

Lógica Proposicional

IIC1253

Inicio de la lógica

Originalmente, la lógica trataba con argumentos en el lenguaje natural.

Ejemplo

¿Es el siguiente argumento válido?

Todos los hombres son mortales.

Sócrates es hombre.

Por lo tanto, Sócrates es mortal.

La lógica debería poder usarse para demostrar que sí.

Ejemplo

¿Qué pasa con el siguiente caso?

Algunas personas son mujeres.

Sócrates es una persona.

Por lo tanto, Sócrates es mujer.

En este caso deberíamos decir que el argumento no es válido.

Pero los argumentos pueden ser más complejos ...

Creo que todos los hombres son mortales.

Creo que Sócrates es hombre.

Por lo tanto, creo que Sócrates es mortal.

¿Es este argumento válido? ¿Por qué?

¿Qué significa **creo**? ¿Qué pasaría si reemplazamos **creo que** por **no se si**?

Un día de la próxima semana les voy a hacer una interrogación,
y les aseguro que el día que se las haga van a estar sorprendidos.

¿Qué día voy a hacer la interrogación?

Matemática en el lenguaje natural: Paradoja de Berry

Podemos representar los números naturales usando oraciones del lenguaje natural: “Mil quinientos veinte”, “el primer número”, ...

El número de palabras en el Diccionario de la Real Academia es finito.

El número de oraciones con a los más 50 palabras también es finito.

Matemática en el lenguaje natural: Paradoja de Berry

Sea B el siguiente número natural:

El primer número natural que no puede ser definido por una oración con a lo más cincuenta palabras tomadas del Diccionario de la Real Academia.

B está bien definido, pero con sólo 25 palabras. ¡Tenemos una contradicción!

- ¿Por qué se produjo esta paradoja?

Más paradojas: Russell (1902)

También pueden aparecer paradojas usando lenguaje matemático.

Sea $A = \{1, 2, 3\}$

► ¿ $A \in A$? No

Sea $B = \{\{1, 2, 3\}, \{4, 5\}\}$

► ¿ $A \in B$? Sí

► ¿ $B \in B$? No

Más paradojas: Russell (1902)

Sea C el conjunto de todos los conjuntos que tienen a lo menos dos elementos: $C = \{A, B, \dots\}$

► ¿ $C \in C$? Sí

Entonces podemos definir el siguiente conjunto: $U = \{X \mid X \notin X\}$

► Tenemos: $A \in U$, $B \in U$, $C \notin U$.

¿ $U \in U$? Por definición, $U \in U$ si y sólo si $U \notin U$. ¡Tenemos una contradicción!

¿Cómo definimos la noción de conjunto?

¿Por qué necesitamos la lógica?

Necesitamos un lenguaje con una sintaxis precisa y una semántica bien definida.

Queremos usar este lenguaje en matemáticas.

- ▶ Definición de objetos matemáticos: conjunto, números naturales, números reales.
- ▶ Definición de teorías matemáticas: teoría de conjuntos, teoría de los números naturales.
- ▶ Definición del concepto de demostración.

También queremos usar este lenguaje en ingeniería. ¿Por qué?

Lógica proposicional: sintaxis

Tenemos los siguientes elementos:

- ▶ Variables proposicionales (P): $p, q, r, \dots$
- ▶ 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

$$P = \{socrates_es_hombre, socrates_es_mortal\}$$

Lógica proposicional: sintaxis

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

Ejemplo

socrates_es_hombre $\rightarrow$ *socrates_es_mortal*

socrates_es_hombre $\rightarrow$ ($\neg$ *socrates_es_mortal*)

Símbolos de puntuación son usados para evitar ambigüedades.

Sintaxis de la lógica proposicional: Definición

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.

La naturaleza de la definición es inductiva.

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

Definiciones inductivas

Queremos definir una función la que indica cuantos 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.

Demostraciones inductivas

Lo siguiente parece ser cierto: Cada fórmula contiene el mismo número de paréntesis izquierdos y derechos.

$$pi(\varphi) = pd(\varphi), \text{ para cada fórmula } \varphi.$$

¿Cómo podemos demostrar esto?

Podemos usar inducción ...

Inducción en la lógica proposicional

Principio de inducción: Para cada $A \subseteq L(P)$ tal que

Caso base : $p \in A$, para cada $p \in P$,

Caso inductivo : si $\varphi, \psi \in A$, entonces $(\neg\varphi) \in A$ y $(\varphi \star \psi) \in A$, donde $\star \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$,

se tiene que $A = L(P)$.

¿Por qué funciona?

Ejercicio

Demuestre que cada fórmula contiene el mismo número de paréntesis izquierdos y derechos.

Inducción en la lógica proposicional: Ejercicios

1. Defina $v(\varphi)$ como el número de ocurrencias de variables proposicionales en φ .
2. Demuestre que para cada fórmula proposicional φ que no contiene el símbolo $\neg$ se tiene que $la(\varphi) \leq 4 \cdot v(\varphi)^2$.

¿Qué sucede si φ contiene el símbolo $\neg$?

¿Qué sucede si las fórmulas de la forma $(\neg(\neg\varphi))$ no son permitidas?

3. Demuestre que un prefijo propio de una fórmula no es una fórmula.

Eliminación de paréntesis

Si eliminamos paréntesis de una fórmula entonces no necesariamente vamos a tener una manera única de leer la fórmula

- ▶ Vamos a eliminar paréntesis para simplificar la notación, pero teniendo cuidado de no producir ambigüedades

Ejemplo

Tenemos dos maneras de leer la expresión $\alpha = (p \wedge q \wedge r)$:

- ▶ $\alpha = (\beta_1 \wedge \gamma_1)$, donde $\beta_1 = p$ y $\gamma_1 = q \wedge r$
- ▶ $\alpha = (\beta_2 \wedge \gamma_2)$, donde $\beta_2 = p \wedge q$ y $\gamma_2 = r$

Notación

Vamos a omitir los paréntesis exteriores de una fórmula: Usamos $p \wedge q$ en lugar de $(p \wedge q)$

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): $\sigma : P \rightarrow \{0, 1\}$

Ejemplo

$\sigma(\text{socrates_es_hombre}) = 1$ y $\sigma(\text{socrates_es_mortal}) = 0$

Semántica: Definición

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

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

Definición

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

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

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

- ▶ 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 (continuación)

- ▶ 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 $\varphi = (\alpha \rightarrow \beta)$, 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

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

Semántica: Ejemplos

Supongamos que $\sigma(\text{socrates_es_hombre}) = 1$ y
 $\sigma(\text{socrates_es_mortal}) = 0$.

Entonces:

$$\hat{\sigma}(\text{socrates_es_hombre} \rightarrow \text{socrates_es_mortal}) = 0$$

$$\hat{\sigma}(((\text{socrates_es_hombre} \rightarrow \text{socrates_es_mortal}) \wedge \\ \text{socrates_es_hombre}) \rightarrow \text{socrates_es_mortal})) = 1$$

Tablas de verdad

Cada fórmula se puede representar y analizar en una tabla de verdad.

p	q	$\neg p$	$p \vee q$	$p \wedge q$	$p \rightarrow q$	$p \leftrightarrow q$
0	0	1	0	0	1	1
0	1	1	1	0	1	0
1	0	0	1	0	0	0
1	1	0	1	1	1	1

Ejercicio

Suponga que P contiene n variables. ¿Cuántas tablas de verdad distintas existen para $L(P)$?

Tablas de verdad: Comparación de fórmulas

Podemos comparar fórmulas usando tablas de verdad:

p	q	r	$(p \wedge q) \wedge r$	$p \wedge (q \wedge r)$
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	0	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Concluimos que $\wedge$ es asociativo (vamos a enunciar esto de manera formal más adelante)

Notación

Vamos a omitir paréntesis para las fórmulas que sólo contienen $\wedge$: Usamos $p \wedge q \wedge r$ en lugar de $(p \wedge q) \wedge r$ y $p \wedge (q \wedge r)$

Tablas de verdad: Comparación de fórmulas

Otra comparación de fórmulas:

p	q	r	$(p \vee q) \vee r$	$p \vee (q \vee r)$
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Concluimos que $\vee$ también es asociativo (también vamos a enunciar esto de manera formal más adelante)

Notación

Vamos a omitir paréntesis para las fórmulas que sólo contienen $\vee$: Usamos $p \vee q \vee r$ en lugar de $(p \vee q) \vee r$ y $p \vee (q \vee r)$

Conectivos ternarios

Queremos definir el conectivo lógico: **si p entonces q si no r**

p	q	r	si p entonces q si no r
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

¿Cómo se puede representar este conectivo usando $\neg$, $\wedge$ y $\rightarrow$?

Conectivos ternarios (continuación)

Solución: $(p \rightarrow q) \wedge ((\neg p) \rightarrow r)$

p	q	r	si p entonces q si no r	$(p \rightarrow q) \wedge ((\neg p) \rightarrow r)$
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	0	1	1
1	1	1	1	1

¿Por qué el conectivo es equivalente a la fórmula?

- Porque tienen la misma tabla de verdad

Conectivos n -arios

Usando tablas de verdad podemos definir conectivos n -arios:

$C(p_1, \dots, p_n)$.

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

¿Es posible representar $C(p_1, \dots, p_n)$ usando $\neg$, $\vee$, $\wedge$, $\rightarrow$ y $\leftrightarrow$?

Conectivos n -arios

Veamos un ejemplo: $C_1(p, q, r, s)$.

p	q	r	s	$C_1(p, q, r, s)$	p	q	r	s	$C_1(p, q, r, s)$
0	0	0	0	0	1	0	0	0	1
0	0	0	1	1	1	0	0	1	0
0	0	1	0	0	1	0	1	0	0
0	0	1	1	0	1	0	1	1	0
0	1	0	0	0	1	1	0	0	0
0	1	0	1	0	1	1	0	1	0
0	1	1	0	1	1	1	1	0	1
0	1	1	1	0	1	1	1	1	0

¿Cómo definimos $C_1(p, q, r, s)$ usando $\neg$, $\vee$, $\wedge$, $\rightarrow$ y $\leftrightarrow$?

Solución: $C_1(p, q, r, s)$ es equivalente a la siguiente fórmula

$$((\neg p) \wedge (\neg q) \wedge (\neg r) \wedge s) \vee ((\neg p) \wedge q \wedge r \wedge (\neg s)) \vee \\ (p \wedge (\neg q) \wedge (\neg r) \wedge (\neg s)) \vee (p \wedge q \wedge r \wedge (\neg s))$$

Notación

Desde ahora en adelante $\neg$ tiene mayor precedencia que los conectivos binarios. Así por ejemplo, $(\neg p) \rightarrow q$ es lo mismo que $\neg p \rightarrow q$ y la fórmula anterior es lo mismo que:

$$(\neg p \wedge \neg q \wedge \neg r \wedge s) \vee (\neg p \wedge q \wedge r \wedge \neg s) \vee \\ (p \wedge \neg q \wedge \neg r \wedge \neg s) \vee (p \wedge q \wedge r \wedge \neg s)$$

Conectivos n -arios

Solución a nuestro problema original:

Suponiendo que σ_i es la valuación correspondiente a la fila i de la tabla de verdad de $C(p_1, \dots, p_n)$, 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.

Satisfacción de una fórmula

Definición

Una fórmula φ es satisfacible si existe una valuación σ tal que $\hat{\sigma}(\varphi) = 1$.

Ejemplo

Las siguientes fórmulas son satisfacibles:

$$(p \vee q) \rightarrow r$$

$$p \rightarrow \neg p$$

Las siguientes fórmulas no son satisfacibles:

$$p \wedge \neg p$$

$$(p \vee q) \leftrightarrow \neg(p \vee q)$$

El problema de satisfacción y el poder expresivo de la lógica proposicional

Problema: Dada una fórmula φ , queremos verificar si φ es satisfacible.

- ▶ ¿Por qué nos interesa este problema? ¿Es este un problema importante?
- ▶ ¿Es este un problema difícil? ¿Puede dar un algoritmo para resolver este problema?

El problema de satisfacción es un problema fundamental tanto en ciencia de la computación como en ingeniería.

- ▶ Muchos otros problemas pueden ser resueltos usando este problema, lo que muestra el poder expresivo de la lógica proposicional.

El problema de satisfacción y problemas en grafos

Paréntesis de teoría de conjuntos

Dados conjuntos A y B :

$$A \times B = \{(a, b) \mid a \in A \text{ y } b \in B\}$$

Por ejemplo, si $A = \{1, 2\}$ y $B = \{1, 3, 4\}$:

$$A \times B = \{(1, 1), (1, 3), (1, 4), (2, 1), (2, 3), (2, 4)\}$$

$$B \times A = \{(1, 1), (1, 2), (3, 1), (3, 2), (4, 1), (4, 2)\}$$

Nótese que $A \times B \neq B \times A$

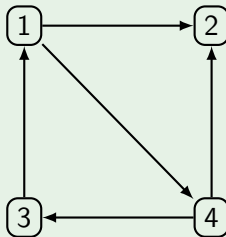
El problema de satisfacción y problemas en grafos

Un grafo es una tupla $G = (N, A)$, donde:

- ▶ N es un conjunto de nodos
- ▶ $A \subseteq (N \times N)$ es un conjunto de arcos

Ejemplo

$N = \{1, 2, 3, 4\}$ y $A = \{(1, 2), (1, 4), (3, 1), (4, 2), (4, 3)\}$:

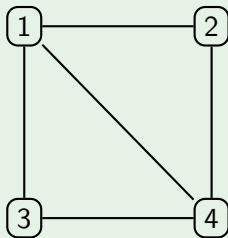


El problema de satisfacción y problemas en grafos

Un grafo $G = (N, A)$ es no-dirigido si para cada $(u, v) \in A$, se tiene que $(v, u) \in A$.

Ejemplo

$N = \{1, 2, 3, 4\}$ y $A = \{(1, 2), (2, 1), (1, 4), (4, 1), (3, 1), (1, 3), (4, 2), (2, 4), (4, 3), (3, 4)\}$:



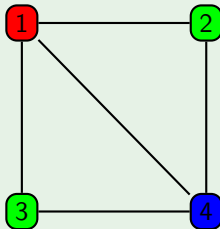
El problema de satisfacción y la coloración de grafos

Un grafo no-dirigido $G = (N, A)$ es 3-coloreable si existe una función $f : N \rightarrow \{R, G, B\}$ tal que:

Para cada $(u, v) \in A$: $f(u) \neq f(v)$

Vale decir, nodos adyacentes tienen que ser pintados con colores distintos.

Ejemplo



El problema de satisfacción y la coloración de grafos

Vamos a mostrar como codificar el problema de 3-coloración de un grafo como un problema de satisfacción de una fórmula.

Supongamos que $G = (N, A)$, donde $N = \{1, \dots, n\}$

Para codificar el problema de 3-coloración de G consideramos las siguientes variables proposicionales:

- $a_{i,j}$: indica que hay un arco de i a j , donde $i, j \in \{1, \dots, n\}$
- $c_{i,x}$: indica que el color asignado al nodo i es x , donde $i \in \{1, \dots, n\}$ y $x \in \{R, G, B\}$

El problema de satisfacción y la coloración de grafos

Utilizamos las siguientes fórmulas para codificar el problema:

- Fórmula que representa G :

$$\left(\bigwedge_{(i,j) \in A} a_{i,j} \right) \wedge \left(\bigwedge_{(i,j) \notin A} \neg a_{i,j} \right)$$

- Fórmula que indica que cada nodo debe tener asociado un único color:

$$\left(\bigwedge_{i=1}^n \bigvee_{x \in \{R,G,B\}} c_{i,x} \right) \wedge \left(\bigwedge_{i=1}^n c_{i,R} \rightarrow (\neg c_{i,G} \wedge \neg c_{i,B}) \right) \wedge \\ \left(\bigwedge_{i=1}^n c_{i,G} \rightarrow (\neg c_{i,R} \wedge \neg c_{i,B}) \right) \wedge \left(\bigwedge_{i=1}^n c_{i,B} \rightarrow (\neg c_{i,R} \wedge \neg c_{i,G}) \right)$$

El problema de satisfacción y la coloración de grafos

- Fórmula que indica que dos nodos adyacentes deben ser pintados con colores distintos:

$$\bigwedge_{i=1}^n \bigwedge_{j=1}^n \bigwedge_{x \in \{R, G, B\}} (a_{i,j} \wedge c_{i,x}) \rightarrow \neg c_{j,x}$$

Si la fórmula φ_G es la conjunción de las fórmulas anteriores, entonces:

G es 3-coloreable si y sólo si φ_G es satisfacible

El problema de satisfacción y la existencia de cliques

Otro problema sobre grafos: **Existencia de cliques**

- ▶ Este es un problema muy útil cuando analizamos redes sociales

Dados grafos $G = (N, A)$, $G' = (N', A')$ tales que $N' \subseteq N$ y $A' \subseteq A$, y dado un número k , decimos que G' es un clique de G de tamaño k si:

- ▶ N' contiene k elementos
- ▶ Para cada $u \in N'$ y $v \in N'$ tales que $u \neq v$, se tiene que $(u, v) \in A'$

Problema a resolver: Dado G y k , determinar si existe un clique de tamaño k

Ejercicio

Muestre como codificar este problema usando el problema de satisfacción.

Tautologías y contradicciones

Si una fórmula no es satisfacible, entonces decimos que es una contradicción.

- ▶ $p \wedge \neg p$ es una contradicción

Definición

Una fórmula φ es una tautología si para toda valuación σ se tiene que $\hat{\sigma}(\varphi) = 1$.

Ejemplo

Las siguientes fórmulas son tautologías:

$$p \vee \neg p$$

$$p \leftrightarrow p$$

Equivalencia de fórmulas

Definición

Dos fórmulas φ , ψ son *equivalentes*, denotado como $\varphi \equiv \psi$, si para toda valuación σ se tiene que $\hat{\sigma}(\varphi) = \hat{\sigma}(\psi)$.

Una definición alternativa de la noción de equivalencia:

φ , ψ son equivalentes si $\varphi \leftrightarrow \psi$ es una tautología.

Ejercicio

1. Demuestre que las definiciones anteriores coinciden.
2. Defina la noción de equivalencia usando tablas de verdad.
 - ¿Qué pasa si las fórmulas no usan las mismas variables? ¿Puede ocurrir esto?

Algunas equivalencias útiles

Ley de la doble negación:

$$\neg(\neg\varphi) \equiv \varphi$$

Leyes de De Morgan:

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

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

Leyes de conmutatividad:

$$\varphi \wedge \psi \equiv \psi \wedge \varphi$$

$$\varphi \vee \psi \equiv \psi \vee \varphi$$

Algunas equivalencias útiles

Leyes de asociatividad:

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

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

Leyes de distributividad:

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

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

Ley de implicancia:

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

Ley de doble implicancia:

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

Ejercicios

1. Demuestre las leyes enunciadas en las transparencias anteriores.
2. ¿Es $\rightarrow$ asociativo? Vale decir, ¿Es cierto que $(\alpha \rightarrow \beta) \rightarrow \gamma \equiv \alpha \rightarrow (\beta \rightarrow \gamma)$?
3. ¿Cuántas fórmulas no equivalentes puede construir con n variables proposicionales?

Formas normales: DNF

Decimos que una fórmula φ está en **forma normal disyuntiva (DNF)** si φ es de la forma:

$$\bigvee_{i=1}^m \left(\bigwedge_{j=1}^{n_i} l_{i,j} \right),$$

donde cada $l_{i,j}$ es un **literal**, es decir, una variable proposicional o la negación de una variable proposicional.

Ejemplo

$$(p \wedge q) \vee (\neg p \wedge r \wedge s)$$

Teorema

Toda fórmula es equivalente a una fórmula en DNF.

Ya demostramos este teorema, ¿cierto?

Formas normales: CNF

Decimos que una fórmula φ está en **forma normal conjuntiva (CNF)** si φ es de la forma:

$$\bigwedge_{i=1}^m \left(\bigvee_{j=1}^{n_i} l_{i,j} \right),$$

donde cada $l_{i,j}$ es un literal.

Ejemplo

$$(p \vee \neg q) \wedge (\neg p \vee \neg r \vee s) \wedge (\neg r \vee s)$$

Teorema

Toda fórmula es equivalente a una fórmula en CNF.

Ejercicio

Haga dos demostraciones del teorema.

- ▶ En la primera sólo utilice las leyes de equivalencia. ¿Qué leyes necesita utilizar?
- ▶ En la segunda utilice el resultado de que toda fórmula es equivalente a una fórmula en DNF. ¿Qué leyes de equivalencia necesita utilizar en este caso?

Conectivos funcionalmente completos

Definición

Un conjunto C de conectivos es **funcionalmente completo** si para conjunto P de variables proposicionales y para cada $\varphi \in L(P)$, existe una fórmula ψ tal que ψ sólo usa los conectivos en C y $\varphi \equiv \psi$.

Ya demostramos que $\{\neg, \vee, \wedge\}$ es funcionalmente completo. ¿Son $\{\neg, \vee\}$ y $\{\neg, \wedge\}$ funcionalmente completos?

Ejercicio

1. Demuestre que $\{\neg, \rightarrow\}$ es funcionalmente completo.
2. Demuestre que $\{\neg\}$ no es funcionalmente completo.
3. ¿Es $\{\wedge, \vee, \rightarrow, \leftrightarrow\}$ funcionalmente completo?

Conectivos funcionalmente completos

El conectivo NAND se define como:

p	q	$p \text{ NAND } q$
0	0	1
0	1	1
1	0	1
1	1	0

¿A qué corresponde este conectivo?

NAND es funcionalmente completo

Teorema

$\{\text{NAND}\}$ es funcionalmente completo.

Para demostrar el teorema usamos el hecho de que $\{\neg, \wedge\}$ es funcionalmente completo y las siguientes equivalencias:

$$\neg\varphi \quad \equiv \quad \varphi \text{ NAND } \varphi$$

$$\begin{aligned} \varphi \wedge \psi &\equiv \neg(\varphi \text{ NAND } \psi) \\ &\equiv (\varphi \text{ NAND } \psi) \text{ NAND } (\varphi \text{ NAND } \psi) \end{aligned}$$

La noción de consecuencia lógica

Una valuación σ satisface un conjunto de fórmulas Σ si para cada $\varphi \in \Sigma$, se tiene que $\hat{\sigma}(\varphi) = 1$.

Notación: $\hat{\sigma}(\Sigma) = 1$

¿Cuándo decimos que una fórmula ψ se deduce desde Σ ?

Definición

ψ es *consecuencia lógica* de Σ si para cada valuación σ tal que $\hat{\sigma}(\Sigma) = 1$, se tiene que $\hat{\sigma}(\psi) = 1$.

Notación: $\Sigma \models \psi$

La noción de consecuencia lógica: Ejemplos

Modus ponens:

$$\{p, p \rightarrow q\} \models q$$

Demostración por partes:

$$\{p \vee q \vee r, p \rightarrow s, q \rightarrow s, r \rightarrow s\} \models s$$

Ejercicio

1. Demuestre que si $\Sigma \models \alpha \wedge \beta$, entonces $\Sigma \models \alpha$ y $\Sigma \models \beta$.
2. ¿Es cierto que si $\Sigma \models \alpha \vee \beta$, entonces $\Sigma \models \alpha$ o $\Sigma \models \beta$?

Ejercicios finales

1. Un conjunto de fórmulas Σ es satisfacible si existe una valuación σ tal que $\hat{\sigma}(\Sigma) = 1$. En caso contrario, Σ es inconsistente (o contradictorio).
Demuestre que $\Sigma \models \varphi$ si y sólo si $\Sigma \cup \{\neg\varphi\}$ es inconsistente.
2. Encuentre una fórmula ψ tal que Σ es satisfacible si y sólo si $\Sigma \not\models \psi$.
3. Demuestre que si Σ es un conjunto finito de fórmulas, entonces el problema de verificar si $\Sigma \models \varphi$ puede reducirse al problema de verificar si una fórmula es una tautología.
4. Demuestre que φ es una tautología si y solo si $\emptyset \models \varphi$.