Apuntes de Introducción a la Lógica

Maria Luisa Bonet Carbonell

25 de febrero de 2004

Índice

Capítulo 1

Lógica Proposicional

1.1 Sintaxis de la Lógica Proposicional

Definición 1.1 (Lenguaje)

1. *conjunto de símbolos atómicos o variables:* $\{P, Q, R, S, \ldots, P_1, P_2, \ldots\}$.
2. *conjunto de conectivas proposicionales:* $\neg$, $\vee$, $\wedge$, $\to$, $\leftrightarrow$.
3. *paréntesis:* (,).

Definición 1.2 (Fórmula)

1. *Toda variable o átomo es una fórmula.*
2. *Si A y B son fórmulas, entonces* $(\neg A)$, $(A \wedge B)$, $(A \vee B)$, $(A \to B)$, $(A \leftrightarrow B)$ *son fórmulas.*
3. *Nada más es una fórmula*

Ejemplo: $\underbrace{(\underbrace{(\neg \underbrace{(\underbrace{P} \vee \underbrace{Q})})} \to \underbrace{(\underbrace{R} \wedge \underbrace{S})})}$ es una fórmula

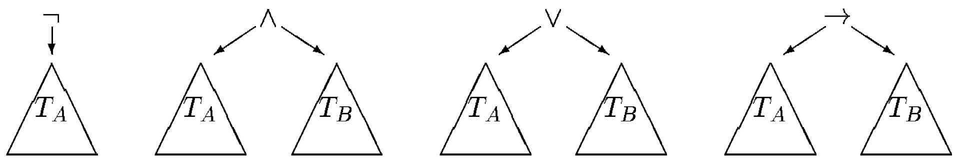
Eliminación de paréntesis

- Los exteriores siempre se pueden eliminar.
- Eliminaremos paréntesis siguiendo las reglas de prioridad: $\neg$, $\wedge/\vee$, $\to$ / $\leftrightarrow$.

Ejemplo: $((\neg(P \vee Q)) \rightarrow (R \wedge S))$ puede escribirse como $(\neg(P \vee Q)) \rightarrow (R \wedge S)$ eliminando los paréntesis exteriores. Esta última puede escribirse como $\neg(P \vee Q) \rightarrow (R \wedge S)$, dado que la negación tiene más prioridad que la implicación, y finalmente se puede sacar el paréntiesis de la conjunción final, dado que la conjunción tiene más prioridad que la implicación. Obtenemos $\neg(P \vee Q) \rightarrow R \wedge S$, y no se pueden sacar más paréntesis.

Definición 1.3 (Forma arbórea)

1. *El árbol correspondiente a una variable proposicional es un nodo.*
2. *Si T_A y T_B son las formas aróreas correspondientes a las fórmulas A y B, entonces*



son las formas arbóreas de $(\neg A)$, $(A \wedge B)$, $(A \vee B)$ y $(A \rightarrow B)$.

1.2 Semántica

Para evaluar si una fórmula es cierta o falsa necesitamos primero asignar valor a las variables. Las interpretaciones son funciones que hacen estas asignaciones.

Una *interpretación* I es una función del conjunto de variables en $\{0, 1\}$

$$I : \text{Conjunto de Variables} \rightarrow \{0, 1\}$$

Una interpretación es adecuada a una fórmula si asigna un valor 0 ó 1 a todas sus variables. Podemos extender cualquier interpretación I a una función I' que asigna valores 0 ó 1 a todas las fórmulas que podemos formar con el conjunto de variables.

1. Para toda variable proposicional P, $I'(P) = I(P)$
2. $I'(F \wedge G) = \begin{cases} 1 & \text{si } I'(F) = 1 \text{ y } I'(G) = 1 \\ 0 & \text{en caso contrario} \end{cases}$
3. $I'(F \vee G) = \begin{cases} 0 & \text{si } I'(F) = 0 \text{ y } I'(f) = 0 \\ 1 & \text{en caso contrario} \end{cases}$

4. $I'(\neg F) = \begin{cases} 0 & \text{si } I'(F) = 0 \\ 1 & \text{si } I'(F) = 1 \end{cases}$

5. $I'(F \to G) = \begin{cases} 0 & \text{si } I'(F) = 1 \text{ y } I'(G) = 0 \\ 1 & \text{en caso contrario} \end{cases}$

Definición 1.4 (Satisfactible e Insatisfactible) *Decimos que una fórmula es* satisfactible *si existe alguna interpretación que la satisface. Una fórmula es* insatisfactible *o* contradicción *sii ninguna interpretación la satisface. Finalmente, una fórmula es* válida *o* tautología *sii toda interpretación la satisface.*

Lema 1.5 *Una fórmula F es válida sii $\neg F$ es insatisfactible.*

Demostración:

$$\begin{aligned} F \text{ es válida} & \text{ sii para toda interpretación } I,\ I'(F) = 1 \\ & \text{ sii para toda interpretación } I,\ I'(\neg F) = 0 \\ & \text{ sii } \neg F \text{ es insatisfactible} \end{aligned}$$

■

Lema 1.6 *Si A y $A \to B$ son tautologías, entonces B es una tautología.*

Demostración: Supongamos que A y $A \to B$ son tautologías. Supongamos que B no es una tautología. Entonces existe una interpretación I que falsifica B. I satisface A y $A \to B$, pero I falsifica B. Contradicción. Por tanto B es una tautología. ■

Definición 1.7 (Equivalencia y Consecuencia lógica) *Una fórmula A es* equivalente *a B ($A \equiv B$) sii para toda interpretación I, I satisface a A sii I satisface a B. Una fórmula A es* consecuencia lógica *de un conjunto de fórmulas Γ ($\Gamma \models A$) sii toda interpretación I que satisface a todas las fórmulas de Γ también satisface a A.*

Ejemplo: Veamos que $A, A \to B \models B$. Sea I una interpretación arbitraria. Si I satisface a A y $A \to B$, entonces está forzada a satisfacer a B.

Lema 1.8 *Sea Γ un conjunto finito de fórmulas, y sea $\bigwedge \Gamma$ la conjunción de todas las fórmulas de Γ. Entonces $\Gamma \models A$ sii $\bigwedge \Gamma \wedge \neg A$ es insatisfactible.*

Demostración: Ejercicio. ■

Teorema 1.9 (Teorema del Reemplazamiento) *Si $F \equiv G$ y H es una fórmula que tiene F como subfórmula, entonces $H \equiv H'$ donde H' es el resultado de reemplazar una ocurrencia de F por G en H.*

Demostración: Por inducción sobre la profundidad del árbol de una fórmula H.

- Caso base (profundidad 0). H es una variable. $H = F \equiv G$, y $H' = G$. Por tanto, $H \equiv H'$.
- Paso de inducción: Suponemos que el enunciado es cierto para toda fórmula de profundidad $\leq k$. Sea H una fórmula de profundidad $k+1$.
- Caso 1: $H = \neg H_1$: H_1 tiene profundidad k: F es subfórmula de H y $F \equiv G$.
  - Si $F = H$, la demostración es como en el caso base.
  - Si F es subfórmula de H_1, por hipotesis de inducción $H_1' \equiv H_1$, donde H_1' es el resultado de reemplazar F por G en H_1. $H' = \neg H_1'$.
    Dado que $H_1' \equiv H_1$, por la definición semántica de la $\neg$, $H \equiv H'$ ($H' = \neg H_1' \equiv \neg H_1 = H$).
- Caso 2: $H = H_1 \vee H_2$. F es una subfórmula de H_1 o de H_2 o es $H_1 \vee H_2$.
  - Si F es una subformula de H_1, como H_1 tiene profundidad $\leq k$, $H_1 \equiv H_1'$. $H' = H_1' \vee H_2$ y $H \equiv H'$ por la definición semántica de $\vee$.
  - Si F es una subfórmula de H_2, lo demostramos como el caso anterior.
  - Si F es $H_1 \vee H_2$, entonces la demostración es como en el caso base.
- El resto de los casos se hace de forma similar.

■

Teorema 1.10 (Equivalencias elementales)

$$\left.\begin{array}{l} F \wedge F \equiv F \\ F \vee F \equiv F \end{array}\right\} \textit{idempotencia}$$

$$\left.\begin{array}{l} F \wedge G \equiv G \wedge F \\ F \vee G \equiv G \vee F \end{array}\right\} \textit{conmutativa}$$

$$\left.\begin{array}{l} (F \wedge G) \wedge H \equiv F \wedge (G \wedge H) \\ (F \vee G) \vee H \equiv F \vee (G \vee H) \end{array}\right\} \textit{asociativa}$$

$$\left.\begin{array}{l} F \wedge (F \vee G) \equiv F \\ (F \vee (F \wedge G) \equiv F \end{array}\right\} \textit{absorción}$$

$$\left.\begin{array}{l} F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H) \\ F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H) \end{array}\right\} \textit{distributiva}$$

$$\neg\neg F \equiv F\} \textit{ doble negación}$$

$$\left.\begin{array}{l} \neg(F \wedge G) \equiv \neg F \vee \neg G \\ \neg(F \vee G) \equiv \neg F \wedge \neg G \end{array}\right\} \textit{leyes deMorgan}$$

$$\left.\begin{array}{l} F \vee G \equiv F \textit{ si } F \textit{ es tautología} \\ F \wedge G \equiv G \textit{ si } F \textit{ es tautología} \end{array}\right\} \textit{leyes de la tautología}$$

$$\left.\begin{array}{l} F \vee G \equiv G \textit{ si } F \textit{ es insatisfactible} \\ F \wedge G \equiv F \textit{ si } F \textit{ es insatisfactible} \end{array}\right\} \textit{leyes de la insatisfactibilidad}$$

$$\left.\begin{array}{l} F \rightarrow G \equiv \neg F \vee G \\ F \leftrightarrow G \equiv (F \rightarrow G) \wedge (G \rightarrow F) \end{array}\right\} \textit{transformación de implicaciones}$$

Demostración: Ejercicio. ∎

Lema 1.11 (Generalizacion de la distributiva) *A partir de las leyes de distributibidad anteriormente dadas, podemos obtener las siguientes:*

$$(\bigvee_{i=1}^{n} F_i) \wedge (\bigvee_{j=1}^{m} G_j) \equiv \bigvee_{i=1}^{n} \bigvee_{j=1}^{m} F_i \wedge G_i$$

$$(\bigwedge_{i=1}^{n} F_i) \vee (\bigwedge_{j=1}^{m} G_j) \equiv \bigwedge_{i=1}^{n} \bigwedge_{j=1}^{m} F_i \vee G_i$$

Demostración: Se demuestra por inducción sobre el número de disyunciones (o conjunciones) n. Ejercicio. ∎

Ejemplo: $(A \vee B) \wedge (C \vee D) \equiv (A \wedge (C \vee D)) \vee (B \wedge (C \vee D)) \equiv (A \wedge C) \vee (A \wedge D) \vee (B \wedge C) \vee (B \wedge D)$

Ejemplo: Demostrar que $(A \vee (B \vee C)) \wedge (C \vee \neg A) \equiv (B \wedge \neg A) \vee C$

$$
\begin{aligned}
(A \vee (B \vee C)) \wedge (C \vee \neg A) &\equiv ((A \vee B) \vee C) \wedge (C \vee \neg A) && : \text{asociativa + teor. del reemplaz.}\\
&\equiv (C \vee (A \vee B)) \wedge (C \vee \neg A) && : \text{conmutativa + teor. del reemplaz.}\\
&\equiv C \vee ((A \vee B) \wedge \neg A) && : \text{distributiva}\\
&\equiv ((A \vee B) \wedge \neg A) \vee C && : \text{conmutativa}\\
&\equiv ((A \wedge \neg A) \vee (B \wedge \neg A)) \vee C && : \text{distributiva + teor. del reemplaz.}\\
&\equiv (B \wedge \neg A) \vee C && : \text{leyes de la contrad. + teor. del reemplaz.}
\end{aligned}
$$

1.3 Formas normales

Definición 1.12 (Literal) *Un* literal *es o una variable o la negación de una variable.*

Definición 1.13 (Forma Normal Conjuntiva (FNC)) *F esta en* FNC *sii F es una conjunción de disyunciones de literales. Es decir:*

$$F = \bigwedge_{i=1}^{n} \bigvee_{j=1}^{m} l_{ij}$$

Definición 1.14 (Forma Normal Disyuntiva (FND)) *F esta en* FND *sii F es una disyunción de conjunciones de literales. Es decir:*

$$F = \bigvee_{i=1}^{n} \bigwedge_{j=1}^{m} l_{ij}$$

Teorema 1.15 *Para toda fórmula F, existe otra equivalente en FNC, y otra en FND.*

Demostración: Por inducción sobre la profundidad de F.

- Caso base, profundidad 0. F es una variable. F esta en FNC y FND.
- Paso de inducción: Supongamos que toda fórmula de profundidad hasta k tiene otra equivalente en FNC y otra en FND. Sea F una fórmula arbitraria de profundidad $k+1$.
- Caso 1: $F = \neg G$. Por la hipotesis de inducción, existen G_1 y G_2 equivalentes a G, estando G_1 en FNC y G_2 en FND.

  Aplicando las leyes de deMorgan a $\neg G_1$ obtenemos una fórmula equivalente a F y en FND.

  Aplicando las leyes de deMorgan a $\neg G_2$ obtenemos una fórmula equivalente a F y en FNC.

- Caso 2: $F = G \vee H$. G y H tienen profundidad $< k + 1$.

  Por la hipótesis de inducción, G y H tienen fórmula equivalente G' y H' en FND. La FND de F es $G' \vee H'$.

  Sean G'' y H'' las FNC de G y H.

$$G'' = \bigwedge_{i=1}^{n} G_i'' \text{ y } H'' = \bigwedge_{j=1}^{m} H_j''$$

  donde G_i'' y H_j'' son disyunciones de literales.

$$F = \bigwedge_{i=1}^{n} G_i'' \vee \bigwedge_{j=1}^{m} H_j'' \equiv \bigwedge_{i=1}^{n} \bigwedge_{j=1}^{m} G_i'' \vee H_j''$$

■

Algoritmo para buscar FNC y FND Para buscar la FNC y la FND de una fórmula dada F, podemos utilizar el siguiente algoritmo:

1. Eliminar $\rightarrow$ y $\leftrightarrow$.

2. Usar las siguientes equivalencias para poner todas las negaciones al lado de las variables.

   - $\neg\neg G \equiv G$
   - $\neg(G \vee H) \equiv \neg G \vee \neg H$
   - $\neg(G \wedge H) \equiv \neg G \wedge \neg H$

3. Usar las leyes distributivas.

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

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

4. Hacer simplificaciones.

1.4 Resolución

Definición 1.16 (Cláusula) *Una* cláusula *es una disyunción de literales.*

Veamos como verificar $\models A$ o bien $A_1, \ldots, A_n \models A$. Sabemos que $A_1, \ldots, A_n \models A$ sii $\bigwedge_{i=1}^{n} A_i \wedge \neg A$ es una contradicción. Ponemos $\bigwedge_{i=1}^{n} A_i \wedge \neg A$ en FNC. Si sacamos las conjunciones tenemos un conjunto de cláusulas, a las que podemos aplicarles la siguiente regla:

$$\frac{C_1 \vee l \qquad C_2 \vee \bar{l}}{C_1 \vee C_2}$$

Aplicamos la regla a las cláusulas iniciales y a las que vamos obteniendo hasta obtener la cláusula vacia o $\Box$. Nota: la única manera para obtener $\Box$ (cláusula vacía) es haber obtenido previamente p y $\neg p$ para alguna variable p.

Lema 1.17 $C_1 \vee l, C_2 \vee \bar{l} \models C_1 \vee C_2$

Demostración: Sea I una interpretación arbitraria que satisface a $C_1 \vee l$ y a $C_2 \vee \bar{l}$

- Caso 1: si I no satisface a l, entonces I ha de satisfacer a algún literal de C_1 y por tanto a $C_1 \vee C_2$
- Caso 2: si I satisface a l, entonces I no satisface $\bar{l}$ y I ha de satisfacer a algún literal de C_2 y por tanto a $C_1 \vee C_2$

En ambos casos I satisface a $C_1 \vee C_2$ ■

Definición 1.18 *Sea F un conjunto de cláusulas (F en FNC):*

$Res(F) = \{C \mid C \text{ puede obtenerse resolviendo 2 cláusulas de } F\}$

$Res^0(F) = F$

$Res^{n+1}(F) = Res(Res^n(F))$

$Res^*(F) = \bigcup_n Res^n(F)$

Teorema 1.19 (Teorema de la Corrección) *Si $\Box \in Res^*(F)$, entonces F es insatisfactible.*

Demostración: Suponemos $\Box \in Res^*(F)$ y que F es satisfactible.

Existe una interpretación I tal que I satisface a todas las cláusulas de F.

- I satisface a $Res(F)$ (por el lema 1.17)
- I satisface a $Res^n(F)$ $\forall n$ (por el lema 1.17)
- I satisface a $Res^*(F)$

$\Box \in Res^*(F)$ y también l y $\bar{l}$ para alguna cláusula. Contradicción. F es insatisfactible. ■

Teorema 1.20 (Teorema de Completitud) *Si F es insatisfactible, entonces $\Box \in Res^*(F)$.*

Demostración: Por inducción sobre n, el número de variables de F. ($\forall n$ si F tiene n variables, entonces si F es insatisfactible, $\Box \in Res^*(F)$)

- Caso $n = 0$. F no tiene variables y es insatisfactible. Por tanto, $F = \{\Box\}$.
- Paso de inducción: Suponemos que la propiedad se cumple para fórmulas de hasta n variables. Sea F una fórmula de $n+1$ variables e insatisfactible. Sea $\{p_1, \ldots, p_{n+1}\}$ el conjunto de variables de F. Definimos ahora las siguientes fórmulas:

$$\begin{cases} F_0 \text{ será el resultado de asignar } 0 \text{ a } p_{n+1} \text{ en } F \\ F_1 \text{ será el resultado de asignar } 1 \text{ a } p_{n+1} \text{ en } F \end{cases}$$

Veamos primero que F_0 y F_1 son insatisfactibles. Supongamos que F_0 es satisfactible. Sea I la interpretación que la satisface. Entonces $I \cup \{p_{n+1} \to 0\}$ satisface a F. Contradicción, y por tanto F_0 es insatisfactible. Un argumento similar demuestra que F_1 es insatisfactible.

Dado que F_0 y F_1 son insatisfactibles, y tienen como máximo n variables, por la hipótesis de inducción, $\Box \in Res^*(F_0)$ y $\Box \in Res^*(F_1)$.

$\exists C_1, \ldots, C_n$ y $\exists C'_1, \ldots, C'_k$ tales que $C_n = \Box$ y $C'_k = \Box$, y C_i es, o bien de F_0 o se obtiene de dos anteriores por resolución (lo equivalente ocurre con las claúsulas C'_j). Si en las cláusulas de F_0 introducimos p_{n+1} donde se había eliminado, las secuencia $C_1, \ldots, C_n$ se convierte en otra que genera p_{n+1}. Y si en las cláusulas de F_1 introducimos $\neg p_{n+1}$ donde se había eliminado, las secuencia $C'_1, \ldots, C'_k$ se convierte en otra que genera $\neg p_{n+1}$. Tomando estas dos nuevas secuencias acabadas en p_{n+1} y $\neg p_{n+1}$ respectivamente, y resolviendo p_{n+1} con $\neg p_{n+1}$ se obtiene $\Box$. Por tanto $\Box \in Res^*(F)$.

■

1.5 Ejercicios

1. Dadas las siguientes fórmulas, poner todos los paréntesis implícitos y dar fu forma arbórea:

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

   (b) $p \to (q \vee r) \wedge p$

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

2. Eliminar el máximo número de paréntesis posible y poner en forma arbórea las siguientes fórmulas:

   (a) $(p \to (q \to (r \wedge s)))$

   (b) $((\neg(p \vee q)) \to ((\neg p) \vee (\neg q)))$

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

3. Sea $\Gamma = \{F_1, F_2, ..., F_k\}$ Demostrar que los 3 enunciados siguientes son equivalentes.

   (a) $\Gamma \models A$

   (b) $\bigwedge_{i=1}^{k} F_i \to A$ es una tautología

   (c) $\bigwedge_{i=1}^{k} F_i \wedge \neg A$ es una contradicción

4. Es el siguiente conjunto satisfactible?

$$M = \{P_1 \vee P_2, \neg P_2 \vee \neg P_3, P_3 \vee P_4, \neg P_4 \vee \neg P_5, ...\}$$

   Si es así, buscar una interpretación que lo satisfaga.

5. Demostrar o dar contraejemplo:

   (a) Si $F \to G$ y F son tautologías, entonces G es tautología

   (b) Si $F \to G$ y F son satisfactibles, entonces G es satisfactible

   (c) Si $F \to G$ es una tautología y F es satisfactible, entonces G es satisfactible

6. Buscar FNC y FND equivalente de

$$(\neg A \to B) \vee ((A \wedge \neg C) \leftrightarrow B)$$

7. La tabla de verdad de una fórmula nos da la FND. Por que? Tambin nos da la FNC?

8. Sea:

$$F = \{p \vee \neg q \vee r, q \vee r, \neg p \vee r, q \vee \neg r, \neg r\}$$

Determinar $Res^n(F)$ para $n = 0$, $n = 1$ y $n = 2$.

9. Demostrar que para todo conjunto finito de cláusulas F, existe $k \geq 0$ tal que

$$Res^k(F) = Res^{k+1}(F) = ... = Res^*(F)$$

10. Demostrar usando resolución:

$$p \vee q \vee r, \neg p \vee q, \neg q \vee r, p \vee \neg r \vdash p \wedge q \wedge r$$

11. Demostrar usando resolución:

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

12. Una clusula es positiva (negativa) si solo contiene literales positivos (negativos). Demostrar que una FNC que no contiene una clusula positiva (negativa) es satisfactible.