Matemática Discreta

- Licenciatura en Informática – Ingeniería en Informática
- 1º año

Lógica de Predicados

¿Qué es la Lógica de Predicados?

En el capítulo anterior dijimos que la Lógica es una disciplina que se encarga del análisis de los razonamientos como objeto de estudio. Formaliza su escritura y establece técnicas para estudiar su validez.

La Lógica de predicados es una lógica más rica que la proposicional. Mantiene todas las propiedades de la lógica proposicional, pero agrega el estudio de los **elementos** sobre los cuales se razona.

El resultado es una lógica **más expresiva**, ya que además de lidiar con frases que expresan razonamientos (en PROP se llamaban proposiciones), también incorpora el estudio de los siguientes aspectos:

- Términos (representan los **elementos** sobre los que se razona)
- Relaciones y funciones (se aplican sobre los términos)
- Cuantificadores ($\forall$ y $\exists$, también se aplican sobre los términos)

<u>¿Qué es la Lógica de Predicados? (continuación)</u>:

A modo de ejemplo, considérense las siguientes frases que expresan razonamientos. En PROP, se representaban con **letras proposicionales**. Sin embargo, en predicados veremos que se expresan con más detalle.

Frase	Proposicional	Predicados
"*2 es un natural*"	**p**	$N(2)$
"*2 es un entero*"	**q**	$E(2)$
"*Todo natural es entero*"	**r**	$\forall x\,(N(x) \rightarrow E(x))$

En PROP, las letras proposicionales representaban frases **atómicas**, que no dejaban ver la naturaleza de los elementos sobre los que se razona.

En Predicados, las frases atómicas dan detalles sobre los elementos involucrados (las dos primeras muestran expresamente que 2 es un natural y un entero). Incluso frases que en PROP eran atómicas, no son atómicas en Predicados (tercer ejemplo).

¿Qué es la Lógica de Predicados? (continuación):

La Lógica de Predicados posee las mismas cuatro áreas de estudio que había en la Lógica Proposicional. La diferencia es que en vez de hablar de **proposiciones**, en predicados se habla de **fórmulas lógicas**.

- **Sintaxis**: Se encarga de definir las reglas necesarias para la correcta escritura de fórmulas lógicas que representen razonamientos.
- **Semántica**: Se encarga de establecer técnicas para estudiar la veracidad o falsedad de las fórmulas lógicas.
- **Deducción natural**: Se encarga de establecer técnicas para la correcta escritura de pruebas matemáticas basadas en fórmulas lógicas.
- **Completitud**: Se encarga de establecer un paralelismo (equivalencia) entre la semántica y la deducción natural.

En este capítulo veremos cada una de las cuatro áreas, y al igual que en PROP lo haremos siguiendo el mismo orden en que fueron presentadas.

UDE
UNIVERSIDAD DE LA EMPRESA

Sintaxis de la Lógica de Predicados:

Al igual que en Proposicional, la sintaxis se encarga de definir los símbolos necesarios para escribir fórmulas lógicas que expresen razonamientos.

Dado que en Predicados escribimos razonamientos sobre **elementos** concretos, lo primero que debemos definir es el **universo** que contiene a dichos elementos. Esto se hace mediante la definición de una **estructura**.

Estructura: Es una secuencia de símbolos que define el conjunto universal que contiene los elementos sobre los que se va a razonar. Toda estructura posee la siguiente forma:

$$\mathcal{M} = < U \,;\, R_1, \ldots, R_N \,;\, F_1, \ldots, F_M \,;\, c_1, \ldots, c_k >$$

- $\mathcal{M}$ es el nombre que le damos a la estructura.
- U es el **conjunto universal** que contiene a los elementos.
- $R_1, \ldots, R_N$ son **relaciones** entre los elementos de U.
- $F_1, \ldots, F_M$ son **funciones** sobre los elementos de U.
- $c_1, \ldots, c_k$ son algunos **valores concretos** de U.

UDE
UNIVERSIDAD DE LA EMPRESA

Sintaxis de la Lógica de Predicados (continuación):

La estructura contiene todo lo necesario para que más adelante podamos escribir fórmulas lógicas que expresen razonamientos sobre los elementos del universo. En PROP no definíamos ninguna estructura porque no interesaba identificar a los elementos sobre los que íbamos a razonar.

Cuando definimos una estructura, debemos definir con todo **detalle** cada uno de los componentes que la integran, como vemos a continuación:

Ejemplo de estructura:

Sea la siguiente estructura: $\mathcal{M} = \langle N ; P, M ; s, p, c; 0, 1, 2 \rangle$ donde:

- **N es el conjunto de los números naturales (el universo).**
- **$P = \{ x \in N \,/\, x \text{ es par} \}$ es la relación "ser par".**
- **$M = \{ (x, y) \in N \times N \,/\, x > y \}$ es la relación "mayor".**
- **$s: N \times N \rightarrow N \,/\, s(x, y) = x + y$ es la función "suma".**
- **$p: N \times N \rightarrow N \,/\, p(x, y) = x * y$ es la función "producto".**
- **$c: N \rightarrow N \,/\, c(x) = x^2$ es la función "cuadrado".**

UNIVERSIDAD DE LA EMPRESA

Sintaxis de la Lógica de Predicados (continuación):

Algunas observaciones acerca del ejemplo de estructura propuesto:

- Queda claramente establecido cuál es el conjunto universal (en este ejemplo es **N**, el conjunto de los naturales).
- Cada **relación** se expresa en forma de **conjunto**. En este ejemplo, son conjuntos **infinitos**, por lo que fueron definidos por **comprensión**.
- Cada **función** se expresa usualmente en **notación prefija**, aunque esto **no** es obligatorio. En el ejemplo tanto dominios como codominios son conjuntos **infinitos**, por lo que se usó la notación prefija.
- Tanto en las **relaciones** como en las **funciones** definidas en la estructura, el conjunto de base sobre el que se trabaja **siempre** debe ser el conjunto universo (en el ejemplo, los naturales).
- Los **nombres** que se dé a las relaciones y funciones en la estructura deben **coincidir** con los nombres usados luego al detallarlas.

Sintaxis de la Lógica de Predicados (continuación):

Veremos ahora un concepto estrechamente vinculado al de **estructura**, se trata del concepto de **tipo de similaridad**:

Tipo de similaridad: Dada una estructura, su tipo de similaridad es una secuencia de números que indica cuál es la **aridad** (unaria, binaria, etc.) de c/u de sus relaciones y funciones, junto con la cantidad de constantes.

El tipo de similaridad nos servirá luego para saber cómo refererirnos a las relaciones, funciones y constantes cuando escribamos **fórmulas lógicas**.

Ejemplo de tipo de similaridad:

Considere nuevamente la estructura: $\mathcal{M}$ = < N ; P, M ; s, p, c; 0, 1, 2 >

Su tipo de similaridad es: < 1, 2 ; 2, 2, 1 ; 3 > debido a que:

- **La relación P es unaria y la relación M es binaria.**
- **Las funciones s y p son binarias y la función c es unaria.**
- **Hay tres constantes definidas para la estructura.**

Sintaxis de la Lógica de Predicados (continuación):

Ahora que ya hemos definido la **estructura** y su **tipo de similaridad**, el siguiente paso es definir qué **nombres** vamos a usar para referirnos a las relaciones, funciones y constantes cuando escribamos **fórmulas lógicas**.

Es fundamental que cuando escribamos fórmulas lógicas usemos siempre los **mismos** nombres en todas las fórmulas que escribamos.

Si no unificamos los nombres, puede no quedar claro a qué nos estamos refiriendo. Por ejemplo, tres personas diferentes podrían escribir las siguientes fórmulas para expresar la **misma** afirmación:

- **∃x (Par (x))**
- **∃x (P (x))**
- **∃x (MultiploDe2 (x))**

Si bien en los tres casos queda claro lo que se trató de expresar, esto no siempre puede ser así. Haber usado tres nombres distintos para nombrar la relación “ser par” puede llegar a causar confusiones.

Sintaxis de la Lógica de Predicados (continuación):

Se llama **símbolos del alfabeto** a los nombres que usamos para llamar a las relaciones, funciones y constantes de una estructura. En Predicados, se acostumbra adoptar el siguiente criterio para definir los nombres:

- Nombres de **relaciones**: Letras castellanas **mayúsculas**.
- Nombres de **funciones**: Letras castellanas **minúsculas**.
- Nombres de **constantes**: Letras castellanas **minúsculas con subíndice**, o bien usar el propio valor de las constantes.

Ejemplo de símbolos del alfabeto:

Considere nuevamente la estructura: $\mathcal{M}$ = < N ; P, M ; s, p, c; 0, 1, 2 >

Definimos los siguientes símbolos del alfabeto:

- **P (relación "ser par"), M (relación "mayor")**
- **s (función "suma"), p (función "producto"), c (función "cuadrado"**
- **c_0, c_1, c_2 (para las constantes 0, 1, 2 respectivamente).**

Sintaxis de la Lógica de Predicados (continuación):

Ahora que ya tenemos definidos los símbolos del alfabeto, estamos en condiciones de empezar a escribir **términos** y **fórmulas lógicas**.

Los **términos** son los elementos del universo sobre los cuales haremos razonamientos. Las **fórmulas lógicas** son las frases que expresan dichos razonamientos. Por ejemplo: $\forall x (P(x) \rightarrow P(c(x)))$

La **fórmula** es la frase en azul, mientras que los **términos** están en rojo. Esta fórmula dice que: "***Para todo* x*, si* x *es par, entonces* x^2 *es par***". En esta frase estamos haciendo una afirmación sobre **x** y **c (x)**, que son los **términos**. La **fórmula** es toda la **frase** que expresa la afirmación.

En PROP **no** hacíamos la distinción entre términos y fórmulas porque lo único que escribíamos eran **proposiciones**. En Predicados sí hacemos la distinción, y veremos **reglas de pertenencia** para escribir correctamente tanto términos como fórmulas.

UNIVERSIDAD DE LA EMPRESA

Sintaxis de la Lógica de Predicados (continuación):

Se llama **TERM** al conjunto de todos los **términos** que existen. Es un conjunto infinito, y está definido por las siguientes **reglas de pertenencia**.

1. $c_n \in$ **TERM (siendo c_n cualquier símbolo de constante)**
2. $x \in$ **TERM (siendo x cualquier variable)**
3. **Si** $t_1, \ldots, t_M \in$ **TERM entonces** $F(t_1, \ldots, t_M) \in$ **TERM**
   (siendo F cualquier símbolo de función M-ario)

Observaciones:

- La regla 1 establece que cualquier símbolo de constante del alfabeto es por sí mismo un término (por ejemplo; **1**, **2**, c_2, c_3)
- La regla 2 establece que cualquier variable es por sí misma un término (por ejemplo: **x**, **y**, **z**)
- La regla 3 establece que cualquier función aplicada a otros términos es también un término (por ejemplo: **c (x)**, **s (1,2)**, **p (x,y)**)

Sintaxis de la Lógica de Predicados (continuación):

Se llama **FORM** al conjunto de todas las **fórmulas lógicas** que existen (tanto atómicas como no atómicas). Es un conjunto infinito, y está definido por las siguientes **reglas de pertenencia**.

1. $\perp \in$ **FORM**
2. **Si** $t_1, \ldots, t_N \in$ **TERM entonces** $R(t_1, \ldots, t_N) \in$ **FORM**
   (siendo R cualquier símbolo de relación N-ario)
3. **Si** $\alpha \in$ **FORM y** $\beta \in$ **FORM entonces** $(\alpha \wedge \beta) \in$ **FORM**
4. **Si** $\alpha \in$ **FORM y** $\beta \in$ **FORM entonces** $(\alpha \vee \beta) \in$ **FORM**
5. **Si** $\alpha \in$ **FORM y** $\beta \in$ **FORM entonces** $(\alpha \rightarrow \beta) \in$ **FORM**
6. **Si** $\alpha \in$ **FORM y** $\beta \in$ **FORM entonces** $(\alpha \leftrightarrow \beta) \in$ **FORM**
7. **Si** $\alpha \in$ **FORM entonces** $(\neg\alpha) \in$ **FORM**
8. **Si** $\alpha \in$ **FORM entonces** $\exists x (\alpha) \in$ **FORM (x variable cualquiera)**
9. **Si** $\alpha \in$ **FORM entonces** $\forall x (\alpha) \in$ **FORM (x variable cualquiera)**

Sintaxis de la Lógica de Predicados (continuación):

Algunas observaciones acerca de la definición de **FORM**:

- El **absurdo** ($\perp$) sigue estando presente en FORM.
- Las **conectivas binarias** ($\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$) y la conectiva unaria ($\neg$) siguen estando presentes en FORM y se utilizan igual que en PROP para construir fórmulas **no atómicas**.
- La regla 2 establece que cualquier **relación** aplicada a términos es una **fórmula atómica**. Por ejemplo: $P(x)$, $P(c(x))$, $M(2,1)$.
- Las reglas 8 y 9 establecen que cualquier fórmula puede ser precedida por un **cuantificador** aplicado a una variable cualquiera. Por ejemplo: $\exists x\,(P(x))$, $\forall x\,(P(x) \rightarrow P(c(x)))$.

Sintaxis de la Lógica de Predicados (continuación):

Ejemplos de fórmulas lógicas:

Considere nuevamente la estructura: $\mathcal{M} = < N ; P, M ; s, p, c; 0, 1, 2 >$ con los siguientes símbolos del alfabeto:

- **P (relación "ser par"), M (relación "mayor")**
- **s (función "suma"), p (función "producto"), c (función "cuadrado")**
- **c_0, c_1, c_2 (para las constantes 0, 1, 2 respectivamente).**

Utilizando <u>únicamente</u> los símbolos definidos, escriba una fórmula lógica que exprese cada una de las siguientes afirmaciones:

1) "***Existe un natural que es par y mayor que cero***"
2) "***Todo natural cumple que es par o no es par***"
3) "***La suma de todo natural consigo mismo es par***"
4) "***Para todo natural existe otro natural mayor que él***"
5) "***Todo natural cumple que si es mayor que cero, entonces su cuadrado también lo es***"

Sintaxis de la Lógica de Predicados (continuación):

Se llama **alcance de un cuantificador** a la zona de la fórmula sobre la que el cuantificador tiene alcance. Toda variable que ocurre dentro de la zona de alcance se dice que queda **ligada** al cuantificador. En caso de que no quedara ligada a ningún cuantificador, se dice que ocurre **libre**.

Ante ausencia de paréntesis, el cuantificador tiene **mayor precedencia** que cualquiera de las conectivas. En presencia de paréntesis, el alcance queda determinado por la ubicación de los paréntesis.

Ejemplos de alcance:

Para cada una de las siguientes fórmulas, determine cuál es el alcance de cada cuantificador. Señale cuáles ocurrencias de variables quedan ligadas y cuáles quedan libres.

1) $\exists x\,(P(x) \wedge M(x,c_0))$
2) $\exists x\,P(x) \wedge M(x,c_0)$
3) $\forall x\,(M(x,y) \rightarrow \exists y\,M(y,x))$

<u>Sintaxis de la Lógica de Predicados (continuación)</u>:

Se llama **fórmula cerrada o sentencia** a cualquier fórmula que **no** tenga ocurrencias de variables libres. Es decir, cualquier fórmula sin variables, o bien con todas sus ocurrencias de variables ligadas a cuantificadores.

Se llama **fórmula abierta** a cualquier fórmula que tenga al menos una ocurrencia de variable libre. Es decir, cualquier fórmula en la que al menos una ocurrencia de variable **no** queda ligada a ningún cuantificador.

<u>Ejemplos de fórmulas cerradas y abiertas</u>:

Considere nuevamente las tres fórmulas del ejemplo anterior. Para cada una de ellas, indique si se trata de una fórmula abierta o cerrada.

1) $\exists x \left(P(x) \wedge M(x, c_0) \right)$

2) $\exists x\, P(x) \wedge M(x, c_0)$

3) $\forall x \left(M(x, y) \rightarrow \exists y\, M(y, x) \right)$

Sintaxis de la Lógica de Predicados (continuación):

Veremos ahora una operación que será de utilidad más adelante en semántica y deducción natural: la **sustitución de términos**.

Sean un término $t \in$ TERM, una fórmula $\alpha \in$ FORM y una variable x. $\alpha[t / x]$ denota la fórmula que resulta de sustituir (remplazar) en α cada ocurrencia **libre** de la variable x por el término t. Las ocurrencias **ligadas** de la variable x **no** se sustituyen.

Para sustituir, hay que recorrer α de modo que cada vez que se encuentre una ocurrencia **libre** de la variable x, se remplaza dicha ocurrencia por t.

Ejemplos de sustituciones:

1) **Sean $\alpha = \forall x\, P(x) \leftrightarrow M(x,y)$, $t = c(2)$. Calcular $\alpha[t / x]$**
2) **Sean $\beta = P(y) \rightarrow \exists y\, M(x,y)$, $t = s(y,z)$. Calcular $\beta[t / y]$**
3) **Sean $\delta = P(x) \wedge \exists y\, M(x,y)$, $t = s(x,y)$. Calcular $\delta[t / x]$**

UNIVERSIDAD DE LA EMPRESA

Sintaxis de la Lógica de Predicados (continuación):

Decimos que un término t es **libre para una variable** x en una fórmula α cuando, tras realizar la sustitución $\alpha[t / x]$, todas las variables de t siguen estando libres.

Si alguna variable de t quedó ligada tras la sustitución, entonces t **no** es libre para x en α. En tal caso, la sustitución es **inválida** y no debe hacerse.

De ahora en más, solamente permitiremos hacer la sustitución cuando el término **efectivamente** sea libre para la variable en la fórmula.

Ejemplos de términos libres para variables:

1) Sean nuevamente $\alpha = \forall x\, P(x) \leftrightarrow M(x,y)$, $t = c(2)$ y $\alpha[t / x]$

2) Sean nuevamente $\beta = P(y) \rightarrow \exists y\, M(x,y)$, $t = s(y,z)$ y $\beta[t / y]$

3) Sean nuevamente $\delta = P(x) \wedge \exists y\, M(x,y)$, $t = s(x,y)$ y $\delta[t / x]$

¿Son válidas las tres sustituciones realizadas? Explicar porqué.

UNIVERSIDAD DE LA EMPRESA

<u>Semántica de la Lógica de Predicados</u>:

Es la segunda área de estudio de la lógica de predicados. Se encarga de definir técnicas para estudiar la veracidad o falsedad de las fórmulas.

Como en PROP, se utilizan **valores de verdad** (0 y 1) para representar la veracidad o falsedad de una fórmula. Sin embargo, en Predicados el valor de verdad de una fórmula depende de la **estructura** en la cual se la lea.

<u>Ejemplo introductorio</u>:

Dado el tipo de similaridad < 1,1; – ;1 > con los siguientes símbolos de relación: A (unaria), B (unaria) y la fórmula $\alpha = \exists x\,(A(x) \wedge B(x))$.

1) **Sea $\mathcal{M}$ = < N ; {x $\in$ N / x es impar} ; {x $\in$ N / x es primo} ; – ; 3 >**
   ¿Qué significa α en la estructura $\mathcal{M}$? ¿Es verdadera o falsa en $\mathcal{M}$?

2) **Sea $\mathcal{N}$ = < R ; {x $\in$ R / x > 0} ; {x $\in$ R / x < 0} ; – ; 0 >**
   ¿Qué significa α en la estructura $\mathcal{N}$? ¿Es verdadera o falsa en $\mathcal{N}$?

Semántica de la Lógica de Predicados (continuación):

En Predicados, puede suceder que una **misma** fórmula sea **verdadera** en una estructura, pero **falsa** en otra, dependiendo del significado que se dé a la fórmula en una u otra estructura.

Por lo tanto, la noción de **valuación** en Lógica de Predicados está **atada** a la estructura en la cual se interprete la fórmula. Veremos que la definición de **valuación** en Predicados se realiza en el contexto de una **estructura**.

Observaciones:

- Cuando leemos una fórmula en una estructura determinada, se dice que **interpretamos** el significado de la fórmula en dicha estructura.
- En Lógica de Predicados, la noción de valuación se define **únicamente** para las **sentencias** (fórmulas cerradas). Es decir, solamente para las fórmulas en las que **no** hay ocurrencias de variables libres.

UDE UNIVERSIDAD DE LA EMPRESA

Semántica de la Lógica de Predicados (continuación):

Se llama **valuación en una estructura** $\mathcal{M}$ a una función $v^{\mathcal{M}}$ **:** SENT $\rightarrow \{0,1\}$ que permite calcular el valor de verdad de una **sentencia** (fórmula cerrada) en la estructura $\mathcal{M}$. Cumple las siguientes propiedades:

1. $v^{\mathcal{M}}(\bot) = 0$
2. $v^{\mathcal{M}}(R(t_1, \ldots, t_N)) = \begin{cases} 1 \text{ si } t_1^{\mathcal{M}}, \ldots, t_N^{\mathcal{M}} \text{ satisfacen la relación R en } \mathcal{M} \\ 0 \text{ si } t_1^{\mathcal{M}}, \ldots, t_N^{\mathcal{M}} \textbf{ no} \text{ satisfacen la relación R en } \mathcal{M} \end{cases}$
3. $v^{\mathcal{M}}(\alpha \wedge \beta) = \min \{ v^{\mathcal{M}}(\alpha), v^{\mathcal{M}}(\beta) \}$
4. $v^{\mathcal{M}}(\alpha \vee \beta) = \max \{ v^{\mathcal{M}}(\alpha), v^{\mathcal{M}}(\beta) \}$
5. $v^{\mathcal{M}}(\alpha \rightarrow \beta) = \max \{ 1 - v^{\mathcal{M}}(\alpha), v^{\mathcal{M}}(\beta) \}$
6. $v^{\mathcal{M}}(\alpha \leftrightarrow \beta) = 1 \Leftrightarrow v^{\mathcal{M}}(\alpha) = v^{\mathcal{M}}(\beta)$, en otro caso es 0.
7. $v^{\mathcal{M}}(\neg\alpha) = 1 - v^{\mathcal{M}}(\alpha)$
8. $v^{\mathcal{M}}(\exists \mathbf{x}\, \alpha) = 1 \Leftrightarrow$ existe algún $a \in U$ / $v^{\mathcal{M}}(\alpha[a/x]) = 1$, en otro caso es 0.
9. $v^{\mathcal{M}}(\forall \mathbf{x}\, \alpha) = 1 \Leftrightarrow$ para todo $a \in U$, $v^{\mathcal{M}}(\alpha[a/x]) = 1$, en otro caso es 0.

Obs: $t_1^{\mathcal{M}}, \ldots, t_N^{\mathcal{M}}$ denotan los valores de los términos en la estructura $\mathcal{M}$.

Semántica de la Lógica de Predicados (continuación):

Vemos ahora un concepto estrechamente ligado a la noción de **valuación**, se trata del concepto de **modelo**. Se dice que una estructura $\mathcal{M}$ es **modelo** de una sentencia α si y sólo si α es verdadera en $\mathcal{M}$. La notación es:

$$\mathcal{M} \models \alpha \iff v^{\mathcal{M}}(\alpha) = 1$$

En resumen, decir que una estructura es **modelo** de una fórmula es sinónimo de decir que la fórmula es **verdadera** en dicha estructura.

Ejemplos de valuación y modelo:

Considere nuevamente la fórmula $\alpha = \exists x\,(A(x) \wedge B(x))$ y las estructuras $\mathcal{M}$ y $\mathcal{N}$ del ejemplo introductorio:

$\mathcal{M}$ **= < N ; {x $\in$ N / x es impar} ; {x $\in$ N / x es primo} ; – ; 3 >**

$\mathcal{N}$ **= < R ; {x $\in$ R / x > 0} ; {x $\in$ R / x < 0} ; – ; 0 >**

Conteste las siguientes preguntas aplicando las definiciones de valuación y modelo: ¿Se cumple que $\mathcal{M} \models \alpha$? ¿Se cumple que $\mathcal{N} \models \alpha$?

Semántica de la Lógica de Predicados (continuación):

Las siguientes definiciones son idénticas a PROP, con la diferencia que ahora involucran **sentencias** y **estructuras** en vez de **proposiciones**.

- **Tautología**: Sea $\alpha \in$ SENT una sentencia. Decimos que α es una tautología $\Leftrightarrow$ Toda estructura $\mathcal{M}$ cumple que que $v^{\mathcal{M}}(\alpha) = 1$.

- **Contradicción**: Sea $\alpha \in$ SENT una sentencia. Decimos que α es una contradicción $\Leftrightarrow$ Toda estructura $\mathcal{M}$ cumple que que $v^{\mathcal{M}}(\alpha) = 0$.

- **Contingencia**: Sea $\alpha \in$ SENT una sentencia. Decimos que α es una contingencia $\Leftrightarrow$ α no es tautología ni contradicción.

Ejemplos de tautología, contradicción, contingencia:

Considere nuevamente el tipo de similaridad < 1,1; – ; 1 > con los símbolos de relación A y B (unarios).

1) $\alpha = \exists x (A(x) \wedge B(x))$ **¿Se cumple que** $\models \alpha$**? Justifique.**
2) $\beta = \forall x (A(x) \vee \neg A(x))$ **¿Se cumple que** $\models \beta$**? Justifique.**

UDE
UNIVERSIDAD DE LA EMPRESA

Semántica de la Lógica de Predicados (continuación):

La siguiente definición también es idéntica a PROP, con la diferencia que ahora involucra **sentencias** y **estructuras** en vez de **proposiciones**.

Consecuencia lógica: Sea $\Gamma \subseteq$ SENT un conjunto de sentencias y sea $\alpha \in$ SENT una sentencia. Se cumple que α es consecuencia lógica de Γ $\Leftrightarrow$ para toda estructura $\mathcal{M}$ tal que $v^{\mathcal{M}}(\Gamma) = 1$ se cumple que $v^{\mathcal{M}}(\alpha) = 1$.

Notación:

- La notación $v^{\mathcal{M}}(\Gamma) = 1$ denota que para toda $\delta \in \Gamma$ se cumple $v^{\mathcal{M}}(\delta) = 1$.
- La notación $\Gamma \models \alpha$ denota que α es consecuencia lógica de Γ.
- La notación $\Gamma \not\models \alpha$ denota que α **no** es consecuencia lógica de Γ.

Ejemplos de consecuencia lógica:

Considere nuevamente el tipo de similaridad < 1,1; – ; 1 > con los símbolos de relación A y B (unarios). Sean $\alpha = \forall x\, A(x)$, $\beta = \exists x\, A(x)$

1) **¿Se cumple que $\alpha \models \beta$? Justifique.**
2) **¿Se cumple que $\beta \models \alpha$? Justifique.**

Semántica de la Lógica de Predicados (continuación):

El siguiente **algoritmo** determina si α es o no consecuencia lógica de Γ:

Entrada: Conjunto de sentencias Γ, sentencia α.

Salida: Indicación de si α es o no consecuencia lógica de Γ.

Método:

¿Hay al menos una estructura $\mathcal{M}$ tal que $v^{\mathcal{M}}(\Gamma) = 1$?

Si no hay ninguna **entonces**

α **es** consecuencia lógica de Γ (o sea, $\Gamma \models \alpha$)

Sino

¿Hay alguna estructura $\mathcal{M}$ tal que $v^{\mathcal{M}}(\Gamma) = 1$ y $v^{\mathcal{M}}(\alpha) = 0$?

Si la hay **entonces**

α **no es** consecuencia lógica de Γ (o sea, $\Gamma \not\models \alpha$)

Sino

α **es** consecuencia lógica de Γ (o sea, $\Gamma \models \alpha$)

Fin

Fin

Deducción Natural de la Lógica de Predicados:

Es la tercera área de estudio de la Lógica de Predicados. Como en PROP, se encarga de establecer técnicas para la correcta escritura de pruebas matemáticas basadas en fórmulas lógicas.

Las 15 reglas de derivación de la Lógica Proposicional **siguen valiendo** en predicados. Se agregan además 4 **nuevas reglas** que veremos en este capítulo, y que involucran el uso de **cuantificadores**:

eliminación de $\forall$	**introducción de $\exists$**
introducción de $\forall$	**eliminación de $\exists$**

Consecuencia sintáctica: Sea $\Gamma \subseteq$ SENT un conjunto de sentencias y sea $\alpha \in$ SENT una sentencia. Se dice que α es consecuencia sintáctica de $\Gamma \Leftrightarrow$ existe un árbol de derivación D tal que $H(D) \subseteq \Gamma$ y $C(D) = \alpha$.

Observación: Esta definición nuevamente es idéntica a PROP, con la diferencia que ahora involucra **sentencias** en vez de **proposiciones**.

<u>Deducción Natural de la Lógica de Predicados (continuación)</u>:

<u>Eliminación de ∀</u>: Esta regla dice que si hemos concluido $\forall x\,\alpha$, entonces podemos concluir $\alpha\,[t/x]$, siendo t un término **libre** para x en α. El término t debe elegirse al momento de aplicar la regla, de modo que sirva al enunciado que se quiere probar.

$$\dfrac{\begin{array}{c} D \\ \forall x\,\alpha \end{array}}{\alpha\,[t/x]}\;\forall E$$

(*) t es libre para x en α

<u>Introducción de ∀</u>: Esta regla dice que si hemos concluido α, entonces podemos concluir $\forall x\,\alpha$, siempre y cuando la variable x **no ocurra libre** en las hipótesis de la rama que hasta el momento permanecen **sin** cancelar.

$$\dfrac{\begin{array}{c} D \\ \alpha \end{array}}{\forall x\,\alpha}\;\forall I\ (*)$$

(*) $x \notin FV(H(D))$

<u>Observación</u>: La notación $FV(H(D))$ quiere decir: "conjunto de variables que ocurren **libres** en las hipótesis de la derivación D"

Deducción Natural de la Lógica de Predicados (continuación):

Introducción de ∃: Esta regla dice que si hemos concluido $\alpha[t/x]$, siendo t un término **libre** para x en α, entonces podemos concluir $\exists x\,\alpha$. Como en la eliminación de ∀, el término se elige convenientemente al aplicar la regla.

$$\dfrac{\begin{array}{c}\nabla D\\ \alpha[t/x]\end{array}}{\exists x\,\alpha}\ \exists I\ (*)$$

(*) t es libre para x en α

Eliminación de ∃: Esta regla dice que si hemos concluido $\exists x\,\alpha$, y que partiendo de α logramos concluir δ, entonces la conclusión final es δ. Esta regla exige cancelar la hipótesis α en la 2º rama.

$$\dfrac{\begin{array}{c}\nabla D\\ \exists x\,\alpha\end{array}\qquad \begin{array}{c}\alpha^{(1)}\\ \nabla D\\ \delta\end{array}}{\delta}\ \exists E\ (*)\ (1)$$

(*) $x \notin FV(H(D_2) - \{\alpha\})$, $x \notin FV(\delta)$

La regla exige realizar **dos** chequeos:

1) La variable x **no puede** ocurrir libre en las hip. de la 2º rama, **excepto** α.

2) La variable x **no puede** ocurrir libre en la conclusión δ.

UNIVERSIDAD DE LA EMPRESA

Deducción Natural de la Lógica de Predicados (continuación):

Mostramos a continuación cuatro ejemplos de derivaciones en los cuales hay algunas reglas que están **mal aplicadas**:

1)

$$\dfrac{\dfrac{P(x) \qquad\qquad Q(x)}{P(x) \wedge Q(x)}\;(\wedge\, I)}{\forall x\,(P(x) \wedge Q(x))}\;(\forall\, I)$$

2)

$$\dfrac{\exists x\,(P(x) \wedge Q(x)) \qquad\qquad \dfrac{\overset{(1)}{\cancel{P(x) \wedge Q(x)}}}{Q(x)}\;(\wedge\, E)}{Q(x)}\;(\exists\, E)\;(1)$$

3)

$$\dfrac{\forall x\, \exists y\, R(x,y)}{\exists y\, R(y,y)}\;(\forall\, E)$$

4)

$$\dfrac{\exists x\, P(x) \qquad\qquad \dfrac{\dfrac{\overset{(2)}{\cancel{P(x)}} \qquad\qquad Q(x)}{P(x) \wedge Q(x)}\;(\wedge\, I)}{\exists x\,(P(x) \wedge Q(x))}\;(\exists\, I)}{\exists x\,(P(x) \wedge Q(x))}\;(\exists\, E)\;(2)$$

¿Qué reglas están mal aplicadas en estos ejemplos y porqué?

UNIVERSIDAD DE LA EMPRESA

Deducción Natural de la Lógica de Predicados (continuación):

Ejemplos de derivaciones:

En los ejemplos siguientes, construya derivaciones que demuestren las consecuencias sintácticas propuestas. Justifique el cumplimiento de las restricciones sobre las variables al aplicar las reglas.

1) **Sea el tipo de similaridad < 1,1; – ; 1 > con símbolos de relación P y Q (unarios) y símbolo de constante c_1:**

   a. $\forall x\, P(x),\ \forall x\, (P(x) \rightarrow Q(x)) \vdash Q(c_1)$

   b. $P(c_1) \wedge \forall x\, Q(x) \vdash \exists x\, (P(x) \wedge Q(x))$

2) **Sean α, β fórmulas tales que la única variable que ocurre libre es x.**

   a. $\forall x\, \alpha \wedge \forall x\, \beta \vdash \forall x\, (\alpha \wedge \beta)$

   b. $\exists x\, (\alpha \rightarrow \beta),\ \forall x\, \alpha \vdash \exists x\, \beta$

<u>Completitud de la Lógica de Predicados</u>:

Es la cuarta y última área de estudio de la lógica de predicados. Como en PROP, establece que las nociones de **consecuencia lógica** (semántica) y **consecuencia sintáctica** (deducción natural) son equivalentes.

Lo anterior significa que, si una proposición α es **consecuencia lógica** de un conjunto de sentencias Γ, entonces necesariamente α debe también ser **consecuencia sintáctica** de Γ (y viceversa).

Nuevamente, como en PROP, esto se resume en el siguiente enunciado:

<u>Teorema de completitud</u>: Sea $\Gamma \subseteq$ SENT un conjunto de sentencias y sea $\alpha \in$ SENT una sentencia. Se cumple entonces que:

$$\Gamma \models \alpha \quad \Leftrightarrow \quad \Gamma \vdash \alpha$$

Completitud de la Lógica de Predicados (continuación):

Al igual que en Proposicional, otros resultados vinculados al área de completitud, también se mantienen en Lógica de Predicados, nuevamente cambiando **proposiciones** por **sentencias**.

Conjunto consistente: Sea $\Gamma \subseteq$ SENT un conjunto de sentencias. Se dice que Γ es consistente $\Leftrightarrow \Gamma \nvdash \perp$. Es decir, si **no** es posible derivar el absurdo partiendo de hipótesis contenidas en Γ. En otro caso, Γ es **inconsistente**.

Condición necesaria y suficiente de consistencia: Como en PROP, este resultado constituye una alternativa para probar la consistencia de un conjunto de **sentencias**. Su enunciado es el siguiente:

Sea $\Gamma \subseteq$ SENT un conjunto de sentencias. Se cumple que:

Γ es consistente $\Leftrightarrow$ Existe una estructura $\mathcal{M}$ tal que $v^{\mathcal{M}}(\Gamma) = 1$.

La notación $v^{\mathcal{M}}(\Gamma) = 1$ denota que para toda $\delta \in \Gamma$ se cumple $v^{\mathcal{M}}(\delta) = 1$.

Completitud de la Lógica de Predicados (continuación):

- Para demostrar que un conjunto Γ es **consistente**, hay que encontrar una **estructura** $\mathcal{M}$ que haga verdaderas a **todas** las fórmulas de Γ. (es decir, aplicar la CN y S de consistencia).

- Para demostrar que un conjunto Γ es **inconsistente**, hay que construir una **derivación** que, partiendo de hipótesis en Γ, concluya el **absurdo** (es decir, aplicar la definición de conjunto inconsistente).

Ejemplos de consistencia:

Sea el tipo de similaridad <1; – ; 1> con símbolo de relación P (unario) y símbolo de constante c_1. Determine si los siguientes conjuntos son consistentes o inconsistentes, justificando apropiadamente en c/u:

1) $\Gamma = \{ \forall x\, P(x) , \forall x \neg P(x) \}$

2) $\Delta = \{ \exists x\, P(x) , \exists x \neg P(x) \}$