

# Satisfacibilidad y consecuencia lógica

Clase 5

IIC 1253

Prof. Sebastián Buggedo

# Outline

Obertura

Satisfacibilidad

Formas normales

Consecuencia lógica

Epílogo

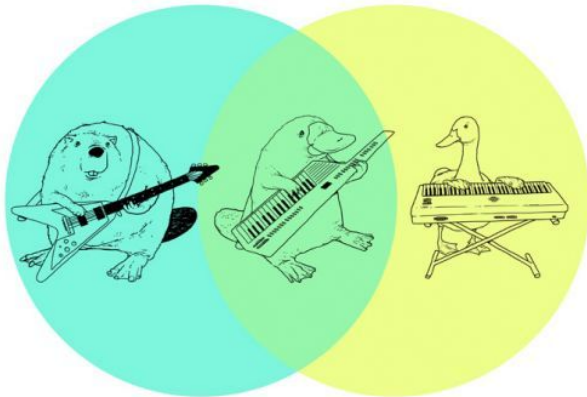


# Primer Acto: Fundamentos

## Inducción y lógica



# Playlist Primer Acto



Playlist del curso: DiscretiWawos

Además sigan en instagram:

@orquesta\_tamen

# Conectivos funcionalmente completos

## Definición

Un conjunto de conectivos lógicos se dice **funcionalmente completo** si toda fórmula en  $\mathcal{L}(P)$  es lógicamente equivalente a una fórmula que sólo usa esos conectivos.

## Ejemplo

El conjunto  $C = \{\neg, \wedge, \vee\}$  es funcionalmente completo, pues para toda fórmula  $\varphi$ , se tiene que

$$\varphi \equiv \bigvee_{\substack{j=1 \dots 2^n \\ \sigma_j(\varphi)=1}} \left( \left( \bigwedge_{\substack{i=1 \dots n \\ \sigma_j(p_i)=1}} p_i \right) \wedge \left( \bigwedge_{\substack{i=1 \dots n \\ \sigma_j(p_i)=0}} (\neg p_i) \right) \right)$$

# Conectivos funcionalmente completos

## Ejercicios

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

Ya demostramos 1.c. Ahora demostraremos 2.

El resto quedan propuestos ★!

# Conectivos funcionalmente completos

## Ejercicio 2.

Demostraremos que  $\{\neg\}$  no es funcionalmente completo.

Dado  $P = \{p\}$ , demostraremos por inducción que toda fórmula en  $\mathcal{L}(P)$  construida usando sólo  $p$  y  $\neg$  es lógicamente equivalente a  $p$  o a  $\neg p$ . Como ninguna de estas fórmulas es equivalente a  $p \wedge \neg p$ , se concluye que  $\{\neg\}$  no puede ser funcionalmente completo.

Esta demostración es “*negativa*”... daremos un caso en que **no se cumple** la definición de funcionalmente completo



# Conectivos funcionalmente completos

## Ejercicio 2.

Demostraremos la propiedad

$$P(\varphi) := \varphi \text{ es equivalente a } p \text{ o a } \neg p$$

- **BI:** Si  $\varphi = p$ , con  $p \in P$ , la propiedad se cumple trivialmente.
- **HI:** Supongamos que  $\varphi \in L(P)$  construida usando sólo  $p$  y  $\neg$  es equivalente a  $p$  o a  $\neg p$ .

Estamos acotando las tablas de verdad que son posibles en  $\{\neg\}$

# Conectivos funcionalmente completos

## Ejercicio 2.

- **TI:** El único caso inductivo que tenemos que demostrar es  $\psi = \neg\varphi$ , pues sólo podemos usar el conectivo  $\neg$ .

Por **HI**, sabemos que para toda valuación  $\sigma$  se cumple que  $\sigma(\varphi) = \sigma(p)$  o  $\sigma(\varphi) = \sigma(\neg p)$ . Podemos hacer una tabla de verdad:

$\varphi$	$\psi = \neg\varphi$
$p$	$\neg p$
$\neg p$	$p$

Concluimos que  $\psi$  es equivalente a  $p$  o a  $\neg p$ .

Como la fórmula  $\psi = p \wedge \neg p$  no es equivalente a ninguna fórmula que solo usa símbolos de  $\{\neg\}$ , tenemos que  $\{\neg\}$  no es funcionalmente completo.  $\square$

# Objetivos de la clase

- Comprender el concepto de satisfacibilidad de fórmulas y conjuntos
- Aplicar lógica para modelar problemas
- Conocer las formas normales
- Comprender concepto de consecuencia lógica
- Demostrar consecuencias lógicas sencillas

# Outline

Obertura

**Satisfacibilidad**

Formas normales

Consecuencia lógica

Epílogo

# Satisfacibilidad

## Definición

Una fórmula  $\varphi$  es **satisfacible** si existe una valuación  $\sigma$  tal que  $\sigma(\varphi) = 1$ .

## Ejemplo

Las siguientes fórmulas son satisfacibles:

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

$$p \rightarrow \neg p$$

Las siguientes fórmulas no son satisfacibles:

$$p \wedge \neg p$$

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

Una fórmula es satisfacible si hay algún “*mundo*” en el cual es verdadera

# El problema de satisfacibilidad

## Problema de satisfacibilidad (SAT)

Sea  $\varphi$  una fórmula proposicional. El **problema de satisfacibilidad** consiste en determinar si  $\varphi$  es satisfacible o no.

Este es un problema central en computación

- Permite resolver problemas fuera de la lógica...
- ... usando modelación en lógica proposicional

¿Es un problema difícil? ¿Cómo se resuelve?

# Modelación en lógica proposicional

## Ejercicio

Sea  $M$  un mapa conformado por  $n$  países. Decimos que  $M$  es 3-coloreable si se pueden pintar todos los países con 3 colores sin que ningún par de países adyacente tenga el mismo color. En otras palabras, los países vecinos deben tener colores distintos.

Dado un mapa  $M$ , construya una fórmula  $\varphi \in \mathcal{L}(P)$  tal que  $M$  es 3-coloreable si y sólo si  $\varphi$  es satisfacible.

La fórmula  $\varphi$  debe **codificar** los requisitos y estructura del problema

# Satisfacibilidad

## Ejercicio (mapa 3-coloreable)

Sea  $M$  el mapa. Consideremos la lista de países  $\{1, 2, \dots, n\}$  y una lista de pares de países adyacentes  $A = \{(i, j), (k, m), \dots\}$ .

Seguiremos la siguiente estrategia para resolver el problema

1. Definición de variables proposicionales
  - Variables predefinidas por el problema
  - Variables que hay que asignar
2. Construcción de restricciones a través de fórmulas proposicionales
3. Demostración de que  $\varphi$  cumple lo pedido (si y solo si)



# Satisfacibilidad

## Ejercicio (mapa 3-coloreable)

Primero, definimos las variables proposicionales. Usamos dos tipos de variables:

- Para  $1 \leq i, j \leq n$  definimos

$$p_{ij} = \begin{cases} 1 & \text{si } i \text{ es adyacente con } j \\ 0 & \text{si no} \end{cases}$$

Observamos que cada  $p_{ij}$  se conoce de antemano una vez que conocemos el mapa  $M$ . Debemos **inicializarlas**.

- Análogamente, para  $1 \leq i \leq n$  definimos

$$r_i \quad b_i \quad g_i$$

que valen 1 si el país  $i$  es pintado rojo, azul o verde respectivamente, y 0 en caso contrario. Estas variables deben ser **determinadas** para resolver el problema.

¿Qué restricciones son naturales para este problema?

## Ejercicio (mapa 3-coloreable)

Para representar el problema vamos a definir  $\varphi$  como la conjunción de las siguientes fórmulas.

- “Cada país tiene uno y solo un color”

$$\varphi_C = \bigwedge_{i=1}^n \left( (r_i \vee b_i \vee g_i) \wedge (r_i \rightarrow (\neg b_i \wedge \neg g_i)) \wedge (b_i \rightarrow (\neg r_i \wedge \neg g_i)) \wedge (g_i \rightarrow (\neg r_i \wedge \neg b_i)) \right)$$

- “Países adyacentes deben tener colores distintos”

$$\varphi_D = \bigwedge_{i=1}^n \bigwedge_{j=1}^n \left( p_{ij} \rightarrow ((r_i \rightarrow \neg r_j) \wedge (b_i \rightarrow \neg b_j) \wedge (g_i \rightarrow \neg g_j)) \right)$$

## Ejercicio (mapa 3-coloreable)

- Inicializamos las variables conocidas por la instancia  $M$  del problema

$$\varphi_M = \bigwedge_{(i,j) \in A} p_{ij} \quad \wedge \quad \bigwedge_{(i,j) \notin A} \neg p_{ij}$$

Entonces, nuestra fórmula será la conjunción de las fórmulas anteriores:

$$\varphi = \varphi_C \wedge \varphi_D \wedge \varphi_M$$

Ahora demostraremos que  $M$  es 3-coloreable si y sólo si  $\varphi$  es satisfacible.

Debemos demostrar dos direcciones

# Satisfacibilidad

## Ejercicio (mapa 3-coloreable)

( $\Rightarrow$ ) **P.D.** Si  $M$  es 3-coloreable, entonces  $\varphi$  es satisfacible.

Supongamos que  $M$  es 3-coloreable. Luego, existe una coloración válida para  $M$ . Construimos una valuación  $\sigma$  según

$$\begin{aligned}\sigma(p_{ij}) &= \begin{cases} 1 & \text{si } i, j \text{ son adyacentes en } M \\ 0 & \text{en otro caso} \end{cases} \\ \sigma(r_i) &= \begin{cases} 1 & \text{si } i \text{ es rojo en la coloración de } M \\ 0 & \text{en otro caso} \end{cases} \\ \sigma(b_i) &= \begin{cases} 1 & \text{si } i \text{ es azul en la coloración de } M \\ 0 & \text{en otro caso} \end{cases} \\ \sigma(g_i) &= \begin{cases} 1 & \text{si } i \text{ es verde en la coloración de } M \\ 0 & \text{en otro caso} \end{cases}\end{aligned}$$

Esta dirección consiste en construir una valuación que satisface  $\varphi$

## Ejercicio (mapa 3-coloreable)

( $\Rightarrow$ ) (continuación) Ahora verificamos que  $\sigma(\varphi) = 1$ :

- $\sigma(\varphi_C)$ : para cada país  $i$ , se debe cumplir que  $\sigma(r_i) = 1$ , o que  $\sigma(g_i) = 1$ , o que  $\sigma(b_i) = 1$ , y solo una de estas, por construcción de  $\sigma$ . Luego, es claro que  $\sigma(\varphi_C) = 1$ .
- $\sigma(\varphi_D)$ : para cada combinación de países  $i, j$ , sabemos que  $\sigma(p_{ij}) = 1$  solo cuando los países son adyacentes. Entonces, como en  $\varphi_D$  tenemos una implicancia, solo nos preocuparemos de los pares de países adyacentes. En el consecuente de la implicancia, sabemos que si por ejemplo  $\sigma(r_i) = 1$ , se debe cumplir que  $\sigma(r_j) = 0$ , dado que construimos  $\sigma$  a partir de una 3-coloración. Para las otras dos implicancias, sabemos que el lado izquierdo va a ser falso (por la fórmula  $\varphi_C$ ), y por lo tanto todo el lado derecho se hace verdadero, y entonces  $\sigma(\varphi_D) = 1$ . El análisis para cuando  $i$  es de otro color es análogo.

# Satisfacibilidad

## Ejercicio (mapa 3-coloreable)

( $\Rightarrow$ ) (continuación)

- $\sigma(\varphi_M)$ : por construcción de  $\sigma$  es claro que  $\sigma(\varphi_M) = 1$ , dado que la construimos precisamente como esta fórmula, asignando 1 a pares de países adyacentes y 0 a los que no.

Finalmente, como  $\varphi$  es la conjunción de las fórmulas anteriores, concluimos que  $\sigma(\varphi) = 1$ , y entonces  $\varphi$  es satisfacible.

La dirección opuesta comienza suponiendo que  $\varphi$  es satisfacible

# Satisfacibilidad

## Ejercicio (mapa 3-coloreable)

( $\Leftarrow$ ) **P.D.** Si  $\varphi$  es satisfacible, entonces  $M$  es 3-coloreable.

Supongamos que  $\varphi$  es satisfacible. Luego, existe una valuación  $\sigma$  tal que  $\sigma(\varphi) = 1$ , y por construcción  $\sigma(\varphi_C) = \sigma(\varphi_D) = \sigma(\varphi_M) = 1$ . Usaremos esta valuación para colorear el mapa.

En primer lugar, como  $\sigma(\varphi_C) = 1$ , sabemos que para cada  $i$ ,  $\sigma(r_i \vee g_i \vee b_i) = 1$ , y por lo tanto cada país tiene asignado al menos un color. Sin pérdida de generalidad, supongamos que  $\sigma(r_k) = 1$ , es decir, pintamos el país  $k$  rojo. Como también se cumple que  $\sigma(r_k \rightarrow (\neg g_k \wedge \neg b_k)) = 1$ , necesariamente  $\sigma(g_k) = 0$  y  $\sigma(b_k) = 0$ , y por lo tanto cada país tiene un único color.

Esta dirección busca deducir la **existencia**  
de una coloración a partir de la valuación

## Ejercicio (mapa 3-coloreable)

( $\Leftarrow$ ) (continuación)

En segundo lugar, como  $\sigma(\varphi_M) = 1$ , sabemos que si  $i, j$  son adyacentes en  $M$ ,  $\sigma(p_{ij}) = 1$ , y si no lo son,  $\sigma(p_{ij}) = 0$ . Ahora, en  $\sigma(\varphi_D) = 1$ , solo nos interesa el primer caso (dado que en el segundo no podemos concluir nada de la implicancia). Tomemos entonces  $i, j$  adyacentes, y sin pérdida de generalidad supongamos que  $\sigma(r_i) = 1$ . Como  $\sigma(r_i \rightarrow \neg r_j) = 1$  para todo  $j$  adyacente a  $i$ , necesariamente  $\sigma(r_j) = 0$ , y entonces los países adyacentes no pueden estar pintados del mismo color.

Concluimos que usando los colores asignados por  $\varphi$  a través de  $r_i, g_i, b_i$ , podemos 3-colorear  $M$ .



# Otros conceptos asociados a satisfacibilidad

## Definición

Una fórmula  $\varphi$  es una **contradicción** si no es satisfacible; es decir, para toda valuación  $\sigma$  se tiene que  $\sigma(\varphi) = 0$ .

## Ejemplo

$$p \wedge \neg p$$

## Definición

Una fórmula  $\varphi$  es una **tautología** si para toda valuación  $\sigma$  se tiene que  $\sigma(\varphi) = 1$ .

## Ejemplo

$$p \vee \neg p$$

$$p \leftrightarrow p$$

# Otros conceptos asociados a satisfacibilidad

## Definición

Una fórmula  $\varphi$  es una **tautología** si para toda valuación  $\sigma$  se tiene que  $\sigma(\varphi) = 1$ .

Podemos definir la equivalencia lógica de una manera alternativa:

## Teorema

Dos fórmulas  $\varphi, \psi \in L(P)$  son lógicamente equivalentes si  $\varphi \leftrightarrow \psi$  es una tautología.

Demuestre el teorema (★)

# Outline

Obertura

Satisfacibilidad

**Formas normales**

Consecuencia lógica

Epílogo

# Formas normales

Concluiremos nuestra revisión sintáctica con un resumen sintáctico jj

## Definición

Un literal es una variable proposicional o la negación de una variable proposicional.

## Ejemplo

Para el conjunto de variables  $P = \{p, q, r\}$ , las fórmulas  $p$  y  $\neg r$  son literales.

Los literales son los átomos de la construcción que mostraremos

# 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,  $B_i = (l_{i1} \wedge \dots \wedge l_{ik_i})$

## Ejemplo

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

¿Hemos visto fórmulas en DNF antes?

# 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.

Ya demostramos esto y ¡mostramos cómo construir tal fórmula!

# 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,  $C_i = (l_{i1} \vee \dots \vee l_{ik_i})$

## Observaciones

- Una disyunción de literales se llama una **cláusula**.
  - Los  $C_i$  anteriores son cláusulas.
- Una fórmula en CNF es una conjunción de cláusulas.

## Ejemplo

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

¿Podríamos obtener una fórmula en CNF para una tabla de verdad?

# 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.

Demuestre el teorema (★○)



# Outline

Obertura

Satisfacibilidad

Formas normales

**Consecuencia lógica**

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**

## Demostración

Próxima clase!

# Outline

Obertura

Satisfacibilidad

Formas normales

Consecuencia lógica

**Epílogo**

# Objetivos de la clase

- Comprender el concepto de satisfacibilidad de fórmulas y conjuntos
- Aplicar lógica para modelar problemas
- Conocer las formas normales
- Comprender concepto de consecuencia lógica
- Demostrar consecuencias lógicas sencillas