejemplo.

Ejemplo de razonamiento lógicamente correcto:

Algunos mamíferos leen	$\exists$ x (M(x) $\wedge$ L(x))
Todos los que leen disfrutan	$\forall$ x (L(x) $\rightarrow$ D(x))
$\therefore$ Algunos mamíferos disfrutan	$\therefore \exists$ x (M(x) $\wedge$ D(x))

donde escogemos: $\Sigma = \{\underbrace{M/1, D/1, L/1}_{P_\Sigma}\}$

y nos basamos en la : Σ-estructura $\mathcal{S} = (\mathrm{U}, \emptyset, \{\mathrm{M}^{\mathcal{S}}/1, \mathrm{D}^{\mathcal{S}}/1, \mathrm{L}^{\mathcal{S}}/1\})$
U (universo de discurso): Seres vivos
$\mathrm{L}^{\mathcal{S}}$(x): x lee
$\mathrm{D}^{\mathcal{S}}$(x) : x disfruta
$\mathrm{M}^{\mathcal{S}}$(x) : x es mamífero

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Formalización y validez de razonamientos. (3)

$$\begin{array}{c} \varphi_1 : \exists\, x\ (M(x) \wedge L(x)) \\ \varphi_2 : \forall\, x\ (L(x) \to D(x)) \\ \hline \therefore \psi : \exists\, x\ (M(x) \wedge D(x)) \end{array}$$

- $\varphi_1, \varphi_2 \models \psi$

Dem. : Sea $(\mathcal{A}, \sigma) \in Int_\Sigma$ tal que $(\mathcal{A}, \sigma) \in Mod(\{\varphi_1, \varphi_2\})$

$$\begin{aligned} \mathcal{A} \models \varphi_1\ \sigma \quad &\iff [\![M(x) \wedge L(x)]\!]^{\mathcal{A}}_{\sigma[x/a]} = 1 \text{ para algún } a \in A \\ &\iff v_\wedge(M^{\mathcal{A}}(a), L^{\mathcal{A}}(a)) = 1 \text{ para algún } a \in A \\ &\iff M^{\mathcal{A}}(a) = L^{\mathcal{A}}(a) = 1 \text{ para algún } a \in A \end{aligned}$$

$$\begin{aligned} \mathcal{A} \models \varphi_2\ \sigma \quad &\iff [\![L(x) \to D(x)]\!]^{\mathcal{A}}_{\sigma[x/b]} = 1 \text{ para cualquier } b \in A \\ &\iff v_\to(L^{\mathcal{A}}(b), D^{\mathcal{A}}(b)) = 1 \text{ para cualquier } b \in A \\ & \text{En particular para } b = a. \end{aligned}$$

Luego hemos encontrado un $a \in A$ tal que $M^{\mathcal{A}}(a) = D^{\mathcal{A}}(a) = 1$.

Por lo tanto $\mathcal{A} \models \exists x(M(x) \wedge D(x))\ \sigma$

$$\iff (\mathcal{A}, \sigma) \in Mod(\psi)$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Formalización y validez de razonamientos. (4)

Ejemplo de razonamiento lógicamente inválido:

Alguien mató a Trotski	$\exists$ x R(x,a)
Stalin pagó a alguien	$\exists$ x Q(c,x)
$\therefore$ Stalin pagó a alguien que mató a Trotski	$\therefore \exists$ x (Q(c,x)$\wedge$ R(x,a))

donde escogemos: $\Sigma = \{\underbrace{a/0, c/0}_{F_\Sigma}, \underbrace{R/2, Q/2}_{P_\Sigma}\}$

y nos basamos en la : Σ-estructura $\mathcal{S} = (\mathrm{U}, \{\mathrm{a}^{\mathcal{S}}/0, \mathrm{c}^{\mathcal{S}}/0\}, \{\mathrm{R}^{\mathcal{S}}/2, \mathrm{Q}^{\mathcal{S}}/2\})$

U (universo de discurso): Seres vivos

$\mathrm{a}^{\mathcal{S}}$: a representa a Trotski

$\mathrm{c}^{\mathcal{S}}$: c representa a Stalin

$\mathrm{R}^{\mathcal{S}}$(x,y) : x mató a y

$\mathrm{Q}^{\mathcal{S}}$(x,y) : x pagó a y

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Formalización y validez de razonamientos. (5)

$$\begin{array}{c} \varphi_1 : \exists\ x\ R(x,a) \\ \varphi_2 : \exists\ x\ Q(c,x) \\ \hline \therefore \psi : \exists\ x\ (Q(c,x) \wedge R(x,a)) \end{array}$$

- $\varphi_1, \varphi_2 \not\models \psi$

Dem. : Basta dar una Σ-estructura contraejemplo ya que todas las fórmulas son cerradas.

Σ-estructura contraejemplo: $\mathcal{A} = (A, \{a^{\mathcal{A}}/0, c^{\mathcal{A}}/0\}, \{R^{\mathcal{A}}/2, Q^{\mathcal{A}}/2\})$

donde:

$A = \{$Stalin, Trotski, Pepe, Paco $\}$
$a^{\mathcal{A}} =$ Trotski
$c^{\mathcal{A}} =$ Stalin
$R^{\mathcal{A}} = \{$(Pepe, Trotski)$\}$
$Q^{\mathcal{A}} = \{$(Stalin, Paco)$\}$

$\mathcal{A} \models \varphi_1$ ya que (Pepe, Trotski) $\in R^{\mathcal{A}}$
$\mathcal{A} \models \varphi_2$ ya que (Stalin, Paco) $\in Q^{\mathcal{A}}$
$\mathcal{A} \not\models \psi$ ya que Pepe $\neq$ Paco y $R^{\mathcal{A}}$ y $Q^{\mathcal{A}}$ son unitarios.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Equivalencia lógica

DEF:

Dos fórmulas φ y ψ son lógicamente equivalentes, $\varphi \sim \psi$, si $Mod(\varphi) = Mod(\psi)$ *(i.e. $[\![\varphi]\!]^{\mathcal{A}}_{\sigma} = [\![\psi]\!]^{\mathcal{A}}_{\sigma}$ para cualquier interpretación $(\mathcal{A}, \sigma)$).*

PROP.:

A) $\sim \subseteq L_\Sigma \times L_\Sigma$ es una relación de equivalencia

B) Dadas $\varphi, \psi \in L_\Sigma$ se tiene

$$\begin{aligned} \varphi \sim \psi &\Leftrightarrow \models \varphi \leftrightarrow \psi \\ &\Leftrightarrow \models \varphi \rightarrow \psi \text{ y } \models \psi \rightarrow \varphi \\ &\Leftrightarrow \varphi \models \psi \text{ y } \psi \models \varphi \end{aligned}$$

C) (Propiedades de reemplazamiento):

[c.1)] (Reemplazamiento de subfórmulas equivalentes): Si $\chi(\varphi)$ es una fórmula que contiene a φ y $\varphi \sim \psi$, entonces $\chi(\varphi) \sim \chi(\psi)$ siendo $\chi(\psi)$ el resultado de reemplazar una o varias apariciones de φ en χ por ψ

[c.2)] (Reemplazamiento en fórmulas equivalentes de símbolos de proposición por fórmulas): Si $\chi_1 \sim \chi_2$ entonces $\chi_1[\overline{p}/\overline{\varphi}] \sim \chi_2[\overline{p}/\overline{\varphi}] \quad \forall\, \overline{p} \in (\mathrm{P}^0_\Sigma)^n, \forall\, \overline{\varphi} \in L^n_\Sigma$

Equivalencia lógica. (2)

- Se puede demostrar la equivalencia lógica de dos fórmulas "encadenando" equivalencias lógicas de subfórmulas.
- Todas las leyes de equivalencia lógica de la lógica proposicional se mantienen en la lógica de primer orden (consecuencia de que las fórmulas de primer orden que son casos de tautologías son lógicamente válidas).

El lenguaje de la lógica de primer orden

Equivalencia lógica. Leyes de los cuantificadores

Leyes de los cuantificadores. Dadas $\varphi, \psi \in L_\Sigma$

$\exists x\ \exists y\ \varphi \sim \exists y\ \exists x\ \varphi$ $\forall x\ \forall y\ \varphi \sim \forall y\ \forall x\ \varphi$	Cuantificaciones sucesivas
$\neg\exists x\ \varphi \sim \forall x\ \neg\varphi$ $\neg\forall x\ \varphi \sim \exists x\ \neg\varphi$	Negación de cuantificaciones
$\exists x\ \varphi \sim \varphi$ si $x \notin lib(\varphi)$ $\forall x\ \varphi \sim \varphi$ si $x \notin lib(\varphi)$	Cuantificaciones sin efecto
$\exists x\ \varphi\ \vee\ \exists x\ \psi \sim \exists x\ (\varphi \vee \psi)$ $\forall x\ \varphi\ \wedge\ \forall x\ \psi \sim \forall x\ (\varphi \wedge \psi)$ $\exists x\ \varphi\ \vee\ \psi \sim \exists x\ (\varphi \vee \psi)$ si $x \notin lib(\psi)$ $\exists x\ \varphi\ \wedge\ \psi \sim \exists x\ (\varphi \wedge \psi)$ si $x \notin lib(\psi)$ $\forall x\ \varphi\ \vee\ \psi \sim \forall x\ (\varphi \vee \psi)$ si $x \notin lib(\psi)$ $\forall x\ \varphi\ \wedge\ \psi \sim \forall x\ (\varphi \wedge \psi)$ si $x \notin lib(\psi)$	Cuantificaciones dentro de conjunciones o disyunciones
$\exists x\ \varphi\ \rightarrow\ \psi \sim \forall x\ (\varphi \rightarrow \psi)$ si $x \notin lib(\psi)$ $\forall x\ \varphi\ \rightarrow\ \psi \sim \exists x\ (\varphi \rightarrow \psi)$ si $x \notin lib(\psi)$ $\varphi\ \rightarrow\ \exists x\ \psi \sim \exists x\ (\varphi \rightarrow \psi)$ si $x \notin lib(\varphi)$ $\varphi\ \rightarrow\ \forall x\ \psi \sim \forall x\ (\varphi \rightarrow \psi)$ si $x \notin lib(\varphi)$	Cuantificaciones dentro de condicionales
$\exists x\ \varphi \sim \exists y\ \varphi[x/y]$ si $y \neq x$ e $y \notin lib(\varphi)$ $\forall x\ \varphi \sim \forall y\ \varphi[x/y]$ si $y \neq x$ e $y \notin lib(\varphi)$	Renombramiento de variables ligadas

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Equivalencia lógica. Leyes de los cuantificaderes (2)

- $\neg\exists x\ \varphi \sim \forall x\ \neg\varphi$

$$\begin{aligned}
(\mathcal{A},\sigma) \in Mod(\neg\exists x\ \varphi) &\iff [\![\neg\exists x\ \varphi]\!]_{\sigma}^{\mathcal{A}} = 1\\
&\iff v_{\neg}([\![\exists x\ \varphi]\!]_{\sigma}^{\mathcal{A}}) = 1\\
&\iff [\![\exists x\ \varphi]\!]_{\sigma}^{\mathcal{A}} = 0\\
&\iff [\![\varphi]\!]_{\sigma[x/a]}^{\mathcal{A}} = 0 \ \text{ para cualquier } a \in A\\
&\iff [\![\neg\varphi]\!]_{\sigma[x/a]}^{\mathcal{A}} = 1 \ \text{ para cualquier } a \in A\\
&\iff [\![\forall x \neg\varphi]\!]_{\sigma}^{\mathcal{A}} = 1\\
&\iff (\mathcal{A},\sigma) \in Mod(\forall x\ \neg\varphi)
\end{aligned}$$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Equivalencia lógica. Leyes de los cuantificadores (3)

- $\neg\forall x\, \varphi \sim \exists x\, \neg\varphi$

$$\begin{aligned} \neg\forall x\, \varphi &\sim \neg\forall x\, \neg\neg\varphi && \text{doble negación y reemplazamiento c.1)} \\ &\sim \neg\neg\exists x\, \neg\varphi && \text{negación de cuantificaciones y reemplazamiento c.1)} \\ &\sim \exists x\, \neg\varphi && \text{doble negación} \end{aligned}$$

- $\varphi \;\to\; \exists x\, \psi \sim \exists x\, (\varphi \to \psi)$ si $x \;\notin\; lib(\varphi)$

$$\begin{aligned} \varphi \;\to\; \exists x\, \psi &\sim \neg\varphi \vee \exists x\, \psi && \text{relación entre } \to,\ \neg \text{ y } \vee \\ &\sim \exists x\, \psi \vee \neg\varphi && \text{conmutatividad de } \vee \\ &\sim \exists x(\psi \vee \neg\varphi) && \text{cuantificaciones dentro de disyunciones,} x \notin lib(\neg\varphi) \\ &\sim \exists x(\neg\varphi \vee \psi) && \text{conmutatividad de } \vee \text{ y reemplazamiento c.1)} \\ &\sim \exists x(\varphi \to \psi) && \text{relación entre } \to,\ \neg \text{ y } \vee \text{ y reemplazamiento c.1)} \end{aligned}$$

El lenguaje de la lógica de primer orden

Equivalencia lógica. Leyes de los cuantificadores (4)

- Si $x \in lib(\varphi)$ entonces $\varphi \to \exists x\, \psi \not\sim \exists x\, (\varphi \to \psi)$

$$\Sigma = \{\underbrace{M/1, D/1}_{P_\Sigma}\}, \quad \varphi \equiv M(x), \quad \psi \equiv D(x)$$

$$\Sigma\text{-estructura } \mathcal{A} = (\mathrm{A}, \emptyset, \{\mathrm{M}^{\mathcal{A}}/1, \mathrm{D}^{\mathcal{A}}/1\})$$

$$\mathrm{A} = \{a, b\}, \quad \mathrm{M}^{\mathcal{A}} = \{a\}, \quad \mathrm{D}^{\mathcal{A}} = \emptyset, \quad \sigma(x) = a$$

$\mathcal{A} \not\models (\mathrm{M(x)} \to \exists x\, \mathrm{D(x)})\, \sigma$ ya que $[\![\mathrm{M}(x)]\!]^{\mathcal{A}}_{\sigma} = \mathrm{M}^{\mathcal{A}}(\sigma(x)) = 1, \quad [\![\exists x\, \mathrm{D}(x)]\!]^{\mathcal{A}}_{\sigma} = 0$

$\mathcal{A} \models \exists x\, (\mathrm{M(x)} \to \mathrm{D(x)})\, \sigma$ ya que $\mathcal{A} \models (\mathrm{M(x)} \to \mathrm{D(x)})\sigma[x/b]$

puesto que $\mathcal{A} \not\models \mathrm{M(x)}\, \sigma[x/b], \quad ([\![\mathrm{M}(x)]\!]^{\mathcal{A}}_{\sigma[x/b]} = \mathrm{M}^{\mathcal{A}}(b) = 0)$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Formas normales en lógica de primer orden

Formas normales conjuntivas y disyuntivas

1. FORMAS NORMALES CONJUNTIVAS Y DISYUNTIVAS

(Para fórmulas sin cuantificadores)

DEF:

- **Literal:** *cualquier fórmula de la forma* λ *(literal positivo) o* $\neg\lambda$ *(literal negativo), donde* λ *es una fórmula atómica.*
  Un literal se llama ecuacional si es una ecuación y relacional en otro caso.
- **Cláusula conjuntiva:** *cualquier fórmula que es una conjunción de literales.*
- **Cláusula disyuntiva:** *cualquier fórmula que es una disyunción de literales.*
- *Una fórmula está en* **forma normal conjuntiva (FNC)** *si es una conjunción de cláusulas disyuntivas.*
- *Una fórmula está en* **forma normal disjuntiva (FND)** *si es una disyunción de cláusulas conjuntivas.*

Formas normales conjuntivas y disyuntivas (2)

Convenios:

- Un único literal puede considerarse, indistintamente, como conjunción o como disyunción.
- $\perp$ está en FND y es una cláusula disyuntiva (representa una disyunción vacía (trivialmente falsa)).
- $\top$ está en FNC y es una cláusula conjuntiva (representa una conjunción vacía (trivialmente cierta)).

Observaciones:

- Una única cláusula disyuntiva puede considerarse que está en FNC (con una cláusula) o en FND (con varias cláusulas conjuntivas unitarias).
- Una única cláusula conjuntiva puede considerarse que está en FND (con una cláusula) o en FNC (con varias cláusulas disyuntivas unitarias).

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Formas normales en lógica de primer orden

Formas normales conjuntivas y disyuntivas

(3)

TEOREMA: Dada $\varphi \in L_\Sigma$, sin cuantificadores, pueden encontrarse sendas fórmulas FND(φ) y FNC(φ) tales que

$$FND(\varphi) \sim \varphi, \quad FND(\varphi) \text{ está en FND}$$

$$FNC(\varphi) \sim \varphi, \quad FNC(\varphi) \text{ está en FNC}$$

(Obs: Las $FNC(\varphi)$ y $FND(\varphi)$ no son únicas.)

Dem: ...

Ej: $\Sigma = \{P/1, Q/1\}$
$\varphi \equiv \neg(P(x) \wedge Q(y)) \rightarrow (\neg P(u) \vee \neg Q(v))$

$$\varphi \sim (P(x) \wedge Q(y)) \vee \neg P(u) \vee \neg Q(v) \qquad \textbf{que está en FND}$$

$$\sim (P(x) \vee \neg P(u) \vee \neg Q(v)) \wedge (Q(y) \vee \neg P(u) \vee \neg Q(v)) \qquad \textbf{que está en FNC}$$

(Todos los cuantificadores aparecen al principio)

2. FORMA NORMAL PRENEXA

DEF:

Una fórmula $\varphi \in L_\Sigma$ *de primer orden está en* **forma normal prenexa** *si se ajusta al esquema* $K_1 x_1 \cdots K_n x_n \; \eta$ *donde* $n \geq 0$, $K_i \in \{\exists, \forall\}$ *para cada* $i \in \{1, \cdots, n\}$, $x_1, \cdots x_n$ *son variables diferentes y* η *es una fórmula sin cuantificadores, llamada el núcleo de la forma prenexa. A* $K_1 x_1 \cdots K_n x_n$ *se le llama prefijo de la forma prenexa.*

TEOREMA: Dada $\varphi \in L_\Sigma$ de primer orden, puede encontrarse otra fórmula en forma normal prenexa $\mathrm{FP}(\varphi)$ tal que

$$FP(\varphi) \sim \varphi, \quad voc(FP(\varphi)) = voc(\varphi) \text{ y } lib(FP(\varphi)) = lib(\varphi)$$

Dem: ...

Obs: La $FP(\varphi)$ de φ no es única.

El lenguaje de la lógica de primer orden

Formas normales en lógica de primer orden

Forma normal prenexa (2)

Ej: $\Sigma = \{P/1, Q/1\}$ $\quad \varphi \equiv \neg \forall x \, \forall y \, (P(x) \wedge Q(y)) \to (\exists u \, \neg P(u) \vee \exists v \, \neg Q(v))$

$$\varphi \sim \exists x \, \exists y \, \neg \, (P(x) \wedge Q(y)) \to (\exists u \, \neg P(u) \vee \exists v \, \neg Q(v))$$

reemplazamiento c1) y negación de cuantific.

$$\sim \exists x \, \exists y \, \neg \, (P(x) \wedge Q(y)) \to \exists u \, (\neg P(u) \vee \exists v \, \neg Q(v))$$

cuantific. en disyunciones
$u \notin lib(\exists v \, \neg Q(v))$

$$\sim \exists x \, \exists y \, \neg \, (P(x) \wedge Q(y)) \to \exists u \, \exists v \, (\neg Q(v) \vee \neg P(u))$$

conmutatividad $\vee$ y cuantific. en disyunciones
$v \notin lib(\neg P(u))$

$$\sim \forall x \, \forall y \, (\neg(P(x) \wedge Q(y)) \to \exists u \, \exists v \, (\neg Q(v) \vee \neg P(u)))$$

cuantific. en $\to$
$x, y \notin lib(\exists u \exists v (\neg Q(v) \vee \neg P(u)))$

$$\sim \underbrace{\forall x \, \forall y \, \exists u \, \exists v \, (\neg(P(x) \wedge Q(y)) \to (\neg Q(v) \vee \neg P(u)))}_{\textbf{está en FP}}$$

cuantific. en $\to$
$u, v \notin lib(\neg P(x) \wedge \neg Q(y))$

$$\sim \underbrace{\forall x \, \forall y \, \exists u \, \exists v \, ((P(x) \wedge Q(y)) \vee \neg Q(v) \vee \neg P(u))}_{\textbf{está en FP con el núcleo en FND}}$$

doble negación y relación entre $\to$, $\neg$ y $\vee$

$$\sim \underbrace{\forall x \, \forall y \, \exists u \, \exists v \, ((P(x) \vee \neg Q(v) \vee \neg P(u)) \wedge (Q(y) \vee \neg Q(v) \vee \neg P(u)))}_{\textbf{está en FP con el núcleo en FNC}}$$

distributividad

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Tableaux semánticos para la lógica de primer orden.

∗ Extensión del método de los tableaux semánticos de la lógica proposicional a la lógica de primer orden, añadiendo reglas de construcción de tableaux para los cuantificadores y la igualdad.

∗ Método de cálculo lógico que permite:

- Decidir si una fórmula dada es consecuencia lógica de unas premisas y construir un contraejemplo en el caso de que no lo sea.
- Decidir si un conjunto de fórmulas es satisfactible.
- Decidir si una fórmula es lógicamente válida.

∗ **Procedimiento de refutación:** Para demostrar $\Phi \models \psi$ intenta demostrar que $\Phi \cup \{\neg\psi\}$ es insatisfactible. Para demostrar que φ es lógicamente válida, intenta demostrar que $\neg\varphi$ es contradictoria.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Tableaux semánticos para la lógica de primer orden. (2)

Idea base: Cada fórmula de primer orden compuesta o es un literal negativo distinto de $\neg\bot$ y de $\neg\top$ o es simplificable o es lógicamente equivalente a la disyuncción o conjunción de otras dos fórmulas más sencillas o es de uno de los dos tipos recogidos en las tablas siguientes:

Fórmulas universales

$t \in T_\Sigma$

γ	$\gamma(t)$
$\forall x\ \varphi$	$\varphi[x/t]$
$\neg\exists x\ \varphi$	$\neg\varphi[x/t]$

Dado $t \in T_\Sigma$, si una fórmula universal γ es verdadera también lo será su **particularización** $\gamma(t)$

Fórmulas existenciales

$c \in C$

δ	$\delta(c)$
$\exists x\ \varphi$	$\varphi[x/c]$
$\neg\forall x\ \varphi$	$\neg\varphi[x/c]$

C es un conjunto infinito numerable de **constantes auxiliares**, $C \cap \Sigma = \emptyset$

Si una fórmula existencial δ es verdadera, podemos escoger como **testigo** una constante auxiliar c, que no se haya utilizado, y suponer cierta la fórmula $\delta(c)$ (**ejemplo** de δ)

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Tableaux semánticos para la lógica de primer orden. (3)

También se necesitan reglas para tratar la igualdad:

	t = t’	t = t’ t’ = t”	t = t’ $\varphi[x/t]$
t = t	t’ = t	t = t”	$\varphi[x/t']$
reflexividad	simetría	transitividad	sustitutividad

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Reglas de construcción de tableaux.

Un **tableau** T para un conjunto finito de fórmulas $\Phi = \{\varphi_1, \cdots, \varphi_n\}$ es cualquier árbol de fórmulas construído paso a paso mediante las siguientes reglas de formación:

$[R_{ini}], [R_\sigma], [R_\alpha], [R_\beta]$ (las mismas que para la lógica proposicional) y

- **Reglas para los cuantificadores:**
  - $[R_\gamma]$: Si θ es una rama abierta de T con un nodo etiquetado con una fórmula universal γ, se obtiene T' alargando θ con $\gamma(t)$ siempre que $t \in T_\Sigma$ sea **adecuado para θ y $\gamma(t)$ no aparezca ya en θ**.

    Un término t es **adecuado para θ** si está **formado por constantes, símbolos de función y variables libres que aparezcan en las fórmulas de la rama**. Por convenio, la constante auxiliar c_0 se utiliza como término adecuado si no hay ningún otro término adecuado a la rama (es decir cuando la rama no contiene variables libres ni constantes).
  - $[R_\delta]$: Si θ es una rama abierta de T con un nodo etiquetado con una fórmula existencial δ, se obtiene T' alargando θ con un nodo etiquetado con $\delta(c)$ siempre que el testigo c sea una nueva constante auxiliar que no aparezca en θ y supuesto que la rama θ no contiene ya un ejemplo para δ.

y

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Reglas de construcción de tableaux. (2)

- **Reglas para la igualdad:**
  - $[R_{RF}]$: Si θ es una rama abierta de T y $t \in T_\Sigma$ es **adecuado** a θ, se obtiene T' alargando θ con un nuevo nodo etiquetado con $(t = t)$, supuesto que esta fórmula no aparezca ya en θ.
  - $[R_{SM}]$: Si θ es una rama abierta de T con un nodo etiquetado con una fórmula ecuacional $(t = t')$, se obtiene T' alargando θ con un nuevo nodo etiquetado con $(t' = t)$, supuesto que esta fórmula no aparezca ya en θ.
  - $[R_{TR}]$: Si θ es una rama abierta de T con dos nodos etiquetados con las fórmulas ecuacionales $(t = t')$ y $(t' = t'')$ se obtiene T' alargando θ con un nuevo nodo etiquetado con $(t = t'')$, supuesto que esta fórmula no aparezca ya en θ.
  - $[R_{ST}]$: Si θ es una rama abierta de T con dos nodos etiquetados con las fórmulas ecuacionales $(t = t')$ y $\varphi[x/t]$ se obtiene T' alargando θ con un nuevo nodo etiquetado con $\varphi[x/t']$, supuesto que esta fórmula no aparezca ya en θ.

Un tabbleau queda **terminado** cuando no se puede aplicar ninguna regla.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Reglas de construcción de tableaux. (3)

DEF:

Una **rama** θ *de un árbol de fórmulas se llama* **cerrada** *si* $\perp$ *aparece en* θ *o para alguna fórmula* φ *aparecen en* θ *tanto* φ *como* $\neg\varphi$*. A las ramas que no son cerradas se las llama abiertas.*

Un **tableau** *se llama* **cerrado** *si todas sus ramas están cerradas.*

Las ramas cerradas se marcan con $\sharp$ *seguido de un identificador de los nodos que hacen que la rama se cierre; las ramas que quedan abiertas se marcan con* $\Uparrow$*.*

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Reglas de construcción de tableaux. Heurísticas (4)

HEURÍSTICAS

- Dependiendo del orden en que se apliquen las reglas de formación se pueden construir tableaux diferentes para un mismo conjunto de fórmulas:

  1º Conviene intentar cerrar ramas antes de expandir otras.
  2º Aplicar las reglas y "heurísticas proposicionales".
  3º Aplicar $[R_\delta]$ antes que $[R_\gamma]$
  4º Para elegir entre las aplicaciones posibles de $[R_\gamma]$ y $[R_{ST}]$ cuál parece mejor, conviene interpretar qué sentido tiene la regla que va a aplicarse en relación al problema que se quiere resolver.

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Propiedades fundamentales de los tableaux.

DEF:

Si para el conjunto de fórmulas Φ *puede construirse un tableau cerrado* T*, diremos que* T *prueba la insastifactibilidad de* Φ*. En particular si* $\Phi = \Phi_0 \cup \{\neg\psi\}$ *se dice que* T **prueba por refutación** *que* $\Phi_0 \models \psi$ *lo que se escribe* $\Phi_0 \vdash_{tb} \psi$

TEOREMA FUNDAMENTAL DE LOS TABLEAUX

Dado $\Phi \subseteq L_\Sigma$

1. Φ es insatisfactible si y sólo si Φ tiene un tableau cerrado.
2. $\Phi \models \psi$ si y sólo si $\Phi \vdash_{tb} \psi$.
3. Si T es un tableau terminado y no cerrado de Φ, a partir de cada rama abierta θ se puede construir una interpretación $(\mathcal{A}_\theta, \sigma_\theta)$ tal que $\mathcal{A}_\theta \models \Phi\ \sigma_\theta$. En este caso, si $\Phi = \Phi_0 \cup \{\neg\psi\}$, entonces $\Phi_0 \not\models \psi$ y $(\mathcal{A}_\theta, \sigma_\theta)$ construida a partir de cualquier rama abierta θ es una interpretación contraejemplo que prueba $\Phi_0 \not\models \psi$ ya que $\mathcal{A}_\theta \models \Phi_0\ \sigma_\theta$ pero $\mathcal{A}_\theta \not\models \psi\ \sigma_\theta$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

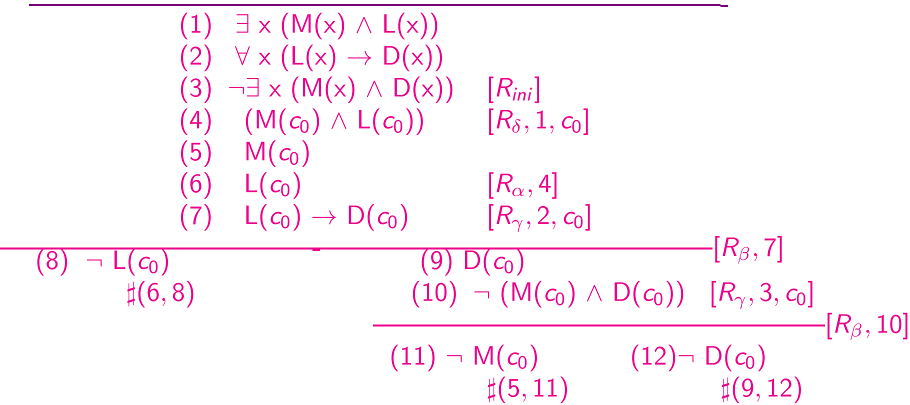
Reglas de construcción de tableaux. (5)

Ej:

$$\begin{array}{c} \varphi_1 : \exists\, x\ (M(x) \wedge L(x)) \\ \varphi_2 : \forall\, x\ (L(x) \to D(x)) \\ \hline \therefore\ \psi : \exists\, x\ (M(x) \wedge D(x)) \end{array}$$

$\Phi = \{\varphi_1, \varphi_2, \neg\psi\}$

Veamos mediante tableaux que $\varphi_1, \varphi_2 \models \psi$



(1) $\exists\, x\ (M(x) \wedge L(x))$

(2) $\forall\, x\ (L(x) \to D(x))$

(3) $\neg\exists\, x\ (M(x) \wedge D(x))$ $[R_{ini}]$

(4) $(M(c_0) \wedge L(c_0))$ $[R_\delta, 1, c_0]$

(5) $M(c_0)$

(6) $L(c_0)$ $[R_\alpha, 4]$

(7) $L(c_0) \to D(c_0)$ $[R_\gamma, 2, c_0]$

$[R_\beta, 7]$

(8) $\neg\ L(c_0)$

$\sharp(6, 8)$

(9) $D(c_0)$

(10) $\neg\ (M(c_0) \wedge D(c_0))$ $[R_\gamma, 3, c_0]$

$[R_\beta, 10]$

(11) $\neg\ M(c_0)$

$\sharp(5, 11)$

(12) $\neg\ D(c_0)$

$\sharp(9, 12)$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Reglas de construcción de tableaux. (6)

Ej: $\varphi_1 : \exists$ x R(x,a)

$\varphi_2 : \exists$ x Q(c,x)

$\therefore \psi : \exists$ x (Q(c,x)$\wedge$ R(x,a))

$\Sigma = \{a/0, c/0, R/2, S/2\}$

Veamos mediante tableaux que $\varphi_1, \varphi_2 \not\models \psi$

(1) $\exists$ x R(x,a)
(2) $\exists$ x Q(c,x)
(3) $\neg\exists$ x (Q(c,x)$\wedge$ R(x,a)) $[R_{ini}]$
(4) R(c_0,a) $[R_\delta, 1, c_0]$
(5) Q(c,c_1) $[R_\delta, 2, c_1]$
(6) $\neg$(Q(c,c_1)$\wedge$ R(c_1,a)) $[R_\gamma, 3, c_1]$

———————— $[R_\beta, 6]$

(7) $\neg$Q(c,c_1)
$\sharp(7,5)$

(8) $\neg$R(c_1,a)
(9) $\neg$(Q(c,c_0)$\wedge$ R(c_0,a)) $[R_\gamma, 3, c_0]$

———————— $[R_\beta, 9]$

(10) $\neg$ Q(c,c_0)
(12) $\neg$ (Q(c,c)$\wedge$ R(c,a)) $[R_\gamma, 3, c]$

(11) $\neg$ R(c_0,a)
$\sharp(4,11)$

———————— $[R_\beta, 12]$

(13) $\neg$ Q(c,c)
(15) $\neg$ (Q(c,a)$\wedge$ R(a,a)) $[R_\gamma, 3, a]$

(14) $\neg$ R(c,a)
(18) $\neg$ (Q(c,a)$\wedge$ R(a,a)) $[R_\gamma, 3, a]$

———————— $[R_\beta, 15]$

(16) $\neg$Q(c,a)
$\Uparrow$
θ_1

(17) $\neg$R(a,a)
$\Uparrow$
θ_2

———————— $[R_\beta, 18]$

(19) $\neg$Q(c,a)
$\Uparrow$
θ_3

(20) $\neg$R(a,a)
$\Uparrow$
θ_4

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Reglas de construcción de tableaux. (7)

Ej:

$$\begin{array}{c} \varphi_1 : \exists\ \mathsf{x}\ \mathsf{R(x,a)} \\ \varphi_2 : \exists\ \mathsf{x}\ \mathsf{Q(c,x)} \\ \hline \therefore\ \psi :\ \exists\ \mathsf{x}\ (\mathsf{Q(c,x)} \wedge\ \mathsf{R(x,a)}) \end{array}$$

$\Sigma = \{a/0, c/0, \mathrm{R}/2, \mathrm{S}/2\}$

$\boldsymbol{\varphi_1, \varphi_2 \not\models \psi}$

Basta dar una Σ-estructura contraejemplo ya que todas las fórmulas son cerradas.

Σ-estructura contraejemplo construida a partir de la rama abierta θ_1:

$$\mathcal{A} = (\mathrm{A}, \{\mathrm{a}^{\mathcal{A}}/0, \mathrm{c}^{\mathcal{A}}/0\}, \{\mathrm{R}^{\mathcal{A}}/2, \mathrm{Q}^{\mathcal{A}}/2\})$$

donde:

$\mathrm{A} = \{\mathrm{c}, \mathrm{a}, c_0, c_1\}$ (los cuatro términos adecuados a la rama θ_1)

$\mathrm{R}^{\mathcal{A}} = \{(c_0,\mathrm{a})\}$ ya que en θ_1 aparecen los literales $\mathrm{R}(c_0,\mathrm{a})$ y $\neg\ \mathrm{R}(c_1,\mathrm{a})$

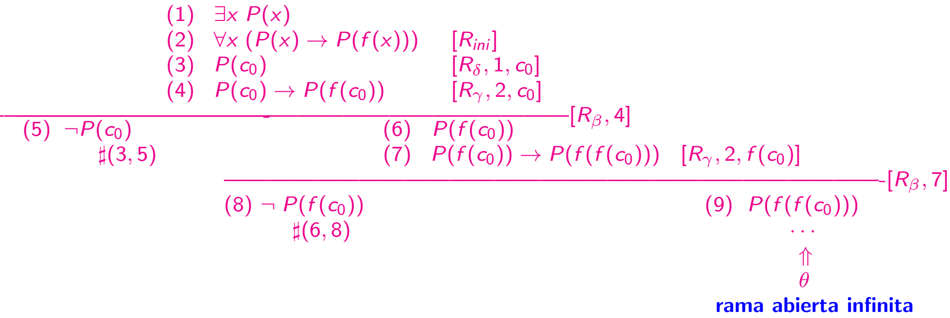
$\mathrm{Q}^{\mathcal{A}} = \{(\mathrm{c},c_1)\}$ ya que en θ_1 aparecen los literales
$\mathrm{Q}(\mathrm{c},c_1)$, $\neg\ \mathrm{Q}(\mathrm{c},c_0)$, $\neg\ \mathrm{Q}(\mathrm{c},\mathrm{c})$ y $\neg\ \mathrm{Q}(\mathrm{c},\mathrm{a})$

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Reglas de construcción de tableaux. (8)

Una diferencia importante con respecto al caso de la lógica proposicional es que puede ocurrir que la construcción de un tableau no termine (tableau infinito)

Veamos mediante tableaux que $\Phi = \{\exists x\, P(x), \forall x\, (P(x) \to P(f(x)))\}$ es satisfactible.



(1) $\exists x\, P(x)$
(2) $\forall x\, (P(x) \to P(f(x)))$ $[R_{ini}]$
(3) $P(c_0)$ $[R_\delta, 1, c_0]$
(4) $P(c_0) \to P(f(c_0))$ $[R_\gamma, 2, c_0]$

$[R_\beta, 4]$

(5) $\neg P(c_0)$
$\sharp(3,5)$

(6) $P(f(c_0))$
(7) $P(f(c_0)) \to P(f(f(c_0)))$ $[R_\gamma, 2, f(c_0)]$

$[R_\beta, 7]$

(8) $\neg\, P(f(c_0))$
$\sharp(6,8)$

(9) $P(f(f(c_0)))$
$\cdots$
$\Uparrow$
θ
rama abierta infinita

EL LENGUAJE DE LA LÓGICA DE PRIMER ORDEN

Reglas de construcción de tableaux. (9)

$\Phi = \{\exists x\, P(x), \forall x\, (P(x) \to P(f(x)))\}$ $\Sigma = \{f/1, P/1\}$

Modelo para Φ: Basta dar una Σ-estructura ya que todas las fórmulas de Φ son cerradas.

Σ-estructura modelo construida a partir de la rama abierta θ:

$$\mathcal{A} = (\mathrm{A}, \{f^{\mathcal{A}}/1\}, \{P^{\mathcal{A}}/1\})$$

donde:

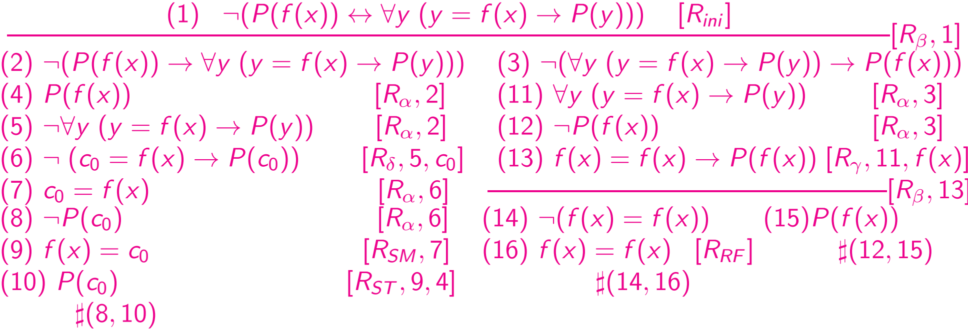
$\mathrm{A} = \{c_0, f(c_0), f(f(c_0)), \cdots\}$ (los términos adecuados a la rama θ)

$f^{\mathcal{A}} = f$

$P^{\mathcal{A}} = \{c_0, f(c_0), f(f(c_0)), \cdots\}$ ya que en θ aparecen los literales $P(c_0), P(f(c_0)), P(f(f(c_0))), \cdots$

Reglas de construcción de tableaux. (10)

Ej: Veamos mediante tableaux que $P(f(x)) \leftrightarrow \forall y\ (y = f(x) \rightarrow P(y))$ es lógicamente válida.



(1) $\neg(P(f(x)) \leftrightarrow \forall y\ (y = f(x) \rightarrow P(y)))$ $[R_{ini}]$

$[R_\beta, 1]$

(2) $\neg(P(f(x)) \rightarrow \forall y\ (y = f(x) \rightarrow P(y)))$
(4) $P(f(x))$ $[R_\alpha, 2]$
(5) $\neg\forall y\ (y = f(x) \rightarrow P(y))$ $[R_\alpha, 2]$
(6) $\neg\ (c_0 = f(x) \rightarrow P(c_0))$ $[R_\delta, 5, c_0]$
(7) $c_0 = f(x)$ $[R_\alpha, 6]$
(8) $\neg P(c_0)$ $[R_\alpha, 6]$
(9) $f(x) = c_0$ $[R_{SM}, 7]$
(10) $P(c_0)$ $[R_{ST}, 9, 4]$
$\sharp(8, 10)$

(3) $\neg(\forall y\ (y = f(x) \rightarrow P(y)) \rightarrow P(f(x)))$
(11) $\forall y\ (y = f(x) \rightarrow P(y))$ $[R_\alpha, 3]$
(12) $\neg P(f(x))$ $[R_\alpha, 3]$
(13) $f(x) = f(x) \rightarrow P(f(x))$ $[R_\gamma, 11, f(x)]$

$[R_\beta, 13]$

(14) $\neg(f(x) = f(x))$
(16) $f(x) = f(x)$ $[R_{RF}]$
$\sharp(14, 16)$

(15) $P(f(x))$
$\sharp(12, 15)$