Introdución a la Lógica Proposicional

Pablo Barceló

Lógica proposicional: Sintaxis

Tenemos los siguientes elementos:

- Variables proposicionales (P): $p, q, r, \ldots$
- Conectivos lógicos: $\neg$, $\vee$, $\wedge$, $\rightarrow$, $\leftrightarrow$
- Símbolos de puntuación: (,)

Cada variable proposicional representa una proposición **completa** e **indivisible**, que puede ser **verdadera** o **falsa**.

Ejemplo:

$$P = \{Bateria\_OK,\ Robot\_se\_mueve,\ Objeto\_elevable\}.$$

Lógica proposicional: Sintaxis

Conectivos lógicos son usados para construir expresiones complejas, que también pueden ser verdaderas o falsas.

Ejemplos:

$$\mathit{Bateria\_OK} \quad \wedge \quad \mathit{Objeto\_elevable}$$

$$\mathit{Bateria\_OK} \quad \wedge \quad (\neg\, \mathit{Objeto\_elevable}) \;\rightarrow\; \mathit{Robot\_se\_mueve}$$

Símbolos de puntuación son usados para evitar ambigüedades.

Definición de sintaxis de la lógica proposicional

Dado: Conjunto P de variables proposicionales.

Definición

$L(P)$ es el menor conjunto que satisface las siguientes reglas:

1. $P \subseteq L(P)$.
2. *Si* $\varphi \in L(P)$, *entonces* $(\neg\varphi) \in L(P)$.
3. *Si* $\varphi, \psi \in L(P)$, *entonces* $(\varphi \vee \psi) \in L(P)$, $(\varphi \wedge \psi) \in L(P)$, $(\varphi \rightarrow \psi) \in L(P)$ *y* $(\varphi \leftrightarrow \psi) \in L(P)$.

Ejercicio: Verifique que $((\neg p) \rightarrow (q \vee r))$ es una fórmula.

Naturaleza de la definición

La naturaleza de la definición es inductiva.

- Permite construir programas recursivos para chequear si una fórmula está bien construida.
- Permite definir recursivamente conceptos asociados a las fórmulas.
- Permite demostrar inductivamente propiedades de las fórmulas.

Demostraciones inductivas

Queremos definir una función *la* que indica cuántos símbolos tiene una fórmula: $la((p \wedge q)) = 5$.

Caso base: Para cada $p \in P$, $la(p) = 1$.

Caso inductivo: $la((\neg\varphi)) = 3 + la(\varphi)$ y
$la((\varphi \star \psi)) = 3 + la(\varphi) + la(\psi)$, donde $\star$ corresponde a $\vee$, $\wedge$, $\rightarrow$ o $\leftrightarrow$.

En el ejemplo: $la((p \wedge q)) = 3 + la(p) + la(q) = 3 + 1 + 1 = 5$.

Demostraciones inductivas

Queremos definir una función *la* que indica cuántos símbolos tiene una fórmula: $la((p \wedge q)) = 5$.

Caso base: Para cada $p \in P$, $la(p) = 1$.

Caso inductivo: $la((\neg\varphi)) = 3 + la(\varphi)$ y
$la((\varphi \star \psi)) = 3 + la(\varphi) + la(\psi)$, donde $\star$ corresponde a $\vee$, $\wedge$, $\rightarrow$ o $\leftrightarrow$.

En el ejemplo: $la((p \wedge q)) = 3 + la(p) + la(q) = 3 + 1 + 1 = 5$.

Ejercicio: Defina las funciones *pi* y *pd* que indican cuáles son los números de paréntesis izquierdos y derechos en una fórmula, respectivamente. Demuestre que para cada fórmula ϕ se tiene que $pi(\phi) = pd(\phi)$.

Demostraciones inductivas: Ejercicio

Ejercicio: Defina $v(\varphi)$ como el número de ocurrencias de variables proposicionales en φ;

- Demuestre que para cada fórmula proposicional φ que no contiene el smbolo $\neg$ se tiene que $la(\varphi) \leq 3 \cdot v(\varphi)^2$.
- ¿Qué sucede si φ contiene el símbolo $\neg$?
- ¿Qué sucede si las fórmulas de la forma $(\neg(\neg\varphi))$ no son permitidas?

Semántica de la lógica proposicional

¿Cómo podemos determinar si una fórmula es verdadera o falsa?

Este valor de verdad depende de los valores de verdad asignados a las variables proposicionales y de los conectivos utilizados.

Valuación (asignación): $\sigma : P \to \{0, 1\}$.

Ejemplo: $\sigma(Robot\_se\_mueve) = 1$, $\sigma(Bateria\_OK) = 0$ y $\sigma(Objeto\_elevable) = 1$.

Semántica: Definición

Dado $\sigma : P \to \{0,1\}$, queremos extender σ:

$$\hat{\sigma} : L(P) \to \{0,1\}.$$

Definición

Dado $\varphi \in L(P)$,

- *Si* $\varphi = p$, *entonces* $\hat{\sigma}(\varphi) := \sigma(p)$.
- *Si* $\varphi = (\neg\alpha)$, *entonces*

$$\hat{\sigma}(\varphi) = \begin{cases} 1 & si\ \hat{\sigma}(\alpha) = 0 \\ 0 & si\ \hat{\sigma}(\alpha) = 1 \end{cases}$$

Semántica: Definición

- Si $\varphi = (\alpha \wedge \beta)$, entonces

$$\hat{\sigma}(\varphi) = \begin{cases} 1 & \text{si } \hat{\sigma}(\alpha) = 1 \text{ y } \hat{\sigma}(\beta) = 1 \\ 0 & \text{si } \hat{\sigma}(\alpha) = 0 \text{ o } \hat{\sigma}(\beta) = 0 \end{cases}$$

- Si $\varphi = (\alpha \vee \beta)$, entonces

$$\hat{\sigma}(\varphi) = \begin{cases} 1 & \text{si } \hat{\sigma}(\alpha) = 1 \text{ o } \hat{\sigma}(\beta) = 1 \\ 0 & \text{si } \hat{\sigma}(\alpha) = 0 \text{ y } \hat{\sigma}(\beta) = 0 \end{cases}$$

Semántica: Definición

- Si $\varphi = (\alpha \rightarrow \beta)$, entonces

$$\hat{\sigma}(\varphi) = \begin{cases} 1 & \text{si } \hat{\sigma}(\alpha) = 0 \text{ o } \hat{\sigma}(\beta) = 1 \\ 0 & \text{si } \hat{\sigma}(\alpha) = 1 \text{ y } \hat{\sigma}(\beta) = 0 \end{cases}$$

- Si $\varphi = (\alpha \leftrightarrow \beta)$, entonces

$$\hat{\sigma}(\varphi) = \begin{cases} 1 & \text{si } \hat{\sigma}(\alpha) = \hat{\sigma}(\beta) \\ 0 & \text{si } \hat{\sigma}(\alpha) \neq \hat{\sigma}(\beta) \end{cases}$$

Por simplicidad, de ahora en adelante usaremos σ en vez de $\hat{\sigma}$.

Ejemplos de semántica

Supongamos que $\sigma(\mathit{Bateria\_OK}) = 1$ y $\sigma(\mathit{Robot\_se\_mueve}) = 0$.

Entonces:

$$\begin{aligned}
&\sigma((\mathit{Bateria_OK} \rightarrow \mathit{Robot_se_mueve})) = 0 \\
&\sigma((((\mathit{Bateria_OK} \rightarrow \mathit{Robot_se_mueve}) \wedge \\
&\qquad\qquad \mathit{Bateria_OK}) \rightarrow \mathit{Robot_se_mueve})) = 1
\end{aligned}$$

Satisfacibilidad

Una fórmula ϕ es satisfacible si existe σ tal que $\sigma(\phi) = 1$.

Una fórmula es una tautología si su negación es insatisfacible.

Ejemplo: La fórmula $(p \wedge (p \rightarrow q))$ es satisfacible. La fórmula $(p \wedge (p \rightarrow q) \wedge (\neg q))$ no lo es. La fórmula $((p \wedge (p \rightarrow q)) \rightarrow q)$ es una tautología.

Satisfacibilidad

Una fórmula ϕ es satisfacible si existe σ tal que $\sigma(\phi) = 1$.

Una fórmula es una tautología si su negación es insatisfacible.

Ejemplo: La fórmula $(p \wedge (p \rightarrow q))$ es satisfacible. La fórmula $(p \wedge (p \rightarrow q) \wedge (\neg q))$ no lo es. La fórmula $((p \wedge (p \rightarrow q)) \rightarrow q)$ es una tautología.

Ejercicio: Sea $G = (V, E)$ un grafo. Demuestre que existe una fórmula ϕ_G de la lógica proposicional tal que:

$$G \text{ es 3-coloreable} \iff \phi_G \text{ es satisfacible.}$$

¿Cuántos recursos (tiempo y espacio) se utilizaron para construir ϕ_G en términos de G?

Consecuencia lógica

Para conjunto de fórmulas Σ escribimos $\sigma(\Sigma) = 1$ si para toda fórmula ψ en Σ se tiene que $\sigma(\psi) = 1$.

Definición

La fórmula ϕ es consecuencia lógica de Σ si para toda valuación σ:

$$\sigma(\Sigma) = 1 \implies \sigma(\phi) = 1.$$

Ejemplo: Demuestre que $(\neg \mathit{Objeto\_elevable})$ es consecuencia lógica del conjunto de fórmulas:

$$\{\mathit{Bateria\_OK},\ (\neg \mathit{Robot\_se\_mueve})$$
$$((\mathit{Bateria\_OK} \wedge \mathit{Objeto\_elevable}) \rightarrow \mathit{Robot\_se\_mueve})\}$$

Dos fórmulas ϕ y ψ son equivalentes si $\phi \models \psi$ y $\psi \models \phi$.

- De la misma forma, si para toda valuación σ,

$$\sigma(\phi) = 1 \;\Leftrightarrow\; \sigma(\psi) = 1.$$

Equivalencia lógica

Dos fórmulas ϕ y ψ son equivalentes si $\phi \models \psi$ y $\psi \models \phi$.

- De la misma forma, si para toda valuación σ,

$$\sigma(\phi) = 1 \;\Leftrightarrow\; \sigma(\psi) = 1.$$

Algunas equivalencias útiles:

$$(\neg(\varphi \wedge \psi)) \equiv ((\neg\varphi) \vee (\neg\psi))$$
$$(\neg(\varphi \vee \psi)) \equiv ((\neg\varphi) \wedge (\neg\psi))$$
$$(\varphi \wedge (\psi \wedge \theta)) \equiv ((\varphi \wedge \psi) \wedge \theta)$$
$$(\varphi \vee (\psi \vee \theta)) \equiv ((\varphi \vee \psi) \vee \theta)$$

$$(\varphi \rightarrow \psi) \equiv ((\neg\varphi) \vee \psi)$$
$$(\varphi \leftrightarrow \psi) \equiv ((\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi))$$
$$(\neg(\neg\varphi)) \equiv \varphi$$

Equivalencia de fórmulas

Desde ahora en adelante:

- Vamos a omitir los paréntesis externos;
- vamos a escribir $\varphi \wedge \psi \wedge \theta$ en lugar de $(\varphi \wedge \psi) \wedge \theta$ (lo mismo para $\vee$).

Ejercicio: Es $\rightarrow$ asociativo? Vale decir, ¿Es cierto que $(\alpha \rightarrow \beta) \rightarrow \gamma \equiv \alpha \rightarrow (\beta \rightarrow \gamma)$?

Tablas de verdad

Como sabemos, cada fórmula tiene una tabla de verdad relacionada.

Ejercicio: Asuma que $P = \{p, q\}$. ¿Cuántas fórmulas contiene $L(P)$? ¿Cuántas fórmulas no equivalentes contiene este conjunto?

Tablas de verdad

Como sabemos, cada fórmula tiene una tabla de verdad relacionada.

Ejercicio: Asuma que $P = \{p, q\}$. ¿Cuántas fórmulas contiene $L(P)$? ¿Cuántas fórmulas no equivalentes contiene este conjunto?

Lo contrario también es cierto:

Para cada tabla de verdad existe una fórmula que tiene exactamente esa tabla de verdad.

Conectivos n-arios

Una tabla de verdad con n variables proposicionales es un conectivo n-ario (Nota: Nuestros conectivos lógicos son precisamente conectivos unarios ($\neg$) y binarios ($\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$)):

p_1	p_2	$\cdots$	p_{n-1}	p_n	$C(p_1, p_2, \ldots, p_{n-1}, p_n)$
0	0	$\cdots$	0	0	b_1
0	0	$\cdots$	0	1	b_2
$\vdots$	$\vdots$	$\ldots$	$\vdots$	$\vdots$	$\vdots$
1	1	$\cdots$	1	1	b_{2^n}

¿Qué fórmula de la lógica proposicional tiene exactamente esta tabla de verdad?

Conectivos funcionalmente completos

Asumiendo que σ_i es la valuación correspondiente a la fila i de la tabla de verdad de $C(p_1, \ldots, p_n)$, este conectivo es equivalente a:

$$\bigvee_{i \,:\, b_i = 1} \left(\left(\bigwedge_{j \,:\, \sigma_i(p_j) = 1} p_j \right) \wedge \left(\bigwedge_{k \,:\, \sigma_i(p_k) = 0} \neg p_k \right) \right).$$

Asumiendo que σ_i es la valuación correspondiente a la fila i de la tabla de verdad de $C(p_1, \ldots, p_n)$, este conectivo es equivalente a:

$$\bigvee_{i\,:\,b_i=1} \left(\left(\bigwedge_{j\,:\,\sigma_i(p_j)=1} p_j \right) \wedge \left(\bigwedge_{k\,:\,\sigma_i(p_k)=0} \neg p_k \right) \right).$$

Conclusión: Basta con los conectivos lógicos $\neg, \vee, \wedge$ para representar cualquier tabla de verdad.

Decimos que el conjunto de conectivos $\{\neg, \vee, \wedge\}$ es funcionalmente completo.

Conectivos funcionalmente completos

Pregunta: ¿Es cierto que los conjuntos $\{\neg, \vee\}$, $\{\neg, \wedge\}$ y $\{\neg, \rightarrow\}$ son funcionalmente completos?

Ejercicio: ¿Existe algún conectivo binario que sea funcionalmente completo por si solo?

Ejercicio: ¿Es el conjunto $\{\vee, \wedge, \rightarrow, \leftrightarrow\}$ funcionalmente completo?

Propiedades de la lógica proposicional: Monotonía

Presentaremos algunas propiedades interesantes de la lógica proposicional, partiendo por monotonía.

Teorema (Monotonía)

Si $\Sigma \models \psi$, *entonces para cada fórmula* θ *se tiene que* $\Sigma \cup \{\theta\} \models \psi$.

Ejercicio: Demuestre el teorema.

Propiedades de la lógica proposicional: Monotonía

Sabemos que $\{p,\ p \rightarrow q\} \models q$. Usando el teorema de monotonía deducimos que $\{p,\ p \rightarrow q,\ \neg q\} \models q$. ¿Cómo es esto posible?

Propiedades de la lógica proposicional: Monotonía

Sabemos que $\{p,\ p \rightarrow q\} \models q$. Usando el teorema de monotonía deducimos que $\{p,\ p \rightarrow q,\ \neg q\} \models q$. ¿Cómo es esto posible?

¿Puede usarse la lógica proposicional para modelar razonamiento con sentido común?

Consecuencia lógica y satisfacibilidad

Existe una estrecha relación entre las nociones de consecuencia lógica y satisfacibilidad.

Teorema

$\Sigma \models \varphi$ *si y sólo si* $\Sigma \cup \{\neg\varphi\}$ *es insatisfacible.*

Además:

Teorema

Para todo conjunto finito de fórmulas Σ *y f'ormula* ϕ *existe* α *tal que* $\Sigma \models \phi$ *si y sólo si* α *es tautología.*

Ejercicio: Demuestre los teoremas.

Además:

Teorema

Σ *es insatisfacible si y sólo si para cada fórmula* ϕ, $\Sigma \models \phi$.

Ejercicio: Demuestre el teorema.

Además:

Teorema

Σ es insatisfacible si y sólo si para cada fórmula ϕ, $\Sigma \models \phi$.

Ejercicio: Demuestre el teorema.

O equivalentemente, Σ es insatisfacible si y sólo si para cualquier fórmula insatisfacible ϕ, $\Sigma \models \phi$.