Universidad Complutense de Madrid

Facultad de Ciencias Matemáticas

Lógica Matemática

Curso académico 2019-2020

Autor: Javier López

Versión Febrero 2020

Índice

1. Lógica de proposiciones

Definición 1. Diremos que una **proposición** es un enunciado que puede ser verdadero o falso. Nunca será una proposición cualquier enunciado que expresa duda o sentimientos. Tampoco lo serán aquellos enunciados que no tengan sentido lógico.

Un ejemplo de lo que no es proposición sería
p ≡ "Juan se cae"
q ≡ "Yo me río"
φ ≡ Juan se cae y yo me río

Conectivas lógicas. Son los símbolos que utilizamos para formalizar las proposiciones. Estos son

$$\neg \rightsquigarrow \text{ negación} \quad \wedge \rightsquigarrow \text{ conjunción} \quad \vee \rightsquigarrow \text{ disyunción} \quad \rightarrow \rightsquigarrow \text{ implicación}$$

$$\leftrightarrow \rightsquigarrow \text{ implicación} \quad \perp \rightsquigarrow \text{ falso} \quad \top \rightsquigarrow \text{ cierto}$$

Definición 2. Se denomina **formalizar** una proposición, a escribirla mediante conectivas lógicas.

Ejemplo 1: Formalizar las siguientes frases

1. Si llueve se suspende el partido.
2. Solo si llueve se suspende el partido.

tomando como proposiciones p ≡ " Llueve" y q ≡ "se suspende el partido".

(1) $p \to q$

(2) $q \to p$

Definición 3. Llamaremos **formula** a una cadena de símbolos.

Definición 4. Denotamos el **conjunto de** todos los **símbolos de proposición** como

$$\mathrm{SP} = \{p,\, q, \ldots\}$$

que es un conjunto numerable (no necesariamente finito).

Definición 5. Al conjunto formado por SP y las conectivas lógicas se le denomina **alfabeto** y lo denotamos como

$$\mathrm{A} = \mathrm{SP} \cup \{\neg,\, \wedge,\, \vee\ \rightarrow,\, \leftrightarrow,\, \top,\, \perp,\, (,\,)\}$$

denotamos por A^* al **conjunto de cadenas de símbolos** de A

$$\mathrm{A}^* = \{\varepsilon,\, a_1,\, a_2, \ldots, a_n : a_n \geq 0,\, a_i \in A,\, 1 \leq j \leq n\}$$

donde ε es la cadena vacía.

Ejemplo 2: Dado el vocabulario A $= \{a,\, b\}$ su conjunto de cadena de símbolos será el conjunto

$$\mathrm{A}^* = \{\varepsilon,\, a, b, ab, ba, aaa, aab, \ldots\}$$

Definición 6. Dado SP un conjunto de símbolos de proposición, tomamos el alfabeto A_{SP} y definimos PROP_{SP} como el menor subconjunto de A^*_{SP} que verifica

1. $SP \subseteq \mathrm{PROP}_{SP}$
2. Si $\varphi \in \mathrm{PROP}_{SP}$, entonces $(\neg\varphi) \in \mathrm{PROP}_{SP}$
3. Si $\varphi, \psi \in \mathrm{PROP}_{SP}$, entonces $(\varphi \,\square\, \psi) \in \mathrm{PROP}_{SP}$, donde

$$\square \in \{\wedge,\, \vee\, \rightarrow,\, \leftrightarrow\}$$

Veamos como se construye esta definición. Sean

$$P_0 = \mathrm{SP}$$

$$P_{n+1} = P_n \cup \{(\neg\varphi),\, (\varphi \,\square\, \psi) : \square \in \{\wedge,\, \vee\, \rightarrow,\, \leftrightarrow\},\, \varphi, \psi \in P_n\}$$

$$P = \bigcup_{i \geq 0}$$

veamos como P cumple las propiedades 1, 2 y 3 de la definición anterior. De forma trivial se verifica que $\mathrm{PROP}_{SP} \subseteq P$ y nos faltaría por demostrar la inclusión en el otro sentido.

Demostración. Sea $\varphi \in P$ entonces $\exists k$ tal que $\varphi \in P_k$ y aplicamos inducción sobre k para ver que $\varphi \in \mathrm{PROP}_{SP}$.

Para $k = 0$, por la propiedad 1 de la definición se tienen que $\varphi \in \mathrm{PROP}_{SP}$. Para $k \geq 0$

(i) $\varphi \in P_{k-1}$

(ii) $\psi \in P_{k-1}$ tal que $\varphi = (\neg\psi)$

(iii) $\psi_1, \psi_2 \in P_{k-1}$ entonces $\varphi = (\psi_1 \,\square\, \psi_2)$

□

1.1. Inducción estructural

Supongamos que queremos probar una propiedad P que cumpla $P(\varphi), \forall\varphi \in \mathrm{PROP}_{SP}$. Para ello vamos a usar una estructura basada en el *método de inducción* usual sobre $\mathbb{N}$ aplicado sobre las proposiciones. El método tiene la siguiente estructura

(1) Demostrar la **base inductiva**. Lo haremos sobre las atómicas (*i.e.* SP, $\perp, \top$)

(AT) Se cumple $P(\varphi), \forall\varphi \in \mathrm{AT}$.

(2) **Paso inductivo**. Una vez que tenemos la propiedad P probada para el caso base, suponemos la cierta la hipótesis de inducción, es decir que se cumple $P(\varphi)$, y la utilizamos para los dos casos siguientes

($\neg\varphi$) Utilizando la *h.i.* demostraremos que se cumple $P((\neg\varphi))$.

($\Box$) Suponemos que φ_1 cumple P y que φ_2 cumple P, es decir, se verifican $P(\varphi_1)$ y $P(\varphi_2)$ y entonces hay que demostrar $P(\varphi_1 \Box \varphi_2)$ con $\Box \in \{\wedge, \vee \ \rightarrow, \leftrightarrow\}$. Dependiendo de la propiedad que queramos demostrar, podremos, o bien agrupar la conectivas lógicas en un sólo caso, o bien separarlas de forma en casos particulares.

Ejemplo 3: Vamos a demostrar por inducción estructural la siguiente propiedad

P: *Toda fórmula tiene el mismo número de paréntesis abiertos y cerrados*

Para ello, vamos a denotar $|\varphi|_($ al número de paréntesis abiertos de φ y, análogamente, denotamos $|\varphi|_)$ al número de paréntesis cerrados de φ.

(AT) Si $\varphi \in SP$ ó $\varphi = \bot$ ó $\varphi = \top$, en cualquiera de los casos no hay paréntesis, luego $|\varphi|_(= |\varphi|_)$ y por tanto se verifica

$$P(\varphi), \forall\varphi \in \mathrm{AT}$$

($\neg\varphi$) Sea $\varphi \in \mathrm{PROP}_{SP}$ tal que se verifica $P(\varphi)$, es decir

$$|\varphi|_(= |\varphi|_)$$

ahora, el $|(\neg\varphi)|_(= |\varphi|_(+ 1$ y analogamente $|(\neg\varphi)|_) = |\varphi|_) + 1$, luego por *h.i.* se tiene que

$$|(\neg\varphi)|_) = |(\neg\varphi)|_($$

luego se verifica $P(\neg\varphi), \forall\varphi \in \mathrm{PROP}_{SP}$.

($\Box$) Sean φ_1, $\varphi_2 \in \mathrm{PROP}_{SP}$, supongamos que

$$\text{Se verifica } P(\varphi_1) \Rightarrow |\varphi_1|_(= |\varphi_1|_)$$

$$\text{Se verifica } P(\varphi_2) \Rightarrow |\varphi_2|_(= |\varphi_2|_)$$

y veamos que ocurre con $P(\varphi_1 \Box \varphi_2)$

$$|\varphi_1 \Box \varphi_2|_(= |\varphi_1|_(+ |\varphi_2|_(+ 1$$

$$|\varphi_1 \Box \varphi_2|_) = |\varphi_1|_) + |\varphi_2|_) + 1$$

y, por *h.i.* se tiene que

$$|\varphi_1 \Box \varphi_2|_) = |\varphi_1 \Box \varphi_2|_($$

finalizando así la demostración.

Definición 7. Sea A un alfabeto y $\omega \in A^*$ decimos que ω' es **prefijo** de ω si $\exists\omega''$ tal que

$$\omega = \omega'\omega''$$

con, $\omega = a_1 \dots a_n$, entonces $\exists k$, $0 \leq k \leq n$ tal que $\omega' = a_1 \dots a_k$. Diremos que ω' es **prefijo propio** si $\omega' \neq \varepsilon$ y $\omega' \neq \omega$.

Ejemplo 4: Sea $A = \{a, b\}$ y $\omega = aababb$ entonces

- Si $k = 0 \Rightarrow \omega' = \varepsilon$
- Si $k = 1 \Rightarrow \omega' = a$
- Si $k = 2 \Rightarrow \omega' = aa$
- Si $k = 3 \Rightarrow \omega' = aab$
- Si $k = 4 \Rightarrow \omega' = aaba$
- Si $k = 5 \Rightarrow \omega' = aabab$
- Si $k = 5 \Rightarrow \omega' = \omega$

Ejemplo 5: Sea φ' prefijo propio de φ, vamos a probar por inducción estructural la propiedad

P: *El número de paréntesis cerrados de φ' es menor que el número de paréntesis abiertos.*

utilizando la notación del ejemplo (III).

(AT) Sea $\varphi \in SP$, supongamos $\varphi = p$ entonces, o bien $\varphi' = \epsilon$ o $\varphi' = p$ luego φ no tiene prefijos propios de modo que se cumple la propiedad. Si $\varphi = \perp$ o $\varphi = \top$ de nuevo $\varphi' = \epsilon$ o $\varphi' = \perp$ o $\varphi' = \top$ que no son prefijos propios, luego φ tampoco tiene prefijos.

$$P(\varphi), \forall \varphi \in \text{AT}$$

($\neg\varphi$) Supongamos que φ' es prefijo propio de $(\neg\varphi)$, y supongamos que todo prefijo propio de φ cumple la propiedad.

(CASO 1) Si $\varphi' = ($, entonces $|\varphi'|_(= 1 > |\varphi'|_) = 0$

(CASO 2) Si $\varphi' = (\neg$, entonces $|\varphi'|_(= 1 > |\varphi'|_) = 0$

(CASO 3) Si $\varphi' = (\neg\varphi''$, siendo φ'' prefijo de φ luego cumple la propiedad, si además le sumamos uno la cumple también.

(CASO 4) Si $\varphi' = (\neg\varphi$, entonces $|\varphi'|_(= 1 = |\varphi'|_) = 0$

($\square$) Supongamos que todo prefijo propio de φ_1, φ_2 cumple la propiedad. Hay que ver entonces, que los prefijos lo cumplen

$$(,\ (\varphi_1,\ \varphi_1,\ \varphi_1 \square,\ (\varphi_1 \square \varphi_2',\ (\varphi_1 \square \varphi_2$$

Se deja como ejercicio.

1.2. Definiciones recursivas

Supongamos que queremos definir una función

$$H : \text{PROP}_{SP} \to A$$

donde A puede tomar diferentes tipos de conjunto: $\mathbb{N}$, $\mathcal{P}(\text{PROP}_{SP})$,... para hacerlo de forma recursiva, vamos a necesitar las siguientes funciones auxiliares

- Caso atómico

  $$H_{\text{AT}} : \text{AT} \to A$$

  dada por $H(\varphi) = H_{\text{AT}}(\varphi), \forall \varphi \in \text{AT}$

- Paso recursivo: donde diferenciamos entre la negación y las conectivas binarias.

  (i) Negación
$$H_{\neg} : A \to A$$
  dada por
$$H|(\neg\varphi)| = H_{\neg}|(H(\varphi)|$$

  (ii) Conectivas binarias
$$H_{\square} : A \times A \to A$$
  dada por
$$H|(\varphi_1 \,\square\, \varphi_2)| = H_{\square}|(H(\varphi_1,\, \varphi_2)|$$

En los casos (i) e (ii) ni la entrada ni la salida de la función está formada por formulas.

Ejemplo 6: Si queremos definir de forma recursiva el número de paréntesis abiertos de una proposición
$$H_{(} : \mathrm{PROP}_{SP} \to \mathbb{N}$$
se tendría
$$H_{\mathrm{AT}\,(} : \mathrm{AT} \to \mathbb{N}$$
dada por
$$H_{\mathrm{AT}\,(}(\varphi) = 0, \forall\varphi \in \mathrm{AT}$$
la negación
$$\begin{array}{rccc} H_{\neg(} : & \mathbb{N} & \to & \mathbb{N} \\ & n & \longmapsto & n+1 \end{array}$$
finalmente, el resto de conectivas
$$\begin{array}{rccc} H_{\square(} : & \mathbb{N}\times\mathbb{N} & \to & \mathbb{N} \\ & (n,\, m) & \longmapsto & n+m+1 \end{array}$$

Escribirlo así es un poco farragoso por lo que, generalmente, escribiremos las funciones haciendo uso del propio concepto de recursión
$$H_{(}(\varphi) = 0, \forall\varphi \in \mathrm{AT}$$
$$H_{\neg(}((\neg\varphi)) = H_{(}(\varphi) + 1$$
$$H_{\square(}(\varphi_1 \,\square\, \varphi_2) = H_{(}(\varphi_1) + H_{(}(\varphi_2) + 1$$
además, en la escritura, se suprime la función auxiliar. **Ejemplo 7**: definiremos de forma recursiva el número de apariciones de $\wedge$ en φ y lo denotamos como $|\varphi|^{\wedge}$.

(AT) $|\varphi|^{\wedge} = 0, \forall\varphi \in \mathrm{AT}$

($\neg$) $|(\neg\varphi)|^{\wedge} = |\varphi|^{\wedge}$

($\square$) Hay que diferenciar dos casos

$$|(\varphi_1 \square \varphi_2)|^{\wedge} = \begin{cases} |\varphi_1|^{\wedge} + |\varphi_2|^{\wedge} & \text{si} \quad \square \in \{\vee, \rightarrow, \leftrightarrow\} \\ \varphi_1|^{\wedge} + |\varphi_2|^{\wedge} + 1 & \text{si} \qquad \square = \wedge \end{cases}$$

Definición 8. Dadas dos fórmulas φ y ψ decimos que ψ es una **subformula** de φ si una parte de φ formada por símbolos consecutivos es idéntica a ψ.

Ejemplo 8: dada la formula,

$$\varphi \equiv (p \vee (q \rightarrow (\neg r)))$$

observamos que esta formada por las subformulas siguientes

$$p,\ q,\ r\ (\neg r),\ (q \rightarrow (\neg r)), \varphi$$

Ejemplo 9: Creamos una función recursiva que devuelva las diferentes subformulas de una formula dada.

$$\text{SUB} : \text{PROP}_{SP} \rightarrow \mathcal{P}(\text{PROP}_{SP})$$

dada por

(AT) $\text{SUB}(\varphi) = \{\varphi\}, \forall \varphi \in \text{AT}$

($\neg$) $\text{SUB}((\neg\varphi)) = \{(\neg\varphi)\} \cup \text{SUB}(\varphi)$

($\square$) $\text{SUB}((\varphi_1 \square \varphi_2)) = \text{SUB}(\varphi_1) + \text{SUB}(\varphi_2) \cup \{(\varphi_1 \square \varphi_2)\}$

Proposición 1. *El esquema de definición recursiva da como resultado una única función. Esto es, dadas*

$$H_{AT} : AT \rightarrow A$$

$$H_{\neg} : A \rightarrow A$$

$$H_{\square} : A \times A \rightarrow A$$

existe una única función $H : PROP_{SP} \rightarrow A$ que verifica

$$H(\varphi) = H_{AT}(\varphi), \forall \varphi \in AT$$

$$H((\neg\varphi)) = H_{\neg}(H(\varphi))$$

$$H((\varphi_1 \square \varphi_2)) = H_{\square}(H(\varphi_1,\ \varphi_2))$$

1.3. Eliminación de paréntesis

Como en la aritmética básica, cuando escribimos una operación podemos utilizar las reglas de prioridad, asociatividad, etc. para escribir el menor número de paréntesis posible. Así por ejemplo en

$$(((2 \cdot 3) + 5) - (3 \cdot 2)) = 2 \cdot 3 + 5 - 3 \cdot 3$$

la idea es, por tanto, dar una serie de reglas para poder hacer lo mismo con nuestras formulas de proposición.

1.3.1. Reglas de eliminación de paréntesis

1. **Elminación de paréntesis externos**. No aportan información.
2. **Prioridad entre conectivas**. En la siguiente lista, las conectivas, aparecen de más prioridad a menos
$$(+) \quad \neg,\, \wedge,\, \vee,\, \to,\, \leftrightarrow \quad (-)$$
3. **Asociatividad**. Adoptamos el convenio de asociar por la izquierda. De este modo si tenemos $p \to q \to r$ daremos como asociación válida
$$(p \to q) \to r$$
siendo, por tanto, errónea
$$p \to (q \to r)$$
Esto mismo se aplica para $\leftrightarrow$. En los casos de $\vee$ y $\wedge$ no se presenta problema pues son asociativas.

1.4. Valoraciones: Tablas de verdad

Todo lenguaje se subdivide en

1. Sintaxis: reglas de formación de las frases o fórmulas.
2. Semántica: significado de las frases o fórmulas.

de este modo, se conforma un lenguaje formal.

Definición 9. Denotaremos por BOOL al conjunto formado por dos elementos[1]
$$\mathrm{BOOL} = \{\mathrm{Verdadero},\ \mathrm{Falso}\} = \{\mathrm{V},\ \mathrm{F}\} = \{\mathrm{T},\ \mathrm{F}\} = \{1,\, 0\}$$
con los que vamos a *valorar* las fórmulas.

Para dar un sentido más formal a la idea de *valorar*, vamos a construir una serie de aplicaciones sobre el conjunto BOOL en la siguiente definición.

Definición 10. Diremos que una **valoración** es una aplicación de la forma
$$v: \ \mathrm{SP} \ \to \ \mathrm{BOOL}$$
cuya extensión da lugar a
$$\begin{array}{rccc} \widehat{v}: & \mathrm{PROP}_{SP} & \to & \mathrm{BOOL} \\ & \varphi & \mapsto & \widehat{v}(\varphi) \end{array}$$
que definiremos de forma recursiva

(AT) $\widehat{v}(\top) = V$, $\widehat{v}(\bot) = F$, $\widehat{v}(p) = v(p)$, si $p \in \mathrm{SP}$

($\neg$) $\widehat{v}(\neg) = v_{\neg}(\widehat{v}(\varphi))$

($\square$) $\widehat{v}((\varphi \,\square\, \psi)) = v_{\square}(\widehat{v}(\varphi),\, \widehat{v}(\psi))$ con $\square \in \{\wedge,\, \vee\ \to,\, \leftrightarrow\}$

Estas funciones de valoración dan lugar a las tablas de verdad de cada una de las conectivas lógicas.

[1] Durante este curso elegimos, generalmente, como notación para el conjunto BOOL = {V, F}.

$p \wedge q$

		p	
	$v_\wedge$	**V**	**F**
q	**V**	V	F
	F	F	F

$p \vee q$

		p	
	$v_\vee$	**V**	**F**
q	**V**	V	V
	F	V	F

$p \to q$

		p	
	$v_\to$	**V**	**F**
q	**V**	V	F
	F	V	V

$p \leftrightarrow q$

		p	
	$v_\leftrightarrow$	**V**	**F**
q	**V**	V	F
	F	V	V

$\neg p$

	$v_\neg$	
p	**V**	F
	F	V

Las aplicaciones son

$$v_\square : \ \text{BOOL} \times \text{BOOL} \ \to \ \text{BOOL}$$

si $\square \in \{\wedge,\ \vee\ \to,\ \leftrightarrow\}$ En el caso de la negación

$$v_\neg : \ \text{BOOL} \ \to \ \text{BOOL}$$

Donde encontremos matrices simétricas, podemos decir que ese operador es conmutativo. Los operadores que conmutan son: la disyunción y la conjunción.

Ejemplo 10: utilizando las aplicaciones de la definición (10), vamos a valor la fórmula

$$\varphi = (q \vee r) \to (q \to r)$$

partiendo de las valoraciones

$$v(p) = V, \quad v(q) = F, \quad v(r) = V$$

$$\widehat{v} = v_\to(\widehat{v}(q \vee r), \widehat{v}(p \to q)) = v_\to(v_\vee(\widehat{v}(q), \widehat{v}(r)), v_\to(\widehat{v}(p), \widehat{v}(q)) =$$
$$= v_\to(v_\vee(F, V), v_\to(V, F)) = v_\to(V, F) = F$$

Es claro que no es un método muy óptimo si queremos ver todas las posibles valoraciones de las proposiciones que formen parte de la fórmula. Para ello se utiliza la tabla de verdad.

Ejemplo 11: Completar la tabla de verdad de la fórmula φ del ejemplo anterior. (*Se deja como ejercicio)*.

Definición 11. Dada $v : \text{SP} \to \text{BOOL}$ valoración y $\varphi \in \text{PROP}_{SP}$, v **satisface** φ si y sólo si $\widehat{v}(\varphi) = V$; y denotamos $v \models \varphi$. En caso contrario, v no satisface φ, si $\widehat{v}(\varphi) = F$, denotamos $v \not\models \varphi$. Al símbolo $\models$ se le denomina **símbolo de satisficidad**.

1.4.1. Clasificación de fórmulas

Definición 12. Según como sean las valoraciones de una fórmula, podemos clarificarlas en

1. **Satisfactible**. Existe alguna valoración v tal que $v \models \varphi$.
2. **Tautología**. Siempre es cierto, es decir, $\forall v$ se tiene que $v \models \varphi$.
3. **Contingencia**. φ se dice contingencia si es satisfactible, pero no tautología.
4. **Contradicción**. Siempre es falso, es decir, $\forall v$ se tiene que $v \not\models \varphi$.

Observación. Todas las valoraciones posibles que hay en una fórmula viene dado por 2^ρ donde ρ es el número de proposiciones que tiene la fórmula.

Definición 13. Sea $\Phi \subseteq \mathrm{PROP}_{SP}$ un conjuntos de fórmulas, definimos

$$\mathrm{Mod}(\Phi) = \{v / v \models \varphi, \forall \varphi \in \Phi\}$$

entonces diremos que

1. Φ es **satisfactible**, si $\mathrm{Mod}(\Phi) \neq \emptyset$ y denotamos $\mathrm{Sat}(\Phi)$
2. Φ es **insatisfactible**, si $\mathrm{Mod}(\Phi) = \emptyset$ y denotamos $\mathrm{Insat}(\Phi)$
3. $\varphi \in \mathrm{PROP}_{SP}$ es **consecuencia lógica** de Φ si y sólo si $\mathrm{Mod}(\Phi) \subseteq \mathrm{Mod}(\{\varphi\})$, denotamos $\Phi \models \varphi$.

Observación: si partimos de una premisa falsa podemos concluir cualquier cosa. Esto es

$$\mathrm{Insat}(\Phi) \text{ entonce } \Phi \models \varphi$$

además, podemos expresar una tautología como

$$\text{Si } \Phi \neq \emptyset \text{ y } \Phi \models \varphi \text{ entonce } \models \varphi$$

En este sentido se pueden definir todo tipo de propiedades, como las que vienen a continuación.

Proposición 2. *Se tiene que*

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

Demostración. Vamos a demostrar el apartado (1) (*el (2) queda como ejercicio.* Comenzamos con la implicación hacía la derecha.

Supongamos que $\Phi \cup \{\varphi\} \models \psi$ y hay que ver si $\mathrm{Mod}(\Phi) \subseteq \mathrm{Mod}(\varphi \to \psi)$. Si $v \in \mathrm{Mod}(\Phi)$ entonces

(i) $\widehat{v}(\varphi) = V \Rightarrow v \in \mathrm{Mod}(\varphi) \Rightarrow v \in \mathrm{Mod}(\Phi \cup \{\varphi\} \Rightarrow v \in \mathrm{Mod}(\psi) \Rightarrow \widehat{v}(\psi) = V$ de modo que $\widehat{v}(\varphi \to \psi) = V$ en consecuencia $v \in \mathrm{Mod}(\varphi \to \psi)$.

(ii) $\widehat{v}(\varphi) = F$, entonces $\widehat{v}(\varphi \to \psi) = V \Rightarrow v \in \mathrm{Mod}(\varphi \to \psi)$

Para la implicación en el otro sentido. Si $v \in \mathrm{Mod}(\Phi \cup \{\varphi\})$ entonces

$$\left.\begin{array}{lclcl} v \in \mathrm{Mod}(\Phi) & \Rightarrow & v \in \mathrm{Mod}(\varphi \to \psi) & \Rightarrow & \widehat{v}(\varphi \to \psi) = V \\ v \in \mathrm{Mod}(\varphi) & \Rightarrow & \widehat{v}(\varphi) = V & & \end{array}\right\} \Rightarrow \widehat{v}(\psi) = V \Rightarrow v \in \mathrm{Mod}(\psi)$$

□

1.5. Equivalencias lógicas

Acabamos de ver en la *proposición 2* que podemos escribir propiedades acerca de un conjunto de proposiciones. Pero, de momento, los métodos que disponemos para demostrar este tipo de propiedades no siempre son los más óptimos. Veamoslo en con el siguiente ejemplo.

Ejemplo 12: a través de una tabla de verdad, partiendo de

$$\Phi = \{\neg p \wedge q \to r,\ \neg p,\ \neg r\}$$

$$\varphi = \neg q$$

hay que demostrar que

$$\Phi \models \varphi$$

Para la tabla coloreamos en rojo las premisas que deben satisfacer la consecuencia, pintada de azul. Satisfacer implica que la valoración debe ser V en todas las premisas. Esta situación sólo se da para las valoraciones p, q y r todas ellas F (fila señalada con una flecha).

	p	**q**	**r**	$\neg p$	$\neg p \wedge q$	$\neg p \wedge q \to r$	$\neg r$	$\neg q$
	V	**V**	**V**	F	F	V	F	F
	V	**V**	**F**	F	F	V	V	F
	V	**F**	**V**	F	F	V	F	V
	V	**F**	**F**	F	F	V	V	V
	F	**V**	**V**	V	V	V	F	F
	F	**V**	**F**	V	V	F	V	F
	F	**F**	**V**	V	F	V	F	V
$\longrightarrow$	**F**	**F**	**F**	V	F	V	V	V

puesto que la consecuencia es también verdadera, hemos demostrado que $\Phi \models \varphi$.

Esta forma no es efectiva y vamos a dar mecanismos para no tener que hacer uso de valoraciones y tablas de verdad.

Definición 14. Sean $\varphi,\ \psi \in \mathrm{PROP}_{SP}$ son **lógicamente equivalentes** y lo denotamos como $\varphi \sim \psi$, si y sólo si $\forall v, \widehat{v}(\varphi) = \widehat{v}(\psi)$.

Lema 1. *Diremos que $\sim$ es una* ***relación de equivalencia****; más aún, es una congruencia respecto a dos operadores lógicos (i.e. la relación se respeta con los operadores lógicos).*

$$\left.\begin{array}{cc} \varphi \sim \psi & \varphi_1 \sim \varphi_1' \\ \neg\varphi \sim \neg\psi & \varphi_2' \sim \varphi_2 \end{array}\right\} \Rightarrow \quad \varphi_1 \,\square\, \varphi_2 \sim \varphi_1' \,\square\, \varphi_2'$$

con $\square \in \{\wedge,\ \vee\ \to,\ \leftrightarrow\}$.

1.5.1. Leyes de equivalencia lógica

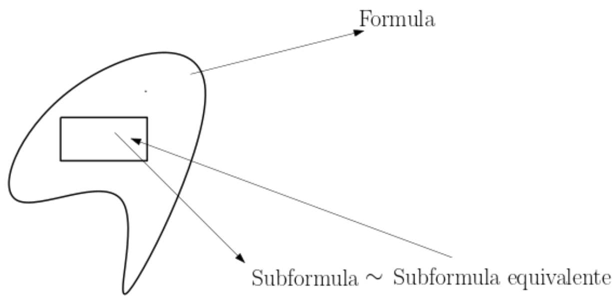
1. **Conmutatividad**
$$p \wedge q \sim q \wedge p$$
$$p \vee q \sim q \vee p$$
2. **Asociatividad**
$$p \wedge (q \wedge r) \sim (p \wedge q) \wedge r$$
$$p \vee (q \vee r) \sim (p \vee q) \vee r$$
3. **Idempotencia**
$$p \wedge p \sim p$$
$$p \vee p \sim p$$
4. **Distributiva**
$$p \wedge (q \vee r) \sim (p \wedge q) \vee (p \wedge r)$$
$$p \vee (q \wedge r) \sim (p \vee q) \wedge (p \vee r)$$
5. **Absorción**
$$p \wedge (q \vee p) \sim p$$
$$p \vee (q \wedge p) \sim p$$
6. **Cero y uno**
$$p \wedge \top \sim p$$
$$p \vee \top \sim \top$$
$$p \wedge \bot \sim \bot$$
$$p \vee \bot \sim p$$
7. **Contradicción**
$$p \wedge (\neg p) \sim \bot$$
8. **Tercio excluido**
$$p \vee (\neg p) \sim \top$$
9. **Doble negación**
$$\neg(\neg p) \sim p$$
10. **Reducción**
$$p \rightarrow q \sim \neg p \vee q$$
$$p \leftrightarrow q \sim (p \rightarrow q) \wedge (q \rightarrow p)$$
11. **De Morgan**
$$\neg(p \wedge q) \sim \neg p \vee \neg q$$
$$\neg(p \vee q) \sim \neg p \wedge \neg q$$

Se deja como ejercicio simplificar las expresiones siguientes mediante las leyes que acabamos de ver

1. $\{\varphi_1, \varphi_2\} \models \psi$
2. $\text{Insat}(\Phi \cup \{\psi\})$
3. $\varphi_1 \wedge \varphi_2 \wedge \neg\psi \sim \bot$

1.6. Sustitución

La idea es tomar una parte de una formula y sustituirla por algo equivalente que sea más manejable. Definamos de forma rigurosa el concepto.



Definición 15. Sean φ, $\psi \in \text{PROP}_{SP}$ y $p \in \text{SP}$, se denota

$$\psi[\varphi/p]$$

a sustituir en la fórmula ψ las apariciones de p por φ. Para definir la sustitución utilizamos el método de recursión. De este modo, obtenemos

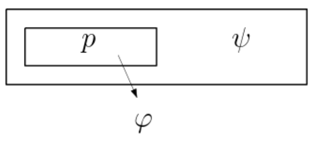
(AT) Para las formulas atómicas se tiene que

- $p[\varphi/p] = \varphi$.
- $q[\varphi/p] = q \quad q \neq p$.
- $\top[\varphi/p] = \top$.
- $\bot[\varphi/p] = \bot$.

($\neg$) $(\neg\psi)[\varphi/p] = \neg(\psi[\varphi/p])$.

($\Box$) $(\psi_1 \Box \psi_2)[\varphi/p] = (\psi_1[\varphi/p] \Box \psi_2[\varphi/p])$.

De manera esquemática



Teorema 1. *Sean $\varphi, \varphi', \psi \in PROP_{SP}$, $p \in SP$. Si $\varphi \sim \varphi'$, entonces $\psi[\varphi/p] \sim \psi[\varphi'/p]$.*

Notación. Dada

$$f : A \to B$$

se tiene que

$$f[b/a](x) = \begin{cases} f(x) & \text{si} \quad x \neq a \\ b & \text{si} \quad x = a \end{cases}$$

Lema 2. *Sean $\varphi, \psi \in PROP_{SP}$, $p \in SP$ y v valoración. Entonces:*

$$\widehat{v}(\psi[\varphi/p]) = v[\widehat{\widehat{v}(\varphi)/p}](\psi)$$

Demostración. Procedemos por inducción estructural sobre ψ. Comenzamos por las atómicas

(AT)
- Si $\psi = p$ entonces, $\psi[\varphi/p] = \varphi$. Luego
$$\widehat{v}(\psi[\varphi/p]) = \widehat{v}(\varphi)$$
- Si $\psi = q$ con $q \neq p$ entonces $\psi[\varphi/p] = q$, con lo que $\widehat{v}(\psi[\varphi/p]) = \widehat{v}(q)$, luego
$$v[\widehat{\widehat{v}(\varphi)/p}](q) = \widehat{v}(q)$$
- Si $\psi = \top$, $\psi[\varphi/p] = \top$, luego
$$v[\widehat{\widehat{v}(\varphi)/p}](\top) = v(\top) = V$$
- Si $\psi = \bot$, $\psi[\varphi/p] = \bot$, luego
$$v[\widehat{\widehat{v}(\varphi)/p}](\bot) = v(\bot) = F$$

($\neg$) Si $\psi = \neg\psi_1$, entonces $\widehat{v}((\neg\psi_1)[\varphi/p]) = \widehat{v}(\neg(\psi_1[\varphi/p])) = v_\neg(\widehat{v}(\psi_1[\varphi/p]))$ por hipótesis de inducción tenemos que

$$v_\neg(v[\widehat{\widehat{v}(\varphi)/p}])(\psi_1) = v[\widehat{\widehat{v}(\varphi)/p}](\neg\psi_1)$$

($\Box$) Si $\psi = \psi_1 \Box \psi_2$, entonces

$$\widehat{v}(\psi[\varphi/p]) = \widehat{v}((\psi_1[\varphi/p] \Box \psi_2[\varphi/p])) = v_\Box(\widehat{v}(\psi_1[\varphi/p]),\ \widehat{v}(\psi_2[\varphi/p]))$$

por hipótesis de inducción

$$v_\Box(v[\widehat{\widehat{v}(\varphi)/p}](\psi_1),\ v[\widehat{\widehat{v}(\varphi)/p}](\psi_2)) = v[\widehat{\widehat{v}(\varphi)/p}](\psi_1 \Box \psi_2)$$

$\square$

Demostración. (*Teorema 1*).

$$\widehat{v}(\psi[\varphi/p]) = v[\widehat{\widehat{v}(\varphi)/p}](\psi) =_1 v[\widehat{\widehat{v}(\varphi')/p}](\psi) = \widehat{v}(\psi[\varphi'/p])$$

en (1) utilizamos que $\widehat{v}(\varphi) = \widehat{v}(\varphi')$. $\square$

Ejemplo 13: Sean $\varphi = p \to q$ y $\psi = r \to t$ y tomamos la sustitución $\psi[\varphi/r] = (p \to q) \to t$. Entonces, es indiferente valorar $\psi[\varphi/r]$ en

p	**q**	**r**	**t**
V	F	V	V

que valorarlo, siendo $\widehat{v}(\varphi) = F$, entonces $v[\widehat{v}(\varphi)/r]$

p	**q**	**r**	**t**
V	F	F	V

Lema 3. *Si* $\psi_1 \sim \psi_2$ *entonces* $\psi_1[\varphi/p] \sim \psi_2[\varphi/p]$.

Demostración. En efecto

$$\widehat{v}(\psi_1[\varphi/p]) = \widehat{v[\widehat{v}(\varphi)/p]}(\psi_1) =$$

como $\psi_1 \sim \psi_2$

$$\widehat{v[\widehat{v}(\varphi)/p]}(\psi_2) = \widehat{v}(\psi_2[\varphi/p])$$

□

Ejemplo 14: Si queremos ver $\varphi \wedge \psi \sim \psi \wedge \varphi$ basta tomar $\psi_1 = p \wedge \psi$ y $\psi_2 = \psi \wedge p$ y aplicar el lema anterior.

Observación. Esto implica que, gracias al lema de sustitución, las leyes de equivalencia vistas en la sección 1.5.1, es también aplicable a fórmulas completas.

1.7. Completitud funcional

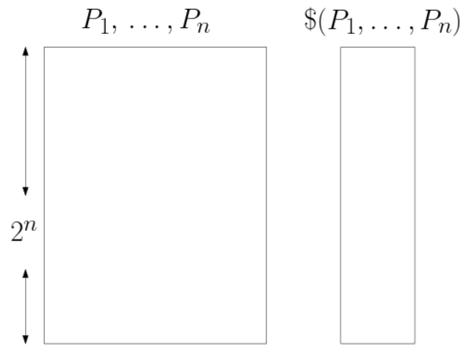
Debido a las leyes de reducción los operadores $\to$, $\leftrightarrow$ se dicen secundarios. También podemos definir nuevos operadores como

Disyunción exclusiva	$\veebar$	$p \veebar q \sim (p \vee \neg q) \vee (\neg p \vee q)$
NAND	$\downarrow$	$p \downarrow q \sim \neg(p \vee q)$
NOR	$\uparrow$	$p \uparrow q \sim \neg(p \wedge q)$

o a través de una tabla de verdad

p	q	$p \veebar q$	$p \downarrow q$	$p \uparrow q$
V	V	F	F	F
V	F	V	V	F
F	V	V	V	F
F	F	F	V	V

Podríamos construir un operador lógico con tantos argumentos como deseáramos. Pero, vamos a ver que todos ellos se pueden expresar de forma equivalente como conjunción, negación y disyunción.



Definición 16. Un conjunto de conectivas $\mathcal{C}$ es **funcionalmente completo** si cualquier otra conectiva \$ de aridad n se puede expresar con las conectivas de $\mathcal{C}$.

Teorema 2. *Sea* $\mathcal{C} = \{\vee, \wedge, \neg\}$ *es funcionalmente completo.*

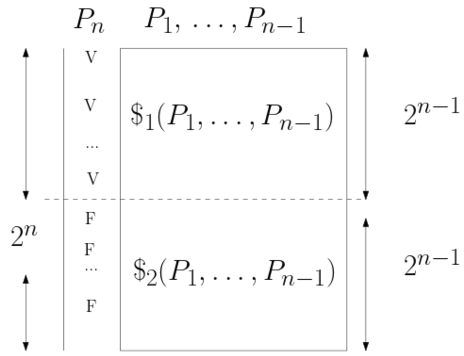
Demostración. Sea \$ conectiva de aridad n y aplicamos inducción sobre n.

Si $n = 1$. Hay cuatro posibles conectivas de aridad 1, veamos sus tablas de verdad

p	$\$_1(p)$	$\$_2(p)$	$\$_3(p)$	$\$_4(p)$
V	V	V	F	F
F	V	F	V	F

Si nos fijamos, $\$_1(p) \sim \top$, $\$_4(p) \sim \bot$, $\$_2(p) \sim p$ y $\$_3(p) \sim \neg p$. Se cumple, por tanto, el caso base.

Si $n > 1$. Se da la situación $\$(P_1, \ldots, P_n) = (P_n \wedge \$_1(P_1, \ldots P_{n-1})) \vee (\neg P_n \wedge \$_2(P_1, \ldots P_{n-1}))$



por hipótesis de inducción existen φ_i, $i = 1, 2$ construidas con las conectivas de $\mathcal{C}$ tales que $\varphi_i \sim \$_i$ con $i = 1, 2$, entonces $\$(P_1, \ldots, P_n) \sim (P_n \wedge \varphi_1) \vee (\neg P_n \wedge \varphi_2)$ □

Se puede demostrar aplicando las leyes de De Morgan que el conjunto $\{\wedge, \neg\}$ también es funcionalmente completo. Otros que también lo son y se deja la comprobación como ejercicio son

$$(1)\ \{\vee, \neg\} \quad (2)\ \{\rightarrow, \bot\}, \quad (3)\ \{\uparrow\} \quad (4)\ \{\downarrow\}$$

2. Lógica de primer orden

2.1. Primeras definiciones

La idea es extender las posibilidades de la lógica de primer orden. Si pensamos en predicados como

1. *Todos los primos mayores que 2 son impares*
2. *El 3 es un primo mayor que 2*
3. *El 3 es impar*

es claro que no podemos expresarlo con las herramientas de la lógica proposicional.

La lógica de primer orden nos habla de *elementos* de un *conjunto* dado. Empezamos introduciendo algunas herramientas básicas

- **Constantes**: en nuestros predicados anteriores serían el 2 y el 3.
- **Variables**: aquellos elementos del conjunto a los que aplicamos las propiedades. Denotamos por x, y, $z\dots$
- **Símbolos de función**: para el conjunto de los números naturales, por ejemplo, hablaríamos de la suma $+$, del producto $*$ o de la función sucesor de un número $s(x)$. Gracias a ellas podremos expresar cosas como: $s(+(2,3))$ o en notación habitual $s((2+3))$.

estos objetos nos van a permitir construir los **términos**.

Los símbolos que teníamos hasta ahora son insuficientes, por ello vamos a introducir algunos más

- Símbolo de igualdad: $\doteq$
- Símbolos de predicado: que establecen una propiedad. Por ejemplo, ser impar que podemos denotar como $\mathrm{im}(x)$ o que un elemento sea menor que otro $<(x,y)$. El primer de ellos decimos que tiene aridad 1 (un argumento de entrada) y el segundo tienen aridad 2 (pues necesita dos entradas).
- Cuantificadores: $\forall$ y $\exists$
- Conectivas lógicas: $\{\wedge,\vee\ \rightarrow,\leftrightarrow\}$ y $\top$, $\bot$

Ejemplo 15: A partir de las frases que hemos dado al inicio de la sección, vamos a formalizarlas con los nuevos elementos que hemos introducido.

1. $\forall x\,(\mathrm{pr}(x)\wedge 2<x\rightarrow \mathrm{im}(x))$
2. $\mathrm{pr}(3)\wedge 2<3$
3. $\mathrm{im}(3)$

Definición 17. Una **signatura** S

$$S = \langle \mathrm{Cts}_S,\, \mathrm{Fn}_S,\, \mathrm{Pd}_S \rangle$$

donde

- $\mathrm{Cts}_S \equiv$ conjunto de constantes de S.
- $\mathrm{Fn}_S \equiv$ conjunto de símbolos de función con aridad.
- $\mathrm{Pd}_S \equiv$ conjunto de símbolos de predicado con aridad.

Ejemplo 16: Tomando el conjunto de los números naturales, obtenemos la signatura

$$\mathrm{Nat} = \langle \mathrm{Cts}_{\mathrm{Nat}}, \mathrm{Fn}_{\mathrm{Nat}}, \mathrm{Pd}_{\mathrm{Nat}} \rangle$$

donde

- $\mathrm{Cts}_{\mathrm{Nat}} \equiv \{0, 1, \ldots\}$
- $\mathrm{Fn}_{\mathrm{Nat}} \equiv \{+|_2, s|_1\}$
- $\mathrm{Pd}_{\mathrm{Nat}} \equiv \{\mathrm{im}|_2, < |_1\}$

El número escrito tras la barra indica su aridad.

Consideramos el conjunto de **variables**

$$\mathrm{Var} = \{x, y, z, \ldots\}$$

siempre será numerable.

Definición 18. Dada una signatura S, el **alfabeto**

$$A_S = \mathrm{Cts}_S \cup \mathrm{Fn}_S \cup \mathrm{Pd}_S \cup \mathrm{Var} \cup \{\wedge,\ \vee\ \to,\ \leftrightarrow\} \cup \{\top,\ \bot,\ \forall,\ \exists,\ \doteq,\ (,\ )\}$$

Definición 19. Dada una signatura S, el conjunto de **términos** de S, TERM_S, es el menor subconjunto X de A_S^* que cumple

- Caso base

  (T1) Si $c \in \mathrm{Cts}_s$ entonces $c \in X$

  (T2) Si $x \in \mathrm{Var}$ entonces $x \in X$

- Paso recursivo

  (T3) $f|_n \in \mathrm{Fn}_s$ y $t_1, \ldots, t_n \in X$ entonces $f \in X$

Ejemplo 17: En el ejemplo anterior definimos Nat, algunos términos suyos son

$$0, 1, 2, ... \in \mathrm{TERM}_{Nat}$$

$$x, y, z, ... \in \mathrm{TERM}_{Nat}$$

$$s(x), s(1), +(s(x), s(0)), ... \in \mathrm{TERM}_{Nat}$$

Definición 20. Dada una signatura S, el conjunto de fórmulas de primer orden, FORM_s, es el menor subconjunto $X \subseteq A_S^*$ que verifica

- Fórmulas atómicas, que denotamos como FORMAT_S

  (F1) t_1, $t_2 \in \mathrm{TERM}_s$ entonces $t_1 \doteq t_2 \in X$

  (F2) $p \in \mathrm{Pd}_s|_n$ y $t_1, \ldots, t_n \in \mathrm{TERM}_s$ entonces $p(t_1, \ldots, t_n) \in X$

  (F3) $\top, \bot \in X$

(F4) $\varphi_1, \varphi_2 \in X$ entonces $(\varphi_1 \square \varphi_2) \in X$ también $\neg\varphi_i \in X$ con $i = 1, 2$.

(F5) $\varphi \in X$, $x \in \mathrm{Var}$ entonces[2] $\forall x : \varphi$ y $\exists x : \varphi$.

Ejemplo 18: Si escribimos

$$\exists x : \varphi \to \psi$$

el existe afecta a todo, en cambio, si escribimos

$$(\exists x : \varphi) \to \psi$$

solo afecta a φ.

Ejemplo 19: dada la fórmula

$$\exists x\, p(x) \to \forall x\, p(x)$$

si $[\forall x\, p(x)] = V$ entonces la premisa es cierta para toda valoración. Ahora, si $[\forall x\, p(x)] = F$ entonces (teniendo en cuenta que siempre consideramos conjuntos de elementos no vacíos) existe un elemento c tal que $[\neg p(c)] = V$ entonces $p(c) \to \forall x\, p(x)$, por tanto llegamos a la conclusión de que

$$\exists x\, p(x) \to \forall x\, p(x)$$

es una tautología.

Dado un conjunto A y en él, el producto cartesiano A^n, veamos algunos casos partículares de este tipo de conjuntos

- El conjunto $A^0 = \{\emptyset\}$, la justificación más sencilla viene de considerar

$$A^0 \times A$$

  que debe ser igual a A. Para que esto ocurra, el cardinal debe mantenerse tras realizar la operación, luego, no queda otra opción más que el conjunto vacío.

[2]El $\forall$ significa que todo elementos de X cumple φ. Por otro lado, el $\exists$ significa que un elemento de X cumple φ

- Las aplicaciones que se crean a partir de este conjunto pueden inducir a error

$$f : A^0 \to A$$

puede interpretarse como

$$f_1 : \{\emptyset\} \to A, \quad f_2 : \emptyset \to A$$

la diferencia radica en que para f_2 sólo hay una función que será $f = \emptyset$, en cambio, para f_1 hay tantas funciones como elementos tiene A, basta con asociar $\emptyset \mapsto a$ con $a \in A$.

También a tener en cuenta, $f|_0 \in \mathrm{Fn}_S$ es una función de aridad 0, es decir, una constante. De esto modo podriamos quitar el conjunto de las constantes Cts_S de la signatura. Del mismo modo podemos definir, símbolos de proposición de aridad cero, $p|_0 \in \mathrm{Pd}_S$ que nos darán sentido a expresiones del tipo $p \to q(x)$.

Definición 21. Sea $S = \langle \mathrm{Cts}_S, \mathrm{Fn}_S, \mathrm{Pd}_S \rangle$, se definen los conjuntos

$$T_S^0 = \{c / \, c \in \mathrm{Cts}_S\} \cup \mathrm{Var}_S$$

$$T_S^{n+1} = T_n^S \cup \bigcup_{k \in \mathbb{N}} \{f(t_1, \ldots, t_k) / \, f|_k \in \mathrm{Fn}_S, \, t_1, \ldots, t_n \in T_n^S\}$$

$$T_S = \bigcup_{i \in \mathbb{N}} T_S^i$$

Proposición 3. *Dada la definición (21), se verifica que*

$$T_S = TERM_S$$

Demostración. Es igual que la demostración del capítulo primero que se hace dentro de la definición 6. □

2.2. Inducción estructural y recursividad

La **recursion** la definimos como

1. Caso base

$$F_0 : \{c / \, c \in \mathrm{Cts}_S\} \cup Var \to A$$

2. Caso recursivo

$$f|_k \in \mathrm{Fn}_S, \, F_f : A^k \to A$$

Ejemplo 20: para ilustrar la recursividad en, introducimos la función var

$$\mathrm{var} : \mathrm{TERM}_S \to \mathcal{P}(\mathrm{var})$$

1. Caso base, $var_0 : \mathrm{Cts}_S \cup \mathrm{Var} \to A$
   - $\mathrm{var}_0(c) = \emptyset, \quad c \in \mathrm{Cts}_S$
   - $\mathrm{var}_0(x) = \{x\}, \quad x \in \mathrm{var}$
2. Caso recursivo,
   - $\mathrm{var}_f : \mathcal{P}(\mathrm{Var})^k \to \mathcal{P}(\mathrm{Var})$, dada por

$$(A_1, \ldots, A_k) \mapsto \bigcup_{i=1}^{k} A_k$$

en consecuencia

$$\mathrm{var}_f(f(t_1, \ldots, t_k)) = \bigcup_{i=1}^{k} \mathrm{var}(t_i)$$

El esquema de recursión para fórmulas queda de la manera

1. Fórmulas básicas

$$F_S^0 = \{\top, \bot\} \cup \bigcup_{k \in \mathbb{N}} \{p(t_1, \ldots, t_k) / \, t_1, \ldots, t_k \in \mathrm{TERM}_S, \, p_k \in \mathrm{Pd}_S\} \cup \{t \doteq s / \, t, s \in \mathrm{TERM}_S\}$$

2. Fórmulas recursivas

$$F_S^{n+1} := F_S^n \cup \{\neg\varphi / \, \varphi \in F_S^n\} \cup \{(\varphi \Box \psi) / \, \varphi, \psi \in F_S^n\} \cup \{(Qx \, \varphi) / \, x \in Var, \, \varphi \in F_S^n\}$$

Ejemplo 21: Definimos ahora var de forma recursiva en $FORM_S$

1. Caso base:
   - $\mathrm{var}(\top) = \mathrm{var}(\bot) = \emptyset$
   - $\mathrm{var}(t \doteq s) = \mathrm{var}(t) \cup \mathrm{var}(s)$
   - $\mathrm{var}(p(t_1, \ldots, t_k)) = \bigcup_{i=1}^{k} \mathrm{var}(t_i)$
2. Caso recursivo:
   - $\mathrm{var}((\neg\varphi)) = \mathrm{var}(\varphi)$
   - $\mathrm{var}((\phi \, \Box \, \psi)) = \mathrm{var}(\varphi) \cup \mathrm{var}(\psi)$
   - $\mathrm{var}((Qx \, \varphi)) = \{x\} \cup \mathrm{var}(\varphi)$

Para la **inducción estructural** queremos demostrar una propiedad P, para cada uno de los términos. Seguimos los pasos de manera análoga a como lo hacemos en lógica proposicional.

1. Caso base: hay que demostrar P para $\mathrm{Cts}_S \cup \mathrm{Var}$
2. Paso inductivo: Suponemos cierta la propiedad P para $t_1, \ldots, t_k$. P es cierta para T_S^n y siendo $f|_k \in \mathrm{Fn}_S$ lo demostramos para $f(t_1, \ldots, t_k)$, esto es, para T_s^{n+1}

Definición 22. Introducimos un nuevo concepto asociado a las variables

1. Una **variable ligada** es aquella está afectada por un cuantificador.
2. Una **variable libre** es aquella que no es ligada, es decir, que no se ve afectada por un cuantificador.

aunque las fórmulas con variables libres no son *buenas*, las vamos a necesitar para poder llegar a fórmulas que carezcan de ellas.

Debido a la importancia de conocer el número de variables libres que contiene una fórmula, vamos a definir una función que cumpla con este objetivo utilizando el esquema recursivo

1. Caso base
$$\mathrm{lib}(\varphi) = \mathrm{var}(\varphi)$$
2. Caso recursivo
   - $\mathrm{lib}(\neg\varphi) = \mathrm{lib}(\varphi)$
   - $\mathrm{lib}(\varphi \,\square\, \psi) = \mathrm{lib}(\varphi) \cup \mathrm{lib}(\psi)$ con $\square \in \{\wedge,\, \vee\, \rightarrow,\, \leftrightarrow\}$
   - $\mathrm{lib}(Qx\, \varphi) = \mathrm{lib}(\varphi) \setminus \{x\}$ con $Q \in \{\exists, \forall\}$

Definición 23. Diremos que φ es una **sentencia** si $\mathrm{lib}(\varphi) = \emptyset$ y denotamos como $\varphi \in \mathrm{Sent}_S$.

2.3. Semántica

Vamos a darle un *sentido* a las fórmulas que ya hemos definido. Lo primero que necesitamos es una signatura.

Ejemplo 22: dada la signatura $S = \langle \mathrm{Cts}_{ar},\, \mathrm{Fn}_{ar},\, \mathrm{Pd}_{ar} \rangle$ como

$$\mathrm{Cts}_{ar} = \{0\}$$

$$\mathrm{Fn}_{ar} = \{s|_1,\, +|_2,\, *|_2\}$$

$$\mathrm{Pd}_{ar} = \{< |_2\}$$

y llevamos a cabo una escritura usual, en el sentido de que $+(0, s(s(0))) \equiv 0 + 2$. Esto lo hacemos para todo operador, función o termino con variable.

También vamos a necesitar un soporte y entender como este interacciona con las distintas funciones y predicados.

Ejemplo 23: para las constantes

$$\begin{array}{ccc} \mathrm{Cts}_{ar} & \to & \mathbb{N} \\ 0 & \mapsto & 0 \end{array}$$

para las funciones

$$\begin{array}{cccc} s: & \mathbb{N} & \to & \mathbb{N} \\ & n & \mapsto & n+1 \end{array}$$

$$\begin{array}{cccc} +: & \mathbb{N} & \to & \mathbb{N} \\ & (m,\, n) & \mapsto & m+n \end{array}$$

$$\begin{array}{cccc} *: & \mathbb{N} & \to & \mathbb{N} \\ & (m,\, n) & \mapsto & m*n \end{array}$$

finalmente para símbolos de predicados

$$\begin{array}{cccc} <: & \mathbb{N} & \to & \mathrm{BOOL} \\ & (m,\, n) & \mapsto & \begin{cases} V & \text{si} \quad m<n \\ F & \text{e.c.c.} \end{cases} \end{array}$$

Vamos a dotar a estas funciones y conjuntos de una estructura formal.

Definición 24. Sea $S = \langle \mathrm{Cts}_S,\, \mathrm{Fn}_S,\, \mathrm{Pd}_S \rangle$ una signatura. Una **S-álgebra** es una tupla

$$\mathfrak{a} := \langle A, \{c^{\mathfrak{a}} / \, c \in \mathrm{Cts}_S\}, \{f^{\mathfrak{a}} / \, f \in \mathrm{Fn}_S\}, \{p^{\mathfrak{a}} / \, p \in \mathrm{Pd}_S\} \rangle$$

De modo que:

- $A \neq \emptyset$, denominado **conjunto soporte**
- Si $c \in \mathrm{Cts}_S$, entonces $c^{\mathfrak{a}} \in A$.
- Si $f|_k \in \mathrm{Fn}_S$, entonces $f^{\mathfrak{a}} : A^k \to A$.
- Si $p|_k \in \mathrm{Pd}_S$, entonces $p^{\mathfrak{a}} : A^k \to \mathrm{BOOL}$.

Como consecuencia, se tiene que

1. $\exists x\, x \doteq x$ es tautologia
2. $\forall x\, \neg x \doteq x$ es contradicción

Entonces, lo que hemos dado anteriormente en el ejemplo XXIII es una S_{ar}-álgebra. En el siguiente ejemplo, vamos a tomar como soporte un conjunto menos usual que los números naturales.

Ejemplo 24: sea $\mathfrak{a} = \langle\{\triangle, \bigcirc\}, \{0^{\mathfrak{a}}\}, \{+^{\mathfrak{a}}, *^{\mathfrak{a}}, s^{\mathfrak{a}}\}, \{<^{\mathfrak{a}}\}\rangle$ tal que

- $0^{\mathfrak{a}} = \triangle$.
- Aplicaciones para cada fórmula y símbolo de proposición

$$\begin{array}{rccc} s^{\mathfrak{a}}: & \{\triangle, \bigcirc\} & \rightarrow & \{\triangle, \bigcirc\} \\ & x & \mapsto & \triangle \end{array}$$

$$\begin{array}{rccc} +^{\mathfrak{a}}: & \{\triangle, \bigcirc\}^2 & \rightarrow & \{\triangle, \bigcirc\} \\ & (x,\, y) & \mapsto & \triangle \end{array}$$

$$\begin{array}{rccc} *^{\mathfrak{a}}: & \{\triangle, \bigcirc\}^2 & \rightarrow & \{\triangle, \bigcirc\} \\ & (x,\, y) & \mapsto & \bigcirc \end{array}$$

$$\begin{array}{rccc} <^{\mathfrak{a}}: & \{\triangle, \bigcirc\}^2 & \rightarrow & \mathrm{BOOL} \\ & (x,\, y) & \mapsto & V \end{array}$$

de forma intuitiva, dadas las dos álgebras para la misma signatura, vemos que tienen *interpretaciones diferentes* a una misma fórmula

$$*(+(s(0),\, s(0)),\, s(s(s(0))))$$

- Para $\mathbb{N}$ obtenemos
$$*(+(1,\, 1),\, 3) = *(2,\, 3) = 6$$
- Mientras que si tomamos $\{\triangle, \bigcirc\}$ entonces
$$*(+(\triangle,\, \triangle),\, \triangle) = *(\bigcirc,\, \triangle) = \bigcirc$$

Formalicemos el concepto de interpretación mediante la siguiente definición.

Definición 25. Sea S signatura. Una **S-interpretación** es una tupla $\mathfrak{Y} = \langle\mathfrak{a}, \sigma\rangle$ donde

- $\mathfrak{a}$ es una S-álgebra con soporte A.
- $\sigma : \mathrm{Var} \rightarrow A$.

Definimos recursivamente la interpretación a través de la siguiente definición

Definición 26. Sea $S = \langle \mathrm{Cts}_S,\, \mathrm{Fn}_S,\, \mathrm{Pd}_S\rangle$ una signatura, $\mathfrak{Y}$ S-interpretación y $t \in \mathrm{TERM}_S$. Sea $t^{\mathfrak{Y}}$

- Caso base:
  - $t = c^{\mathfrak{Y}} = c^{\mathfrak{a}}$ si $c \in \mathrm{Cts}_S$
  - $t = x^{\mathfrak{Y}} = \sigma(x)$ si $x \in \mathrm{Var}$
- Caso recursivo:
  - Si $f|_k \in \mathrm{Fn}_S$, $t_1, \ldots, t_k \in \mathrm{TERM}_S$
$$f(t_1, \ldots, t_k)^{\mathfrak{Y}} = f^{\mathfrak{a}}(t_1^{\mathfrak{Y}}, \ldots, t_k^{\mathfrak{Y}})$$

Del mismo modo, podemos definir recursivamente la *valoración*. Pero antes damos una aclaración en cuestión de notación, abusando del concepto de sustitución definido en la lógica proposicional se tiene que $\sigma[a/x]$, $x \in \mathrm{Var}$, $a \in A$, $\mathfrak{Y} = \langle\mathfrak{a}, \sigma\rangle$, entonces $\mathfrak{Y}[a/x] = \langle\mathfrak{a}, \sigma[a/x]\rangle$.

Definición 27. Sea $S = \langle \mathrm{Cts}_S, \mathrm{Fn}_S, \mathrm{Pd}_S \rangle$, $\mathfrak{Y} := \langle \mathfrak{a}, \sigma \rangle$ S-interpretación, con A conjunto soporte. La **interpretación de una fórmula** se define recursivamente como sigue $\varphi \in \mathrm{FORM}_S$, $\varphi^{\mathfrak{Y}}$, como:

- Caso base:
  - Si $\varphi = \top$, entonces $\varphi^{\mathfrak{Y}} = V$.
  - Si $\varphi = \bot$, entonces $\varphi^{\mathfrak{Y}} = F$.
  - Si $\varphi = t \doteq s$, con $t, s \in \mathrm{TERM}_S$, entonces

$$\begin{array}{ll} \varphi^{\mathfrak{Y}} = V & \text{si} \quad t^{\mathfrak{Y}} = s^{\mathfrak{Y}} \\ \varphi^{\mathfrak{Y}} = F & \text{e.c.c} \end{array}$$

  - Si $\varphi = p(t_1, \dots, t_k)$, con $t_1, \dots, t_k \in \mathrm{TERM}_S$, $p|_k \in \mathrm{Pd}_S$, entonces

$$\begin{array}{ll} \varphi^{\mathfrak{Y}} = V & \text{si} \quad p^{\mathfrak{a}}(t_1^{\mathfrak{Y}}, \dots, t_k^{\mathfrak{Y}}) = V \\ \varphi^{\mathfrak{Y}} = F & \text{e.c.c} \end{array}$$

- Caso recursivo:
  - Si $\varphi = \neg\varphi_1$, entonces $\varphi^{\mathfrak{Y}} = v_{\neg}(\varphi_1^{\mathfrak{Y}})$
  - Si $\varphi = \varphi_1 \,\square\, \varphi_2$, entonces $\varphi^{\mathfrak{Y}} = v_{\square}(\varphi_1^{\mathfrak{Y}}, \varphi_2^{\mathfrak{Y}})$
  - Si $\varphi = \forall x\, \varphi_1$, con $x \in \mathrm{Var}$, entonces

$$\begin{array}{ll} \varphi^{\mathfrak{Y}} = V & \text{si} \quad \text{p.t. } a \in A,\ \varphi_1^{\mathfrak{Y}[a/x]} = V \\ \varphi^{\mathfrak{Y}} = F & \text{e.c.c} \end{array}$$

  - Si $\varphi = \exists x\, \varphi_1$, con $x \in Var$, entonces

$$\begin{array}{ll} \varphi^{\mathfrak{Y}} = V & \text{si} \quad \text{existe } a \in A,\ \varphi_1^{\mathfrak{Y}[a/x]} = V \\ \varphi^{\mathfrak{Y}} = F & \text{e.c.c} \end{array}$$

2.3.1. Interpretación

Definición 28. Enumeramos las diferentes consecuencias lógicas para la lógica de primer orden. Sea $\Phi \subset \mathrm{FORM}_S$, $\varphi \in \mathrm{FORM}_S$ e $\mathfrak{Y}$ una S-interpretación.

1. $\mathfrak{Y} \models \varphi$ significa que $\varphi^{\mathfrak{Y}} = V$
2. $\mathfrak{Y} \models \Phi$ si $\mathfrak{Y} \models \varphi$ para todo $\varphi \in \Phi$
3. $\Phi \models \varphi$ si todo $\mathfrak{Y}$ tal que $\mathfrak{Y} \models \Phi$ entonces $\mathfrak{Y} \models \varphi$
4. Φ es **satisfactible** si existe $\mathfrak{Y}$ tal que $\mathfrak{Y} \models \Phi$
5. Φ es **insatisfactible** si no existe $\mathfrak{Y}$ tal que $\mathfrak{Y} \models \Phi$
6. φ es **tautología** si toda $\mathfrak{Y}$ verifica $\mathfrak{Y} \models \varphi$
7. φ es **contradicción** si no existe $\mathfrak{Y}$ tal que $\mathfrak{Y} \models \varphi$
8. φ es **contingencia** si existe $\mathfrak{Y}$ tal que $\mathfrak{Y} \models \varphi$

Ejemplo 25: Sea la signatura $S = \langle \emptyset, \emptyset, \{R|_2\}\rangle$ y las fórmulas

$$\varphi = \exists x\, \forall y\, R(x,y) \tag{1}$$

$$\psi = \forall y\, \exists x\, R(x,y) \tag{2}$$

veamos que $\varphi \models \psi$.

Sea $\mathfrak{Y}$ S-interpretación de soporte A tal que $\mathfrak{Y} \models \varphi$, entonces existe un $a \in A$ tal que

$$\mathfrak{Y}[a/x] \models \forall y\, R(x,y)$$

por tanto, para todo $b \in A$ se cumple que

$$\mathfrak{Y}[a/x][b/y] \models R(x,y)$$

Hay que ver si $\mathfrak{Y} \models \psi$, entonces para todo $c \in A$ tal que

$$\mathfrak{Y}[c/y] \models \exists x,\ R(x,y)$$

por tanto, para todo $d \in A$ se cumple que

$$\mathfrak{Y}[c/y][d/x] \models R(x,y)$$

basta considerar $d = a$ se cumple que

$$\mathfrak{Y}[c/y][a/x] \models R(x,y)$$

entonces

$$\varphi \models \psi \Leftrightarrow \varphi \to \psi$$

es tautología.

Veamos ahora que $\psi \not\models \varphi$. Basta dar un contraejemplo, para ello proponemos la interpretación $\mathfrak{a} = \langle A, \emptyset, R^{\mathfrak{a}}\rangle$, con $A = \{1,2,3\}$ y $R^{\mathfrak{a}} = \{(1,2),(2,3),(3,1)\}$[3].

Veamos que $\mathfrak{Y} \models \psi$, para todo $a \in A$, tenemos que

$$\mathfrak{Y}[a/y] \models \exists x\, R(x,y)$$

habrá que estudiar

- $\mathfrak{Y}[1/y] \models \exists x\, R(x,y)$
- $\mathfrak{Y}[2/y] \models \exists x\, R(x,y)$
- $\mathfrak{Y}[3/y] \models \exists x\, R(x,y)$

[3]La notación para $R^{\mathfrak{a}}$ esta dada en un sentido conjuntista, expresando sólo aquellos pares que hacen la valoración cierta

Demostramos el primer caso. Necesitamos un b tal que

$$\mathfrak{Y}[1/y][b/x] \models R(x,y)$$

$$(R^{\mathfrak{a}})^{\exists[1/y][b/x]} = V$$

$$R^{\mathfrak{a}}(b,1) = V \Leftrightarrow b = 3$$

por simetría

(2) $c = 1$

(3) $d = 2$

ahora tenemos que $\mathfrak{Y} \models \varphi$, donde φ es la ecuación (1). La pregunta que nos hacemos es

$$c \in A, \mathfrak{Y}[c/x] \models \forall y\, R(x,y)$$

y para todo $d \in A$

$$\mathfrak{Y}[c/x][d/y] \models R(x,y)$$

finalmente lo que buscamos es

$$R^{\mathfrak{a}}(c,d) = V$$

pero este elemento no existe, luego hemos demostrado que $\psi \not\models \varphi$.

Ejemplo 26: Dada la fórmula

$$\varphi = (\exists z\, \forall x\, \neg f(x) \doteq z) \wedge (\forall x\, \forall y\, f(x) \doteq f(y) \rightarrow x \doteq y) \tag{3}$$

que dividimos en dos como

$$\varphi = \varphi_1 \wedge \varphi_2$$

por φ_2 deducimos que f es inyectiva y por φ_1 deducimos que f no es sobreyectiva. Por tanto, φ es una contingencia. Para que esto ocurra el cardinal de A debe ser infinito.

Si ahora observamos la fórmula $\varphi_{=2} = (\exists x\, \exists y\, \neg x \doteq y) \wedge \forall z\, (z \doteq y \vee z \doteq x)$ vemos que su cardinal debe ser 2. Por tanto, del mismo modo se puede definir $\varphi_{=n}$ para todo $n < \infty$, siendo $\varphi_{=\infty}$ la fórmula (3).

2.3.2. Axiomas de Peano

Dada la signatura Nat de los ejemplos XVI y XVII podemos formalizar los axiomas de Peano de la siguiente manera.

1. $\forall x\, \neg s(x) \doteq 0$
2. $\forall x\, \forall y\, s(x) \doteq s(y) \rightarrow x \doteq y$
3. $\forall x\, x + 0 \doteq x$
4. $\forall x\, \forall y\, x + s(y) \doteq s(x + y)$

5. $\forall x\, x * 0 \doteq 0$

6. $\forall x \forall y\, x * s(y) \doteq (x * y) + x$

7. $\forall x \forall y\, (x < y \leftrightarrow \exists z\, x + s(z) \doteq y)$

Se cumple

$$\mathcal{N} = \{\mathbb{N}, \{0^{\mathcal{N}}\}, \{+^{\mathcal{N}}, *^{\mathcal{N}}\}, \{<^{\mathcal{N}}\}\}$$

entonces

$$0^{\mathcal{N}} = 0$$

$$\begin{array}{rccc} +^{\mathcal{N}}: & \mathbb{N}^2 & \to & \mathbb{N} \\ & (m,\, n) & \mapsto & m+n \end{array} \qquad \begin{array}{rccc} *^{\mathcal{N}}: & \mathbb{N}^2 & \to & \mathbb{N} \\ & (m,\, n) & \mapsto & m*n \end{array}$$

$$\begin{array}{rccc} <^{\mathcal{N}}: & \mathbb{N}^2 & \to & \text{BOOL} \\ & (m,\, n) & \mapsto & \begin{cases} V & \text{si} \quad m < n \\ F & \text{e.c.c} \end{cases} \end{array}$$

Si $\mathfrak{Y} : \langle \mathfrak{a},\, \sigma \rangle$ entonces $\mathfrak{Y} \models \Phi_{\text{Peano}}$.

Teorema 3 (de Incompletitud). *No se puede encontrar un conjunto de axiomas que identifique solo a* $\mathbb{N}$.

♣

Definición 29. Vocabulario. Conjunto de constantes, símbolos de función y símbolos de predicado que aparecen en términos y fórmulas.

Definición 30. Sea $S = \langle \text{Cts}_S,\, \text{Fn}_S,\, \text{Pd}_S \rangle$ una signatura. El vocabulario **para términos** se define recursivamente como

- Base
  - $\text{voc}(c) = \{c\}, \quad c \in \text{Cts}_S$
  - $\text{voc}(x) = \{\emptyset\}, \quad x \in \text{Var}$
- Recursivo
  - $\text{voc}(f(t_1, \ldots, t_n)) = \{f\} \cup \bigcup_{i=1}^{k} \text{voc}(t_i), \quad f|_k \in \text{Fn}_S \quad t_i \in \text{TERM}_S$

para fórmulas se define como

- Base
  - $\text{voc}(p(t_1, \ldots, t_k)) = \{p\} \cup \bigcup_{i=1}^{k} \text{voc}(t_i), \quad p \in \text{Pd}_S \quad t_i \in \text{TERM}_S$
  - $\text{voc}(t_1 \doteq t_2) = \text{voc}(t_1) \cup \text{voc}(t_2) \quad t_1,\, t_2 \in \text{TERM}_S$
  - $\text{voc}(\top) = \text{voc}(\bot) = \emptyset$
- Recursivo

- $\mathrm{voc}(\neg\varphi) = \mathrm{voc}(\varphi), \quad \varphi \in \mathrm{FORM}_S$
- $\mathrm{voc}(\varphi \,\square\, \psi) = \mathrm{voc}(\varphi) \cup \mathrm{voc}(\psi), \quad \varphi,\, \psi \in \mathrm{FORM}_S$
- $\mathrm{voc}(Qx\varphi) = \mathrm{voc}(\varphi), \quad Q \in \{\exists, \forall\}, \quad x \in \mathrm{Var} \quad \varphi \in \mathrm{FORM}_S$

Notación. Sean S_1 y S_2 son signaturas tal que

$$S_1 = \langle \mathrm{Cts}_{S_1},\, \mathrm{Fn}_{S_1},\, \mathrm{Pd}_{S_1} \rangle$$

$$S_2 = \langle \mathrm{Cts}_{S_2},\, \mathrm{Fn}_{S_2},\, \mathrm{Pd}_{S_2} \rangle$$

entonces

$$S_1 \cup S_2 = \{\mathrm{Cts}_{S_1} \cup \mathrm{Cts}_{S_2},\, \mathrm{Fn}_{S_1} \cup \mathrm{Fn}_{S_2},\, \mathrm{Pd}_{S_1} \cup \mathrm{Pd}_{S_2}\}$$

$$S_1 \cap S_2 = \{\mathrm{Cts}_{S_1} \cap \mathrm{Cts}_{S_2},\, \mathrm{Fn}_{S_1} \cap \mathrm{Fn}_{S_2},\, \mathrm{Pd}_{S_1} \cap \mathrm{Pd}_{S_2}\}$$

Definición 31. Sean S_1 y S_2 dos signaturas $\mathfrak{Y}_1 = \langle \mathfrak{a}_1,\, \sigma_1 \rangle$ una s_1-interpretación e $\mathfrak{Y}_2 = \langle \mathfrak{a}_2,\, \sigma_2 \rangle$ una s_2-interpretación. $\mathfrak{Y}_1$ e $\mathfrak{Y}_2$ tienen el mismo soporte. Sea $S = S_1 \cap S_2$, $t \in \mathrm{TERM}_S$ y $\varphi \in \mathrm{FORM}_S$

- $\mathfrak{Y}_1$ e $\mathfrak{Y}_2$ coinciden en t ($\mathfrak{Y}_1 \sim_t \mathfrak{Y}_2$) si
  a) Para todo $c \in \mathrm{voc}(t), \quad c^{\mathfrak{Y}_1} = c^{\mathfrak{Y}_2}$
  b) Para todo $x \in \mathrm{Var}(t), \quad x^{\mathfrak{Y}_1} = x^{\mathfrak{Y}_2}$
  c) Para todo $f \in \mathrm{voc}(t), \quad f \in \mathrm{Fn}_S, \quad f^{\mathfrak{a}_1} = f^{\mathfrak{a}_2}$
- $\mathfrak{Y}_1$ e $\mathfrak{Y}_2$ coinciden en φ ($\mathfrak{Y}_1 \sim_\varphi \mathfrak{Y}_2$) si
  a) Para todo $c \in \mathrm{voc}(\varphi), \quad c \in \mathrm{Cts}_S : \; c^{\mathfrak{a}_1} = c^{\mathfrak{a}_2}$
  b) Para todo $x \in \mathrm{lib}(\varphi) : \; \sigma_1(x) = \sigma_2(x)$
  c) Para todo $f \in \mathrm{voc}(\varphi), \quad f \in \mathrm{Fn}_S : \; f^{\mathfrak{a}_1} = f^{\mathfrak{a}_2}$
  d) Para todo $p \in \mathrm{voc}(\varphi), \quad p \in \mathrm{Pd}_S : \; p^{\mathfrak{a}_1} = p^{\mathfrak{a}_2}$

2.4. Lema de coincidencia

Teorema 4 (*Lema de coincidencia*)**.** *Sean S_1 y S_2 dos signaturas, $\mathfrak{Y}_1$ una S_1-interpretación e $\mathfrak{Y}_2$ una S_2-interpretación con el mismo soporte y $S = S_1 \cap S_2$.*

1. *$t \in TERM_S$ tal que $\mathfrak{Y}_1 \sim_t \mathfrak{Y}_2$ entonces*

$$t^{\mathfrak{Y}_1} = t^{\mathfrak{Y}_2}$$

2. *$\varphi \in FORM_S$ tal que $\mathfrak{Y}_1 \sim_\varphi \mathfrak{Y}_2$ entonces*

$$\varphi^{\mathfrak{Y}_1} = \varphi^{\mathfrak{Y}_2}$$

Demostración. *Esta en el foro, cuando Luis de el OK, la añado a los apuntes.* □

Notación. El lema de coincidencia nos permite ignorar el significado de los símbolos que no aparecen en fórmulas y términos, en particular las variables. Tomemos $t \in \mathrm{TERM}_S$, $\varphi \in \mathrm{FORM}_S$ tales que existe $V = \{x_1, \dots, x_n\} \subseteq \mathrm{Var}$ y $\mathrm{Var}(\varphi) \subseteq V$ y $\mathrm{lib}(\varphi) \subseteq V$. El lema de coincidencia nos permite ignorar el significado de las variables que no están en V. Tomamos una interpretación $\mathfrak{Y} = \langle \mathfrak{a}, \sigma \rangle$ y $a_i = \sigma(x_i)$ $(1 \leq i \leq n)$. En lugar de tomar $\mathfrak{Y}$, podemos escribir $\mathfrak{a}[a_1/x_1, \dots, a_n/x_n]$ o simplemente $\mathfrak{a}[\bar{a}/\bar{x}]$ donde $\bar{a} = (a_1, \dots, a_n)$ y $\bar{x} = (x_1, \dots, x_n)$

$$t^{\mathfrak{a}[\bar{a}/\bar{x}]} = t^{\mathfrak{Y}} \qquad \varphi^{\mathfrak{a}[\bar{a}/\bar{x}]} = \varphi^{\mathfrak{Y}}$$

Es particularmente útil para fórmulas cerradas (*i.e.* sin variables libres) en las que podemos tomar $n = 0$ $(v = \emptyset)$. En ese caso podemos escribir directamente

$$\varphi^{\mathfrak{a}}$$

2.5. Isomorfía

Definición 32. Sean $\mathfrak{a}_1$ y $\mathfrak{a}_2$ dos S-estructuras con soportes A_1 y A_2. La aplicación

$$h: \; A_1 \to A_2$$

es un **isomorfismo** si

(a) h es biyectiva

(b) Si c $\in \mathrm{Cts}_S$: $h(c^{\mathfrak{a}_1}) = c^{\mathfrak{a}_2}$

(c) Si $f|_k \in \mathrm{Fn}_S$ y $a_1, \dots, a_n \in A$:

$$h(f^{\mathfrak{a}_1}(a_1, \dots, a_k)) = f^{\mathfrak{a}_2}(h(a_1), \dots, h(a_k))$$

(d) Si $p|_k \in \mathrm{Pd}_S$ y $a_1, \dots, a_n \in A$:

$$p^{\mathfrak{a}_1}(a_1, \dots, a_k) = p^{\mathfrak{a}_2}(h(a_1), \dots, h(a_k))$$

$\mathfrak{a}_1$ y $\mathfrak{a}_2$ son isomorfas ($\mathfrak{a}_1 \simeq \mathfrak{a}_2$)

además, h es un **isomorfismo entre dos interpretaciones**

$$\mathfrak{Y}_1 = \langle \mathfrak{a}_1, \sigma_1 \rangle \quad \text{e} \quad \mathfrak{Y}_2 = \langle \mathfrak{a}_2, \sigma_2 \rangle$$

si

- h es isomorfismo entre $\mathfrak{a}_1$ y $\mathfrak{a}_2$
- $h(\sigma_1(x)) = \sigma_2(x)$ para $x \in \mathrm{Var}$

$$\mathfrak{Y}_1 \simeq \mathfrak{Y}_2$$

$*$	**a**	**b**
a	a	b
b	b	a

Ejemplo 27: Sean los grupos $(\mathbb{Z}_2, +) \simeq (A, *)$ donde $A = \{a, b\}$

$$S = \langle \{e\}, \{*|_2\}, \emptyset \rangle$$

$$\mathfrak{a}_1 = \langle \mathbb{Z}_2, \{e^{\mathfrak{a}_1}\}, \{*^{\mathfrak{a}_1}\}, \emptyset \rangle$$

$$\mathfrak{a}_2 = \langle A, \{e^{\mathfrak{a}_2}\}, \{*^{\mathfrak{a}_1}\}, \emptyset \rangle$$

$$h(0) = a, \quad h(1) = b, \quad \text{es isomorfía}$$

$$\mathfrak{a}_1 \simeq \mathfrak{a}_2$$

Teorema 5. *Sea h un isomorfismo entre $\mathfrak{Y}_1$ e $\mathfrak{Y}_2$*

1. *Si $t \in TERM_S$ es un término*
$$t^{\mathfrak{Y}_1} = t^{\mathfrak{Y}_2}$$

2. *Si $\varphi \in FORM_S$ es una fórmula*
$$\varphi^{\mathfrak{Y}_1} = \varphi^{\mathfrak{Y}_2} \quad (\mathfrak{Y}_1 \models \varphi \Leftrightarrow \mathfrak{Y}_2 \models \varphi)$$

Demostración. Se deja como ejercicio. La demostración para términos es inmediata por inducción estructural. Para fórmulas es sencilla, en el caso recursivo para cuantificadores hay que utilizar lo siguiente
$$\text{si } a \in A \text{ y } \mathfrak{Y}_1 \simeq \mathfrak{Y}_2$$
entonces
$$\mathfrak{Y}_1[a/x] \simeq \mathfrak{Y}_2[a/x]$$
□

Lema 4. *Los isomorfismos son funciones biyectivas.*

Demostración. Ejercicio □

2.6. Sustitución

Definición 33 (Sustitución)**. Para términos**. Sea S signatura. Sean $t \in \mathrm{TERM}_S$, $\bar{x} = (x_1, \ldots, x_n) \in \mathrm{Var}$ y $\bar{s} = (s_1, \ldots, s_n) \in \mathrm{TERM}_S$. Definimos la *sustitución* de $\bar{x}$ por $\bar{s}$ en t como

- Si $t \in \mathrm{Cts}_S$, $t[\bar{s}/\bar{x}] = t$
- Si $t \in \mathrm{Var}$, $t[\bar{s}/\bar{x}] = s_i$ si $t = x_i$, para cierto i, y $t[\bar{s}/\bar{x}] = t$ si $t \neq x_i$, para todo i
- $f(t_1, \ldots t_n)[\bar{s}/\bar{x}] = f(t_1[\bar{s}/\bar{x}], \ldots, t_n[\bar{s}/\bar{x}])$

Para fórmulas. Sea S una signatura, $\varphi \in \mathrm{FORM}_S$, definimos la **sustitución** de $\bar{x}$ por $\bar{s}$ en φ como

- Caso base:
  - $\top[\bar{s}/\bar{x}] = \top$
  - $\bot[\bar{s}/\bar{x}] = \bot$
  - $(t_1 \doteq t_2)[\bar{s}/\bar{x}] = (t_1[\bar{s}/\bar{x}] \doteq t_2[\bar{s}/\bar{x}])$
  - Si $p|_k \in \mathrm{Pd}_S$ y $t_1, \ldots, t_k \in \mathrm{TERM}_S$, $p(t_1, \ldots, t_k)[\bar{s}/\bar{x}] = p(t_1[\bar{s}/\bar{x}], \ldots, t_k[\bar{s}/\bar{x}])$
- Caso recursivo
  - $(\neg\varphi)[\bar{s}/\bar{x}] = (\neg\varphi[\bar{s}/\bar{x}])$
  - $(\varphi_1 \Box \varphi_2)[\bar{s}/\bar{x}] = (\varphi_1[\bar{s}/\bar{x}] \Box \varphi_2[\bar{s}/\bar{x}])$
  - Se define por casos

$$(Qx\ \varphi)[\bar{s}/\bar{x}] = \begin{cases} (Qx\ \varphi[\bar{s}/\bar{x}]) & \text{si} \quad x \notin \bar{x},\ x \notin \bigcup_{i=1}^n var(t_i) \\ \left(Qx\ \varphi \left[\left(\dfrac{s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n)}{(x_1, \ldots, x_{i-1}, x_{i+1} \ldots, x_n)}\right)\right]\right. & \text{si} \quad x = x_i \in \bar{x},\ x \notin \bigcup_{i=1}^n var(t_i) \\ (Qz\ \varphi[z/x][\bar{s}/\bar{x}]) & \text{si} \quad x \in \bigcup_{i=1}^n var(t_i),\ z \notin \bar{x} \cup \bigcup_{i=1}^n var(t_i) \end{cases}$$

Teorema 6 (Lema de sustitución). *Sea S una signatura, $\mathfrak{Y}$ una interpretación, $\bar{x} \in Var^n$, $\bar{t} \in TERM_S^n$. Denotamos $\bar{t}^{\mathfrak{Y}} = (t_1^{\mathfrak{Y}}, \ldots, t_n^{\mathfrak{Y}})$ y $\mathfrak{J} = \mathfrak{Y}[\bar{t}^{\mathfrak{Y}}/\bar{x}]$*

1. *Si $s \in TERM_S$ entonces*
$$(s[\bar{t}/\bar{x}])^{\mathfrak{Y}} = s^{\mathfrak{J}}$$
2. *Si $\varphi \in FORM_S$ entonces*
$$(\varphi[\bar{t}/\bar{x}])^{\mathfrak{Y}} = \varphi^{\mathfrak{J}} \quad (\mathfrak{Y} \models \varphi[\bar{t}/\bar{x}] \Leftrightarrow \mathfrak{J} = \varphi)$$

2.7. Equivalencias Lógicas

Definición 34. Sean $\varphi, \psi \in \mathrm{FORM}_S$. Decimos que φ y ψ son **lógicamente equivalentes**, escrito $\varphi \sim \psi$, si y sólo si para toda interpretación $\mathfrak{Y}$, se verifica $\varphi^{\mathfrak{Y}} = \psi^{\mathfrak{Y}}$ (equivalentemente $\mathfrak{Y} \models \varphi$ si y solo si $\mathfrak{Y} \models \psi$

Proposición 4. *Sean $\varphi, \psi \in FORM_S$, las siguientes afirmaciones son equivalentes*

1. $\varphi \sim \psi$.
2. $\models \varphi \leftrightarrow \psi$.
3. $\varphi \models \psi$ *y* $\psi \models \varphi$.

Demostración. Se deja como ejercicio □

Teorema 7 (Leyes de equivalencia lógica con cuantificadores). *Se tiene que*

1. *Todas las leyes de equivalencia lógica de la lógica proposicional (véase 1.5) siguen siendo validas.*

2. *Siendo u variable nueva*
$$\forall x\,\varphi \sim \forall u\,\varphi[u/x],$$
$$\exists x\,\varphi \sim \exists u\,\varphi[u/x]$$

3. *Si $\varphi \sim \psi$, entonces*
$$\forall x\,\varphi \sim \forall x\,\psi,$$
$$\exists x\,\varphi \sim \exists x\,\psi$$

4. $$\neg\forall x\,\varphi \sim \exists x\,\neg\varphi,$$
$$\neg\exists x\,\varphi \sim \forall x\,\neg\varphi$$

5. $$\forall x\,\varphi \sim \neg\exists x\,\neg\varphi,$$
$$\exists x\,\varphi \sim \neg\forall x\neg\varphi$$

6. *Si $x \neq y$, entonces*
$$\forall x\,\forall y\,\varphi \sim \forall y\,\forall x\,\varphi,$$
$$\exists x\,\exists y\,\varphi \sim \exists y\,\exists x\,\varphi$$

7. $$\forall x\,\varphi \wedge \psi \sim \forall x\,\varphi \wedge \forall x\,\psi,$$
$$\exists x\,\varphi \vee \psi \sim \exists x\,\varphi \vee \exists x\,\psi$$

En las reglas que siguen suponemos que $x \notin lib(\psi)$.

8. $$\forall x\,\psi \sim \psi,$$
$$\exists x\,\psi \sim \psi$$

9. $$(\forall x\,\varphi) \wedge \psi \sim \forall x\,(\varphi \wedge \psi),$$
$$(\exists x\,\varphi) \vee \psi \sim \exists x\,(\varphi \vee \psi)$$

10. $$\forall x\,(\varphi \vee \psi) \sim \forall x\,\varphi \vee \forall x\,\psi,$$
$$\exists x\,(\varphi \wedge \psi) \sim \exists x\,\varphi \wedge \exists x\,\psi$$

11. $$(\forall x\,\varphi) \to \psi \sim \exists x\,(\varphi \to \psi),$$
$$(\exists x\,\varphi) \to \psi \sim \forall x(\varphi \to \psi)$$

Demostración. Se tiene que

1. Ya han sido justificadas anteriormente.

2. Veamos el caso $\forall$, el $\exists$ se hace de forma análoga (*se recomienda escribirlo)*.

$$\mathfrak{Y} \models \forall u\, \varphi[u/x]$$

   por definición es,
$$\text{para todo } a \in A,\ \mathfrak{Y}[a/u] \models \varphi[u/x]$$
   Por el lema de sustitución $u^{\mathfrak{Y}[a/u]} = a$,

$$\text{para todo } a \in A,\ (\mathfrak{Y}[a/u])[a/x] \models \varphi$$

   puesto que u es nueva,
$$(\mathfrak{Y}[a/u])[a/x] \sim_\varphi \mathfrak{Y}[a/x]$$
   por el lema de coincidencia lo anterior es equivalente a,

$$\text{para todo } a \in A,\ \mathfrak{Y}[a/x] \models \varphi$$

   que es la definición de $\mathfrak{Y} \models \forall x\, \varphi$.

3. Tenemos $\mathfrak{Y}$ tal que $\mathfrak{Y} \models \exists x\, \varphi$, por definición existe $a \in A$ tal que $\mathfrak{Y}[a/x] \models \varphi$ puesto que $\varphi \sim \psi$, lo anterior es equivalente a $a \in A$ con $\mathfrak{Y}[a/x] \models \psi$, que es la definición de $\mathfrak{Y} \models \exists x\, \psi$. De nuevo el caso $\forall$ es análogo (*Escribirlo*).

4. Tenemos $\mathfrak{Y}$ tal que $\mathfrak{Y} \models \neg\forall x\, \varphi$ que es lo mismo que $\mathfrak{Y} \not\models \forall x\, \varphi$. Por tanto, debe existir $a \in A$ tal que $\mathfrak{Y}[a/x] \not\models \varphi$, de forma equivalente $\mathfrak{Y} \models \neg\varphi$ que es la definición de $\mathfrak{Y} \models \exists x \neg\varphi$. (*Escribir el caso de la negación*).

5. $\varphi \sim \neg\neg\varphi$ por tanto, aplicando (3), $\forall x\, \varphi \sim \forall x\, \neg\neg\varphi$, aplicando (4), $\forall x\, \neg\neg\varphi \sim \neg\exists x\, \neg\varphi$ por tanto
$$\forall x\, \varphi \sim \neg\exists x\, \neg\varphi$$
   El caso $\exists$ es análogo.

6. Si $x = y$, ver regla (8) más adelante que implica

$$\forall x \forall x \varphi \sim \forall x \varphi$$

$$\exists x \exists x \varphi \sim \exists x \varphi$$

7. Sea $\mathfrak{Y}$ tal que $\mathfrak{Y} \models \forall x\, \varphi \wedge \forall x\, \psi$ es equivalente a que $\mathfrak{Y} \models \forall x\, \varphi$ y $\mathfrak{Y} \models \forall x\, \psi$.

   Entonces, en el primer caso tenemos que

$$\text{para todo } a \in A,\ \mathfrak{Y}[a/x] \models \varphi$$

   y en el segundo que
$$\text{para todo } b \in A,\ \mathfrak{Y}[b/x] \models \psi$$

obtenemos
$$\text{para todo } c \in A,\ \mathfrak{Y}[c/x] \models \varphi \text{ y } \mathfrak{Y}[c/x] \models \psi$$
entonces
$$\mathfrak{Y}[c/x] \models \varphi \wedge \psi$$
finalmente
$$\mathfrak{Y} \models \varphi \wedge \psi$$

8. Hay que usar que $\mathfrak{Y}[a/x] \sim_\psi \mathfrak{Y}$
9. Aplicando (7) y (8).
10. Hay que demostrar que $\mathfrak{Y} \models \forall x\, \varphi \vee \psi$ si y solo si $\mathfrak{Y} \models \forall x\, \varphi \vee \forall x \psi$

(⇒) $\mathfrak{Y} \models \forall x\, (\varphi \vee \psi)$
$$\text{para todo } a \in A,\ \mathfrak{Y}[a/x] \models \varphi \vee \psi$$
diferenciamos dos casos

- Existe $a \in A$, $\mathfrak{Y}[a/x] \not\models \varphi$ entonces $\mathfrak{Y}[a/x] \models \psi$. Por el lema de coincidencia $gY \models \psi$. Por el lema de coincidencia
$$\text{para todo } c \in A,\ \mathfrak{Y}[c/x] \models \psi$$
que es a definición de
$$\mathfrak{Y} \models \forall x \psi$$
por tanto $\mathfrak{Y} \models \forall x \varphi \vee \forall x \psi$
- para todo $a \in A$, $\mathfrak{Y}[a/x] \models \varphi$ definición de $\mathfrak{Y} \models \forall x \varphi$

(⇐) De nuevo dividimos en dos casos

- $\mathfrak{Y} \models \forall x \varphi$ sii para todo $a \in A$, $\mathfrak{Y}[a/x] \models \varphi$ por definición de $\forall$, para todo $a \in A$, $\mathfrak{Y}[a/x] \models \varphi \vee \psi$ por definición de para todo $\mathfrak{Y} \models \forall x(\varphi \vee \psi)$
- $\mathfrak{Y} \models \forall x \psi$ (*análogo al anterior.*

Escribir los casos para el $\exists$

□

Las leyes de equivalencia permiten *sacar* los cuantificadores al inicio de la fórmula.

Definición 35. Sea $\varphi \in \mathrm{FORM}_S$ se dice que está en forma prenexa si es de la forma
$$\varphi = Q_1 x_1, \dots, Q_n x_n\, \varphi, \quad Q_i \in \{\forall, \exists\} \quad 1 \leq i \leq n$$
ψ (el núcleo) está libre de cuantificadores.

Proposición 5. *Sea $\varphi \in FORM_S$. Existe $\psi \in FORM_S$ en forma prenexa tal que $\psi \sim \varphi$*

Demostración. Antes de nada observar lo siguiente. Si $x \in \mathrm{Lib}(\psi)$

$$\forall x\varphi \vee \psi \sim \forall u\varphi[u/x] \vee \psi \sim \forall u(\varphi[u/x] \vee \psi)$$

u variable nueva.

La demostración se hace por inducción estructural. Los casos complicados se relación con los conectivos. Se hace para $\{\neg, \vee\}$ (los demás casos son análogos, y a la vez, innecesarios pues ya vimos que el conjunto $\{\neg, \vee\}$ es funcionalmente completo).

($\neg$) Por hipótesis de inducción ψ en forma prenexa tal que $\varphi \sim \psi$

$$\psi = Q_1x_1, \dots, Q_nx_n\,\psi$$

$$\neg\varphi \sim \neg\psi \sim Q'_1x_1, \dots, Q'_nx_n\,\psi_1$$

donde

$$Q'_i = \begin{cases} \forall & \text{si} \quad Q_i = \exists \\ \\ \exists & \text{si} \quad Q_i = \forall \end{cases}$$

($\vee$) Por hipótesis de inducción existen φ' y ψ' en forma prenexa tal que

$$\varphi \sim \varphi' \qquad \psi \sim \psi'$$

$$\varphi' = Q'_1x_1, \dots, Q'_nx_n\,\varphi''$$
$$\psi' = Q^2_1y_1, \dots, Q^2_my_m\,\psi''$$

consideramos variables nuevas $u_1, \dots, u_n$ y $v_1, \dots, v_m$, entonces

$$\varphi' = Q'_1u_1, \dots, Q'_nu_n\,\varphi''[\bar{u}/\bar{x}]$$

$$\psi' = Q^2_1v_1, \dots, Q^2_mv_m\,\psi''[\bar{v}/\bar{y}]$$

puesto que $\bar{u}$ y $\bar{v}$ son nuevas

$$\varphi' \vee \psi' = Q'_1u_1, \dots, Q'_nu_n\,Q^2_1v_1, \dots, Q^2_mv_m\,\varphi''[\bar{u}/\bar{x}] \vee \psi''[\bar{v}/\bar{y}]$$

□

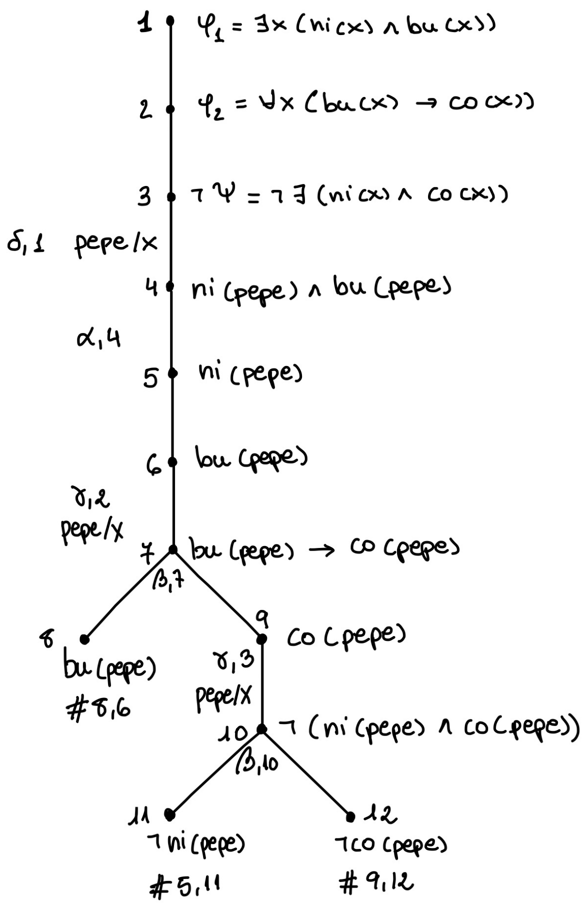
2.8. Tableaux para la lógica de primer orden

Empezamos con un ejemplo sencillo. Sean

$$\varphi_1 \equiv \text{ Algunos niños son buenos}$$

$$\varphi_2 \equiv \text{ Todo lo bueno se come}$$

$$\therefore \text{ Algunos niños se comen}$$



para formalizar se ha utilizado

$$\text{ni}(x) \equiv x \text{ es niño}$$

$$\text{co}(x) \equiv x \text{ se come}$$

$$\text{bu}(x) \equiv x \text{ es bueno}$$

En el ejemplo anterior hemos usado la constante *pepe*. En general supondremos que existe un conjunto de constantes auxiliares

$$C_A = \{c_0, c_1, c_2, \ldots\}$$

En los tableaux supondremos que estas constantes están siempre en nuestra signatura.

2.8.1. Clasificación de fórmulas

Tomamos de la lógica proposicional las α, β y σ fórmulas. Además, consideramos

- γ-fórmulas, universales

$$\begin{array}{c|c} \gamma & \gamma(t) \\ \hline \forall x\,\varphi & \varphi[t/x] \\ \neg\exists x\,\varphi & \neg\varphi[t/x] \end{array} \quad t \in \text{TERM}_S.$$

- δ-fórmulas, existenciales

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

- Axiomas de igualdad ($\theta \in \mathrm{EQ}_S$)

(RF) $\forall x\;\; x \doteq x$

(IM) $\forall x \forall y\; x \doteq y \to y \doteq x$

(IM) $\forall x \forall y \forall z\; x \doteq y \wedge y \doteq z \to x \doteq z$

(ST_1) Sustitución para funciones. Si $f|_k \in \mathrm{Fn}_S$

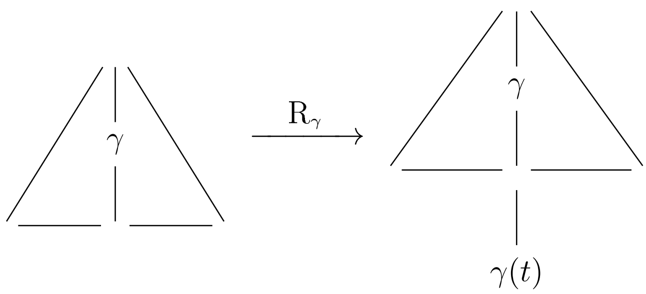
$$\forall x_1, \ldots x_k \forall y_1 \ldots, y_k\; x_1 \doteq y_1 \wedge \ldots \wedge x_k \doteq y_k \to f(x_1, \ldots x_k) = f(y_1 \ldots, y_k)$$

(ST_2) Sustitución para predicados. Si $p|_k \in \mathrm{Pd}_S$

$$\forall x_1, \ldots x_k \forall y_1 \ldots, y_k\; x_1 \doteq y_1 \wedge \ldots \wedge x_k \doteq y_k \to p(x_1, \ldots x_k) = p(y_1 \ldots, y_k)$$

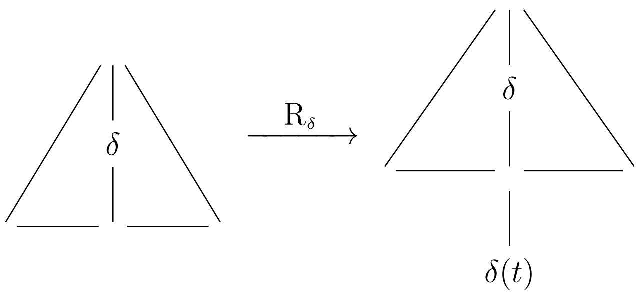
2.8.2. Reglas para la construcción de tableaux

- Las reglas que ya fueron utilizadas en la lógica proposicional aquí son exactamente igual de válidas: $\mathrm{R_{ini}}$, R_α, R_β, R_σ, $\mathrm{R_{hip}}$.
- R_γ, si γ es fórmula universal y $t \in \mathrm{TERM}_S$ un término



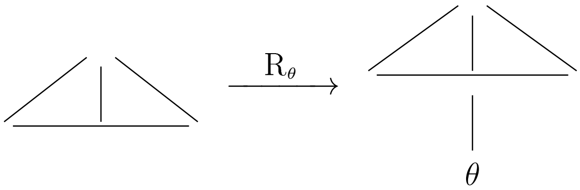
  si r es una rama abierta del árbol y $\gamma \in \Gamma_r$, se puede extender la rama r con $\gamma(t)$.

- R_δ, si δ es fórmula existencial y $c \in C_A$ una constante



  si r es una rama abierta del árbol, se puede extender la rama r con $\delta(c)$. Donde c debe ser nueva en la rama.

- R_θ, $\theta \in \mathrm{EQ}_S$



en cualquier momento se pueden usar los axiomas de igualdad.

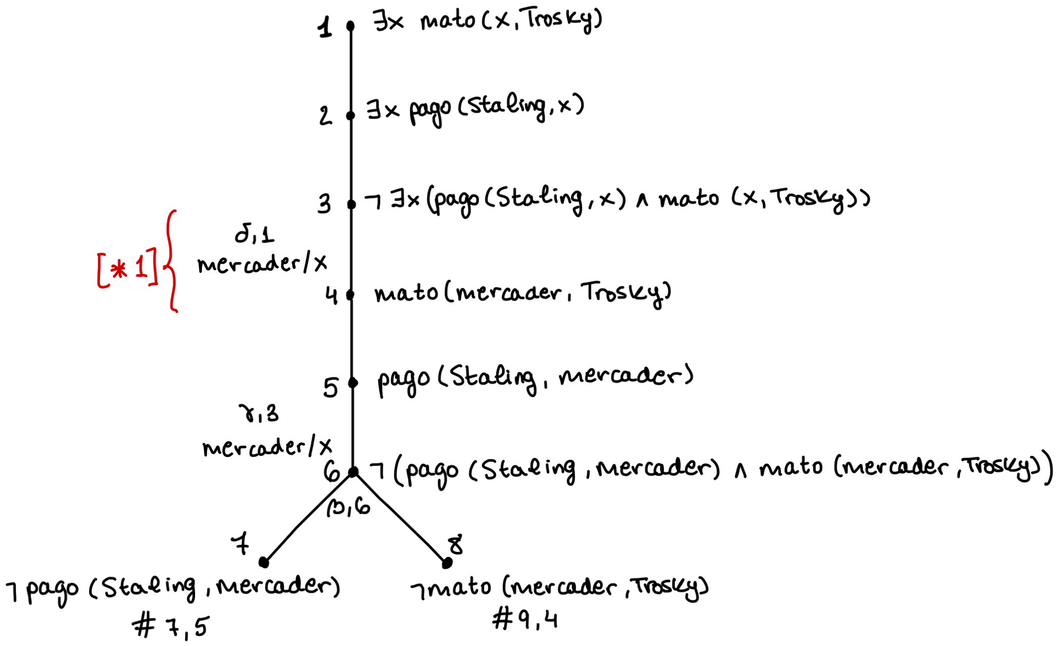
Una vez vistas estas reglas, surge de forma natural la cuestión

¿Por qué la constante c debe ser nueva?

veámoslo a través de un ejemplo.

Ejemplo 28:

Alguien mató a Trosky Estalín pagó a Alguien	$\exists x$ mato$(x$, Trosky$)$ $\exists x$ pago(Stalin, $x)$
$\therefore$ Estalín pagó a alguien que mató a Trosky	$\therefore \exists x$ pago(Stalin, $x) \wedge$ mato$(x$, Trosky$)$



[$*1$] En el punto 4 hemos aplicado correctamente una regla δ introduciendo la constante auxiliar *Mercader*, pero después hemos vuelto a introducirla, por tanto, estamos introduciendo una constante auxiliar que no es nueva en la rama y como consecuencia la deducción original es válida, lo que no tiene sentido.

Definición 36. Dados $\Phi \subseteq \mathrm{FORM}_S$ y $\varphi \in \mathrm{FORM}_S$, escribimos $\Phi \overset{\mathrm{tb}}{\vdash} \varphi$ cuando existe un **tableau cerrado** para $\Phi \cup \{\neg\varphi\}$.

Lema 5 (**Correción**).

$$\Phi \overset{tb}{\longmapsto} \varphi \implies \Phi \models \varphi$$

Lema 6 (**Completitud**).

$$\Phi \models \varphi \implies \Phi \overset{tb}{\longmapsto} \varphi$$

Lema 7.

$$\theta \in EQ_S, \ \models \theta$$

Los axiomas de igualdad, son axiomas.

Demostración. *Ejercicio* □

Para demostrar la corrección habrá que ver que las reglas de construcción de tableaux son correctas. Informalmente si una rama es satisfactible, alguna de sus extensiones lo es.

Proposición 6. *Sean* $\Phi \subseteq FORM_S$ *e* $\mathfrak{Y}$ *una* S*-interpretación tal que* $\mathfrak{Y} \models \Phi$. *Entonces existe* $\mathfrak{Y}'$ *tal que* $\mathfrak{Y} \sim_\varphi \mathfrak{Y}'$ *para toda* $\varphi \in \Phi$ *y*

- *Sea* $\sigma \in \Phi$, *entonces* $\mathfrak{Y}' \models \Phi \cup \{\sigma_1\}$. *(*$\sigma$ *fórmula simplificable)*
- *Si* $\alpha \in \Phi$, *entonces* $\mathfrak{Y}' \models \Phi \cup \{\alpha_1, \alpha_2\}$. *(*$\alpha$ *una* α*-fórmula)*
- *Si* $\beta \in \Phi$, *entonces* $\mathfrak{Y}' \models \Phi \cup \{\beta_1\}$ *o* $\mathfrak{Y}' \models \Phi \cup \{\beta_2\}$.
- *Si* $\gamma \in \Phi$, *entonces* $\mathfrak{Y}' \models \Phi \cup \{\gamma(t)\}$ *para* $t \in TERM_S$.
- *Si* $\delta \in \Phi$, *entonces* $\mathfrak{Y}' \models \Phi \cup \{\delta(c)\}$ *para* $c \in C_A$ *nueva en* Φ, $c \in voc(\Phi)$.
- *Si* $\theta \in EQ_S$, *entonces* $\mathfrak{Y}' \models \Phi \cup \{\theta\}$.

Demostración. Los casos para α, β y σ fórmulas son similares a los de la lógica proposicional. Basta usar que $\mathfrak{Y}' = \mathfrak{Y}$ y

$$\sigma \sim \sigma_1$$
$$\alpha \sim \alpha_1 \wedge \alpha_2$$
$$\beta \sim \beta_1 \vee \beta_2$$

Se deja su construcción en detalle como ejercicio.

El caso para $\theta \in \mathrm{EQ}_S$ también es trivial por el lema anterior.

Supongamos $\gamma \in \Phi$. Supongamos $\gamma = \forall x\, \varphi$ (si $\gamma = \neg\exists\varphi$ basta usar el hecho de que es equivalente a $\forall x \neg\varphi$ y proceder como se indica). En este caso tomamos $\mathfrak{Y}' = \mathfrak{Y}$.

Tomemos $t \in \mathrm{TERM}_S$. Sea $a = t^{\mathfrak{Y}} \in A$. Puesto que $\forall x\varphi \in \Phi$, tenemos $\mathfrak{Y} \models \forall x\varphi$ y por tanto

$$\mathfrak{Y}[a/x] \models \varphi$$

Por el lema de sustitución tendremos

$$\mathfrak{Y} \models \varphi[t/x] y \varphi[t/x] = \gamma(t)$$

Supongamos que $\delta \in \Phi$. Supongamos $\delta = \exists x\, \varphi$ (si $\delta = \neg\forall\varphi$ basta usar el hecho de que es equivalente a $\exists x \neg\varphi$ y proceder como se indica). Este es el único caso en el que $\mathfrak{Y}' \neq \mathfrak{Y}$.

Tomemos $c \in C_A$ nueva, $c \in \mathrm{voc}(\Phi)$ ($c \notin \mathrm{voc}\psi, \quad \psi \in \Phi$)). $\delta(c) = \varphi[c/x]$.

Puesto que $\mathfrak{Y} \models \exists x\varphi$ y por definición existe $a \in A$ tal que $\mathfrak{Y}[a/x] \models \varphi$.

Supongamos que $\mathfrak{Y} = \langle \mathfrak{a}, \sigma \rangle$. Tomemos $\mathfrak{a}'$, igual que $\mathfrak{a}$ salvo que $c^{\mathfrak{a}'} = a$. Tomemos $\mathfrak{Y}' = \langle \mathfrak{a}', \sigma \rangle$.

Puesto que $c \notin \mathrm{voc}(\psi)$ para $\psi \in \Phi$, tenemos que

$$\mathfrak{Y}' \sim_\psi \mathfrak{Y}$$

Por el lema de sustitución

$$(\varphi[c/x])^{\mathfrak{Y}'} = \varphi^{\mathfrak{Y}'[c^{\mathfrak{Y}'}/x]} = \varphi^{\mathfrak{Y}'[a/x]}$$

Al ser c nueva

$$\mathfrak{Y}'[a/x] \sim_\varphi \mathfrak{Y}[a/x]$$

y de nuevo, obtenemos

$$\varphi^{\mathfrak{Y}'[a/x]} = \varphi^{\mathfrak{Y}[a/x]} = \mathrm{V}$$

□

Teorema 8. *Sea $\Phi \subseteq FORM_S$ un conjunto de fórmulas satisfactible (Sat(Φ)) y T un tableau abierto para Φ. ENtonces existe una rama abierta r de T tal que Sat(Γ_r).*

Demostración. Desmotraremos que existe una interpretación $\mathfrak{Y}$ y una rama r tales que $\mathfrak{Y} \models \Gamma$ e $\mathfrak{Y} \models \Phi$.

Si T es un tableau finito para Φ se ha construido en número finito de pasos

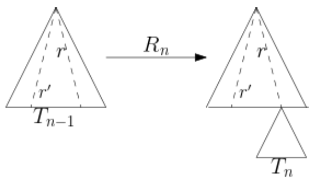
$$T_0 \overset{R_1}{\rightsquigarrow} T_1 \overset{R_2}{\rightsquigarrow} T_2 \;\ldots\; T_{n-1} \overset{R_n}{\rightsquigarrow} T_n$$

de forma que T_0 es un tableau inicial para Φ y R_i ($1 \leq i \leq n$) es una regla. Se hace la demostración por inducción sobre n.

$n = 0$) T_0 es un tableau inicial donde $\gamma = \{\varphi_1, \varphi_2, \ldots, \varphi_n \subseteq \Phi\}$. Puesto que Sat$(\Phi)$, tenemos que Sat$(\Gamma)$.

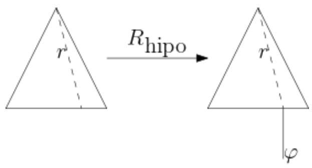
$n \geq 1$) $T_{n-1} \overset{R_n}{\rightsquigarrow} T_n$. La regla R_n extiende una rama r de T_{n-1}. Por otro lado hay una rama r de T_{n-1} tal que Sat(Γ'_r).

- Si $r \neq r'$ entonces



r' también es una rama.

Si $R_n = R_{\text{hipo}}$, existe $\varphi \in \Phi$ tal que



Por hipótesis de inducción existe $\mathfrak{Y}$ tal que $\mathfrak{Y} \models \Gamma_r^{n-1}$ y $\mathfrak{Y} \models \Phi$. Puesto que $\varphi \in \Phi$, tenemos que $\mathfrak{Y} \models \varphi$ por tanto

$$\mathfrak{Y} \models \Gamma_r^{n-1} \cup \{\varphi\} = \Gamma_r^n$$

Los casos $R_n = R_\alpha$, $R_n = R_\beta$ y $R_n = R_\sigma$ son similares al caso de la lógica proposicional.

$R = R_\delta$ en este caso existe $\delta \in \Gamma_r^{n-1}$ y $\Gamma_r^n = \Gamma_r^{n-1} \cup \{\delta(c)\}$. Por hipótesis de inducción existe $\mathfrak{Y}$ tal que

$$\mathfrak{Y} \models \Gamma_r^{n-1} \cup \{\delta(c)\} = \Gamma_r^n$$

En el lema anterior la única modificación entre $\mathfrak{Y}$ e $\mathfrak{Y}'$ es la constante c. Puesto que es auxiliar $c \notin \text{voc}(\psi)$ para $\psi \in \Phi$. Por tanto $\mathfrak{Y}' \sim_\psi \mathfrak{Y}$ para $\psi \in \Phi$ y tenemos $\mathfrak{Y}' \models \psi$ para $\psi \in \Phi$. Es decir

$$\mathfrak{Y}' \models \Phi$$

Se dejan como ejercicio los casos $R_n = R_\gamma$ y $R_n = R_\theta$

□

Corolario 1. *Se tiene que*

1. *Si $\Phi \subseteq FORM_S$ y Φ tiene un tableau cerrado, entonces $InSat(\Phi)$.*

2. $\Phi \overset{tb}{\longmapsto} \varphi$ *entonces,* $\Phi \models \varphi$.

2.9. Completitud

Para poder demostrar la completitud debemos considerar árboles infinitos. ¿Qué es un árbol infinito?

Definición 37. Un árbol se puede ver como un grafo no dirigido $T = \langle N, E\rangle$ donde el conjunto N esta formado por los nodos y el conjunto E por las aristas. Un árbol es un **grafo conexo acíclico** (entre cada para de nodos hay una sola secuencia de aristas que los unen. Si a un árbol se le quita una arista dejada de ser un grafo conexo.

En un árbol finito se verifica la propiedad $|\overline{\tau}| = |N| - 1$ como un si y sólo si.

Definición 38. Dados 2 árboles $T_1 = \langle N_1, E_1\rangle$ y $T_2 = \langle N_2, E_2\rangle$, diremos que T_1 está incluido en T_2 y escribimos $T_1 \leq T_2$ si

- $N_1 \subseteq N_2$ y la raíz coincide
- $E_1 \subseteq E_2$

Definición 39. Dada una secuencia de árboles

$$T_0 \leq T_1 \leq \ldots \leq T_k \leq T_{k+1} \leq \ldots$$

definimos

$$\lim_{n\in\mathbb{N}} T_n = \langle N, E\rangle, \text{ donde } N = \bigcup_{n\in\mathbb{N}} N_n \text{ y } E = \bigcup_{n\in\mathbb{N}} E_n$$

y la raíz, es la raíz de T_0.

Proposición 7. *Si $T_0 \leq T_1 \leq \ldots \leq T_k \leq T_{k+1} \leq \ldots$ es una secuencia de árbol. Sea $T = \lim_{n\in\mathbb{N}} T_n$, entonces*

(a) $T_i \leq T$ para $i \in \mathbb{N}$

(b) T es un árbol

Demostración. La afirmación (a) de la proposición es consecuencia directa de las definiciones. Hay que demostrar que T es conexo. En efecto, tomemos n_1, $n_2 \in N$ (nodos de T). Puesto que los nodos de T es la unión de todos los nodos y $N_i \subseteq N_j$ para $j > i$. Debe existir k tal que n_1, $n_2 \in N_k$. Puesto que N_k es conexo, debe existir una secuencia de aristas $e_1, \ldots, e_l \in E_k$ que unen n_1 y n_2. Puesto que $E_k \leq \bigcup_{i\in\mathbb{N}} E_i = E$, $e_1, \ldots, e_l \in E$ y por tanto n_1 y n_2 están conectados en T.

Supongamos que existen 2 nodos n_1 y n_2 de forma que hay 2 caminos $e'_1, \ldots, e'_k$ y $e''_1, \ldots, e''_l$ diferentes que los conectan. De existir $i \in N$ tal que n_1, $n_2 \in N_i$ de forma que hay 2 caminos $e'_1, \ldots, e'_k$, $e''_1, \ldots, e''_l \in E_i$. Por tanto en T_i habría 2 formas de conectar n_1 y n_2. Lo que contradice el hecho de que T_i es un árbol. □

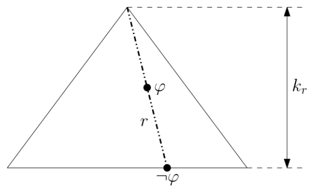
Las reglas de construcción de Tableaux solo extienden ramas abiertas. Es absurdo expandir una rama cerrada. Por tanto si una rama está cerrada, tiene profundidad finita.

Proposición 8. *Si T es un tableau cerrado entonces es finito.*

Demostración. En primer lugar observemos que en un tableau, el número máximo de descendientes es 2 (cada nodo tiene como mucho 2 hijos correspondientes a la aplicación de una regla de tipo β).

En toda rama hay 2 nodos n_1 y n_2 que se contradicen $h_r = \text{máx}\{\text{altura}n_1, \text{altura}n_2\}$. Tomamos

$$h = \text{máx}\{h_r/r \text{ rama de T}\}$$



- Si $h < \infty$ como cada nodo tiene como mucho 2 descendientes el árbol es finito.
- Si $h = \infty$ el árbol ha de tener ramas de altura arbitraria.
  Puesto que cada nodo tiene como mucho 2 descendientes, por el lema de Köning, debe tener una rama infinita. Esa rama infinita no puede estar cerrada puesto que las ramas se cierran siempre a profundidad finita.

□

Lema 8 (König). *[1] Sea T cualquier árbol localmente finito, esto es, con la propiedad de que cada nodo interno tiene un número finito de hijos. Si T es infinito, entonces existe en T al menos una rama infinita.*

2.10. Conjuntos de Hintikka para lógica de primer Orden

Para continuar con el teorema de completitud, lo siguiente es definir el análogo a los conjuntos de Hintikka que vimos en la lógica proposicional, pero ahora para la lógica de primer orden.

Lo primero que necesitamos es el concepto de **término adecuado** para un conjunto de fórmulas $\Phi \subseteq \text{FORM}_S$.

Definición 40. Sea S una signatura, $t \in$ y $\Phi \subseteq \text{FORM}_S$. Diremos que t es **adecuado** para Φ si

- $\text{voc}(t) \subseteq \text{voc}(\Phi)$
- $\text{Var}(t) \subseteq \text{lib}(\Phi)$

Observación. Puede darse el caso de que no encontremos ningún término adecuado para un conjunto de fórmulas. Por ejemplo consideremos el conjunto de fórmulas

$$\Phi = \{\exists x\ (p(x) \rightarrow q(x))\}$$

En este caso $(\text{voc}(\Phi) \cap Ct_S) \cup \text{lib}(\Phi) = \emptyset$. En este caso tomaremos como término adecuado cualquier constante auxiliar $c \in C_A$ nueva.

Definición 41. Sea S una signatura, $\Phi \subseteq \text{FORM}_S$ y $T_H = \{t\,/t \in \text{TERM}_S \text{ adecuado para } \Phi\}$. Diremos que Φ es de **Hintikka** si y sólo si

- Φ es coherente, es decir
  - $\perp \notin \Phi$
  - no existe $\varphi \in \Phi$ tal que $\varphi \in \Phi$ y $\varphi \in \Phi$
- Φ es α-saturado: si $\alpha \in \Phi$ es una α-fórmula, $\alpha_1, \alpha_2 \in \Phi$

- Φ es β-saturado: si $\beta \in \Phi$ es una β-fórmula, $\beta_1 \in \Phi$ o $\beta_2 \in \Phi$
- Φ es σ-saturado: si $\sigma \in \Phi$ es una σ-fórmula, $\sigma_1 \in \Phi$
- Φ es γ-saturado: si $\gamma \in \Phi$ es una γ-fórmula y $t \in T_H$, entonces $\gamma(t) \in \Phi$
- Φ es δ-saturado: si $\delta\Phi$ es una γ-fórmula, entonces existe una constante auxiliar $c \in CA$ tal que $\delta(c) \in \Phi$
- Φ es θ-saturado: si $\theta \in \mathrm{EQ}_S$ es un axioma de igualdad, entonces $\theta \in \Phi$.

A continuación veremos que un conjunto de Hintikka es satisfactible. El primer paso para llegar a ello es establecer cuál va a ser el soporte. Al ser TERM_S un conjunto, nos puede servir para construir el soporte. Es lo que se llama el **álgebra libre de términos**. Pero este álgebra no es suficiente. Supongamos que tenemos la signatura de la aritmética. Según las fórmulas, en cual modelo de la aritmética podemos tener que los términos $s(0) + 0$ y $s(0)$ representan el mismo elemento. Eso se traduce que en nuestro álgebra necesitamos establecer una relación de equivalencia de términos. Esa relación la denotaremos como $\equiv_\Phi$, la idea es que $t \equiv_\Phi s$ si $\Phi \models t \doteq s$.

Definición 42. Sea S una signatura, $\Phi \subseteq \mathrm{FORM}_S$ y $s, t \in \mathrm{TERM}_S$. Definimos

$$t \equiv_\Phi s \quad \text{si y sólo si} \quad t \doteq s \in \Phi$$

Lo primero que vamos a hacer es demostrar que, si Φ es un conjunto de fórmulas de Hintikka, $\equiv_\Phi$ es una relación de equivalencia en el conjunto T_Φ.

Proposición 9. *Sea S una signatura, $\Phi \subseteq FORM_S$ un conjunto de fórmulas de Hintikka. Entonces la relación de $\equiv_\Phi \subseteq T_\Phi \times T_\Phi$ es una relación de equivalencia.*

Demostración. Sean $t, s \in T_\Phi$ tales que $t \equiv_\Phi s$, es decir, $t \doteq s \in \Phi$. Por ser Φ un conjunto de Hintikka tenemos que contiene el axioma de simetría

$$\forall x \forall y\, x \doteq y \to y \doteq x \ \in \Phi$$

Resulta que ese axioma es una γ-fórmula. Por tanto $\gamma(t) \in \Phi$

$$\gamma(t) = \forall y\, t \doteq y \to t \doteq x \ \in \Phi$$

Que vuelve a ser una γ-fórmula, por tanto $\gamma(t)(s) \in \Phi$

$$\gamma(t)(s) = t \doteq s \to t \doteq s \ \in \Phi$$

Que es una β-fórmula, $\beta_1 = \neg t \cdot s$ y $\beta_2 = s \cdot t$. Puesto que $t \doteq s \in \Phi$, deducimos $\beta_1 \notin \Phi$. De lo que se deduce $\beta_2 = s \cdot t \in \Phi$. Es decir $s \equiv_\Phi t$ $\square$

Todos sabemos que dado una relación de equivalencia induce una partición en el conjunto y por tanto podemos considerar el conjunto cociente, es decir, el conjunto de particiones inducido por la relación de equivalencia. Dado $t \in T_\Phi$,

$$[t] \quad = \quad \{s \,/ s \in T_\Phi, t \equiv_\Phi s\}$$

Entonces tenemos

$$T_\Phi / \equiv_\Phi \quad = \quad \{[t] \,/\, t \in T_\Phi\}$$

El soporte del modelo que vamos a construir es este conjunto cociente: $T_\Phi / \equiv_\Phi$.

En este conjunto cociente debemos definir el valor de las constantes, las funciones y los símbolos de predicado.

- Si $c \in Ct_S$ definimos
$$c^{\mathfrak{T}_\Phi} = \begin{cases} [c] & \text{si } c \in T_\Phi \\ \text{arbitrario} & \text{e.o.c} \end{cases}$$
Por el lema de coincidencia no importa el valor que tenga.
- Si $f|_k \in \mathrm{Fn}_S$ y $t_1, \cdots t_k \in T_\Phi$
$$f^{\mathfrak{T}_\Phi}([t_1], \cdots [t_k]) = \begin{cases} [f(t_1, \cdots t_k)] & \text{si } f(t_1, \cdots t_k) \in T_\Phi \\ \text{arbitrario} & \text{e.o.c} \end{cases}$$
- Si $p|_k \in \mathrm{Pd}_S$ y $t_1, \cdots t_k \in T_\Phi$
$$p^{\mathfrak{T}_\Phi}([t_1], \cdots [t_k]) = \begin{cases} V & \text{si } p(t_1, \cdots t_k) \in \Phi \\ F & \text{e.o.c} \end{cases}$$

Antes de seguir hay que demostrar que las funciones y símbolos de predicado están bien definidos. Al estar definidos a partir de los representantes de las clases, hay que ver que no dependen de éstos.

Proposición 10. *Sea S una signatura, $\Phi \subseteq FORM_S$ un conjunto de fórmulas de Hintikka, $t_1, \ldots, t_k, s_1, \ldots, s_k \in T_\Phi$ términos tales que $[t_i] = [s_i]$ para $1 \leq i \leq k$.*

- *Si $f|_k \in Fn_S$ entonces $[f(t_1, \ldots, t_k)] = [f(s_1, \ldots, s_k)]$*
- *Si $p|_k \in Pd_S$ entonces $p(t_1, \ldots t_k) \in \Phi$ si y sólo si $p(s_1, \ldots, s_k) \in \Phi$*

Demostración. En primer lugar veamos el caso de los símbolos de función. Puesto que Φ es de Hintikka tenemos que el axioma de igualdad

$$\gamma_0 = \forall x_1 \cdots \forall x_k \forall y_1 \cdots \forall y_k \, x_1 \doteq y_1 \wedge \cdots \wedge x_k \doteq y_k \to f(x_1, \cdots, x_k) \doteq f(y_1, \cdots, y_k) \in \Phi$$

Tenemos ahora una secuencia de fórmulas

$$\begin{aligned} &\gamma_1 = \gamma_0(t_1), \gamma_2 = \gamma_1(t_2), \cdots \gamma_k = \gamma_{k-1}(t_k) \\ &\gamma_{k+1} = \gamma_k(s_1), \gamma_{k+2}(s_2), \cdots \gamma_{2k} = \gamma_{2k-1}(s_k) \\ &\gamma_{2k} \;=\; t_1 \doteq s_1 \wedge \cdots t_k \doteq s_k \to f(t_1, \cdots, t_k) \doteq f(s_1, \cdots, s_k) \end{aligned}$$

Resulta que γ_{2k} es una β-fórmula. Por tanto $\neq (t_1 \doteq s_1 \wedge \cdots t_k \doteq s_k) \in \Phi$ o $f(t_1, \cdots, t_k) \doteq f(s_1, \cdots, s_k) \in \Phi$. Hay 2 casos

- $\neq (t_1 \doteq s_1 \wedge \cdots t_k \doteq s_k) \in \Phi$. Esta es una β-fórmula. En este caso alguna de las fórmulas $\neg(t_i \doteq s_i) \in \Phi$ $(1 \leq i \leq k)$. Por otro lado $[t_i] = [s_i]$ significa $t_i \doteq s_i \in \Phi$. Pero al ser Φ de Hintikka, es coherente y no puede ser que $\neg(t_i \doteq s_i), t_i \doteq s_i \in \Phi$. Por tanto, $\neq (t_1 \doteq s_1 \wedge \cdots t_k \doteq s_k) \notin \Phi$

- $f(t_1, \cdots, t_k) \doteq f(s_1, \cdots, s_k) \in \Phi$. Esta es la definición de $f(t_1, \cdots, t_k) \equiv_\Phi f(s_1, \cdots, s_k$, por tanto $[f(t_1, \cdots, t_k)] = [f(s_1, \cdots, s_k)]$.

El caso de los símbolos de proposición es similar. Hay que tener en cuenta que en este caso hay que demostrar un **si y sólo si**. □

Ahora podemos definir el álgebra que vamos a usar como modelo

$$\mathfrak{T}_\Phi \;=\; \left\langle T_\Phi/\equiv_\Phi, \{[c] \mid c \in Ct_S\}, \{f^{\mathfrak{T}_\Phi} \mid f|_k \in \mathrm{Fn}_S\}, \{p^{\mathfrak{T}_\Phi} \mid p|_k \in \mathrm{Pd}_S\}\right\rangle$$

Para dar una interpretación, necesitamos una asignación de variables

$$\sigma_\Phi = \left\{ \begin{array}{ll} [x] & \text{si} x \in T_\Phi \\ \text{arbitrario} & \text{e.o.c} \end{array} \right\}$$

Tomamos ahora la interpretación

$$\mathfrak{I}_\Phi = \langle \mathfrak{T}_\Phi, \sigma_\Phi \rangle$$

A continuación demostraremos $\mathfrak{I}_\Phi \models \Phi$. En primer lugar tenemos

Proposición 11. *Sea S una signatura, $\Phi \subseteq FORM_S$ un conjunto de fórmulas de Hintikka y $t \in T_\Phi$. Entonces $\mathfrak{I}_\Phi = [t]$.*

Demostración. Se deja como ejercicio. Es sencilla por inducción estructural. □

Tenemos ahora que demostrar $\mathfrak{I} \models \varphi$ para cada $\varphi\Phi$. La demostración no se puede hacer por inducción estructural. Debemos extender la norma que vimos en lógica proposicional

Definición 43. Sea S una signatura. Definimos la normal de una fórmula

$$\| \bullet \| : \mathrm{FORM}_S \mapsto \mathbb{N}$$

de forma recursiva como sigue

Caso base $\|\varphi\| = 0$ si φ es atómica.

Casos recursivos

- $\|\neg\varphi\| = 1 + \|\varphi\|$
- $\|\varphi \wedge \psi\| = 1 + \|\varphi\| + \|\psi\|$
- $\|\varphi\psi\| = 1 + \|\varphi\| + \|\psi\|$
- $\|\varphi \to \psi\| = 2 + \|\varphi\| + \|\psi\|$
- $\|\varphi \leftrightarrow \psi\| = 5 + \|\varphi\| + \|\psi\|$
- $\|\forall x\, \varphi\| = 1 + \|\varphi\|$
- $\|\exists x\, \varphi\| = 1 + \|\varphi\|$

Observación.

- $\|\varphi \to \psi\| = \|\neg\varphi\psi\|$
- $\|\varphi \leftrightarrow \psi\| = \|(\varphi \to \psi) \wedge (\psi \to \varphi)\|$

Antes de continuar necesitamos varios resultados previos

Lema 9.

- *Si α es una α-fórmula,* $\|\alpha\| > \|\alpha_1\|$ *y* $\|\alpha\| > \|\alpha_2\|$

Si β es una β-fórmula, $\|\beta\| > \|\beta_1\|$ *y* $\|\beta\| > \|\beta_2\|$

Si σ es una σ-fórmula, $\|\sigma\| > \|\sigma\|$

Si $\varphi \in FORM_S$ es una fórmula y $t \in TERM_S$ y $x \in Var$,$\|\varphi[t/x]\| = \|\varphi\|$

Demostración. Las tres primeras están vistas en lógica proposicional. La última se hace por inducción estructural y queda como ejercicio. Intuitivamente la norma de una fórmula no depende de los términos involucrados. □

Lema 10. *Sea S una asignatura, $\Phi \subseteq FORM_S$ un conjunto defórmulas de Hintikka y $\varphi \in FORM_S$ una fórmula atómica. $\varphi \in \Phi$ si y sólo si $\mathfrak{I}_\Phi \models \varphi$.*

Demostración.

- $\varphi = \top$. Trivial
- $\varphi = t \doteq s$. En este caso $t, s \in \Phi$ si y sólo si $t \equiv_\Phi s$, que es lo mismo que $[t] = [s]$. Por la proposición (11) tenemos $t^{\mathfrak{I}_\Phi} = s^{\mathfrak{I}_\Phi}$. Por tanto $\mathfrak{I}_\Phi \models t \doteq s$.
- $\varphi = p(t_1, \cdots, t_k)$ para $p|_k \in \mathrm{Pd}_S$ y $t_1, \cdots, t_k \in T_\Phi$.
  Por definición $p(t_1, \cdots, t_k) \in \Phi$ si y sólo si $p^{\mathfrak{T}_\Phi}([t_1], \cdots, [t_k]) = V$. Por tanto tenemos
  $$p^{\mathfrak{T}_\Phi}([t_1], \cdots, [t_k]) = p^{\mathfrak{T}_\Phi}(t_1^{\mathfrak{I}_\Phi}, \cdots, t_k^{\mathfrak{I}_\Phi}) = (p(t_1, \cdots, t_k))^{\mathfrak{I}_\Phi}$$
  Puesto que $(p(t_1, \cdots, t_k))^{\mathfrak{I}_\Phi} = V$ es la definición de $\mathfrak{I}_\Phi \models p(t_1, \cdots, t_k) \in \Phi$, tenemos $p(t_1, \cdots, t_k) \in \Phi$ si y sólo si $\mathfrak{I}_\Phi \models p(t_1, \cdots, t_k) \in \Phi$.

□

Ahora podemos demostrar el resultado que buscamos.

Teorema 9. *Sea S una asignatura y $\Phi \subseteq FORM_S$ un conjunto de fórmulas de Hintikka, entonces $\mathfrak{I}_\Phi \models \Phi$.*

Demostración. Sea $\varphi \in \Phi$, probaremos $\mathfrak{I}_\Phi \models \varphi$ por inducción sobre $\|\varphi\|$.

$\|\varphi\| = 0$ Es una de las implicaciones del lema 43.

$\|\varphi\| > 0$ En ese caso tenemos varias opciones

- $\varphi = \alpha$ es una α-fórmula. Al ser Φ un conjunto α-saturado, tenemos $\alpha_1, \alpha_2 \in \Phi$. Puesto que $\|\alpha\| > \|\alpha_1\|$ y $\|\alpha\| > \|\alpha_2\|$ podemos aplicar hipótesis de inducción y tenemos $\mathfrak{I}_\Phi \models \alpha_1$ y $\mathfrak{I}_\Phi \models \alpha_2$. Puesto que $\varphi = \alpha \sim \alpha_1 \wedge \alpha_2$ tenemos $\mathfrak{I}_\Phi \models \varphi$.
- $\varphi = \beta$ es una β-fórmula.
- $\varphi = \sigma$ es una σ-fórmula. *(Se deja como ejercicio)*

- $\varphi = \gamma$ es una γ-fórmula. Hay 2 casos
  - $\varphi = \forall x\, \psi$. *(Se deja como ejercicio)*
  - $\varphi = \neg\exists x\, \psi$. Puesto que Φ es saturado tenemos que para todo término $t \in T_\Phi$ se verifica $\neg\psi[t/x] \in \Phi$. Por otro lado

$$\|\neg\forall x\, \psi\| = 2 + \|\psi\| = 1 + \|\neg\psi\| > \|\neg\psi\| = \|\neg\psi[t/x]\|$$

  Por podemos aplicar inducción a los términos $\neg\psi[t/x]$para $t \in T_\Phi$. Por lo que podemos decir que para cada $t \in T_\Phi$ tenemos $\mathfrak{I}_\Phi \models \neg\psi[t/x]$ y, aplicando el lema de sustitución, tenemos $\mathfrak{I}_\Phi[t^{\mathfrak{I}_\Phi}/x] \models \neg\psi$. Puesto que $t^{\mathfrak{I}_\Phi} = [t]$, tenemos que para cada $[t] \in T_\Phi/\equiv_\Phi$ se verifica$\mathfrak{I}_\Phi[[t]/x] \models \neg\psi$. Puesto que cada elemento de $t \in T_\Phi$ está en una y solo una clase de equivalencia, podemos decir que para cada $a \in T_\Phi/\equiv_\Phi$ se verifica $\mathfrak{I}_\Phi[a/x] \models \neg\psi]$, que es la definición de que $\mathfrak{I}_\Phi \models \forall x\, \neg\psi$. Puesto que $\forall x\, \neg\psi \sim \neg\exists x\, \psi$ tenemos el resultado

- $\varphi = \delta$ es una δ-fórmula. *(Se deja como ejercicio)*

$\square$

Referencias

[1] María Teresa. Hortalá González, Javier Leach Albert, Mario. Rodríguez Artalejo, and Mario. Rodríguez Artalejo. Matemática discreta y lógica matemática. Estudios complutenses. Editorial Complutense, Madrid, 1998.