Pontificia Universidad Católica de Chile
Facultad de Ingeniería
Departamento de Ciencias de la computación
Curso: Matemáticas discretas
Ayudantes: Francisca Caprile, Catalina Ortega, Matías Fernández e Ignacio Vergara

Ayudantía 3

1 de septiembre de 2023

2º semestre 2023 - Profesores G. Diéguez - S. Bugedo - N. Alvarado- B. Barías

---

Resumen

- **Conceptos importantes de lógica proposicional:**
    - Tautología: Una fórmula es una tautología si su valor de verdad es siempre 1, para cualquier valuación.
    - Contradicción: Una fórmula es una contradicción si su valor de verdad es siempre 0, para cualquier valuación.
    - Forma normal conjuntiva (CNF): Una fórmula está en forma normal conjuntiva si es una conjunción de disyunciones de literales. Es decir, es de la forma $C_1 \wedge C_2 \wedge \ldots \wedge C_k$, donde cada C_i es una disyunción de literales, es decir, $C_i = (l_{i1} \vee \ldots \vee l_{iki})$.
    - Forma normal disyuntiva (DNF): Una fórmula está en forma normal disyuntiva si es una disyunción de conjunciones de literales. Es decir, es de la forma $B_1 \vee B_2 \vee \ldots \vee B_k$, donde cada B_i es una conjunción de literales, es decir, $B_i = (l_{i1} \wedge \ldots \wedge l_{iki})$.
    - Satisfacibilidad: Un conjunto de fórmulas Σ es satisfacible si existe una valuación σ tal que $\sigma(\Sigma) = 1$. En caso contrario, Σ es inconsistente.
- **Consecuencia lógica:**

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

  Notación: $\Sigma \models \psi$.

Ejercicio 1 — Consecuencia lógica

Sean α y β fórmulas proposicionales y $\Sigma = \{\phi_1, \ldots, \phi_n\}$ un conjunto de fórmulas proposicionales. Demuestre o dé un contraejemplo para las siguientes afirmaciones:

(a) Sea α y β dos fórmulas proposicionales sin variables proposicionales en común tal que $\alpha \models \beta$. Si α es satisfacible, entonces β es una tautología.

Solución:

Es verdadero y lo demostraremos.

Sean $\alpha(x, y)$, $\beta(x, y)$ fórmulas proposicionales tales que α y β solo mencionan los valores de x y y respectivamente. Asumimos que α es satisfacible y que $\alpha \models \beta$, queremos demostrar que β es tautología.

Sea u, v una valuación cualquiera de x, y. Como α es satisfacible, existe una valuación u^*, v^* tal que $\alpha(u^*, v^*) = 1$. Dado que α solo menciona a x, entonces $\alpha(u^*, v) = 1$.

Luego, como $\alpha \models \beta$, tenemos que $\beta(u^*, v)$. Siendo v una valuación cualquiera de y. Luego, como β solo depende de y, tenemos que:

$$\beta(u, v) \equiv \beta(u^*, v) = 1$$

Finalmente, llegamos a que para cualquier valuación u, v se tiene $\beta(u, v) = 1$. Concluimos que β es tautología.

(b) Si $\Sigma \models \alpha$, entonces $\neg\alpha \models \neg\phi_i$ para cualquier fórmula ϕ_i en Σ

Solución:
Se desea buscar el caso borde que hace verdadero a $\Sigma \models \alpha$ pero falso a $\neg\alpha \models \neg\phi_i$.

Para que la segunda expresión sea falsa, $\neg\alpha$ debe ser verdadero y $\neg\phi_i$ tiene que ser falso. Dicho lo anterior, sean $\alpha = q$ y $\Sigma = \{p, p \Rightarrow q\}$, donde $\phi_1 = p$ y $\phi_2 = p \Rightarrow q$. Se sabe de clases que el caso anterior es una consecuencia lógica (Modus ponens). Notemos que $\neg\alpha = \neg q$, $\neg\phi_1 = \neg p$ y $\neg\phi_2 = \neg(p \Rightarrow q)$.

Ya sabemos que $\Sigma \models \alpha$ es verdadero, así que falta por encontrar un caso donde $\neg\alpha \models \neg\phi_i$ es falso. Analizamos la tabla de verdad del caso anterior:

p	q	$\neg\alpha = \neg q$	$\neg\phi_1 = \neg p$
1	1	0	0
1	0	1	0
0	1	0	1
0	0	1	1

Notemos que la valuación $(1, 0)$ hace que $\neg\alpha = 1$ y que $\neg\phi_1 = 0$. Por lo tanto, $\neg\alpha \models \neg\phi_i$ es falso. De esa forma, se concluye que la expresión es falsa.

Ejercicio 2 — Equivalencia lógica, inconsistencia y resolución

(a) Sean α y β fórmulas proposicionales y $\Sigma = \{\phi_1, \ldots, \phi_n\}$ un conjunto de fórmulas proposicionales. Demuestre o dé un contraejemplo para la siguiente afirmación

Si $\alpha \not\equiv \beta$, entonces $\alpha \equiv \neg\beta$

Solución:

Se desea buscar el caso borde que hace verdadero a $\alpha \not\equiv \beta$ pero falso a $\alpha \equiv \neg\beta$: Sean $\alpha = p \wedge q$ y $\beta = p \vee q$ que cumplen $\alpha \not\equiv \beta$. Lo anterior se puede verificar con la siguiente tabla de verdad:

p	q	$\alpha = p \wedge q$	$\beta = p \vee q$	$\neg\beta$
1	1	1	1	0
1	0	0	1	0
0	1	0	1	0
0	0	0	0	1

Esto ocurre porque α no es igual a β en todas las valuaciones. Basta tomar las valuaciones $(1, 0)$ y $(0, 1)$ de p y q para corroborarlo. Además, por la misma tabla de verdad se puede ver que $\alpha \equiv \neg\beta$ no se cumple en las valuaciones $(1, 1)$ y $(0, 0)$. Por lo tanto, la expresión es falsa, pues se ha encontrado un contraejemplo con los α y β definidos anteriormente.

(b) Demuestre que $\Sigma = \{p \Leftrightarrow q, p \veebar q\}$, con '$\veebar$' la disyunción exclusiva, es inconsistente.

Observación: La disyunción exclusiva es similar a la disyunción, la única diferencia es que cuando los dos valores (p y q) son verdad, esta es falsa.

Tabla de verdad de la disyunción exclusiva:

p	q	$p \veebar q$
0	0	0
0	1	1
1	0	1
1	1	0

Solución:
Podemos demostrar con resolución que Σ es inconsistente. Para ello podemos reescribir $\{p \Leftrightarrow q\}$ y $\{p \veebar q\}$ como,

- $p \Leftrightarrow q \equiv (p \to q) \wedge (q \to p) \equiv (\neg p \vee q) \wedge (\neg q \vee p) \equiv \{\neg p \vee q, \neg q \vee p\}$
- $p \veebar q \equiv p \vee q \wedge \neg(p \wedge q) \equiv (p \vee q) \wedge (\neg p \vee \neg q) \equiv \{p \vee q, \neg p \vee \neg q\}$

Luego, demostrar que $\Sigma' := \{\neg p \vee q, \neg q \vee p, p \vee q, \neg p \vee \neg q\}$ es inconsistente es equivalente a demostrar que Σ es inconsitente (pues $\Sigma \equiv \Sigma'$). Por resolución,

(1)	$\neg q \vee p$	$\in \Sigma$
(2)	$p \vee q$	$\in \Sigma$
(3)	$p \vee p$	resolución de (1) y (2)
(4)	p	factorización de (3)
(5)	$\neg p \vee q$	$\in \Sigma$
(6)	$\neg p \vee \neg q$	$\in \Sigma$
(7)	$\neg p \vee \neg p$	resolución de (5) y (6)
(8)	$\neg p$	factorización de (7)
(9)	$\square$	resolución de (4) y (8)

Obteniendo así que Σ' es inconsistente por lo cual Σ es inconsistente.

Ejercicio 3 — Modelamiento

Tras la pandemia, la ciudad de Venecia tendrá un recorte de presupuesto y, en consecuencia, se tomó la decisión de disminuir la cantidad de generadores eléctricos. La ciudad se puede modelar como un conjunto de islotes conectados por puentes. El gobierno de la ciudad quiere abastecer a todos los puentes de luz eléctrica. Para ello, cada puente debe incidir en algún islote equipado con un generador. Tenga en consideración que la ciudad consta de 10 islotes y usted cuenta con 5 generadores para distribuir entre las islas.

En concreto, usted debe construir una fórmula $\varphi \in L(P)$ tal que:

$$\varphi \text{ es satisfacible}$$
$$\Leftrightarrow$$
$$\text{es posible asignar los generadores a los islotes y tener luz eléctrica en todos los puentes.}$$

Solución:

Tenemos 10 islas denotadas del 1 al 10. Definamos las siguientes variables proposicionales:

$$q_i := \ 1 \text{ si la isla } i \in \{1, \ldots, 10\} \text{ tiene asignado un generador; 0, e.o.c.}$$

Además, como tenemos el plano entregado por el gobierno, tenemos conocimiento de las islas y sus conexiones. Consideraremos un conjunto $\mathcal{M}$ que contiene tuplas (i, j) que indican que existe un puente entre la isla i y j.

Ahora consideremos las siguientes fórmulas:

1. Todo puente debe estar conectado a una isla con generador.

$$\varphi_1 = \bigwedge_{(i,j)\in\mathcal{M}} (q_i \vee q_j)$$

2. Hay a lo más 5 generadores asignados. Definamos el conjunto

$$D = \left\{(l_1, l_2, l_3, l_4, l_5) \in \{1, \ldots, 10\}^5 : l_1 < l_2 < \cdots < l_5\right\}$$

   Entonces:

$$\varphi_2 = \bigwedge_{(l_1,\ldots,l_5)\in D} \left[\left(\bigwedge_{i=1}^{5} q_{l_i}\right) \rightarrow \left(\bigwedge_{l\in\{1,\ldots,10\}\setminus\{l_1,\ldots,l_5\}} \neg q_l\right)\right]$$

Finalmente, obtenemos la fórmula deseada:

$$\varphi = \varphi_1 \wedge \varphi_2$$