Tema 7: Lógica Proposicional

María Inés Fernández Camacho

MATEMÁTICA DISCRETA Y LÓGICA MATEMÁTICA
(GRUPOS E y F)
UCM Curso 18/19

Lógica:

- Fundamentación del concepto de certeza y todo lo que ésta involucra.
- Estudia las reglas que debe respetar todo razonamiento válido:
  - Discernir lo que con seguridad es cierto a partir de premisas que damos por buenas.
  - Distinguir entre razonamientos que son lógicamente válidos y los que no lo son.

Argumentación lógica

- Premisas y conclusión
- (A1)
  $$\frac{\text{Mario compró un coche} \quad \text{Luisa saludó a Mario}}{\therefore \text{Luisa saludó a uno que compró un coche}}$$
- Una argumentación es lógicamente válida si la verdad de las premisas conlleva necesariamente la verdad de la conclusión (i.e. si no podemos concebir circunstancias que hagan verdaderas las premisas y falsa la conclusión)
- La validez lógica de una argumentación no depende de la verdad o falsedad de sus premisas y conclusión, sino **de la relación entre la hipotética verdad de las premisas y la verdad de la conclusión.**

ARGUMENTACIÓN LÓGICA (2)

Mario compró un coche Luisa saludó a Mario	Pepe lleva sombrero Juan contrató a Pepe
∴ Luisa saludó a uno que compró un coche	∴ Juan contrató a uno que lleva sombrero
(A1)	(A2)

- (A1) y (A2) son razonamientos válidos.

(A3)

Alguien lleva bufanda
Pedro pagó a alguien

∴ Pedro pagó a uno que lleva bufanda

- (A3) no es un razonamiento lógicamente válido.

Argumentaciones con igual forma "superficial" en lenguaje natural pueden diferir en su validez lógica.

INTRODUCCIÓN

ORACIONES DECLARATIVAS

Los razonamientos estudiados por la lógica se refieren a oraciones declarativas de las que tiene sentido preguntarse si son verdaderas o falsas.

DEF:

Una **proposición** *es una afirmación (declaración) que o bien es cierta o bien es falsa (pero no ambas).*

Ejemplos:

- Proposiciones:
  - París es la capital de Francia
  - 8 es un número primo
  - 9 no es un número primo
  - $(2 < 3)$ y 5 es primo
- Oraciones que no son proposiciones:
  - ¡Cállate!
  - ¿Qué hay en la bolsa?
  - 4 - 2
  - x es par

Lenguaje natural versus lenguaje formal

- El lenguaje natural es ambiguo e impreciso.
- El análisis lógico de una lengua natural a distintos niveles de detalle da lugar a distintos lenguajes formales (lógica proposicional, lógica de predicados o de primer orden, $\cdots$)

Def:

Lenguaje formal: *Conjunto de palabras finitas construídas sobre un alfabeto aplicando ciertas reglas de formación (gramática que fija la sintaxis del lenguaje)*

- **Sintaxis:** Reglas de formación. Gramática. **"El cómo".**
- **Semántica:** Interpretaciones que dan un significado a las construcciones sintácticamente correctas. **"El qué".**

Surge de un análisis muy simple del lenguaje natural basado en la distinción entre dos tipos de proposiciones:

- **proposiciones atómicas:** No se pueden descomponer en otras más simples. Se las denota con letras minúsculas, p,q, r, s, ... con o sin subíndices (símbolos proposicionales).

Mario compró un coche	p
Luisa saludó a Mario	q
Luisa conoce a Mario	r

- **proposiciones compuestas:** Construidas combinando otras más simples mediante conectivas lógicas: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$.
  - Conectiva unaria: $\neg$ (negación).
  - Conectivas binarias: $\wedge$(conjunción), $\vee$(disyunción), $\rightarrow$ (condicional o implicación) y $\leftrightarrow$ (bicondicional o biimplicación).

Mario compró un coche y Luisa saludó a Mario	$(p \wedge q)$
Luisa no saludó a Mario	$\neg q$
Mario compró un coche y Luisa no saludó a Mario aunque le conoce	$((p \wedge \neg q) \wedge r)$

- **Fórmulas:** cualquier enunciado formalizado ya sea simple o compuesto
  - Se las denota habitualmente con letras griegas $\varphi, \psi, \chi, \ldots$ con o sin subíndices.

- **Constantes lógicas:**
  - Sirven para representar respectivamente un enunciado que siempre es cierto o que siempre es falso.
    - $\perp$ (falsedad)
    - $\top$ (certeza)
  - Se pueden considerar conectivas 0-ádicas

Principales conectivas lógicas

Nombre	Notación	Significado
Negación	$\neg\varphi$	"no φ"
Conjunción	$(\varphi_1 \wedge \varphi_2)$	" φ_1 y φ_2 "
Disyunción	$(\varphi_1 \vee \varphi_2)$	" φ_1 o φ_2"
Implicación	$(\varphi_1 \to \varphi_2)$	"si φ_1 entonces φ_2 " " φ_2 si φ_1" " φ_2 siempre que φ_1" " φ_1 sólo si φ_2 " " φ_1 es condición suficiente para φ_2 " " φ_2 es condición necesaria para φ_1 "
Bicondicional	$(\varphi_1 \leftrightarrow \varphi_2)$	" φ_1 si y sólo si φ_2" "φ_1 es condición necesaria y suficiente para φ_2"

- **Símbolos primitivos:**
  - **Símbolos lógicos**: $\{\bot, \top, \neg, \wedge, \vee \rightarrow, \leftrightarrow\}$
    - Constantes lógicas (conectivas 0-ádicas): $\bot$ (falsedad), $\top$ (certeza)
    - Conectiva unaria: $\neg$ (negación).
    - Conectivas binarias:$\wedge$(conjunción), $\vee$(disyunción),$\rightarrow$ (condicional o implicación) y $\leftrightarrow$ (bicondicional o biimplicación).

  Notación: $\square$ denotará cualquier conectiva binaria.
  - **Símbolos auxiliares**: **(** y **)**
  - **Signatura, Σ** : Conjunto de símbolos de proposición.
    Sus elementos se denotan con letras minúsculas, p,q,r,... con o sin subíndices.

  Σ no incluye los símbolos lógicos ni los auxiliares.

- **Alfabeto de símbolos primitivos:**

$$A_{\Sigma} = \mathbf{\Sigma} \cup \{\bot, \top, \neg, \wedge, \vee \rightarrow, \leftrightarrow\} \cup \{(,)\}$$

- **Reglas de formación:**

Llamamos **fórmulas** a aquellas palabras sobre A_{Σ} que se construyen aplicando un número finito de veces las siguientes reglas:

(At): $\bot, \top$ y $p \in \Sigma$ son fórmulas (fórmulas atómicas)
($\neg$): si φ es una fórmula, entonces $\neg\varphi$ es una fórmula (negaciones)
($\wedge$): si φ_1 y φ_2 son fórmulas, entonces $(\varphi_1 \wedge \varphi_2)$ es una fórmula (conjunciones)
($\vee$): si φ_1 y φ_2 son fórmulas, entonces $(\varphi_1 \vee \varphi_2)$ es una fórmula (disyunciones)
($\rightarrow$): si φ_1 y φ_2 son fórmulas, entonces $(\varphi_1 \rightarrow \varphi_2)$ es una fórmula (condicionales)
A $\boldsymbol{\varphi_1}$ se le llama **antecedente** y a $\boldsymbol{\varphi_2}$ **consecuente**.
($\leftrightarrow$): si φ_1 y φ_2 son fórmulas, entonces $(\varphi_1 \leftrightarrow \varphi_2)$ es una fórmula (bicondicionales)

- **El lenguaje de la lógica proposicional con signatura $\boldsymbol{\Sigma}$, $L_{\boldsymbol{\Sigma}}$,** es el conjunto de todas las fórmulas con signatura $\boldsymbol{\Sigma}$

  A las fórmulas se las denota habitualmente con letras griegas φ, ψ, χ,... con o sin subíndices.

- $L_{\boldsymbol{\Sigma}} \subset A^*_{\boldsymbol{\Sigma}}$.

  Si $p, q, r \in \Sigma$,

$$(\neg p \rightarrow (q \wedge r)) \in L_\Sigma$$

$$(\neg)p \vee (q \rightarrow r)) \notin L_\Sigma$$

Principio de inducción estructural para fórmulas proposicionales(PIE)

Dada una propiedad P que tiene sentido para palabras $u \in A_{\Sigma}^{*}$, podemos concluir que toda fórmula $\varphi \in L_{\Sigma}$ tiene la propiedad P siempre que demostremos:

- **Casos base:**

  (At): Toda fórmula atómica tiene la propiedad P

- **Pasos inductivos:**

  ($\neg$): si φ tiene la propiedad P (hipótesis de inducción) , entonces $\neg\varphi$ también tiene la propiedad P.

  ($\Box$): si φ_1 y φ_2 tienen la propiedad P (hipótesis de inducción), entonces $(\varphi_1 \Box \varphi_2)$ también tiene la propiedad P.

Ej.: Demostrar que cualquier fórmula proposicional contiene igual número de veces los símbolos auxiliares "(" y ")".

- **Casos base:**

  (At): Toda fórmula atómica contiene exactamente 0 veces tanto el símbolo "(" como el símbolo ")".

- **Pasos inductivos:**

  ($\neg$): $\varphi = \neg\varphi_1$

  Si **HI:** φ_1 contiene n veces "(" y n veces ")",
  entonces φ también contiene n veces "(" y n veces ")" .

  ($\square$): $\varphi = (\varphi_1 \square \varphi_2)$

  Si **HI:** φ_i contiene n_i veces "(" y n_i veces ")', $i \in \{1, 2\}$,
  entonces φ contiene $n_1 + n_2 + 1$ veces "(" y $n_1 + n_2 + 1$ veces ")".

La construcción de cualquier fórmula determina unívocamente su estructura sintáctica.

Principio de unicidad estructural para fórmulas proposicionales (PUE)

Toda fórmula $\varphi \in L_\Sigma$ cae dentro de uno y sólo uno de los casos siguientes:

(At): φ es atómica

(¬): $\varphi = \neg\varphi_1$ para cierta fórmula φ_1 unívocamente determinada.

(□): $\varphi = (\varphi_1 \square \varphi_2)$ para cierta conectiva $\square$ y ciertas fórmulas φ_1 y φ_2 unívocamente determinadas.

El lenguaje de la lógica proposicional

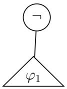
Árboles estructurales de las fórmulas proposicionales

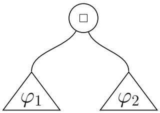
A cada $\varphi \in L_\Sigma$ se le puede asociar un árbol unívocamente determinado por φ que representa su estructura de construcción y que se denomina **árbol estructural** de φ.

Dem:

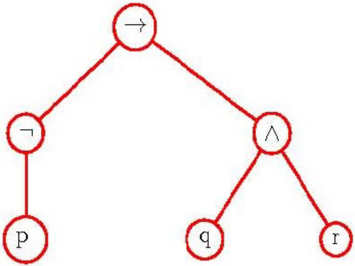
Por PIE y PUE:

(At)	$(\neg)$	$(\square)$
$\varphi \in \{\bot, \top\} \cup \Sigma$	$\varphi = \neg\varphi_1$	$\varphi = (\varphi_1 \square \varphi_2)$





Ej: Árbol estructural de $(\neg p \to (q \wedge r))$:



Principio de recursión estructural para fórmulas proposicionales (PRE)

Dado cualquier conjunto A , para definir una función $f : L_\Sigma \to A$ es válido utilizar el siguiente esquema recursivo:

- **Casos base:**

  (At): Para φ atómica:
  $f(\varphi) = \cdots$ valor explícito $\cdots$

- **Casos recursivos:**

  (¬): $f(\neg\varphi) =$ valor dependiendo de $f(\varphi)$.

  (□): $f((\varphi_1 \square \varphi_2)) =$ valor dependiendo de $f(\varphi_1), f(\varphi_2)$ y $\square$.

Def:

El **vocabulario** *de una fórmula* $\varphi \in L_\Sigma$ *es el conjunto finito formado por todos los símbolos de proposición* $p \in \Sigma$ *que aparecen en* φ.

Definición recursiva de $voc : L_\Sigma \to \wp(\Sigma)$ que asocia a cada fórmula proposicional su vocabulario:

- **Casos base:**

  (At):

  $voc(\top) = \emptyset$
  $voc(\bot) = \emptyset$
  $\forall p \in \Sigma \quad voc(p) = \{p\}$

- **Casos recursivos:**

  ($\neg$ **)**: $voc(\neg\varphi) = voc(\varphi)$.

  ($\square$ **)**: $voc((\varphi_1 \square \varphi_2)) = voc(\varphi_1) \cup voc(\varphi_2)$.

Def:

Dadas $\varphi, \psi \in L_\Sigma$, ψ es una **subfórmula de** φ *si una parte de φ formada por símbolos consecutivos es idéntica a ψ. En este caso el árbol estructural de ψ es un subárbol del árbol estructural de φ.*

Definición recursiva de $CSub : L_\Sigma \to \wp(L_\Sigma)$ que asocia a cada fórmula su conjunto de subfórmulas:

- **Casos base:**

  (At):

  $$CSub(\top) = \{\top\}$$
  $$CSub(\bot) = \{\bot\}$$
  $$\forall p \in \Sigma \quad CSub(p) = \{p\}$$

- **Casos recursivos:**

  $(\neg)$: $CSub(\neg\varphi) = \{\neg\varphi\} \cup CSub(\varphi)$.

  $(\square)$: $CSub((\varphi_1 \square \varphi_2)) = \{(\varphi_1 \square \varphi_2)\} \cup CSub(\varphi_1) \cup CSub(\varphi_2)$.

Ej.: Dada $\varphi = (\neg p \rightarrow (q \wedge \neg r))$:

$$voc(\varphi) = \{p, q, r\}$$

$$CSub(\varphi) = \{\neg p, p, q, r, \neg r, (q \wedge \neg r), \varphi\}$$

Escritura abreviada de fórmulas proposicionales

• Una fórmula está **correctamente** escrita en **forma abreviada** si cumple los siguientes convenios:

- Omite los paréntesis externos.
- Las conectivas tienen el siguiente orden de prioridad:

$$\neg > \wedge > \vee > \rightarrow > \leftrightarrow$$

- Las conectivas $\wedge, \vee, \rightarrow$ asocian por la derecha.

Ej.:

- $\neg p \wedge q \vee \neg r \rightarrow s$ abrevia $(((\neg p \wedge q) \vee \neg r) \rightarrow s)$
- $p \rightarrow q \rightarrow r$ abrevia $(p \rightarrow (q \rightarrow r))$

Interpretaciones que dan un significado a las construcciones sintácticamente correctas.

- **Valores veritativos:** 0 para falso y 1 para cierto.
- **Valoración de la signatura Σ:** Una aplicación $v : \Sigma \to \{0, 1\}$.

  Obs.: Si $|\Sigma| = n$, entonces hay 2^n valoraciones posibles para Σ
- **Interpretaciones de las conectivas** o **tablas de verdad** de las conectivas:
  - Tabla de verdad de la conectiva unaria $\neg$: Es la aplicación $v_\neg : \{0, 1\} \to \{0, 1\}$ dada por la tabla:

x	$v_\neg(x)$
0	1
1	0

  - Tabla de verdad de las conectivas binarias $\square$: Son las aplicaciones $v_\square : \{0, 1\} \times \{0, 1\} \to \{0, 1\}$ dadas por la tabla:

x	y	$v_\wedge(x, y)$	$v_\vee(x, y)$	$v_\to(x, y)$	$v_\leftrightarrow(x, y)$
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	0	0
1	1	1	1	1	1

Def:

Dadas $\varphi \in L_\Sigma$ *y* $v : \Sigma \to \{0,1\}$, **el valor veritativo de** φ **en** v *se define recursivamente por la función* $[\![\cdot]\!]^v : L_\Sigma \to \{0,1\}$ *así:*

- **Casos base:**

$$
\begin{array}{ll}
(\top) & [\![\top]\!]^v = 1 \\
(\bot) & [\![\bot]\!]^v = 0 \\
(\Sigma) & [\![p]\!]^v = v(p) \quad \forall p \in \Sigma
\end{array}
$$

- **Casos recursivos:**

$$(\ \neg\ ): [\![\neg\varphi]\!]^v = v_\neg\,([\![\varphi]\!]^v)$$

$$(\ \square\ ): [\![(\varphi_1 \,\square\, \varphi_2)]\!]^v = v_\square\,([\![\varphi_1]\!]^v, ([\![\varphi_2]\!]^v)$$

Tabla de verdad de una fórmula proposicional: da los valores veritativos de la fórmula para todas las valoraciones posibles de los elementos de su vocabulario.

Ej.:

Dadas $\varphi = (p \to (q \leftrightarrow \neg r))$ y $v : \Sigma \to \{0,1\}$ tal que
$v(p) = v(r) = 0, v(q) = 1$:

$$[\![\varphi]\!]^v = v_\to ([\![p]\!]^v, [\![(q \leftrightarrow \neg r)]\!]^v) = v_\to (0, v_\leftrightarrow ([\![q]\!]^v, [\![\neg r]\!]^v)) = 1$$

p	q	r	¬ r	(q ↔ ¬ r)	φ
0	0	0	1	0	1
0	0	1	0	1	1
0	1	0	1	1	1
0	1	1	0	0	1
1	0	0	1	0	0
1	0	1	0	1	1
1	1	0	1	1	1
1	1	1	0	0	0

Tabla : Tabla de verdad de $(p \to (q \leftrightarrow \neg r))$

Def:

Dadas $\varphi \in L_\Sigma$ *y* $v : \Sigma \to \{0,1\}$ *:*

- *Si* $[\![\varphi]\!]^v = 1$ *, decimos que* v **satisface** φ*, o que* v **es modelo de** φ *y escribimos* $v \models \varphi$*.*
  $Mod(\varphi)$ *denota el conjunto de todos los modelos de* φ
- *Si* $[\![\varphi]\!]^v = 0$ *, decimos que* v **no satisface** φ*, o que* v **no es modelo de** φ *y escribimos* $v \not\models \varphi$*.*
- φ *es* **satisfactible** *si existe una valoración* v *que satisface* φ*.*
- φ *es* **insatisfactible** *si no existe ninguna valoración* v *que satisfaga* φ*.*

Ej.:

Dada $\varphi = (p \to (q \leftrightarrow \neg r))$:

p	q	r	¬ r	(q ↔ ¬ r)	φ
0	0	0	1	0	1
0	0	1	0	1	1
0	1	0	1	1	1
0	1	1	0	0	1
1	0	0	1	0	0
1	0	1	0	1	1
1	1	0	1	1	1
1	1	1	0	0	0

TABLA : Tabla de verdad de $(p \to (q \leftrightarrow \neg r))$

φ es satisfactible, pero no toda valoración la satisface.

Los modelos de φ son las valoraciones que aparecen sombreadas en azul en la tabla anterior.

Def:

Dados $\Phi \subseteq L_\Sigma$ *y* $v : \Sigma \to \{0, 1\}$ *:*

- *Si* $v \models \varphi$, *para* **cada** $\varphi \in \Phi$ *decimos que* v **satisface** Φ, *o que* v **es modelo de** Φ *y escribimos* $v \models \Phi$.
  $Mod(\Phi)$ *denota el conjunto de todos los modelos de* Φ
- *Si* $v \not\models \varphi$,*para* **alguna** $\varphi \in \Phi$ *decimos que* v **no satisface** Φ, *o que* v **no es modelo de** Φ *y escribimos* $v \not\models \Phi$.
- Φ *es* **satisfactible** *si existe una valoración* v *que satisface* Φ.
  (Obs.: $\Phi = \emptyset$ *es satisfactible)*
- Φ *es* **insatisfactible** *si no existe ninguna valoración* v *que satisfaga* Φ.

Prop.: Dado $\Phi = \{\varphi_1, \varphi_2, \cdots \varphi_n\} \subseteq L_\Sigma$,
Φ es satisfactible si y sólo si $\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n$ es satisfactible.

Dem: ...

Ej.:

$\Phi = \{p \to q, \neg q\}$ es satisfactible pero $\Phi_1 = \{p \to q, \neg q, p\}$ es insatisfactible.

p	q	¬ q	(p → q)
0	0	1	1
0	1	0	1
1	0	1	0
1	1	0	1

- $v(p) = v(q) = 0, \quad v \models \Phi.$
- $v \not\models \Phi_1,$ para toda valoración v. (Ninguna valoración satisface las tres fórmulas a la vez)

Def:

- *Una fórmula* φ *es una* **tautología** *si* $v \models \varphi$ *(es cierta) para toda valoración* $v : \Sigma \to \{0,1\}$.
- *Una fórmula* φ *es una* **contradicción** *si* $v \not\models \varphi$ *(es falsa) para toda valoración* $v : \Sigma \to \{0,1\}$.
- *Una fórmula* φ *es una* **contingencia** *si existen al menos dos valoraciones distintas* $v_1, v_2 : \Sigma \to \{0,1\}$ *tales que* $v_1 \models \varphi$ *y* $v_2 \not\models \varphi$ *(no es ni tautología ni contradicción).*

p	$\neg p$	$(p \vee \neg p)$	$(p \wedge \neg p)$	$(p \to \neg p)$
1	0	1	0	0
0	1	1	0	1
$\uparrow$	$\uparrow$	$\uparrow$	$\uparrow$	$\uparrow$
Contingencia	Contingencia	Tautología	Contradicción	Contingencia

Ej.:
$((p \to q) \wedge (p \wedge \neg q))$ es contradicción.

Dem.:
Supongamos que existe $v : \Sigma \to \{0, 1\}$ tal que $[\![\varphi]\!]^v = 1$

$[\![\varphi]\!]^v = v_\wedge\ ([\![(p \to q)]\!]^v, [\![(p \wedge \neg q)]\!]^v) = 1$

Luego $\quad [\![(p \to q)]\!]^v = 1, [\![(p \wedge \neg q)]\!]^v = 1$

Pero $\quad [\![(p \to q)]\!]^v = v_\to\ ([\![p]\!]^v, [\![q]\!]^v),\ \ [\![(p \wedge \neg q)]\!]^v = v_\wedge\ ([\![p]\!]^v, [\![\neg q]\!]^v)$

Luego debería cumplirse $v_\to\ ([\![p]\!]^v, [\![q]\!]^v) = 1, v(p) = 1, v_\neg(v(q)) = 1$

Y llegamos al absurdo $v_\to\ ([\![p]\!]^v, [\![q]\!]^v) = 1, v(p) = 1, v(q) = 0$

Def:

Dadas $\chi, \varphi_1, \cdots \varphi_n \in L_\Sigma$ y n símbolos de proposición diferentes $p_1, p_2 \cdots p_n \in \Sigma$, $\chi' = \chi[p_1/\varphi_1, p_2/\varphi_2, ...p_n/\varphi_n]$ designa a la fórmula resultante de sustituir simultáneamente en χ todas las apariciones de p_i por φ_i y decimos que χ' es un caso particular de χ.

Teorema: Si χ es tautología entonces cualquier caso particular χ' de χ es tautología.

Dem: El valor veritativo de $\chi' = \chi[p_1/\varphi_1, p_2/\varphi_2, ...p_n/\varphi_n]$ bajo cualquier valoración v dada, será el mismo que el de χ bajo la valoración v' que asigna a cada p_i el valor $[\![\varphi_i]\!]^v$ y tal que $v'(q)=v(q)\ \ \forall q \in \Sigma \setminus \{p_1, p_2, ...p_n\}$.

Prop.: Una fórmula proposicional φ es contradicción si y sólo si $\neg\varphi$ es una tautología, y viceversa.

Dem: ...

Corolario.: Cualquier caso particular de una contradicción es una contradicción.

Dem: ...

Prop.: Dado $\Phi = \{\varphi_1, \varphi_2, \cdots \varphi_n\} \subseteq L_\Sigma$,
Φ es insatisfactible si y sólo si $\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n$ es una contradicción.

Dem: ...

Def:

Dados $\Phi \in L_\Sigma$ (conjunto de premisas o hipótesis) y $\psi \in L_\Sigma$ (conclusión o tesis), decimos que ψ es **consecuencia lógica** *de Φ, lo que escribiremos $\Phi \models \psi$, si todo modelo de Φ lo es de ψ (i.e. si $v \models \Phi$ entonces $v \models \psi$, $\forall v : \Sigma \to \{0, 1\}$)*

- A $\Phi \models \psi$ se le llama regla de inferencia.
- $\Phi \not\models \psi$ denota que ψ no es consecuencia lógica de Φ (i.e. hay al menos un modelo de Φ que no es modelo de ψ)
- $\models \psi$ abrevia $\emptyset \models \psi$
- $\varphi_1, \varphi_2 \cdots \varphi_n \models \psi$ abrevia $\{\varphi_1, \varphi_2 \cdots \varphi_n\} \models \psi$

$$\begin{array}{ll} \text{Premisas} & \left\{ \begin{array}{l} \varphi_1 \\ \varphi_2 \\ \vdots \\ \varphi_n \end{array} \right. \\ \hline \text{Conclusión} & \therefore \psi \end{array}$$

Podemos establecer la validez lógica de un razonamiento formalizándolo en la lógica proposicional y comprobando que la conclusión es consecuencia lógica de las premisas.

Ej.: Si encuentras esto difícil entonces no eres inteligente o no lo has trabajado. Lo has trabajado y eres inteligente luego no lo encontrarás difícil.

$$\begin{array}{ll} \text{Premisas} & \left\{ \begin{array}{l} (p \rightarrow (\neg q \vee \neg r)) \\ (r \wedge q) \end{array} \right. \\ \hline \text{Conclusión} & \therefore \neg p \end{array}$$

p :Encuentras esto difícil

q :Eres inteligente

r :Lo has trabajado

Ejemplo de razonamiento lógicamente correcto:

Si encuentras esto difícil entonces no eres inteligente o no lo has trabajado.
Lo has trabajado y eres inteligente luego no lo encontrarás difícil.

$$\text{Premisas}\begin{cases}(p \to (\neg q \vee \neg r)) \\ (r \wedge q)\end{cases}$$

$$\text{Conclusión} \quad \therefore \ \neg p$$

p :Encuentras esto difícil
q :Eres inteligente
r :Lo has trabajado

p	q	r	¬ q	¬ r	(¬ q ∨ ¬ r)	((p → (¬ q ∨ ¬ r))	(r ∧ q)	¬ p
0	0	0	1	1	1	1	0	1
0	0	1	1	0	1	1	0	1
0	1	0	0	1	1	1	0	1
0	1	1	0	0	0	1	1	1
1	0	0	1	1	1	1	0	0
1	0	1	1	0	1	1	0	0
1	1	0	0	1	1	1	0	0
1	1	1	0	0	0	0	1	0

Regla de inferencia: $(p \to (\neg q \vee \neg r)), (r \wedge q) \models \neg p$

Prop.: $\models \psi$ si y sólo si ψ es una tautología.

Dem: ...

Prop.: Si $\Phi \subseteq L_\Sigma$ es insatisfactible entonces $\Phi \models \psi$ para cualquier fórmula $\psi \in L_\Sigma$.
(De falso puede concluirse cualquier cosa)

Dem:
$\Phi \subseteq L_\Sigma$ es insatisfactible si y sólo si no existe $v : \Sigma \to \{0, 1\}$ tal que $v \models \Phi$. Luego $\Phi \models \psi$ pues no hay ningún modelo de Φ para el que ψ sea falso .

Teorema de la deducción: Sean $\Phi \subseteq L_\Sigma$ y $\varphi, \psi, \varphi_1, \cdots \varphi_n \in L_\Sigma$. Entonces

1) $\Phi \models (\varphi \to \psi)$ si y sólo si $\Phi \cup \{\varphi\} \models \psi$
2) $\Phi \models \varphi_1 \wedge \cdots \wedge \varphi_n \to \psi$ si y sólo si $\Phi \cup \{\varphi_1, \cdots, \varphi_n\} \models \psi$
3) $\varphi_1, \cdots, \varphi_n \models \psi \Leftrightarrow \ \models \varphi_1 \wedge \cdots \wedge \varphi_n \to \psi$
$\Leftrightarrow \varphi_1 \wedge \cdots \wedge \varphi_n \to \psi$ es una tautología.

Dem: ...

Relación entre consecuencia lógica, tautologías y contradicciones. (3)

Ej.: Si encuentras esto difícil entonces no eres inteligente o no lo has trabajado. Lo has trabajado y eres inteligente luego no lo encontrarás difícil.

$$\text{Premisas} \begin{cases} (p \to (\neg q \vee \neg r)) \\ (r \wedge q) \end{cases}$$

$$\text{Conclusión} \quad \therefore\ \neg p$$

p :Encuentras esto difícil
q :Eres inteligente
r :Lo has trabajado

Implicación lógica relacionada: φ: $(((p \to (\neg q \vee \neg r)) \wedge (r \wedge q)) \to \neg p)$

p	q	r	¬ q	¬ r	¬ q ∨¬ r	p→ (¬ q ∨¬ r)	r ∧ q	¬ p	(p → (¬ q ∨¬ r)) ∧ (r ∧ q)	φ
0	0	0	1	1	1	1	0	1	0	1
0	0	1	1	0	1	1	0	1	0	1
0	1	0	0	1	1	1	0	1	0	1
0	1	1	0	0	0	1	1	1	1	1
1	0	0	1	1	1	1	0	0	0	1
1	0	1	1	0	1	1	0	0	0	1
1	1	0	0	1	1	1	0	0	0	1
1	1	1	0	0	0	0	1	0	0	1

Teorema de la reducción al absurdo: Sean $\Phi \subseteq L_\Sigma$ y $\varphi, \psi, \varphi_1, \cdots \varphi_n \in L_\Sigma$. Entonces

1) $\Phi \models \psi$ si y sólo si $\Phi \cup \{\neg\psi\}$ es insatisfactible

2) $\varphi_1, \cdots, \varphi_n \models \psi \Leftrightarrow \varphi_1 \wedge \cdots \wedge \varphi_n \wedge \neg\psi$ es una contradicción.

Dem: ...

Corolario: Si existe $v : \Sigma \to \{0, 1\}$ tal que $v \models \varphi_1 \wedge \cdots \wedge \varphi_n \wedge \neg\psi$ entonces ψ no es consecuencia lógica de $\{\varphi_1, \cdots, \varphi_n\}$ y a dicha v se la llama valoración contraejemplo

Ej.: Refuta $(p \to q) \models q$ mediante una valoración contraejemplo.

Valoración contraejemplo: $v : \Sigma \to \{0,1\}, \quad v(p) = v(q) = 0$

$$\begin{aligned} [\![(p \to q) \wedge \neg q]\!]^v &= v_\wedge(v_\to(v(p), v(q)), v_\neg(v(q))) \\ &= v_\wedge(v_\to(0,0), v_\neg(0)) \\ &= v_\wedge(1,1) = 1 \end{aligned}$$

Def:

Dos fórmulas φ y ψson lógicamente equivalentes, y se escribe $\varphi \sim \psi$, si $Mod(\varphi) = Mod(\psi)$ (i.e. $\forall v : \Sigma \to \{0,1\}$ $[\![\varphi]\!]^v = [\![\psi]\!]^v$).

p	q	¬ p	(p → q)	(¬p∨q)	((p → q) ↔ (¬p∨q))
0	0	1	1	1	1
0	1	1	1	1	1
1	0	0	0	0	1
1	1	0	1	1	1

Tabla : $(p \to q) \sim (\neg p \vee q)$

- $\varphi \sim \psi$ si y sólo si $(\varphi \leftrightarrow \psi)$ es una tautología.

$\varphi \not\approx \psi$ denota que φ y ψ no son lógicamente equivalentes.

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

TABLA : $((p \to q) \to r) \not\approx (p \to (q \to r))$

Prop.:

1) $\sim \subseteq L_\Sigma \times L_\Sigma$ es una relación de equivalencia.

2) Dadas $\varphi, \psi \in L_\Sigma$ se tiene
$$\begin{aligned} \varphi \sim \psi &\Leftrightarrow \models \varphi \leftrightarrow \psi \\ &\Leftrightarrow \models \varphi \to \psi \text{ y } \models \psi \to \varphi \\ &\Leftrightarrow \varphi \models \psi \text{ y } \psi \models \varphi \end{aligned}$$

3) Dada $\varphi \in L_\Sigma$, φ es tautología si y sólo si $\varphi \sim \top$

4) Dada $\varphi \in L_\Sigma$, φ es contradicción si y sólo si $\varphi \sim \bot$

5) (Propiedad de reemplazamiento): Dadas $\varphi, \psi \in L_\Sigma$, si $\chi(\varphi)$ es una fórmula que contiene a φ y $\varphi \sim \psi$, entonces $\chi(\varphi) \sim \chi(\psi)$ siendo $\chi(\psi)$ el resultado de reemplazar <u>una o varias</u> apariciones de φ en χ por ψ

6) Dadas $\chi_1, \chi_2 \in L_\Sigma$, si $\chi_1 \sim \chi_2$ entonces $\chi_1[\overline{p}/\overline{\varphi}] \sim \chi_2[\overline{p}/\overline{\varphi}] \quad \forall\, \overline{p} \in \Sigma^n, \forall\, \overline{\varphi} \in L_\Sigma^n$

Corolario: Se puede demostrar la equivalencia lógica de dos fórmulas "encadenando" equivalencias lógicas de subfórmulas.

Leyes algebraicas de Boole. Dadas $\varphi, \psi, \chi \in L_\Sigma$

$(\varphi \vee \psi) \vee \chi \sim \varphi \vee (\psi \vee \chi)$ $(\varphi \wedge \psi) \wedge \chi \sim \varphi \wedge (\psi \wedge \chi)$	Leyes de asociatividad
$\varphi \vee \psi \sim \psi \vee \varphi$ $\varphi \wedge \psi \sim \psi \wedge \varphi$	Leyes de conmutatividad
$\varphi \vee (\psi \wedge \chi) \sim (\varphi \vee \psi) \wedge (\varphi \vee \chi)$ $\varphi \wedge (\psi \vee \chi) \sim (\varphi \wedge \psi) \vee (\varphi \wedge \chi)$	Leyes de distributividad
$\neg(\varphi \vee \psi) \sim \neg\varphi \wedge \neg\psi$ $\neg(\varphi \wedge \psi) \sim \neg\varphi \vee \neg\psi$	Leyes de De Morgan
$\varphi \vee \varphi \sim \varphi$ $\varphi \wedge \varphi \sim \varphi$	Leyes de idempotencia
$\varphi \vee (\varphi \wedge \psi) \sim \varphi$ $\varphi \wedge (\varphi \vee \psi) \sim \varphi$	Leyes de absorción
$\varphi \vee \bot \sim \varphi$ $\varphi \wedge \top \sim \varphi$	Elemento neutro.Leyes de identidad
$\varphi \vee \top \sim \top$ $\varphi \wedge \bot \sim \bot$	Elemento nulo.Leyes de dominación
$\neg\top \sim \bot$ $\neg\bot \sim \top$ $\neg\neg\varphi \sim \varphi$ (Doble negación) $\varphi \vee \neg\varphi \sim \top$ (Tercio excluído) $\varphi \wedge \neg\varphi \sim \bot$ (Contradicción)	Leyes de negación

Demostración de la ley de doble negación $\neg\neg\varphi \sim \varphi$:

p	$\neg p$	$\neg\neg$ p
0	1	0
1	0	1

Tabla : $\neg\neg p \sim p$

$$\neg\neg p[p/\varphi] \sim p[p/\varphi]$$

$$\neg\neg\varphi \sim \varphi$$

Leyes de relación entre conectivas

$\varphi \to \psi \sim \neg\varphi \vee \psi$	
$\varphi \to \psi \sim \neg\psi \to \neg\varphi$ (Contrarrecíproco o trasposición de $\varphi \to \psi$)	
$\varphi \to \psi \sim \neg(\varphi \wedge \neg\psi)$	
$\varphi \leftrightarrow \psi \sim (\varphi \to \psi) \wedge (\psi \to \varphi)$	
$\varphi \leftrightarrow \psi \sim \neg\varphi \leftrightarrow \neg\psi$	
$\varphi \vee \psi \sim \neg\varphi \to \psi \sim \neg\psi \to \varphi$	
$\varphi \wedge \psi \sim \neg(\varphi \to \neg\psi) \sim \neg(\psi \to \neg\varphi)$	
$\varphi \to \bot \sim \neg\varphi$	
$\varphi \to \top \sim \top$	
$\bot \to \varphi \sim \top$	
$\top \to \varphi \sim \varphi$	
$\varphi \wedge \psi \to \varphi \sim \top$	
$\varphi \wedge \psi \to \psi \sim \top$	Leyes de simplificación

Dem. de $\varphi \to \psi \sim \neg(\varphi \wedge \neg\psi)$:

$\varphi \to \psi \sim \neg\varphi \vee \psi$	relación entre $\to, \neg$ y $\vee$
$\sim \neg\varphi \vee \neg\neg\psi$	doble negación y reemplazamiento
$\sim \neg(\varphi \wedge \neg\psi)$	De Morgan

Ejemplo de razonamiento lógicamente correcto:

Si encuentras esto difícil entonces no eres inteligente o no lo has trabajado.
Lo has trabajado y eres inteligente luego no lo encontrarás difícil.

$$\text{Premisas} \begin{cases} (p \to (\neg q \vee \neg r)) \\ (r \wedge q) \end{cases}$$

$$\text{Conclusión} \quad \therefore\ \neg p$$

p :Encuentras esto difícil
q :Eres inteligente
r :Lo has trabajado

Implicación lógica relacionada: φ: $(((p \to (\neg\, q \vee \neg\, r)) \wedge (r \wedge q)) \to \neg\, p\ )$

$((p \to (\neg\, q \vee \neg\, r)) \wedge (r \wedge q)) \to \neg\, p$	
$\sim \neg((p \to (\neg\, q \vee \neg\, r)) \wedge (r \wedge q)) \vee \neg\, p$	relación entre $\to, \neg$ y $\vee$
$\sim \neg((\neg p \vee (\neg\, q \vee \neg\, r)) \wedge (r \wedge q)) \vee \neg\, p$	relación entre $\to, \neg$ y $\vee$ y reemplazamiento
$\sim (\neg(\ \neg p \vee (\neg\, q \vee \neg\, r)) \vee \neg\ (r \wedge q)) \vee \neg\, p$	De Morgan y reemplazamiento
$\sim (\neg\neg p \wedge \neg\ (\neg\, q \vee \neg\, r) \vee \neg\ (r \wedge q)) \vee \neg\, p$	De Morgan y reemplazamiento
$\sim (p \wedge \neg\neg\, q \wedge \neg\neg\, r) \vee \neg\, r \vee \neg\, q \vee \neg\, p$	De Morgan, doble negación y reemplazamiento
$\sim (p \wedge q \wedge r) \vee \neg\, r \vee \neg\, q \vee \neg\, p$	doble negación y reemplazamiento
$\sim (p \wedge q \wedge r) \vee \neg\ (p \wedge q \wedge r)$	De Morgan, conmutatividad y reemplazamiento
$\sim \top$	tercio excluido

Def:

- **Literal:** *cualquier fórmula de la forma p (literal positivo) o $\neg\, p$ (literal negativo), con $p \in \Sigma$*
- **Cláusula conjuntiva:** *cualquier fórmula que es una conjunción de literales.*
- **Cláusula disyuntiva:** *cualquier fórmula que es una disyunción de literales.*
- *Una fórmula está en* **forma normal conjuntiva (FNC)** *si es una conjunción de cláusulas disyuntivas.*
- *Una fórmula está en* **forma normal disjuntiva (FND)** *si es una disyunción de cláusulas conjuntivas.*

Convenios:

- Un único literal puede considerarse, indistintamente, como conjunción o como disyunción.
- $\perp$ está en FND y es una cláusula disyuntiva (representa una disyunción vacía (trivialmente falsa)).
- $\top$ está en FNC y es una cláusula conjuntiva (representa una conjunción vacía (trivialmente cierta)).

Observaciones:

- Una única cláusula disyuntiva puede considerarse que está en FNC (con una cláusula) o en FND (con varias cláusulas conjuntivas unitarias (literal o $\perp$ o $\top$))
- Una única cláusula conjuntiva puede considerarse que está en FND (con una cláusula) o en FNC (con varias cláusulas disyuntivas unitarias (literal o $\perp$ o $\top$))

Formas normales conjuntiva y disyuntiva (3)

TEOREMA: Dada $\varphi \in L_\Sigma$, pueden construirse, usando sólo símbolos del vocabulario de φ , sendas fórmulas $\mathsf{FND}(\varphi)$ y $\mathsf{FNC}(\varphi)$ tales que

$$FND(\varphi) \sim \varphi, \quad FND(\varphi) \text{ está en FND}$$

$$FNC(\varphi) \sim \varphi, \quad FNC(\varphi) \text{ está en FNC}$$

(Obs: Las $FNC(\varphi)$ y $FND(\varphi)$ no son únicas.)

Dem:

- Si $\varphi = \bot$ entonces $\mathsf{FND}(\varphi) = \bot$ y $\mathsf{FNC}(\varphi) = \bot \wedge \bot$
- Si $\varphi = \top$ entonces $\mathsf{FNC}(\varphi) = \top$ y $\mathsf{FND}(\varphi) = \top \vee \top$
- En otro caso, si $Mod(\varphi) = \{v_1, \cdots, v_m\}$ con $m > 0$ y $voc(\varphi) = \{p_1, \cdots, p_n\}$ con $n > 0$, definimos

$$FND(\varphi) = \bigvee_{v \in Mod(\varphi)} \varphi_v = \varphi_{v_1} \vee \cdots \vee \varphi_{v_m} \quad \text{siendo} \quad \varphi_{v_i} = \lambda_{i,1} \wedge \lambda_{i,2} \wedge \cdots \wedge \lambda_{i,n} \quad \text{donde}$$

$$\lambda_{i,j} = \begin{cases} \mathsf{p}_j & \text{si} \quad \mathsf{v}_i(\mathsf{p}_j) = 1 \\ \neg \mathsf{p}_j & \text{si} \quad \mathsf{v}_i(\mathsf{p}_j) = 0 \end{cases}$$

$$FNC(\varphi) = \neg FND(\neg \varphi)$$

Ej.: Escritura en FND y FNC de $\varphi = ((p \to q) \to r)$

a) **A partir de su tabla de verdad**

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

$$FND(\varphi) = (\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge r) \vee (p \wedge \neg q \wedge \neg r) \vee (p \wedge \neg q \wedge r) \vee (p \wedge q \wedge r)$$

$$FND(\neg\varphi) = (\neg p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge q \wedge \neg r) \vee (p \wedge q \wedge \neg r)$$

$$FNC(\varphi) = (p \vee q \vee r) \wedge (p \vee \neg q \vee r) \wedge (\neg p \vee \neg q \vee r)$$

Ej.: Escritura en FND y FNC de $\varphi = ((p \to q) \to r)$

b) **A partir de equivalencias lógicas**

$\varphi = ((p \to q) \to r)$	
$\sim (\neg(p \to q) \lor r)$	relación entre $\to, \neg$ y $\lor$
$\sim (\neg\ (\neg\ p \lor q) \lor r)$	relación entre $\to, \neg$ y $\lor$ y reemplazamiento
$\sim (p \land \neg\ q) \lor r = \text{FND}(\varphi)$	De Morgan, doble negación y reemplazamiento
$\sim (p \lor r) \land (\neg q \lor r) = \text{FNC}(\varphi)$	distributividad

Leyes de equivalencia lógica para simplificación de fórmulas en forma normal.

$$(DIS) \quad (\varphi \wedge \psi) \vee (\varphi \wedge \neg\psi) \sim \varphi$$

$$(CON) \quad (\varphi \vee \psi) \wedge (\varphi \vee \neg\psi) \sim \varphi$$

y en ambos casos se dice que las dos cláusulas asocian con respecto a ψ

Ej.: $\varphi = ((p \rightarrow q) \rightarrow r)$

$$FND(\varphi) = \underbrace{(\neg p \wedge \neg q \wedge r)}_{C_1} \vee \underbrace{(\neg p \wedge q \wedge r)}_{C_2} \vee \underbrace{(p \wedge \neg q \wedge \neg r)}_{C_3} \vee \underbrace{(p \wedge \neg q \wedge r)}_{C_4} \vee \underbrace{(p \wedge q \wedge r)}_{C_5}$$

$$\sim \underbrace{(\neg p \wedge r)}_{C_6} \vee \underbrace{(p \wedge \neg q)}_{C_7} \vee \underbrace{(p \wedge r)}_{C_8}$$

por idempotencia, reemplazamiento y
(DIS): C_4 y C_3 asocian respecto a r
C_1 y C_2 asocian respecto a q
C_4 y C_5 asocian respecto a q

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

por (DIS): C_6 y C_8 asocian respecto a p

Ej.: $\varphi = ((p \to q) \to r)$

$$FNC(\varphi) = \underbrace{(p \vee q \vee r)}_{C_1} \wedge \underbrace{(p \vee \neg q \vee r)}_{C_2} \wedge \underbrace{(\neg p \vee \neg q \vee r)}_{C_3}$$

$\sim (p \vee r) \wedge (\neg q \vee r)$ por idempotencia, reemplazamiento y

(CON): C_1 y C_2 asocian respecto a q

C_2 y C_3 asocian respecto a p

Método de cálculo lógico que permite:

- Decidir si una fórmula dada es consecuencia lógica de unas premisas y construir un contraejemplo en el caso de que no lo sea.
- Decidir si un conjunto de fórmulas es satisfactible.
- Decidir si una fórmula es tautología.
- Calcular formas normales conjuntivas y disyuntivas.

Método de deducción alternativo a las tablas de verdad:

- Menos costoso en muchos casos que las tablas de verdad.
- Extensible a otras lógicas en las que las tablas de verdad dejan de tener sentido.
- Sirve de base para demostradores automáticos.

- **Procedimiento de refutación:** Para demostrar $\Phi \models \phi$ intenta demostrar que $\Phi \cup \{\neg\phi\}$ es insatisfactible. Para demostrar que φ es tautología, intenta demostrar que $\neg\varphi$ es contradicción.
- **Idea base:** Cada fórmula proposicional compuesta o es un literal negativo o es simplificable o es lógicamente equivalente a la disyuncción o conjunción de otras dos fórmulas más sencillas.

Fórmulas simplificables: $\sigma \sim \sigma_1$		Fórmulas conjuntivas: $\alpha \sim \alpha_1 \wedge \alpha_2$			Fórmulas disyuntivas: $\beta \sim \beta_1 \vee \beta_2$		
σ	σ_1	α	α_1	α_2	β	β_1	β_2
$\neg\top$	$\bot$	$\varphi \wedge \psi$	φ	ψ	$\varphi \vee \psi$	φ	ψ
$\neg\bot$	$\top$	$\neg(\varphi \vee \psi)$	$\neg\varphi$	$\neg\psi$	$\neg(\varphi \wedge \psi)$	$\neg\varphi$	$\neg\psi$
$\neg\neg\varphi$	φ	$\neg(\varphi \to \psi)$	φ	$\neg\psi$	$(\varphi \to \psi)$	$\neg\varphi$	ψ
		$\varphi \leftrightarrow \psi$	$\varphi \to \psi$	$\psi \to \varphi$	$\neg(\varphi \leftrightarrow \psi)$	$\neg(\varphi \to \psi)$	$\neg(\psi \to \varphi)$

$\sigma_1, \alpha_1, \alpha_2, \beta_1$ y β_2 se dice que son los **constituyentes** respectivamente de σ, α y β.

El estudio de σ, α o β se reduce al de sus constituyentes.

Esquemas de reducción

$$\frac{\sigma}{\sigma_1} \qquad \frac{\alpha}{\begin{array}{c}\alpha_1\\ \alpha_2\end{array}} \qquad \frac{\beta}{\beta_1 \mid \beta_2}$$

- Para fórmulas simplificables $\sigma \sim \sigma_1$: Para satisfacer σ basta con satisfacer σ_1
- Para fórmulas conjuntivas $\alpha \sim \alpha_1 \wedge \alpha_2$: Para satisfacer α basta con satisfacer α_1 **y** α_2 conjuntamente.
- Para fórmulas disyuntivas $\beta \sim \beta_1 \vee \beta_2$: Para satisfacer β basta con satisfacer β_1 **o** β_2 . El símbolo $\mid$ denota la alternativa entre β_1 y β_2

DEF:

Un **árbol de fórmulas es** *un árbol unario-binario cuyos nodos están etiquetados por fórmulas. Una* **rama** θ *de un árbol de fórmulas se llama* **cerrada** *si* $\perp$ *aparece en* θ *o para alguna fórmula* φ *aparecen en* θ *tanto* φ *como* $\neg\varphi$. *A las ramas que no son cerradas se las llama abiertas.*

DEF:

Un **tableau** T *para un conjunto finito de fórmulas* $\Phi = \{\varphi_1, \cdots, \varphi_n\}$ *es cualquier árbol de fórmulas construído en un número finito de pasos mediante las siguientes reglas de formación:*

- **Regla de inicialización** $[R_{ini}]$: El árbol de fórmulas

$$\begin{array}{c}\varphi_1\\ |\\ \varphi_2\\ \vdots\\ \varphi_{n-1}\\ |\\ \varphi_n\end{array}$$

formado con una sola rama con nodos etiquetados con las fórmulas de Φ es un tableau para Φ (tableau inicial)

- **Reglas de reducción:** Si T es un tableau para Φ y T' se obtiene a partir de T por alguna de las reglas siguientes, entonces T' también es un tableau para Φ.
  - $[R_\sigma]$: Si θ es una rama abierta de T con un nodo etiquetado con una fórmula simplificable σ, se obtiene T' alargando θ con σ_1.
    No se aplica esta regla si σ_1 ya aparecía en θ.
  - $[R_\alpha]$: Si θ es una rama abierta de T con un nodo etiquetado con una fórmula conjuntiva α, se obtiene T' alargando θ con dos nodos etiquetados con las constituyentes α_1 y α_2.
    No se aplica esta regla si α_1 **y** α_2 ya aparecían en θ. Si sólo aparece en θ uno de los dos constituyentes, se obtiene T' alargando θ con un nodo etiquetado con el otro constituyente.
  - $[R_\beta]$: Si θ es una rama abierta de T con un nodo etiquetado con una fórmula disyuntiva β, se obtiene T' añadiendo como hijos de θ dos nodos etiquetados con las constituyentes β_1 y β_2.
    No se aplica esta regla si β_1 **o** β_2 ya aparecían en θ.

Un tabbleau queda **terminado** cuando no se puede aplicar ninguna regla.

Reglas de construcción de tableaux. (4)

Def:

Un **tableau** *se llama* **cerrado** *si todas sus ramas están cerradas. Las ramas cerradas se marcan con $\sharp$ seguido de un identificador de los nodos que hacen que la rama se cierre; las ramas que quedan abiertas se marcan con $\Uparrow$.*

- Cada rama del árbol representará un escenario en el que estudiamos si se pueden satisfacer todas las fórmulas del conjunto de fórmulas en estudio.
  - Si en una misma rama aparecen una fórmula y su negación, esa rama representa un escenario imposible ("cierra"). La situación que plantea esa rama es insatisfactible.
  - Si todas las ramas cierran, el conjunto es insatisfactible

Ej:

$$\frac{\begin{array}{c}\varphi \to (\neg\psi \lor \neg\gamma) \\ \gamma \land \psi\end{array}}{\therefore \neg\varphi}$$

$$\Phi = \{(\varphi \to (\neg\psi \lor \neg\gamma)), (\gamma \land \psi), \neg\neg\varphi\}$$

Veamos mediante tableaux que $\varphi \to (\neg\psi \lor \neg\gamma), \gamma \land \psi \models \neg\varphi$

(1) $\varphi \to (\neg\psi \lor \neg\gamma)$
(2) $\gamma \land \psi$
(3) $\neg\neg\varphi$ $[R_{ini}]$
(4) φ $[R_\sigma, 3]$
(5) γ
(6) ψ $[R_\alpha, 2]$

$[R_\beta, 1]$

- (7) $\neg\varphi$
  - $\sharp(4,7)$
- (8) $(\neg\psi \lor \neg\gamma)$ $[R_\beta, 8]$
  - (9) $\neg\psi$
    - $\sharp(6,9)$
  - (10) $\neg\gamma$
    - $\sharp(5,10)$

Heurísticas

- Dependiendo del orden en que se apliquen las reglas de formación se pueden construir tableaux diferentes para un mismo conjunto de fórmulas:
  - Cada vez que se reduzca una fórmula conviene trasladar las fórmulas constituyentes a todas las ramas abiertas que pasen por el nodo de la fórmula.
  - Es mejor reducir primero las fórmulas simplificables y conjuntivas para evitar demasiadas bifurcaciones.
  - Entre las disyuntivas conviene reducir primero las que vayan a producir que alguna rama se cierre inmediatamente.
  - Conviene intentar cerrar ramas antes de expandir otras.

Def:

Si para el conjunto de fórmulas Φ *puede construirse un tableau cerrado* T, *diremos que* T *prueba la insastifactibilidad de* Φ. *En particular si* $\Phi = \Phi_0 \cup \{\neg\psi\}$ *se dice que* **prueba por refutación** *que* $\Phi_0 \models \psi$ *lo que se escribe* $\Phi_0 \vdash_{tb} \psi$

Def:

Si θ *es una rama abierta de un tableau terminado, se define(n) la(s) valoración(es) asociada(s):*

$$v_\theta(p) = \begin{cases} 1 & \text{si } p \text{ aparece en } \theta \\ 0 & \text{si } \neg p \text{ aparece en } \theta \\ \text{arbitrario} & \text{o.c.} \end{cases}$$

Teorema fundamental de los tableaux

Dado $\Phi \subseteq L_\Sigma$

1. Φ es insatisfactible si y sólo si Φ tiene un tableau cerrado.
2. $\Phi \models \psi$ si y sólo si $\Phi \vdash_{tb} \psi$.
3. Si T es un tableau terminado y no cerrado de Φ, cada rama abierta θ cumple $v_\theta \models \Phi$. En particular, si $\Phi = \Phi_0 \cup \{\neg\psi\}$, entonces v_θ es una valoración contraejemplo que prueba $\Phi_0 \not\models \psi$

Cálculo de formas normales con tableaux.

Dada $\varphi \in L_\Sigma$

1. Construir un tableau terminado T para φ
2. Si T es cerrado: $FND(\varphi) = \perp$ y $FNC(\varphi) = \perp \wedge \perp$.
3. Si T no es cerrado:
   - Para cada rama abierta θ se construye la conjunción φ_θ de todos los literales que etiquetan nodos de la rama θ.

$$FND(\varphi) = \bigvee_{\theta \text{ rama abierta de } T} \varphi_\theta$$

   - Para calcular $FNC(\varphi)$ se construye $FND(\neg\varphi)$ por tableaux y se termina utilizando $FNC(\varphi) \sim \neg FND(\neg\varphi)$ y las leyes de De Morgan.