

# Consecuencia lógica y resolución proposicional

Clase 6

IIC 1253

Prof. Sebastián Buggedo

# Outline

**Obertura**

Consecuencia lógica

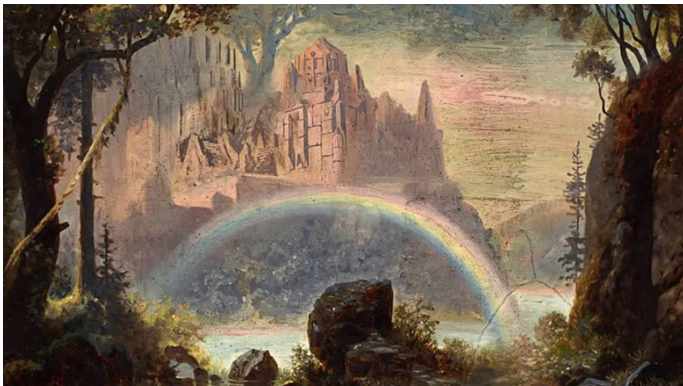
Resolución proposicional

Epílogo

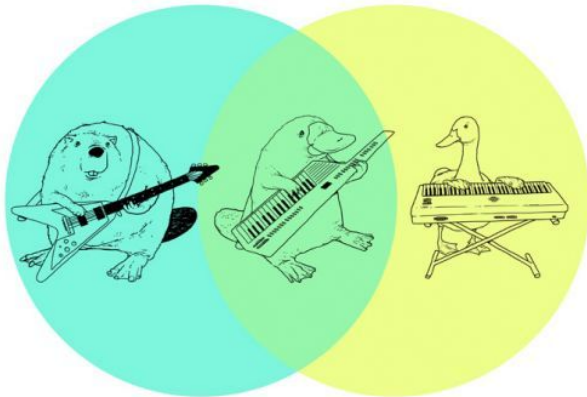


# Primer Acto: Fundamentos

## Inducción y lógica



# Playlist Primer Acto



Playlist del curso: DiscretiWawos

Además sigan en instagram:

@orquesta\_tamen

# Formas normales

## Definición

Decimos que una fórmula  $\varphi$  está en **forma normal disyuntiva (DNF)** si es una disyunción de conjunciones de literales; o sea, si es de la forma

$$B_1 \vee B_2 \vee \dots \vee B_k$$

donde cada  $B_i$  es una conjunción de literales,  $B_i = (l_{i1} \wedge \dots \wedge l_{ik_i})$

## Teorema

Toda fórmula es equivalente a una fórmula en DNF.

# Formas normales

## Definición

Decimos que una fórmula  $\varphi$  está en **forma normal conjuntiva (CNF)** si es una conjunción de disyunciones de literales; o sea, si es de la forma

$$C_1 \wedge C_2 \wedge \dots \wedge C_k$$

donde cada  $C_i$  es una disyunción de literales,  $C_i = (l_{i1} \vee \dots \vee l_{ik_i})$

## Teorema

Toda fórmula es equivalente a una fórmula en CNF.

# Objetivos de la clase

- Comprender el concepto de consecuencia lógica
- Demostrar consecuencias sencillas
- Comprender el método de resolución
- Demostrar consecuencias lógicas usando resolución



# Outline

Obertura

**Consecuencia lógica**

Resolución proposicional

Epílogo

# Conjuntos de fórmulas

## Notación

Dado un conjunto de fórmulas  $\Sigma$  en  $L(P)$ , diremos que una valuación  $\sigma$  **satisface**  $\Sigma$ , denotado por  $\sigma(\Sigma) = 1$ , si para toda fórmula  $\varphi \in \Sigma$  se tiene que  $\sigma(\varphi) = 1$ .

## Definición

Un conjunto de fórmulas  $\Sigma$  es **satisfacible** si existe una valuación  $\sigma$  tal que  $\sigma(\Sigma) = 1$ . En caso contrario,  $\Sigma$  es **inconsistente**.

¿Cuándo decimos que una fórmula se **deduce** de un conjunto?

# Consecuencia lógica

## Definición

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

Lo denotamos por  $\Sigma \models \psi$ .

$\psi$  debe ser satisfecha en cada “*mundo*” donde  $\Sigma$  es verdadero

# Consecuencia lógica

## Ejemplo

La regla de inferencia llamada **Modus ponens** es  $\{p, p \rightarrow q\} \models q$

$p$	$q$	$p$	$p \rightarrow q$	$q$
0	0	0	1	0
0	1	0	1	1
1	0	1	0	0
1	1	1	1	1

Nos tenemos que fijar en las valuaciones que satisfacen al conjunto...  
En esos mundos, la fórmula “*objetivo*” también debe ser satisfecha

## Ejercicio (propuesto ★)

Demuestre las siguientes reglas de inferencia

- **Modus tollens:**  $\{\neg q, p \rightarrow q\} \models \neg p$
- **Demostración por partes:**  $\{p \vee q \vee r, p \rightarrow s, q \rightarrow s, r \rightarrow s\} \models s$

# Un resultado fundamental

## Teorema

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

Este teorema combina dos mundos:  
la **consecuencia lógica** y la **satisfacibilidad**

# Un resultado fundamental

## Teorema

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

## Demostración

( $\Rightarrow$ ) Supongamos que  $\Sigma \models \varphi$ . Por contradicción, supongamos que  $\Sigma \cup \{\neg\varphi\}$  es satisfacible, luego existe una valuación  $\sigma$  tal que  $\sigma(\Sigma \cup \{\neg\varphi\}) = 1$ . Esto implica que  $\sigma(\Sigma) = 1$  y que  $\sigma(\neg\varphi) = 1$ , y por lo tanto  $\sigma(\Sigma) = 1$  y  $\sigma(\varphi) = 0$ , lo que contradice que  $\Sigma \models \varphi$ .

( $\Leftarrow$ ) Sea  $\Sigma \cup \{\neg\varphi\}$  inconsistente. Debemos demostrar que dada una valuación  $\sigma$  tal que  $\sigma(\Sigma) = 1$ , se tiene que  $\sigma(\varphi) = 1$ . Como  $\Sigma \cup \{\neg\varphi\}$  es inconsistente y  $\sigma(\Sigma) = 1$ , necesariamente  $\sigma(\neg\varphi) = 0$ , y luego  $\sigma(\varphi) = 1$ . Hemos demostrado que si  $\sigma$  es tal que  $\sigma(\Sigma) = 1$ , entonces  $\sigma(\varphi) = 1$ , por lo que concluimos que  $\Sigma \models \varphi$ . □

# Un resultado fundamental

El teorema anterior nos permite chequear  $\Sigma \models \varphi$  estudiando la satisfacibilidad de  $\Sigma \cup \{\neg\varphi\}$

- Podemos usar tablas de verdad para esto último...
- ...pero es muy lento!

Estudiaremos un método alternativo que no requiere tablas de verdad

# Outline

Obertura

Consecuencia lógica

**Resolución proposicional**

Epílogo



# Primer ingrediente: Cláusula vacía

Recordemos: una **cláusula** es una disyunción de literales

Notación

Denotaremos por  $\square$  una contradicción cualquiera. La llamaremos **cláusula vacía**

Teorema

Un conjunto de fórmulas  $\Sigma$  es inconsistente si y sólo si  $\Sigma \models \square$ .

# Primer ingrediente: Cláusula vacía

## Teorema

Un conjunto de fórmulas  $\Sigma$  es inconsistente si y sólo si  $\Sigma \models \Box$ .

### Demostración (propuesta ★)

( $\Rightarrow$ ) Dado que  $\Sigma$  es inconsistente, debemos demostrar que  $\Sigma \models \Box$ . Como  $\Sigma$  es inconsistente, sabemos que toda valuación  $\sigma$  es tal que  $\sigma(\Sigma) = 0$ , y luego se cumple trivialmente que  $\Sigma \models \Box$ .

( $\Leftarrow$ ) Dado que  $\Sigma \models \Box$ , debemos demostrar que  $\Sigma$  es inconsistente. Por contradicción, supongamos que  $\Sigma$  es satisfacible. Luego, existe una valuación  $\sigma$  tal que  $\sigma(\Sigma) = 1$ . Como  $\Box$  es una contradicción, tenemos que  $\sigma(\Box) = 0$ , y por lo tanto obtenemos que  $\sigma(\Sigma) = 1$  pero  $\sigma(\Box) = 0$ , lo que contradice que  $\Sigma \models \Box$ .

## Segundo ingrediente: conjuntos equivalentes

### Definición

Los conjuntos de fórmulas  $\Sigma_1$  y  $\Sigma_2$  son **lógicamente equivalentes** ( $\Sigma_1 \equiv \Sigma_2$ ) si para toda valuación  $\sigma$  se tiene que  $\sigma(\Sigma_1) = \sigma(\Sigma_2)$ .

### Observaciones

- Diremos que  $\Sigma$  es lógicamente equivalente a una fórmula  $\varphi$  si

$$\Sigma \equiv \{\varphi\}$$

- Para todo  $\Sigma$  se cumple

$$\Sigma \equiv \bigwedge_{\varphi \in \Sigma} \varphi$$

## Segundo ingrediente: conjuntos equivalentes

### Teorema

Todo conjunto de fórmulas  $\Sigma$  es equivalente a un conjunto de cláusulas.

### Ejemplo

Pasando las fórmulas de un conjunto  $\Sigma$  a CNF, podemos separar sus cláusulas

$$\{p, q \rightarrow (p \rightarrow r), \neg(q \rightarrow r)\} \equiv \{p, \neg q \vee \neg p \vee r, q, \neg r\}$$

obteniendo un **conjunto de cláusulas** que es equivalente al original.

Para determinar si  $\Sigma \models \varphi$ , construiremos un conjunto de cláusulas

$$\Sigma' \equiv \Sigma \cup \{\neg\varphi\}$$

# La regla de resolución

## Notación

Si un literal  $\ell = p$ , entonces  $\bar{\ell} = \neg p$ , y si  $\ell = \neg p$ , entonces  $\bar{\ell} = p$ .

## Regla de resolución

Dadas cláusulas  $C_1, C_2$  y un literal  $\ell$ ,

$$\frac{C_1 \vee \ell \vee C_2 \quad C_3 \vee \bar{\ell} \vee C_4}{C_1 \vee C_2 \vee C_3 \vee C_4}$$

## Observaciones

- La regla es **correcta**:  $\{C_1 \vee \ell \vee C_2, C_3 \vee \bar{\ell} \vee C_4\} \models C_1 \vee C_2 \vee C_3 \vee C_4$
- $\ell$  y  $\bar{\ell}$  se llaman literales **complementarios**

# La regla de resolución

## Regla de resolución

Dadas cláusulas  $C_1, C_2$  y un literal  $\ell$ ,

$$\frac{\begin{array}{c} C_1 \vee \ell \vee C_2 \\ C_3 \vee \bar{\ell} \vee C_4 \end{array}}{C_1 \vee C_2 \vee C_3 \vee C_4}$$

## Ejemplo

Algunos casos particulares de resolución

$$\frac{\begin{array}{c} C_1 \vee \ell \vee C_2 \\ \bar{\ell} \end{array}}{C_1 \vee C_2} \qquad \frac{\begin{array}{c} \ell \\ \bar{\ell} \end{array}}{\square}$$

# La regla de factorización

## Regla de factorización

Dadas cláusulas  $C_1, C_2, C_3$  y un literal  $\ell$ ,

$$\frac{C_1 \vee \ell \vee C_2 \vee \ell \vee C_3}{C_1 \vee \ell \vee C_2 \vee C_3}$$

## Observación

- La regla es **correcta**:  $\{C_1 \vee \ell \vee C_2 \vee \ell \vee C_3\} \models C_1 \vee \ell \vee C_2 \vee C_3$

# Demostraciones por resolución

## Definición

Una **demostración por resolución** de que  $\Sigma$  es inconsistente es una secuencia de cláusulas  $C_1, \dots, C_n$  tal que

- Para cada  $i \leq n$ 
  - $C_i \in \Sigma$  o
  - existen  $j, k < i$  tales que  $C_i$  se obtiene de  $C_j, C_k$  usando la regla de resolución o
  - existe  $j < i$  tal que  $C_i$  se obtiene de  $C_j$  usando la regla de factorización
- $C_n = \square$

Lo denotamos por  $\Sigma \vdash \square$ .



# Resolución proposicional

## Ejemplo

$$\Sigma = \{p \vee q \vee r, \neg p \vee s, \neg q \vee s, \neg r \vee s, \neg s\}$$

- (1)  $p \vee q \vee r \in \Sigma$
- (2)  $\neg p \vee s \in \Sigma$
- (3)  $s \vee q \vee r$  resolución de (1), (2)
- (4)  $\neg q \vee s \in \Sigma$
- (5)  $s \vee s \vee r$  resolución de (3), (4)
- (6)  $s \vee r$  factorización de (5)
- (7)  $\neg r \vee s \in \Sigma$
- (8)  $s \vee s$  resolución de (6), (7)
- (9)  $s$  factorización de (8)
- (10)  $\neg s \in \Sigma$
- (11)  $\square$  resolución de (9), (10)

Es decir, existe una demostración por resolución de que  $\Sigma$  es inconsistente

# Resolución proposicional

## Teorema

Dado un conjunto de cláusulas  $\Sigma$ , se tiene que:

- **Correctitud:** Si  $\Sigma \vdash \square$  entonces  $\Sigma \models \square$ .
- **Compleitud:** Si  $\Sigma \models \square$  entonces  $\Sigma \vdash \square$ .

## Corolario

Si  $\Sigma$  es un conjunto de cláusulas, entonces  $\Sigma \models \square$  si y sólo si  $\Sigma \vdash \square$ .

## Corolario (forma alternativa)

Un conjunto de cláusulas  $\Sigma$  es inconsistente si y sólo si existe una demostración por resolución de que es inconsistente.

¡Resolución resuelve nuestro problema!

# Resolución proposicional

¡Resolución resuelve nuestro problema de consecuencia lógica!

Corolario

Dados un conjunto de fórmulas  $\Sigma$  y una fórmula  $\varphi$  cualesquiera,

$$\Sigma \models \varphi \text{ si y sólo si } \Sigma' \vdash \square$$

donde  $\Sigma'$  es un conjunto de cláusulas tal que  $\Sigma \cup \{\neg\varphi\} \equiv \Sigma'$ .

Este procedimiento nos permite determinar consecuencia lógica

# Resolución proposicional

## Ejemplo

Demostremos que  $\{p, q \rightarrow (p \rightarrow r)\} \models q \rightarrow r$ .

Seguiremos la estrategia planteada

1. Agregar  $\neg\varphi$  al conjunto
2. Transformar todo en  $\Sigma \cup \{\neg\varphi\}$  a CNF y separar cláusulas
3. Obtener una secuencia de cláusulas por resolución para llegar a  $\square$

El desarrollo se deja propuesto ★

# Resolución proposicional

## Ejemplo

Consideremos el conjunto  $\{p, q \rightarrow (p \rightarrow r), \neg(q \rightarrow r)\}$  y llamamos

- $\varphi = q \rightarrow (p \rightarrow r)$

- $\psi = \neg(q \rightarrow r)$

Transformamos cada una a conjuntos de cláusulas

- Para  $\varphi$  usamos ley de implicancia material (dos veces) y asociatividad

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

- Para  $\psi$  usamos implicancia y de Morgan

$$\psi = \neg(q \rightarrow r) \equiv \neg(\neg q \vee r) \equiv q \wedge \neg r \equiv \{q, \neg r\}$$

# Resolución proposicional

## Ejemplo

Tenemos el conjunto de cláusulas

$$\Sigma = \{p, \neg q \vee \neg p \vee r, q, \neg r\}.$$

Basta demostrar que  $\Sigma$  es inconsistente, y como es un conjunto de cláusulas, lo haremos mostrando que  $\Sigma \vdash \square$ :

- |     |                             |                        |
|-----|-----------------------------|------------------------|
| (1) | $p$                         | $\in \Sigma$           |
| (2) | $\neg q \vee \neg p \vee r$ | $\in \Sigma$           |
| (3) | $\neg q \vee r$             | resolución de (1), (2) |
| (4) | $q$                         | $\in \Sigma$           |
| (5) | $r$                         | resolución de (3), (4) |
| (6) | $\neg r$                    | $\in \Sigma$           |
| (7) | $\square$                   | resolución de (5), (6) |

# Outline

Obertura

Consecuencia lógica

Resolución proposicional

**Epílogo**

# Objetivos de la clase

- Comprender el concepto de consecuencia lógica
- Demostrar consecuencias sencillas
- Comprender el método de resolución
- Demostrar consecuencias lógicas usando resolución