Lógica de Primer Orden: Semántica de la Lógica de Primer Orden
Lógica Computacional 2017-2, Nota de clase 4

Favio Ezequiel Miranda Perea Araceli Liliana Reyes Cabello
Lourdes Del Carmen González Huesca Pilar Selene Linares Arévalo

13 de marzo de 2017
Facultad de Ciencias UNAM

Una vez que hemos estudiado a detalle la sintaxis de los lenguajes de primer orden estudiaremos su semántica, la cual es más complicada que en el caso de la lógica de proposiciones debido a la expresividad de la lógica, pues contamos con la presencia de objetos de dos categorías ajenas pero íntimamente relacionadas, a saber la categoría de individuos representada sintácticamente por los términos y la categoría de propiedades o relaciones entre individuos, representada por los predicados. Más aún, la verdad de una fórmula en la lógica de primer orden no queda fija al determinar una interpretación para el lenguaje, como en el caso de las proposiciones, sino que depende de los valores de los términos los cuales a su vez dependen del universo de discurso en cuestión. Dicho universo será un conjunto no vacío, al que tendremos que referirnos para dar significado a las fórmulas. Finalmente observamos que una fórmula de la lógica de primer orden no es necesariamente falsa o verdadera en un mundo dado, contrario a lo que sucedía en la lógica de proposiciones. En particular el hecho de que una fórmula no sea verdadera no implica que sea falsa en general.

Antes de iniciar el estudio de la semántica formal veremos algunos ejemplos de manera informal con números naturales y gráficas.

1. Semántica Informal para $\mathbb{N}$

Considérese el lenguaje $\mathcal{L} = \{P, f, g, a, b, c, d\}$ y el universo $\mathbb{N}$ de los números naturales, con el siguiente significado (interpretación) de los símbolos:

$$P =_{def} < \quad f =_{def} + \quad g =_{def} \cdot \quad a =_{def} 2 \quad b =_{def} 3 \quad c =_{def} 5 \quad d =_{def} 0 \tag{1}$$

Es decir P simboliza a la relación de orden $<$, f a la función suma, g a la función producto y a, b, c, d son constantes que simbolizan a los números $2, 3, 5, 0$ respectivamente.

¿Cuando una fórmula del lenguaje dado es verdadera en $\mathbb{N}$? veamos algunos ejemplos:

1. $P(b, c)$ significa
$$3 < 5$$
lo cual es verdadero en $\mathbb{N}$.

2. $P(f(a, a), b)$ significa
$$2 + 2 < 3$$

lo cual es falso en $\mathbb{N}$.

3. $\forall x \exists y P(x,y)$ lo cual significa[1]
$$\forall x \in \mathbb{N}\, \exists y \in \mathbb{N}(x < y)$$
que es verdadero en $\mathbb{N}$.

4. $\exists y \forall x P(x,y)$ significa
$$\exists y \in \mathbb{N}\, \forall x \in \mathbb{N}\,(x < y)$$
que es falso en $\mathbb{N}$.

5. $\forall x \exists y P(y,x)$ significa
$$\forall x \in \mathbb{N}\, \exists y \in \mathbb{N}\,(y < x)$$
que es falso en $\mathbb{N}$.

6. $\forall x \big(x \neq d \to \exists y P(y,x)\big)$ significa
$$\forall x \in \mathbb{N}\,\big(x \neq 0 \Rightarrow \exists y \in \mathbb{N}\,(y < x)\big)$$
que es verdadero en $\mathbb{N}$.

7. $\forall x \big(P(d,x) \to \exists y (g(y,y) = x)\big)$ significa
$$\forall x \in \mathbb{N}\big(0 < x \Rightarrow \exists y \in \mathbb{N}\,(y \cdot y = x)\big)$$
que es falso en $\mathbb{N}$.

8. $\forall x \exists y \big(g(g(y,y),y) = x\big)$ significa
$$\forall x \in \mathbb{N}\, \exists y \in \mathbb{N}\big((y \cdot y) \cdot y = x\big)$$
que es falso en $\mathbb{N}$.

9. $\forall x \forall y \big(P(x,y) \to \exists z (P(x,z) \wedge P(z,y))\big)$ significa
$$\forall x \in \mathbb{N}\, \forall y \in \mathbb{N}\,\big(x < y \Rightarrow \exists z \in \mathbb{N}\,(x < z \;\&\; z < y)\big)$$
que es falso en $\mathbb{N}$.

Podemos observar que en cada una de las fórmulas anteriores fue posible decidir si la fórmula es falsa o verdadera, ¿Qué otra cosa tienen en común todas las fórmulas anteriores?

Considérese ahora la fórmula $P(x,y)$ ¿Es esta fórmula verdadera o falsa en $\mathbb{N}$?, Con la información dada hasta ahora resulta imposible saberlo puesto que no conocemos de antemano el valor de las variables libres x, y. De hecho la fórmula puede tomar un valor de verdad distinto dependiendo del estado de las variables x, y. Por ejemplo:

- En el estado $\{x \mapsto 2,\ y \mapsto 6\}$ entonces $P(x,y)$ significa $2 < 6$ lo cual es verdadero en $\mathbb{N}$.
- En el estado $\{x \mapsto 8,\ y \mapsto 3\}$ entonces $P(x,y)$ significa $8 < 3$ lo cual es falso en $\mathbb{N}$.

[1] Utilizamos para los significados una mezcla de relaciones, funciones y números con un lenguaje lógico informal con símbolos $\forall$, $\exists$, $\Rightarrow$, etcétera.

De lo anterior se concluye que una fórmula con variables libres NO puede calificarse como verdadera o falsa en general sino que su valor depende del valor particular que tomen las variables libres, estas fórmulas reciben una noción más debil de verdad llamada *satisfacibilidad*. La fórmula será *satisfacible* si algunos valores de sus variables libres la satisfacen, en cuyo caso se dice que la fórmula es satisfacible en o con los valores dados. Si en ciertos valores dados la evaluación resultante es falsa, decimos que la fórmula es no satisfacible o insatisfacible en o con los valores dados.

En el ejemplo anterior se tiene que $P(x, y)$ es satisfacible, en particular es satisfacible en el estado $\{x \mapsto 2, y \mapsto 6\}$ y en el estado $\{x \mapsto 45, y \mapsto 67\}$ pero no es satisfacible en el estado $\{x \mapsto 8, y \mapsto 3\}$ ni en el estado $\{x \mapsto 77, y \mapsto 23\}$.

Veamos otros ejemplos:

1. Sea $D(x, y) =_{def} \exists z(g(x, z) = y)$. Está fórmula significa que

$$\exists z \in \mathbb{N}\, (x \cdot z = y)$$

es decir, x divide a y o $x|y$ en símbolos usuales de la teoría de números.
Es claro que $D(x, y)$ es satisfacible en el estado $\{x \mapsto n, y \mapsto m\}$ si y sólo si $n|m$.

2. La propiedad *"x es par"*, $\exists y \in \mathbb{N}\, (2 \cdot y = x)$, está definida por la fórmula

$$E(x) =_{def} \exists y(g(a, y) = x)$$

que es satisfacible en cualquier estado $\{x \mapsto 2n\}$ e insatisfacible para cualquier estado $\{x \mapsto 2n+1\}$ con $n \in \mathbb{N}$.

3. La relación *"x es primo"* se define con la fórmula

$$Pr(x) =_{def} P(e, x) \land \forall y(D(y, x) \to y = e \lor y = x))$$

donde la constante e representa al 1 y $D(y, x)$ define la relación $y|x$. La fórmula $Pr(a)$ es cerrada por lo tanto podemos decidir su valor de verdad: dado que a significa 2 que es primo, entonces $Pr(a)$ es verdadera. Por otro lado $Pr(x)$ es insatisfacible para el estado $\{x \mapsto 9\}$ y satisfacible en el estado $\{x \mapsto 19\}$.

2. Semántica Informal para Gráficas

Recordemos que una gráfica finita (grafo) consiste de un conjunto finito no vacío V de vértices y un conjunto de aristas E cuyos elementos son parejas de vértices $uv = vu$ si la gráfica es no dirigida o $\langle u, v\rangle \neq \langle v.u\rangle$ si la gráfica es dirigida. Para representar gráfica podemos usar el siguiente lenguaje $\mathcal{L} = \{R^{(2)}\}$ que consta de un sólo símbolo de predicado binario.
Dada una gráfica particular $\mathcal{G} = \langle V, E\rangle$ interpretamos al símbolo R como E, es decir $R(x, y)$ será satisfacible en $\mathcal{G}$ en el estado $\{x \mapsto a, y \mapsto b\}$ si y sólo si $a, b \in V$ son vértices tales que $ab \in E$ (o $\langle a, b\rangle \in E$, si la gráfica es dirigida)

Sean

$$\mathcal{G}_1 = \Big\langle \{a, b, c, d\}, \{\langle a, a\rangle, \langle a, d\rangle, \langle d, b\rangle, \langle d, c\rangle, \langle b, b\rangle, \langle b, c\rangle, \langle c, b\rangle\} \Big\rangle$$

$$\mathcal{G}_2 = \Big\langle \{1, 2, 3, 4, 5\}, \{12, 23, 45, 52, 53\} \Big\rangle$$

$\mathcal{G}_1$ representa a una gráfica dirigida y $\mathcal{G}_2$ a una gráfica no dirigida, ambas con cinco vértices. Como ayuda el lector puede dibujar las gráfica representadas por las interpretaciones dadas.

Analicemos la verdad de los siguientes enunciados en ambas gráfica:

1. $\forall x R(x,x)$. La relación de adyacencia es reflexiva. Falso para $\mathcal{G}_1, \mathcal{G}_2$
2. $\forall x \neg R(x,x)$ La relación de adyacencia es antirreflexiva. Falso para $\mathcal{G}_1$, verdadero para $\mathcal{G}_2$.
3. $\forall x \forall y (R(x,y) \to R(y,x))$. La relación de adyacencia es simétrica. Falso para $\mathcal{G}_1$, verdadero para $\mathcal{G}_2$.
4. $\exists x \exists y R(x,y)$. La gráfica no es discreta, es decir, hay una arista. Verdadero para $\mathcal{G}_1, \mathcal{G}_2$.
5. $\forall x \forall y R(x,y)$. Cualesquiera dos vértices son adyacentes, es decir, la gráfica es completa. Falso para $\mathcal{G}_1, \mathcal{G}_2$.
6. $\forall x \exists y R(x,y)$. No hay vértices aislados. Verdadero para $\mathcal{G}_1, \mathcal{G}_2$.
7. $\exists y \forall x R(x,y)$. Hay un vértice adyacente a todos. Falso para $\mathcal{G}_1, \mathcal{G}_2$.
8. $\exists x \exists y (R(x,y) \wedge R(y,x))$ Hay vértices incidentes en ambas direcciones. Verdadero para $\mathcal{G}_1, \mathcal{G}_2$.
9. $\exists x \exists y \exists z (R(x,y) \wedge R(x,z))$ Hay un vértice incidente con al menos otro vértice (puesto que la fórmula no dice que y y z sean distintos). Es decir, hay un vértice cuyo grado es al menos 1. Verdadero para $\mathcal{G}_1, \mathcal{G}_2$.
10. $\exists x \exists y \exists z (y \neq z \wedge R(x,y) \wedge R(x,z))$. Hay un vértice x adyacente a vértices distintos y, z. Es decir, hay un vértice cuyo grado es al menos dos. Verdadero para $\mathcal{G}_1, \mathcal{G}_2$.
11. $\exists x \exists y \exists z (R(x,y) \wedge R(y,z) \wedge R(z,x))$. Hay vértices x, y, z tales que las aristas $(x,y), (y,z), (z,x)$ están presentes. Verdadero para $\mathcal{G}_1, \mathcal{G}_2$.

Veamos algunos ejemplos de satisfacibilidad con las siguientes fórmulas:

1. $R(x,y)$. x es adyacente a y. Satisfacible en $\mathcal{G}_1$ en el estado $\{x, y \mapsto a, d\}$. Insatisfacible en $\mathcal{G}_2$ en el estado $\{x, y \mapsto 1, 5\}$.
2. $R(x,y) \wedge R(x,z)$. x es adyacente a y y a z. Satisfacible en $\mathcal{G}_1$ con $\{x, y, z \mapsto d, b, c\}$ y con $\{x, y, z \mapsto a, a, d\}$. Satisfacible en $\mathcal{G}_2$ con $\{x, y, z \mapsto 2, 1, 3\}$. Insatisfacible en $\mathcal{G}_2$ con $\{x, y, z \mapsto 4, 5, 3\}$.
3. $R(x,y) \to R(y,x)$. Si x es adyacente a y entonces y es adyacente a x. Insatisfacible en $\mathcal{G}_1$ con $\{x, y \mapsto d, c\}$. Satisfacible en $\mathcal{G}_1$ con $\{x, y \mapsto b, c\}$. Satisfacible en $\mathcal{G}_2$ con cualquier valor de x, y.
4. $R(x,y) \to \neg R(y,x)$. Si x es adyacente a y entonces y no es adyacente a x. Satisfacible en $\mathcal{G}_2$ con $\{x, y \mapsto 3, 3\}$. Satisfacible en $\mathcal{G}_1$ con $\{x, y \mapsto d, c\}$. Insatisfacible en $\mathcal{G}_1$ con $\{x, y \mapsto b, c\}$.
5. $R(x,y) \wedge R(y,z) \wedge x \neq y \wedge y \neq z \wedge x \neq z$. x, y, z son distintos y x es adyacente a y que es adyacente a z. Satisfacible en $\mathcal{G}_1$ con $\{x, y, z \mapsto a, d, c\}$. Insatisfacible en $\mathcal{G}_2$ con $\{x, y, z \mapsto 1, 5, 4\}$.

3. Semántica Formal

Ya que hemos visto ejemplos en naturales y gráficas, acerca de la verdad y la satisfacibilidad de una fórmula. Ahora pasamos a las definiciones formales.

Definición 1 *Sea $\mathcal{L} = \mathcal{P} \cup \mathcal{F} \cup \mathcal{C}$ un lenguaje o signatura de primer orden. Una estructura o interpretación para $\mathcal{L}$ es un par $\mathcal{M} = \langle M, \mathcal{I} \rangle$ donde $M \neq \varnothing$ es un conjunto no vacío llamado el universo de la estructura e $\mathcal{I}$ es una función con dominio $\mathcal{L}$ tal que:*

- *Si $P^{(n)} \in \mathcal{P}$ entonces $\mathcal{I}(P)$ es una relación de m-argumentos sobre M, es decir $\mathcal{I}(P) \subseteq M^n$. Alternativamente podemos definir la interpretación de P como una función booleana que decide si una tupla está o no en la relación deseada, es decir, $\mathcal{I}(P) : M^n \to Bool$.*
- *Si $f^{(n)} \in \mathcal{F}$ entonces $\mathcal{I}(f)$ es una función con dominio M^n y contradominio M, es decir $\mathcal{I}(f) : M^n \to M$.*
- *Si $c \in \mathcal{C}$ entonces $\mathcal{I}(c)$ es un elemento de M, es decir $\mathcal{I}(c) \in M$.*

Si $\mathcal{M} = \langle M, \mathcal{I} \rangle$ es una interpretación para $\mathcal{L}$ también decimos que $\mathcal{M}$ es una $\mathcal{L}$-interpretación o $\mathcal{L}$-estructura.

Dada una interpretación $\mathcal{M} = \langle M, \mathcal{I} \rangle$ la siguiente notación será de utilidad:

$$
\begin{array}{rcl}
|\mathcal{M}| & =_{def} & M \\
P^{\mathcal{I}} & =_{def} & \mathcal{I}(P) \\
f^{\mathcal{I}} & =_{def} & \mathcal{I}(f) \\
c^{\mathcal{I}} & =_{def} & \mathcal{I}(c)
\end{array}
$$

3.1. Implementación

Una interpretación $\mathcal{M} = \langle M, \mathcal{I} \rangle$ se implementa como sigue:

- El universo M se implementará como una lista de un tipo dado, digamos `m::[a]`. por ejemplo,

  $M = \{1, 2, 3\}$ se implementa como `m =[1,2,3]`

  $M = \mathbb{N}$ se implementa como `m=[0,1,2,..]`

  Se observa que esto restringe los universos en la implementación a listas posiblemente infinitas, esto es natural puesto que en un lenguaje de programación como Haskell sólo es posible manejar universos discretos para verificar la validez de un cuantificador universal.

- Las relaciones $R \subseteq M^n$ se implementarán como funciones características para listas.
  El tipo es `[a] ->Bool`. La idea es que si $R \subseteq M^n$ y $(a_1, \ldots, a_n) \in R^{\mathcal{I}}$ y `r` implementa a R entonces se tiene que $\mathtt{r}\,[\mathtt{a_1} \ldots \mathtt{a_n}] = \mathtt{True}$.

- Las funciones $f : M^n - > M$ se implementarán como funciones de listas a individuos.
  El tipo es `[a] ->a`. La idea es que si `f` implementa a f entonces $f(a_1, \ldots, a_n)$ corresponde a $\mathtt{f}\,[\mathtt{a_1}, \ldots, \mathtt{a_n}]$.

Los símbolos del lenguaje se implementarán asociando a sus nombres una relación o función de acuerdo a las implementaciones recien fijadas.

- $\mathcal{I}(f)$ será una función de tipo `[a] ->a`, por lo que definimos el tipo de interpretaciones para símbolos de función como:

```
type IntF a = Nombre ->[a] ->a
```

- $\mathcal{I}(R)$ será una función de tipo `[a] ->Bool`, por lo que definimos el tipo de interpretaciones para símbolos de función como:

```
type IntR a = Nombre ->[a] ->Bool
```

4. Interpretación de términos

Las siguientes definiciones dependen todas de una $\mathcal{L}$-interpretación arbitraria pero fija $\mathcal{M} = \langle M, \mathcal{I} \rangle$.

Definición 2 (Estado o Asignación) *Un estado, asignación o valuación de las variables es una función* $\sigma : \mathsf{Var} \to M$*.*

Definición 3 (Estado modificado o actualizado) *Sea* $\sigma : \mathsf{Var} \to M$ *un estado de las variables. Dadas las variables* $x_1, \ldots, x_n$ *y los elementos del universo* $m_1, \ldots, m_n \in M$ *definimos el estado modificado o actualizado en* $x_1, \ldots, x_n$ *por* $m_1, \ldots, m_n$ *denotado* $\sigma[x_1, \ldots, x_n / m_1, \ldots, m_n]$ *o* $\sigma[\vec{x}/\vec{m}\,]$ *como sigue:*

$$\sigma[\vec{x}/\vec{m}\,](y) = \begin{cases} \sigma(y) & si\ y \notin \{x_1, \ldots, x_n\} \\ \\ m_i & si\ y = x_i \quad 1 \leq i \leq n \end{cases}$$

Obsérvese que $\sigma[\vec{x}/\vec{m}\,]$ es un estado que difiere de σ únicamente en los valores de $\vec{x}$ y que la expresión $\sigma[\vec{x}/\vec{m}\,]$ es sólo el nombre del estado, la aplicación a una variable es $\sigma[\vec{x}/\vec{m}\,](y)$. Como puede observarse esta notación recuerda a una sustitución, esto no es coincidencia, sino que los dos conceptos están fuertemente relacionados como veremos adelante.

Definición 4 (Interpretación de Términos) *Sea* σ *un estado de las variables. Definimos la función de interpretación o significado de los términos con respecto a* σ*,* $\mathcal{I}_\sigma : \mathsf{TERM} \to |\mathcal{M}|$ *como sigue:*

$$\begin{aligned} \mathcal{I}_\sigma(x) &= \sigma(x) \\ \mathcal{I}_\sigma(c) &= \mathcal{I}(c) \\ \mathcal{I}_\sigma\left(f(t_1, \ldots, t_n)\right) &= f^{\mathcal{I}}(\mathcal{I}_\sigma(t_1), \ldots, \mathcal{I}_\sigma(t_n)) \end{aligned}$$

Lema 1 (Lema de coincidencia para términos) *Sean* $t \in \mathsf{TERM}$ *y* σ_1, σ_2 *dos estados de las variables tales que* $\sigma_1(x) = \sigma_2(x)$ *para toda variable* x *que figura en* t*. Entonces* $\mathcal{I}_{\sigma_1}(t) = \mathcal{I}_{\sigma_2}(t)$*.*

Demostración. Inducción sobre t. Ejercicio $\dashv$

Veamos ahora la relación entre los estados modificados y las sustituciones. Supóngase que tenemos un término t que tiene muchas presencias de la variable x y que queremos evaluar $t[x := r]$ en algún estado de cierta interpretación, para esto aplicamos la definición recursiva y cada vez que encontramos una presencia del subtérmino r en $t[x := r]$ debemos calcular el valor de r en el mismo estado; esta situación es poco práctica, pensemos por ejemplo en que r es un término complicado y que hay muchas presencias de r en t. Una mejor manera de evaluar $t[x := r]$ sería evaluar r en el estado dado σ, lo cual nos da cierto valor m y evaluar el término $t[x := r]$ en el estado modificado $\sigma[x/m]$. El resultado será el mismo, como lo asegura el siguiente

Lema 2 (Lema de sustitución para términos) *Sean $r \in \mathsf{TERM}$, σ un estado de las variables, $[\vec{x} := \vec{t}\,]$ una sustitución y $m_1, \ldots, m_n \in M$ tales que $\mathcal{I}_\sigma(t_i) = m_i \;\; 1 \leq i \leq n$. Entonces*

$$\mathcal{I}_\sigma\left(r[\vec{x} := \vec{t}\,]\right) = \mathcal{I}_{\sigma[\vec{x}/\vec{m}\,]}(r)$$

Demostración. Inducción sobre t. Ejercicio $\dashv$

4.1. Implementación

La implementación es como sigue:

- Estados: el tipo de estados en el universo `a` es

  ```
  type Estado a = Nombre ->a
  ```

- Actualización de estados:

  ```
  actEst :: Estado a ->Nombre ->a ->Estado a
  ```

  especificada por

  $$\texttt{actEst}\ \sigma\ \text{“x”}\ \text{m} = \sigma[x/m]$$

  Esta función modifica sólo una variable y puede generalizarse directamente a la modificación de una lista de variables.

- Interpretación de términos: la función de interpretación $\mathcal{I}_\sigma$ se implementa como

  ```
  iTerm :: Estado a ->IntF a ->Term ->a
  ```

  especificada por

  $$\texttt{iTerm}\ \sigma\ \texttt{h}\ \texttt{t} = \mathcal{I}_\sigma(t)$$

  Obsérvese que la función h de la implementación juega el papel de la función $\mathcal{I}$ en la teoría, es decir, $f^{\mathcal{I}}$ se implementa como $h(”f”)$.

5. Interpretación de fórmulas

Ya que tenemos definido el proceso para interpretar términos podemos definir la interpretación de las fórmulas:

Definición 5 (Interpretación de Fórmulas) *Sea σ un estado de las variables. Definimos la función de interpretación o significado de las fórmulas con respecto a σ, $\mathcal{I}_\sigma : \mathsf{FORM} \to \{0,1\}$ como sigue:*

$$
\begin{array}{lll}
\mathcal{I}_\sigma(\bot) \;=\; 0 & & \\
\mathcal{I}_\sigma(\top) \;=\; 1 & & \\
\mathcal{I}_\sigma\left(P(t_1,\ldots,t_m)\right) \;=\; 1 & \textit{si y sólo si} & \left(\mathcal{I}_\sigma(t_1),\ldots,\mathcal{I}_\sigma(t_m)\right) \in P^{\mathcal{I}} \\
\mathcal{I}_\sigma(t_1 = t_2) \;=\; 1 & \textit{si y sólo si} & \mathcal{I}_\sigma(t_1) = \mathcal{I}_\sigma(t_2) \\
\mathcal{I}_\sigma(\neg\varphi) \;=\; 1 & \textit{si y sólo si} & \mathcal{I}_\sigma(\varphi) = 0 \\
\mathcal{I}_\sigma(\varphi \wedge \psi) \;=\; 1 & \textit{si y sólo si} & \mathcal{I}_\sigma(\varphi) = \mathcal{I}_\sigma(\psi) = 1 \\
\mathcal{I}_\sigma(\varphi \vee \psi) \;=\; 0 & \textit{si y sólo si} & \mathcal{I}_\sigma(\varphi) = \mathcal{I}_\sigma(\psi) = 0 \\
\mathcal{I}_\sigma(\varphi \to \psi) \;=\; 0 & \textit{si y sólo si} & \mathcal{I}_\sigma(\varphi) = 1 \textit{ e } \mathcal{I}_\sigma(\psi) = 0 \\
\mathcal{I}_\sigma(\varphi \leftrightarrow \psi) \;=\; 1 & \textit{si y sólo si} & \mathcal{I}_\sigma(\varphi) = \mathcal{I}_\sigma(\psi) \\
\mathcal{I}_\sigma(\forall x\varphi) \;=\; 1 & \textit{si y sólo si} & \mathcal{I}_{\sigma[x/m]}(\varphi) = 1 \;\; \textit{para todo } m \in M \\
\mathcal{I}_\sigma(\exists x\varphi) \;=\; 1 & \textit{si y sólo si} & \mathcal{I}_{\sigma[x/m]}(\varphi) = 1 \;\; \textit{para algún } m \in M
\end{array}
$$

Esta definición es sólo una forma moderna de la definición de satisfacción de Tarski de 1936, la cual es base para las semánticas denotativas de los lenguajes de programación.

Obsérvese que todas las definiciones anteriores dependen del lenguaje $\mathcal{L}$ y de la $\mathcal{L}$-interpretación particular $\mathcal{M}$. Si en dado caso tuvieramos que trabajar con más de una interpretación a la vez debemos indicar cada función de interpretación como $\mathcal{I}_\sigma^{\mathcal{M}}$ dado que $\mathcal{I}_\sigma$ no bastaría para distinguir entre las distintas interpretaciones.

Análogamente al caso de términos se cumplen los lemas de coincidencia y sustitución

Lema 3 (Lema de coincidencia para fórmulas) *Sean φ una fórmula y σ_1, σ_2 dos estados de las variables tales que $\sigma_1(x) = \sigma_2(x)$ para toda variable $x \in FV(\varphi)$. Entonces $\mathcal{I}_{\sigma_1}(\varphi) = \mathcal{I}_{\sigma_2}(\varphi)$.*

Demostración. Inducción sobre φ. Ejercicio $\dashv$

El siguiente lema relaciona la noción sintáctica de sustitución con la noción semántica de estado modificado

Lema 4 (Lema de sustitución para fórmulas) *Sean* $\varphi \in \mathsf{FORM}$*,* σ *un estado de las variables,* $[\vec{x} := \vec{t}\,]$ *una sustitución y* $m_1, \ldots, m_n \in M$ *tales que* $m_i = \mathcal{I}_\sigma(t_i)\ \ 1 \leq i \leq n$*. Entonces*

$$\mathcal{I}_\sigma\left(\varphi[\vec{x} := \vec{t}\,]\right) = \mathcal{I}_{\sigma[\vec{x}/\vec{m}\,]}(\varphi)$$

Demostración. Inducción sobre φ. Ejercicio $\dashv$

5.1. Implementación

La implementación es como sigue:

- Interpretación de fórmulas: la función de interpretación $\mathcal{I}_\sigma$ se implementa como

```
iForm :: Eq a =>[a] ->Estado a ->IntF a ->IntR a ->Form ->Bool
```

  especificada por

  $$\texttt{iForm}\ \sigma\ h\ \varphi = \mathcal{I}_\sigma(\varphi)$$

  Obsérvese que la función h de la implementación juega el papel de la función $\mathcal{I}$ en la teoría, es decir, $R^{\mathcal{I}}$ se implementa como $h(''R'')$.

5.2. Verdad

Definición 6 (Satisfacibilidad y Verdad) *Sean* φ *una fórmula y* $\mathcal{M} = \langle M, \mathcal{I}\rangle$ *una interpretación. Entonces*

- φ *es satisfacible en* $\mathcal{M}$ *si existe un estado de las variables* σ *tal que* $\mathcal{I}_\sigma(\varphi) = 1$*, lo cual suele denotarse con* $\mathcal{M} \models \varphi[\sigma]$ *o con* $\mathcal{M} \models_\sigma \varphi$*.*
- φ *es verdadera en* $\mathcal{M}$ *si para todo estado de las variables* σ *se tiene* $\mathcal{I}_\sigma(\varphi) = 1$*, es decir, si* φ *es satisfacible en* $\mathcal{M}$ *en todos los estados posibles.*
  En tal caso también decimos que $\mathcal{M}$ *es un modelo de* φ *lo cual se denotará con* $\mathcal{M} \models \varphi$*.*

Las definiciones anteriores aplican también para conjuntos de fórmulas, por ejemplo $\mathcal{M} \models \Gamma$ denota el hecho de que Γ tiene un modelo, y $\mathcal{M} \models_\sigma \Gamma$ que todas las fórmulas de Γ se satisfacen en $\mathcal{M}$ en el estado σ.

5.3. Implementación

- Se deja como ejercicio implementar la función `satForm` que verifican la satisfacibilidad de una fórmula en una interpretación dada.
- ¿Será posible implementar una función `verdForm` que verifique si una fórmula es verdadera en una interpretación dada?

5.4. Falsedad

De acuerdo a lo aprendido en lógica proposicional una fórmula φ es falsa si y sólo si su negación es verdadera, por lo que damos la siguiente

Definición 7 *Sean $\mathcal{M} = \langle M, \mathcal{I} \rangle$ una interpretación y φ una fórmula. Decimos que φ es falsa en $\mathcal{M}$ si y sólo si $\mathcal{M} \models \neg\varphi$. Es decir φ es falsa si y sólo si su negación $\neg\varphi$ es verdadera.*

Con la noción de falsedad recien dada hay que tener cuidado. En la lógica proposicional las nociones de ser falsa y no ser verdadera coinciden. Si una fórmula φ es falsa en una interpretación $\mathcal{I}$ entonces $\mathcal{I}(\varphi) = 0$ lo cual sucede si y sólo si $\mathcal{I}(\varphi) \neq 1$ es decir, si y sólo si φ no es verdadera en $\mathcal{I}$.
Por otro lado, en la lógica de predicados tal equivalencia se pierde, puesto que si φ no es verdadera (es decir, si $\mathcal{M} \not\models \varphi$), entonces, de acuerdo a la definición de verdad, existe un estado σ tal que $\mathcal{M} \not\models_\sigma \varphi$, es decir $\mathcal{I}_\sigma(\varphi) = 0$ o bien φ es insatisfacible en el estado σ.
Por lo tanto una fórmula no verdadera es aquella tal que es insatisfacible en algún estado de sus variables, o bien tal que su negación es satisfacible en algún estado de sus variables. Sin embargo, para poder afirmar que φ es falsa, por definición tendríamos que mostrar que $\mathcal{M} \models \neg\varphi$, es decir que $\neg\varphi$ es satisfacible en **todos** los estados posibles. Por lo tanto la noción de falsedad es más fuerte que la noción de no ser verdadera.
En resumen, si una fórmula φ no es verdadera, no tenemos derecho a concluir que φ es falsa.

Ejemplo 5.1 Sean $\mathcal{L} = \{P^{(1)}\}$, $\mathcal{M} = \langle \mathbb{N}, \mathcal{I} \rangle$ donde $P^{\mathcal{I}}$ es la propiedad “ser par”.
Tómense dos estados σ y σ' con $\sigma(x) = 3$ y $\sigma'(x) = 8$ entonces $\mathcal{M} \not\models \varphi$ puesto que $\mathcal{I}_\sigma(\varphi) = 0$ (3 no es par) y además no se cumple $\mathcal{M} \models \neg\varphi$ puesto que $\mathcal{I}_{\sigma'}(\varphi) = 1$ (8 es par).

6. Propiedades de la noción de verdad

Como ya vimos con el caso de la negación, la relación de verdad en lógica de predicados no tiene las mismas propiedades que su contraparte en la lógica proposicional, propiedades que sí hereda la relación de satisfacibilidad. Esta situación se debe a que la relación de verdad involucra una cuantificación implícita de las variables libres de la fórmula φ.
Debemos insistir en esto ya que es error común: aplicar ciertas propiedades de la relación de satisfacibilidad a la relación de verdad lo cual resulta incorrecto, como en el siguiente ejemplo.

Ejemplo 6.1 Sean $\mathcal{L} = \{P^{(1)}, Q^{(1)}\}$, $\mathcal{M} = \langle \mathbb{N}, \mathcal{I} \rangle$ donde $P^{\mathcal{I}}$ es la propiedad “ser par” y $Q^{\mathcal{I}}$ es la propiedad “ser impar”.
Entonces $\mathcal{M} \models P(x) \vee Q(x)$ puesto que cualquier número natural es par o es impar.
Sin embargo no se cumple que $\mathcal{M} \models P(x)$ ni que $\mathcal{M} \models Q(x)$. Puesto que el valor de x no puede ser siempre par o siempre impar, todo depende del estado de las variables.

De este ejemplo podemos concluir que si una disyunción es verdadera en $\mathcal{M}$ entonces no necesariamente lo es alguna de sus componentes o subfórmulas. Es decir, $\mathcal{M} \models \varphi \vee \psi$ no implica que $\mathcal{M} \models \varphi$ o $\mathcal{M} \models \psi$, propiedad que sí era válida en la lógica de proposiciones y sigue siendo válida para la noción de satisfacibilidad.

Veamos ahora algunas propiedades de la relación de verdad.

Proposición 1 *Sean $\mathcal{M}$ una $\mathcal{L}$-interpretación y φ, ψ fórmulas. Entonces*

- *Si $\mathcal{M} \models \varphi$ entonces $\mathcal{M} \not\models \neg\varphi$.*
- *Si $\mathcal{M} \models \varphi \to \psi$ y $\mathcal{M} \models \varphi$ entonces $\mathcal{M} \models \psi$.*
- *Si $\mathcal{M} \models \varphi$ entonces $\mathcal{M} \models \varphi \lor \psi$.*
- *$\mathcal{M} \models \varphi \land \psi$ si y sólo si $\mathcal{M} \models \varphi$ y $\mathcal{M} \models \psi$.*
- *$\mathcal{M} \models \varphi$ si y sólo si $\mathcal{M} \models \forall\varphi$. donde $\forall\varphi$ denota a la cerradura universal de φ, es decir a la fórmula obtenida al cuantificar universalmente todas las variables libres de φ.*

Demostración. Ejercicio $\dashv$

La última propiedad es de gran importancia y nos dice que para mostrar que una fórmula es verdadera basta mostrar que su cerradura universal lo es, pero obsérvese que la cerradura universal de una fórmula es un enunciado y por lo tanto razonar sobre su verdad es más simple puesto que los enunciados se comportan de la misma manera que las fórmulas proposicionales como lo asegura la siguiente

Proposición 2 *Sean φ un enunciado y $\mathcal{M} = \langle M, \mathcal{I} \rangle$ una interpretación. Entonces se cumple una y sólo una de las siguientes condiciones:*

1. *$\mathcal{M} \models \varphi$, es decir, φ es verdadero en $\mathcal{M}$.*
2. *$\mathcal{M} \models \neg\varphi$, es decir, φ es falso en $\mathcal{M}$.*

Demostración. Esto es consecuencia del lema de coincidencia para fórmulas:
Sea σ un estado cualquiera. Si $\mathcal{I}_\sigma(\varphi) = 1$ entonces, como φ no tiene variables libres, cualquier otro estado σ' coincide con σ en $FV(\varphi) = \varnothing$. Por lo tanto por el lema de coincidencia $\mathcal{I}_{\sigma'}(\varphi) = 1$ y así se cumple que $\mathcal{I}_\sigma(\varphi) = 1$ para cualquier estado σ. De aquí que se cumpla el caso 1. Por otra parte, si $\mathcal{I}_\sigma(\varphi) = 0$ entonces $\mathcal{I}_\sigma(\neg\varphi) = 1$ y se procede de igual forma para obtener que $\neg\varphi$ es verdadero, es decir que φ es falso. $\dashv$

Corolario 1 *Si φ es un enunciado entonces φ es falso si y sólo si φ es no verdadero en $\mathcal{M}$*

7. Modelos

Ya mencionamos que un modelo para una fórmula φ es una interpretación $\mathcal{M} = \langle M, \mathcal{I} \rangle$ tal que $\mathcal{M} \models \varphi$, es decir tal que $\mathcal{I}_\sigma(\varphi)$ para cualquier estado σ y que si bien una fórmula cualquiera no necesariamente tiene un modelo, es decir, no necesariamente es verdadera o falsa, esta propiedad sí se cumple para los enunciados, puesto que estos no tienen variables libres que cambien de valor dependiendo del estado.

Uno de los problemas fundamentales en lógica es la búsqueda de modelos de un conjunto dado de fórmulas Γ. Este problema tiene importancia desde el punto de vista puramente teórico, al buscar y clasificar modelos de un conjunto de fórmulas en especial y estudiar como son estos modelos entre si, así como desde el punto de vista computacional donde el objetivo principal es verificar la verdad de una especificación en un modelo dado o bien la existencia de modelos para especificaciones dadas.

El área de las ciencia de la computación conocida como verificación de modelos [2] trata de mecanizar la verificación y construcción de modelos, situación que posible en ciertos casos, aunque imposible en general. La rama de la lógica matemática encargada de estos tópicos se llama *Teoría de Modelos*.

A continuación damos algunos ejemplos sencillos de construcción de modelos para un conjunto de fórmulas dado.

Ejemplo 7.1 Sea $\Gamma = \{P(b), Q(b), R(b), \exists x(P(x) \wedge \neg(Q(x) \vee R(x)),\ \forall x(R(x) \rightarrow P(x))\}$.
Queremos ver si Γ tiene un modelo.
En este caso lo más fácil es ir construyendo un modelo para cada fórmula de Γ, si logramos que todas las fórmulas de Γ sean verdaderas al mismo tiempo entonces habremos construido un modelo para Γ. Analicemos cada fórmula de Γ.

1. $P(b)$. Para que $\mathcal{M} \models P(b)$ se debe cumplir que $b^{\mathcal{I}} \in P^{\mathcal{I}}$. Así que al menos debemos tener $P^{\mathcal{I}} = \{b^{\mathcal{I}}\}$

2. $Q(b)$. Para que $\mathcal{M} \models Q(b)$ se debe cumplir que $b^{\mathcal{I}} \in Q^{\mathcal{I}}$, así que basta $Q^{\mathcal{I}} = \{b^{\mathcal{I}}\}$

3. $R(b)$. Para que $\mathcal{M} \models Q(b)$ se debe cumplir que $b^{\mathcal{I}} \in R^{\mathcal{I}}$ por lo que basta con $R = \{b^{\mathcal{I}}\}$.

4. $\exists x(P(x) \wedge \neg(Q(x) \vee R(x))$. Es decir, hay un elemento de $|\mathcal{M}|$ tal que cumple P y no cumple Q ni R. Claramente este elemento no puede ser $b^{\mathcal{I}}$ por lo que $P^{\mathcal{I}}$ debe tener al menos otro elemento, digamos $P^{\mathcal{I}} = \{b^{\mathcal{I}}, m\}$.

5. $\forall x(R(x) \rightarrow P(x))$. Debemos tener que todo elemento de Γ que cumple $R^{\mathcal{I}}$, debe cumplir $P^{\mathcal{I}}$, pero de acuerdo a como definimos $R^{\mathcal{I}}$ esto ya se cumple pues $b^{\mathcal{I}} \in R^{\mathcal{I}}$ y $b^{\mathcal{I}} \in P^{\mathcal{I}}$.

De manera que $P^{\mathcal{I}} = \{b^{\mathcal{I}}, m\}$ y $Q^{\mathcal{I}} = \{b^{\mathcal{I}}\} = R^{\mathcal{I}}$ y para que el modelo resulte más natural podemos tomar $M = \{0,1\}$, $b^{\mathcal{I}} = 0$ y $m = 1$, con lo que queda $\mathcal{M} = \langle \{0,1\}, \mathcal{I} \rangle$ con $P^{\mathcal{I}} = \{0,1\}, Q^{\mathcal{I}} = \{0\} = R^{\mathcal{I}}, b^{\mathcal{I}} = 0$.

Veamos otro ejemplo.

Ejemplo 7.2 Sea $\Gamma = \{\exists x P(x) \wedge \exists x Q(x),\ \neg\exists x(P(x) \wedge Q(x)),\ \forall x \neg R(x,x),\ \forall x \forall y(R(x,y) \rightarrow R(y,x)),$
$\forall x(P(x) \rightarrow \exists y(Q(y) \wedge R(x,y))),\ \forall x(Q(x) \rightarrow \exists y(Q(y) \wedge R(x,y)))\}$ Queremos hallar un modelo de Γ, analicemos cada fórmula:

1. $\exists x P(x) \wedge \exists x Q(x)$. Es decir, debe existir $a_1 \in |\mathcal{M}|$ tal que $P^{\mathcal{I}}(a_1)$ y además debe existir $a_2 \in |\mathcal{M}|$ tal que $Q^{\mathcal{I}}(a_2)$. Obsérvese que podriamos tener $a_1 = a_2$.

2. $\neg\exists x(P(x) \wedge Q(x))$. Está fórmula indica que no hay un individuo que cumpla $P^{\mathcal{I}}$ y $Q^{\mathcal{I}}$, de manera que los individuos encontrados en (1) deben ser diferentes.

3. $\forall x \neg R(x,x)$. Esto es la relación $R^{\mathcal{I}}$ es antireflexiva, por lo que los pares (a,a) no pueden pertenecer a $R^{\mathcal{I}}$. Obsérvese que esto ya se cumple pues hasta ahora $R^{\mathcal{I}} = \varnothing$, es decir, aún no existe un par en la relación $R^{\mathcal{I}}$.

4. $\forall x \forall y(R(x,y) \rightarrow R(y,x))$. Esta fórmula indica que la relación $R^{\mathcal{I}}$ es simétrica pero todavia no se exige que haya un par relacionado por lo que aún tenemos $R^{\mathcal{I}} = \varnothing$

5. $\forall x(P(x) \rightarrow \exists y(Q(y) \wedge R(x,y)))$. Esto significa que para cada elemento a que cumpla $P^{\mathcal{I}}$ debe existir un elemento b que cumpla $Q^{\mathcal{I}}$ de manera que el par (a,b) esté en la relación $R^{\mathcal{I}}$. Como en (1) obtuvimos que $P^{\mathcal{I}}(a_1)$ entonces debe existir $b \in |\mathcal{M}|$ tal que $Q^{\mathcal{I}}(b)$ y $R^{\mathcal{I}}(a_1, b)$. Obsérvese que podemos tomar $b = a_2$, además, por (4), también debemos tener $R^{\mathcal{I}}(a_2, a_1)$.

[2] Model checking

6. $\forall x(Q(x) \to \exists y(Q(y) \wedge R(x,y)))$. Está fórmula es similar a la anterior. Como tenemos que $Q^{\mathcal{I}}(a_2)$, por (1), entonces debe existir $a_3 \in |\mathcal{M}|$ tal que $Q^{\mathcal{I}}(a_3)$ y $R^{\mathcal{I}}(a_2, a_3)$, además por (3) tenemos que $a_3 \neq a_2$ y por (2), $a_3 \neq a_1$. Finalmente también debemos tener que $R^{\mathcal{I}}(a_3, a_2)$ para que se cumpla (4).

De manera que las interpretaciones pueden quedar como sigue:

$|\mathcal{M}| = \{1, 2, 3\}$

$P^{\mathcal{I}} = \{1\}$

$Q^{\mathcal{I}} = \{2, 3\}$

$R^{\mathcal{I}} = \{(1,2), (2,1), (2,3), (3,2)\}$

7.1. Implementación

¿Será posible implementar la función `esModelo` que verifica si una interpretación es modelo de un conjunto (lista) dado de fórmulas Γ?

8. Validez Universal

El concepto de tautología en lógica proposicional tiene su contraparte en lógica de predicados mediante el concepto de validez universal.

Definición 8 (Validez universal) *Decimos que una fórmula φ es universalmente válida (o simplemente válida) si para toda interpretación $\mathcal{M}$ se cumple que $\mathcal{M} \models \varphi$, es decir, si φ es verdadera en cualquier interpretación posible, lo cual se denota con $\models \varphi$.*

Como consecuencia del lema de sustitución podemos mostrar la validez universal de ciertas fórmulas. Veamos un ejemplo.

Proposición 3 *Sean $\mathcal{M}$ una interpretación, φ una fórmula, t un término y σ un estado de las variables. Entonces $\mathcal{I}_\sigma(\forall x\varphi \to \varphi[x := t]) = 1$. En otras palabras, la fórmula $\forall x\varphi \to \varphi[x := t]$ es universalmente válida.*

Demostración. Si $\mathcal{I}_\sigma(\forall x\varphi) = 0$ terminamos no hay nada que probar pues en tal caso la implicación $\forall x\varphi \to \varphi[x := t]$ se satisface.
Supongamos entonces que $\mathcal{I}_\sigma(\forall x\varphi) = 1$. Queremos demostrar que $\mathcal{I}_\sigma(\varphi[x := t]) = 1$, lo cual, por el lema de sustitución equivale a $\mathcal{I}_{\sigma[x/m]}(\varphi) = 1$ con $m = \mathcal{I}_\sigma(t)$, pero esto es consecuencia de la definición de $\mathcal{I}_\sigma(\forall x\varphi) = 1$. ⊣

Análogamente dejamos como ejercicio mostrar que

$$\models \varphi[x := t] \to \exists x\varphi$$

La noción de validez universal es análoga a la noción de tautología en lógica proposicional. De hecho toda fórmula cuyo esqueleto proposicional es una tautología, resulta ser una fórmula universalmente válida. Por ejemplo sucede que $\models \forall x(P(x) \to P(x) \vee Q(x,y))$ puesto que en lógica proposicional se tiene que $\models P \to P \vee Q$.

Veamos otros ejemplos de formulas universalmente válidas.

Ejemplo 8.1 $\models \neg\exists x\varphi \leftrightarrow \forall x\neg\varphi$.
Sea $\mathcal{M}$ una interpretación, veamos que $\mathcal{M} \models \neg\exists x\varphi \leftrightarrow \forall x\neg\varphi$. Sea σ un estado de las variables, hay 2 casos:

- $\mathcal{I}_\sigma(\neg\exists x\varphi) = 1$. Esto es equivalente con $\mathcal{I}_\sigma(\exists x\varphi) = 0$ cuya definición es que no existe $m \in |\mathcal{M}|$ tal que $\mathcal{I}_{\sigma[x/m]}(\varphi) = 1$, es decir, para todo $m \in |\mathcal{M}|$, $\mathcal{I}_{\sigma[x/m]}(\varphi) = 0$ o equivalentemente para todo $m \in |\mathcal{M}|$, $\mathcal{I}_{\sigma[x/m]}(\neg\varphi) = 1$ pero esto significa que $\mathcal{I}_\sigma(\forall x\neg\varphi) = 1$.

- $\mathcal{I}_\sigma(\neg\exists x\varphi) = 0$. Es decir, $\mathcal{I}_\sigma(\exists x\varphi) = 1$ syss existe $m \in |\mathcal{M}|$ tal que $\mathcal{I}_{\sigma[x/m]}(\varphi) = 1$ syss existe $m \in |\mathcal{M}|$ tal que $\mathcal{I}_{\sigma[x/m]}(\neg\varphi) = 0$ syss no para todo $m \in |\mathcal{M}|$, $\mathcal{I}_{\sigma[x/m]}(\neg\varphi) = 1$ syss $\mathcal{I}_\sigma(\forall x\neg\varphi) = 0$.

Por lo tanto en cualquier caso se tiene que $\mathcal{I}_\sigma(\neg\exists x\varphi) = \mathcal{I}_\sigma(\forall x\neg\varphi)$ es decir, $\mathcal{I}_\sigma(\neg\exists x\varphi \leftrightarrow \forall x\neg\varphi) = 1$. Por lo tanto $\mathcal{M} \models \neg\exists x\varphi \leftrightarrow \forall x\neg\varphi$. Pero como $\mathcal{M}$ era una interpretación arbitraria, se tiene que $\models \neg\exists x\varphi \leftrightarrow \forall x\neg\varphi$.

Ejemplo 8.2 Vamos a demostrar la existencia del dios Baco quién tiene la propiedad de que si él bebe entonces todos beben. Es decir, mostraremos que $\models \exists x(B(x) \to \forall yB(y))$.
Sea $\mathcal{M}$ una interpretación y σ un estado de las variables, veamos que $\mathcal{I}_\sigma\left(\exists x(B(x) \to \forall yB(y))\right) = 1$, es decir, $\mathcal{I}_{\sigma[x/m]}\big(B(x) \to \forall yB(y)\big) = 1$ para algún $m \in |\mathcal{M}|$. Tenemos dos casos:

- $B^{\mathcal{I}} = |\mathcal{M}|$. Es decir, todos los individuos del universo cumplen B. En este caso, como $m \in B^{\mathcal{I}}$ para todo $m \in |\mathcal{M}|$, se tiene que $\mathcal{I}_{\sigma[y/m]}(B(y)) = 1$ para todo $m \in |\mathcal{M}|$. Por lo tanto $\mathcal{I}_\sigma(\forall yB(y)) = 1$, en particular puesto que $x \notin FV(\forall yB(y))$, por el lema de coincidencia para fórmulas,para cualquier $m \in |\mathcal{M}|$ se tiene que $\mathcal{I}_{\sigma[x/m]}(\forall yB(y)) = 1$. Así que $\mathcal{I}_{\sigma[x/m]}\big(B(x) \to \forall yB(y)\big) = 1$.

- $B^{\mathcal{I}} \neq |\mathcal{M}|$. Es decir, al menos un individuo del universo no cumple B. Sea $m \in |\mathcal{M}|$ tal que $m \notin B^{\mathcal{I}}$, es decir $\mathcal{I}_{\sigma[x/m]}(B(x)) = 0$, en tal caso $\mathcal{I}_{\sigma[x/m]}\big(B(x) \to \forall yB(y)\big) = 1$ pues el antecedente de la implicación se evalua a 0.

9. Equivalencia Lógica

Un concepto de gran importancia que depende de la validez universal es el de fórmulas lógicamente equivalentes.

Definición 9 *Sean φ, ψ fórmulas. Decimos que φ es* lógicamente equivalente *a ψ, denotado con $\varphi \equiv \psi$, si y sólo si $\models \varphi \leftrightarrow \psi$, es decir si y sólo si la fórmula $\varphi \leftrightarrow \psi$ es universalmente válida.*

Las siguientes equivalencias lógicas serán de utilidad más adelante.

1. Negación de cuantificaciones (Leyes de De Morgan generalizadas):

   a) $\neg\forall x\varphi \equiv \exists x\neg\varphi$

   b) $\neg\exists x\varphi \equiv \forall x\neg\varphi$

2. Eliminación de cuantificaciones múltiples.

   a) $\forall x\forall x\varphi \equiv \forall x\varphi$.

   b) $\exists x\exists x\varphi \equiv \exists x\varphi$.

3. Renombre de variables. Si y no figura libre en φ entonces:

   a) $\forall x\varphi \equiv \forall y(\varphi[x := y])$.

   b) $\exists x\varphi \equiv \exists y(\varphi[x := y])$.

4. Eliminación de cuantificaciones vacuas. Si x no figura libre en φ entonces:

   a) $\forall x\varphi \equiv \varphi$.

   b) $\exists x\varphi \equiv \varphi$.

10. Consecuencia Lógica

La noción de argumento correcto se formaliza en la lógica de predicados mediante la noción de consecuencia lógica cuya definición es la misma que para lógica de proposiciones y recordamos ahora.

Definición 10 *Sean* Γ *un conjunto de fórmulas y* φ *una fórmula. Decimos que* φ *es* consecuencia lógica *de* Γ*, denotado con* $\Gamma \models \varphi$*, si y sólo si todo modelo de* Γ *es un modelo de* φ*.*
Es decir, si y sólo si para toda interpretación $\mathcal{M}$*, si* $\mathcal{M} \models \Gamma$ *entonces* $\mathcal{M} \models \varphi$*. Si* $\Gamma \models \varphi$ *tambien decimos que* Γ implica lógicamente *a* φ*.*

Las siguientes observaciones son importantes:

1. Al igual que en lógica proposicional el símbolo $\models$ está sobrecargado y se usa para las relaciones *"ser modelo de"* y *"ser consecuencia lógica de"*.

2. Al no cambiar la definición de consecuencia lógica las propiedades de ésta siguen siendo válidas. En particular el argumento con premisas $\Gamma = \{\varphi_1, \ldots, \varphi_n\}$ y conclusión φ, es correcto si y sólo si $\Gamma \models \varphi$.

Veamos un ejemplo sencillo.

Ejemplo 10.1 Vamos a demostrar la correctud del siguiente argumento:

Si sabemos que todos los gatos maullan y que
Cleofas es un gato entonces es correcto concluir que Cleofas maulla.

Las premisas se formalizan como $\{\forall x(G(x) \to M(x)),\ G(c)\}$ y la conclusión $M(c)$.
Entonces demostraremos que:

$$\{\forall x(G(x) \to M(x)),\ G(c)\} \models M(c)$$

Sea $\mathcal{M}$ un modelo de $\{\forall x(G(x) \to M(x)),\ G(c)\}$, es decir, $\mathcal{M} \models \forall x(G(x) \to M(x))$ y $\mathcal{M} \models G(c)$, veamos que $\mathcal{M} \models M(c)$.
Para lo anterior, consideramos a σ un estado (cualquiera) de las variables. Por hipótesis tenemos que para toda $m \in |\mathcal{M}|$, se cumple que $\mathcal{I}_{\sigma[x/m]}(G(x) \to M(x)) = 1$ al igual que $\mathcal{I}_\sigma(G(c)) = 1$.
En particular para el objeto $c^{\mathcal{I}}$ se cumple que $\mathcal{I}_{\sigma[x/c^{\mathcal{I}}]}(G(x) \to M(x)) = 1$, es decir, por el lema de sustitución se tiene $\mathcal{I}_\sigma(G(c) \to M(c)) = 1$. De esto y dado que $\mathcal{I}_\sigma(G(c)) = 1$, se concluye por definición que $\mathcal{I}_\sigma(M(c)) = 1$.
Por lo tanto, como σ era arbitrario, concluimos que $\mathcal{M} \models M(c)$. De manera que en efecto, *Cleofas maulla.*

Finalmente queremos mencionar que la formalización de un criterio para decidir la correctud de argumentos no es la única razón por la que la consecuencia lógica juega un papel central en la lógica. En general, obtener las consecuencias lógicas de un conjunto Γ significa intuitivamente extraer conocimiento que estaba implícito en Γ, problema de gran importancia en ciencias de la computación, cuya aplicación más cercana resulta ser la programación lógica.

La extracción de conocimiento a partir de un conjunto de hechos Γ puede resultar muy simple o sumamente complicada, los casos simples llevan a cierta automatización del proceso de extracción de conocimiento mediante diversas técnicas como la programación lógica o el razonamiento automatizado.

11. Indecidibilidad de la lógica de predicados

Las secciones de validez universal, equivalencia lógica y consecuencia lógica no incluyen una subsección acerca de la implementación debido a que estos problemas son **indecidibles**, es decir, no existe un algoritmo para poder decidir el problema en general. Enunciamos aquí estos famosos resultados.

Teorema 1 (Teorema de Indecidibilidad de Church) *El problema de validez universal es indecidible. Es decir, no puede existir un algoritmo que reciba un enunciado φ como entrada y decida si $\models \varphi$.*

Demostración. Las demostraciones usuales reducen este problema al problema de la detención en máquinas de Turing o bien al problema de correspondencia de Post, ambos indecidibles. $\dashv$

Corolario 2 *La equivalencia lógica es indecidible. Es decir, no puede existir un algoritmo que reciba como entrada dos fórmulas φ y ψ y decida si $\varphi \equiv \psi$.*

Demostración. Basta observar que $\models \varphi$ syss $\varphi \equiv \top$ por lo que cualquier algoritmo que decidiera la equivalencia lógica podría decidir la validez universal.

Corolario 3 *La consecuencia lógica es indecidible. Es decir, no puede existir un algoritmo que reciba como entrada un conjunto finito de fórmulas Γ y una fórmula φ y decida si $\Gamma \models \varphi$.*

Demostración. Basta observar que si $\Gamma = \{\varphi_1, \ldots, \varphi_n\}$ entonces

$$\Gamma \models \varphi \text{ si y sólo si } \models \varphi_1 \land \ldots \land \varphi_n \to \varphi$$

$\dashv$

Corolario 4 *Es indecidible si una fórmula es satisfacible. Es decir, dada una fórmula φ no existe un algoritmo que decida si existe un modelo $\mathcal{M}$ y un estado σ tal que $\mathcal{M} \models_\sigma \varphi$.*

Demostración. Obsérvese que $\mathcal{M} \models_\sigma \varphi$ syss $\not\models \neg\varphi$. Por lo tanto también se cumple que $\mathcal{M} \not\models_\sigma \neg\varphi$ syss $\models \varphi$. Por lo que si $\mathcal{A}$ fuera un algoritmo para decidir satisfacibilidad entonces bastaría darle como entrada $\neg\varphi$ y en caso de que $\mathcal{A}(\neg\varphi)$ devolviera `false`, entonces se decidiría que $\models \varphi$. $\dashv$