

Sintaxis y semántica de la lógica proposicional

Francisco J. Martín Mateos

Dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla

- Objetivos de la lógica
 - Formalización del lenguaje natural
 - Desarrollar métodos de razonamiento independientes del contexto
- Sistemas lógicos
 - Lógica proposicional
 - Lógica de primer orden
 - Lógica de orden superior
 - Lógicas modales
 - Lógicas descriptivas
- Aplicaciones de la lógica
 - Fundamento de lenguajes de programación
 - Verificación y síntesis automática de programas
 - Representación del conocimiento y razonamiento
 - Modelización y razonamiento sobre sistemas

- Las lógicas clásicas son sistemas formales que cumplen:
 - *Principio de tercio excluso*: Cualquier afirmación es verdadera o falsa.
 - *Principio de no contradicción*: No hay afirmaciones que sean verdaderas y falsas al mismo tiempo.
 - *Principio de explosión*: De una afirmación contradictoria se puede deducir cualquier cosa.
 - *Principio de monotonía de la implicación*: Las hipótesis de cualquier hecho derivado pueden ampliarse con supuestos adicionales.
- Ejemplos:
 - **Lógica proposicional**
 - Lógica de primer orden

- Argumentaciones
 - Si el tren llega a las 7 y no hay taxis en la estación, entonces Juan llegará tarde a la reunión. Juan no ha llegado tarde a la reunión. El tren llegó a las 7. Por tanto, había taxis en la estación.
 - Si hay corriente y la lámpara no está fundida, entonces está encendida. La lámpara no está encendida. Hay corriente. Por tanto, la lámpara está fundida.

- Formalización
 - Primer caso
 - $p \equiv$ “el tren llega a las 7”
 - $q \equiv$ “hay taxis en la estación”
 - $r \equiv$ “Juan llegará tarde a la reunión”
 - Segundo caso
 - $p \equiv$ “hay corriente”
 - $q \equiv$ “la lámpara está fundida”
 - $r \equiv$ “la lámpara está encendida”
- En ambos casos
 - Argumentación: Si p y no q , entonces r . No r . p . Por tanto, q .
 - Formulación: $\{p \wedge \neg q \rightarrow r, \neg r, p\} \models q$

- Alfabeto proposicional
 - Variables proposicionales: $p, q, r, \dots, p_1, p_2, \dots$
 - Conectivas lógicas
 - Conectivas monarias:
negación, \neg
 - Conectivas binarias:
conjunción, \wedge
disyunción, \vee
condicional, \rightarrow
bicondicional, \leftrightarrow
 - Símbolos adicionales: (y)

- Fórmulas proposicionales:
 - **Fórmulas atómicas:** Variables proposicionales
 - Si F y G son fórmulas entonces también son fórmulas
 - Negación: $\neg F$
 - Conjunción: $(F \wedge G)$
 - Disyunción: $(F \vee G)$
 - Condicional: $(F \rightarrow G)$
 - Bicondicional: $(F \leftrightarrow G)$
- Ejemplos
 - Son fórmulas: p , $(p \vee \neg q)$, $((p \wedge q) \rightarrow (q \vee p))$
 - No es fórmula: $(p \wedge \vee q)$

- Notación
 - Variables proposicionales: p, q, r
 - Conjunto de todas las variables proposicionales: VP
 - Fórmulas proposicionales: F, G, H
 - Conjunto de todas las fórmulas proposicionales: $Prop$
 - Una conectiva binaria cualquiera: \star
- Los paréntesis se eliminan cuando no hay duda sobre el uso de las conectivas lógicas.
 - Se eliminan los paréntesis externos:
 - $F \vee G$ en lugar de $(F \vee G)$.
 - Se usa una precedencia entre las conectivas: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$.
 - $F \wedge G \rightarrow \neg F \vee G$ en lugar de $(F \wedge G) \rightarrow (\neg F \vee G)$
 - Asociación por la derecha de las conectivas:
 - $F \vee G \vee H$ en lugar de $F \vee (G \vee H)$

Sintaxis de la lógica proposicional

- Definiciones recursivas
 - Comportamiento para las variables proposicionales
 - Comportamiento para las fórmulas dependiendo del tipo de conectiva y usando el comportamiento en las componentes
- Número de variables proposicionales

$$nvp(F) = \begin{cases} 1 & \text{si } F \text{ es atómica} \\ nvp(G) & \text{si } F \text{ es } \neg G \\ nvp(G) + nvp(H) & \text{si } F \text{ es } G \star H \end{cases}$$

- $nvp(p) = 1$
- $nvp(\neg q) = 1$
- $nvp(\neg q \vee p) = 2$
- $nvp(p \rightarrow \neg q \vee p) = 3$

Sintaxis de la lógica proposicional

- Definiciones recursivas
 - Comportamiento para las variables proposicionales
 - Comportamiento para las fórmulas dependiendo del tipo de conectiva y usando el comportamiento en las componentes
- Número de conectivas

$$nc(F) = \begin{cases} 0 & \text{si } F \text{ es atómica} \\ 1 + nc(G) & \text{si } F \text{ es } \neg G \\ 1 + nc(G) + nc(H) & \text{si } F \text{ es } G \star H \end{cases}$$

- $nc(p) = 0$
- $nc(\neg q) = 1$
- $nc(\neg q \vee p) = 2$
- $nc(p \rightarrow \neg q \vee p) = 3$

Sintaxis de la lógica proposicional

- Definiciones recursivas
 - Comportamiento para las variables proposicionales
 - Comportamiento para las fórmulas dependiendo del tipo de conectiva y usando el comportamiento en las componentes
- Número de conectivas binarias

$$ncb(F) = \begin{cases} 0 & \text{si } F \text{ es atómica} \\ ncb(G) & \text{si } F \text{ es } \neg G \\ 1 + ncb(G) + ncb(H) & \text{si } F \text{ es } G \star H \end{cases}$$

- $ncb(p) = 0$
- $ncb(\neg q) = 0$
- $ncb(\neg q \vee p) = 1$
- $ncb(p \rightarrow \neg q \vee p) = 2$

Sintaxis de la lógica proposicional

- Definiciones recursivas
 - Comportamiento para las variables proposicionales
 - Comportamiento para las fórmulas dependiendo del tipo de conectiva y usando el comportamiento en las componentes
- Conjunto de variables proposicionales

$$VP(F) = \begin{cases} \{F\} & \text{si } F \text{ es atómica} \\ VP(G) & \text{si } F \text{ es } \neg G \\ VP(G) \cup VP(H) & \text{si } F \text{ es } G \star H \end{cases}$$

- $VP(p) = \{p\}$
- $VP(\neg q) = \{q\}$
- $VP(\neg q \vee p) = \{p, q\}$
- $VP(p \rightarrow \neg q \vee p) = \{p, q\}$

Sintaxis de la lógica proposicional

- Definiciones recursivas
 - Comportamiento para las variables proposicionales
 - Comportamiento para las fórmulas dependiendo del tipo de conectiva y usando el comportamiento en las componentes
- Conjunto de **subfórmulas**

$$\mathit{subf}(F) = \begin{cases} \{F\} & \text{si } F \text{ es atómica} \\ \{\neg G\} \cup \mathit{subf}(G) & \text{si } F \text{ es } \neg G \\ \{G \star H\} \cup \mathit{subf}(G) \cup \mathit{subf}(H) & \text{si } F \text{ es } G \star H \end{cases}$$

- $\mathit{subf}(p) = \{p\}$
- $\mathit{subf}(\neg q) = \{\neg q, q\}$
- $\mathit{subf}(\neg q \vee p) = \{\neg q \vee p, \neg q, q, p\}$
- $\mathit{subf}(p \rightarrow \neg q \vee p) = \{p \rightarrow \neg q \vee p, p, \neg q \vee p, \neg q, q\}$

- **Principio de inducción estructural:** Si \mathcal{P} es una propiedad sobre las fórmulas que verifica
 - Todas las fórmulas atómicas cumplen la propiedad \mathcal{P}
 - Si F cumple la propiedad \mathcal{P} entonces $\neg F$ también la cumple
 - Si F y G cumplen la propiedad \mathcal{P} entonces $F \star G$ también la cumple

Entonces cualquier fórmula proposicional cumple la propiedad \mathcal{P}

Sintaxis de la lógica proposicional

- El número de variables proposicionales es uno más que el número de conectivas binarias: $nvp(F) = 1 + ncb(F)$
 - **Caso base:** Si F es atómica entonces

$$nvp(F) = 1 = 1 + 0 = 1 + ncb(F)$$

- **Caso de inducción 1:** Supongamos que $nvp(F) = 1 + ncb(F)$ (**hipótesis de inducción**), entonces

$$\begin{aligned}nvp(\neg F) &= nvp(F) && \text{def. de } nvp \\ &= 1 + ncb(F) && \text{hip. de inducción} \\ &= 1 + ncb(\neg F) && \text{def. de } ncb\end{aligned}$$

- **Caso de inducción 2:** Supongamos que $nvp(F) = 1 + ncb(F)$ y $nvp(G) = 1 + ncb(G)$ (**hipótesis de inducción**), entonces

$$\begin{aligned}nvp(F \star G) &= nvp(F) + nvp(G) && \text{def. de } nvp \\ &= 1 + ncb(F) + 1 + ncb(G) && \text{hip. de inducción} \\ &= 1 + ncb(F \star G) && \text{def. de } ncb\end{aligned}$$

Semántica de la lógica proposicional

- Valores de verdad: verdadero (\top o $\mathbf{1}$) y falso (\perp o $\mathbf{0}$)
 - El conjunto de los valores de verdad es \mathbb{B}
- Funciones de verdad asociadas a las conectivas:

$$H_{\neg}(i) = \begin{cases} \mathbf{1} & \text{si } i = \mathbf{0} \\ \mathbf{0} & \text{si } i = \mathbf{1} \end{cases}$$

$$H_{\wedge}(i, j) = \begin{cases} \mathbf{1} & \text{si } i = j = \mathbf{1} \\ \mathbf{0} & \text{en otro caso} \end{cases}$$

$$H_{\vee}(i, j) = \begin{cases} \mathbf{0} & \text{si } i = j = \mathbf{0} \\ \mathbf{1} & \text{en otro caso} \end{cases}$$

$$H_{\rightarrow}(i, j) = \begin{cases} \mathbf{0} & \text{si } i = \mathbf{1} \text{ y } j = \mathbf{0} \\ \mathbf{1} & \text{en otro caso} \end{cases}$$

$$H_{\leftrightarrow}(i, j) = \begin{cases} \mathbf{1} & \text{si } i = j \\ \mathbf{0} & \text{en otro caso} \end{cases}$$

- Tablas de verdad:
 - Conectivas monarias:

i	$\neg i$
1	0
0	1

- Conectivas binarias:

i	j	$i \wedge j$	$i \vee j$	$i \rightarrow j$	$i \leftrightarrow j$
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	0	0
1	1	1	1	1	1

- Una **interpretación** es una función del conjunto de las variables proposicionales en el conjunto de valores de verdad.

$$I : VP \longrightarrow \mathbb{B}$$

- Para cada interpretación I existe una única función que da significado a las fórmulas, $I' : Prop \longrightarrow \mathbb{B}$, tal que:

$$I'(F) = \begin{cases} I(F) & \text{si } F \text{ es atómica} \\ H_{\neg}(I'(G)) & \text{si } F \text{ es } \neg G \\ H_{\star}(I'(G), I'(H)) & \text{si } F \text{ es } G \star H \end{cases}$$

- $I'(F)$ es el **valor de verdad de F con respecto a I**
- Dado que I' es única, escribiremos $I(F)$ en lugar de $I'(F)$

- Ejemplos

- Valor de verdad de $F = (p \vee q) \wedge (\neg q \vee r)$ con respecto a la interpretación I_1 tal que $I_1(p) = I_1(r) = 1$ e $I_1(q) = 0$

p	q	r	$p \vee q$	$\neg q$	$\neg q \vee r$	F
1	0	1	1	1	1	1

- Valor de verdad de $F = (p \vee q) \wedge (\neg q \vee r)$ con respecto a la interpretación I_2 tal que $I_2(r) = 1$ e $I_2(p) = I_2(q) = 0$

p	q	r	$p \vee q$	$\neg q$	$\neg q \vee r$	F
0	0	1	0	1	1	0

- Propiedad: Si dos interpretaciones I_1 e I_2 coinciden en las variables proposicionales de F entonces $I_1(F) = I_2(F)$

- Una interpretación I es **modelo** de una fórmula F ($I \models F$), si F es verdadera con respecto a I
 - Si $I_1(p) = I_1(r) = 1$ e $I_1(q) = 0$, entonces
 $I_1 \models (p \vee q) \wedge (\neg q \vee r)$
 - Si $I_2(r) = 1$ e $I_2(p) = I_2(q) = 0$, entonces
 $I_2 \not\models (p \vee q) \wedge (\neg q \vee r)$
- Una fórmula es **satisfacible** si tiene modelos
 - La fórmula $(p \vee q) \wedge (\neg q \vee r)$ es satisfacible
- Una fórmula es **insatisfacible** si no tiene modelos
 - La fórmula $p \wedge \neg p$ es insatisfacible

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

Tautologías y contradicciones

- Una fórmula F es una **tautología** o **válida** ($\models F$), si toda interpretación es modelo de F
 - La fórmula $(p \rightarrow q) \vee (q \rightarrow p)$ es una tautología
- Una fórmula F es una **contradicción** o **insatisfacible**, si ninguna interpretación es modelo de F
 - La fórmula $(p \rightarrow q) \wedge \neg(p \rightarrow q)$ es una contradicción
- Una fórmula F es **contingente**, si no es una tautología ni una contradicción
 - La fórmula $p \rightarrow q$ es contingente

Clasificaciones de fórmulas

<i>Prop</i>		
Tautologías	Contingentes	Contradicciones
Verdadera en todas las interpretaciones $p \vee \neg p$	Verdadera en algunas interpretaciones y falsa en otras $p \vee q$	Falsa en todas las interpretaciones $p \wedge \neg p$
Satisfacibles		Insatisfacibles
<i>Prop</i>		

- Los problemas SAT y TAUT
 - Problema SAT: Dada una fórmula, determinar si es satisfacible
 - Problema TAUT: Dada una fórmula, determinar si es una tautología
- Relaciones entre estos dos problemas
 - F es una tautología $\iff \neg F$ es insatisfacible
 - F es una tautología $\implies F$ es satisfacible
 - F es satisfacible $\not\implies \neg F$ es insatisfacible
 - $p \rightarrow q$ es satisfacible
 - $\neg(p \rightarrow q)$ es satisfacible
- Hay distintos métodos para comprobar SAT y TAUT, el más simple es el basado en las tablas de verdad

Satisfacibilidad y validez

- Tabla de verdad para $\models (p \rightarrow q) \vee (q \rightarrow p)$

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

- Tabla de verdad simplificada

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

Satisfacibilidad y validez

- Tabla de verdad para $\not\equiv (p \leftrightarrow q) \vee (q \leftrightarrow p)$

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

- Tabla de verdad simplificada

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

- Algunas tautologías importantes
 - Ley de identidad: $\mathbf{p \rightarrow p}$
 - Ley del tercio excluido: $\mathbf{p \vee \neg p}$
 - Principio de no contradicción: $\mathbf{\neg(p \wedge \neg p)}$
 - Ley de Clavius: $\mathbf{(\neg p \rightarrow p) \rightarrow p}$
 - Ley de Duns Scoto: $\mathbf{\neg p \rightarrow (p \rightarrow q)}$
 - Ley de Pierce: $\mathbf{((p \rightarrow q) \rightarrow p) \rightarrow p}$
 - Modus ponens: $\mathbf{(p \rightarrow q) \wedge p \rightarrow q}$
 - Modus tollens: $\mathbf{(p \rightarrow q) \wedge \neg q \rightarrow \neg p}$

Fórmulas equivalentes

- Dos fórmulas F y G son **equivalentes** ($F \equiv G$), si para toda interpretación I se tiene que $I(F) = I(G)$
- Equivalencias importantes:

- Idempotencia:

$$F \vee F \equiv F$$

$$F \wedge F \equiv F$$

- Conmutatividad:

$$F \vee G \equiv G \vee F$$

$$F \wedge G \equiv G \wedge F$$

- Asociatividad:

$$F \vee (G \vee H) \equiv (F \vee G) \vee H$$

$$F \wedge (G \wedge H) \equiv (F \wedge G) \wedge H$$

- Absorción:

$$F \wedge (F \vee G) \equiv F$$

$$F \vee (F \wedge G) \equiv F$$

Fórmulas equivalentes

- Dos fórmulas F y G son **equivalentes** ($F \equiv G$), si para toda interpretación I se tiene que $I(F) = I(G)$
- Equivalencias importantes:

- Distributividad:

$$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$$

$$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$$

- Leyes de tautologías. Si F es una tautología:

$$F \wedge G \equiv G$$

$$F \vee G \equiv F$$

- Leyes de contradicciones. Si F es una contradicción:

$$F \wedge G \equiv F$$

$$F \vee G \equiv G$$

Fórmulas equivalentes

- Dos fórmulas F y G son **equivalentes** ($F \equiv G$), si para toda interpretación I se tiene que $I(F) = I(G)$

- Equivalencias importantes:

- Ley del condicional:

$$F \rightarrow G \equiv \neg F \vee G$$

- Ley del bicondicional:

$$F \leftrightarrow G \equiv (F \rightarrow G) \wedge (G \rightarrow F)$$

- Ley de la doble negación:

$$\neg\neg F \equiv F$$

- Leyes de De Morgan:

$$\neg(F \wedge G) \equiv \neg F \vee \neg G$$

$$\neg(F \vee G) \equiv \neg F \wedge \neg G$$

- La equivalencia lógica es una relación de equivalencia
 - Reflexividad: $F \equiv F$
 - Simetría: Si $F \equiv G$, entonces $G \equiv F$
 - Transitividad: Si $F \equiv G$ y $G \equiv H$, entonces $F \equiv H$
- Relación entre equivalencia y bicondicional

$$F \equiv G \text{ si y solo si } \models F \leftrightarrow G$$

- **Principio de sustitución de fórmulas equivalentes:** Si en una fórmula F se sustituye una subfórmula G por otra lógicamente equivalente G' , entonces la fórmula obtenida F' es lógicamente equivalente a la original
 - $F = \neg(p \wedge q) \rightarrow \neg(p \wedge \neg\neg r)$
 - $G = \neg(p \wedge \neg\neg r)$
 - $G' = \neg p \vee \neg r$
 - $F' = \neg(p \wedge q) \rightarrow \neg p \vee \neg r$

Conjunto completo de conectivas

- Un conjunto de conectivas \mathcal{C} es **funcionalmente completo** si cualquier fórmula proposicional tiene una equivalente en la que sólo se utilizan las conectivas de \mathcal{C}
 - El conjunto $\{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$ es funcionalmente completo por definición de **Prop**
 - El conjunto $\{\neg, \wedge, \vee\}$ es funcionalmente completo (ley del condicional, ley de bicondicional y principio de sustitución)
 - Los conjuntos $\{\neg, \wedge\}$ y $\{\neg, \vee\}$ son funcionalmente completos (ley del condicional, ley del bicondicional, ley de la doble negación, leyes de De Morgan y principio de sustitución)
 - Si definimos la conectiva \uparrow (operador de Sheffer) con la siguiente función de verdad

$$H_{\uparrow}(i, j) = \begin{cases} 0 & \text{si } i = j = 1 \\ 1 & \text{en otro caso} \end{cases}$$

entonces $\{\uparrow\}$ es un conjunto funcionalmente completo

- Una interpretación I es **modelo** de un conjunto de fórmulas S ($I \models S$), si para toda $F \in S$ se tiene que $I \models F$
- Sea $S = \{F_1 = (p \vee q) \wedge (\neg q \vee r), F_2 = p \rightarrow r\}$
 - La interpretación I_1 tal que $I_1(p) = I_1(r) = 1$ e $I_1(q) = 0$ es modelo de S .

p	q	r	$p \vee q$	$\neg q$	$\neg q \vee r$	$p \rightarrow r$	F_1	F_2
1	0	1	1	1	1	1	1	1

- La interpretación I_2 tal que $I_2(p) = I_2(q) = 0$ e $I_2(r) = 1$ no es modelo de S .

p	q	r	$p \vee q$	$\neg q$	$\neg q \vee r$	$p \rightarrow r$	F_1	F_2
0	0	1	0	1	1	1	0	1

Consistencia y consecuencia lógica

- Un conjunto \mathbf{S} es **consistente**, si tiene algún modelo
 - El conjunto $\{(p \vee q) \wedge (\neg q \vee r), p \rightarrow r\}$ es consistente
- Un conjunto \mathbf{S} es **inconsistente**, si no tiene ningún modelo
 - El conjunto $\{p \vee q, \neg p \vee q, p \vee \neg q, \neg p \vee \neg q\}$ es inconsistente.
- Una fórmula \mathbf{F} es **consecuencia lógica** de un conjunto de fórmulas \mathbf{S} ($\mathbf{S} \models \mathbf{F}$) si todos los modelos de \mathbf{S} son a su vez modelos de \mathbf{F} .
 - $\{p \rightarrow q, q \rightarrow r\} \models p \rightarrow r$
 - $\{p\} \not\models p \wedge q$

Propiedades de la consecuencia lógica

- Propiedades básicas de la relación de consecuencia lógica:
 - *Reflexividad*: Para toda $F \in S$, $S \models F$.
 - *Monotonía*: Si $S_1 \models F$ y $S_1 \subseteq S_2$, entonces $S_2 \models F$.
 - *Transitividad*: Si $S \models F$ y $\{F\} \models G$, entonces $S \models G$.
- Relación entre consecuencia, validez, satisfacibilidad y consistencia. Las siguientes condiciones son equivalentes:
 - $\{F_1, \dots, F_n\} \models G$.
 - $\models F_1 \wedge \dots \wedge F_n \rightarrow G$.
 - $\neg(F_1 \wedge \dots \wedge F_n \rightarrow G)$ es insatisfacible.
 - $\{F_1, \dots, F_n, \neg G\}$ es inconsistente.

- Un cálculo lógico es un procedimiento de decisión que permite comprobar la validez de una fórmula.
 - La deducción en un cálculo lógico se representa con el símbolo \vdash .
 - También permiten comprobar la satisfacibilidad, la consistencia y la consecuencia lógica.
- La combinación exhaustiva de todos los valores de verdad de las variables proposicionales y el cálculo del valor de verdad mediante las tablas de verdad es un ejemplo de cálculo lógico.
- Otros cálculos lógicos: tableros semánticos, secuentes, deducción natural, resolución, ...

Propiedades de los cálculos lógicos

- Un cálculo lógico es correcto ($\vdash \implies \models$), si toda fórmula que se puede demostrar que es válida con el cálculo lógico, es válida en la lógica.
 - Todo cálculo lógico debería cumplir esta propiedad.
- Un cálculo lógico es completo ($\models \implies \vdash$), si para toda fórmula válida en la lógica, se puede demostrar su validez utilizando únicamente el cálculo lógico.
- Existen cálculos lógicos para la lógica proposicional que son correctos y completos.
 - Los cálculos lógicos mencionados para la lógica proposicional son correctos y completos: tablas de verdad, tableros semánticos, secuentes, deducción natural, resolución, ...
- La lógica proposicional es decidible: existen procedimientos capaces de decidir si una fórmula es válida o no, en un número finito de pasos.

- Problema de los animales. Se sabe que:
 - Los animales con pelo o que dan leche son mamíferos.
 - Los mamíferos que rumian o tienen pezuñas son ungulados.
 - Los ungulados de cuello largo son jirafas.
 - Los ungulados con rayas negras son cebras.

Se observa que un animal tiene pelos, pezuñas y rayas negras.
Por tanto, se concluye que el animal es una cebra.

- Simbolización
 - ***p***: el animal tiene pelo.
 - ***l***: el animal da leche.
 - ***m***: el animal es un mamífero.
 - ***r***: el animal rumia.
 - ***pz***: el animal tiene pezuñas.
 - ***u***: el animal es ungulado.
 - ***cl***: el animal tiene cuello largo.
 - ***j***: el animal es una jirafa.
 - ***rn***: el animal tiene rayas negras.
 - ***c***: el animal es una cebra.

- Formalización de las premisas y la conclusión.
 - Los animales con pelo o que dan leche son mamíferos.
 $p \vee l \rightarrow m$
 - Los mamíferos que rumian o tienen pezuñas son ungulados.
 $m \wedge (r \vee pz) \rightarrow u$
 - Los ungulados de cuello largo son jirafas.
 $u \wedge cl \rightarrow j$
 - Los ungulados con rayas negras son cebras.
 $u \wedge rn \rightarrow c$
 - Se observa que un animal tiene pelos, pezuñas y rayas negras.
 $p \wedge pz \wedge rn$
 - Por tanto, se concluye que el animal es una cebra.
 c
- Formalización de la argumentación:

$$\{p \vee l \rightarrow m, m \wedge (r \vee pz) \rightarrow u, \\ u \wedge cl \rightarrow j, u \wedge rn \rightarrow c, p \wedge pz \wedge rn\} \models c$$

Problemas lógicos: veraces y mentirosos

- En una isla hay dos tribus, la de los veraces (que siempre dicen la verdad) y la de los mentirosos (que siempre mienten). Un viajero se encuentra con tres isleños A, B y C y cada uno le dice una frase:
 - A dice “B y C son veraces si y solo si C es veraz”.
 - B dice “Si A y C son veraces, entonces B y C son veraces y A es mentiroso”.
 - C dice “B es mentiroso si y solo si A o B es veraz”.
- ¿A qué tribu pertenece cada uno de ellos?
- Simbolización:
 - **a**: A es veraz.
 - **b**: B es veraz.
 - **c**: C es veraz.

- Formalización de las premisas:
 - A dice “B y C son veraces si y solo si C es veraz”.
 $F_1 : a \leftrightarrow (b \wedge c \leftrightarrow c)$
 - B dice “Si A y C son veraces, entonces B y C son veraces y A es mentiroso”.
 $F_2 : b \leftrightarrow (a \wedge c \rightarrow b \wedge c \wedge \neg a)$
 - C dice “B es mentiroso si y solo si A o B es veraz”.
 $F_3 : c \leftrightarrow (\neg b \leftrightarrow a \vee b)$
- Modelos de $S = \{F_1, F_2, F_3\}$:
 - Si $I \models S$, entonces $I(a) = 1, I(b) = 1, I(c) = 0$.
 - Por tanto, A y B son veraces y C es mentiroso

- M. Ben-Ari. *Mathematical logic for computer science*
Capítulos 1 y 2.
- J.A. Díez. *Iniciación a la Lógica*
Capítulos 2 y 3.
- M. Huth y M. Ryan. *Logic in computer science: modelling and reasoning about systems*
Capítulo 1.
- R. Smullyan. *¿Cómo se llama este libro?*