Matemáticas Discretas

Lógica proposicional

Nicolás Alvarado
nfalvarado@mat.uc.cl

Bernardo Barías
bjbarias@uc.cl

Sebastián Bugedo
bugedo@uc.cl

Gabriel Diéguez
gsdieguez@ing.puc.cl

Departamento de Ciencia de la Computación
Escuela de Ingeniería
Pontificia Universidad Católica de Chile

28 de agosto de 2023

Objetivos

1. Formular enunciados formales en notación matemática usando lógica, conjuntos, relaciones, funciones, cardinalidad, y otras herramientas, desarrollando definiciones y teoremas al respecto, así como demostrar o refutar estos enunciados, usando variadas técnicas.
2. Aplicar inducción como técnica para demostración de propiedades en conjuntos discretos y como técnica de definición formal de objetos discretos.
3. Modelar formalmente un problema usando lógica, conjuntos, relaciones, y las propiedades necesarias, y demostrar propiedades al respecto de su modelo.

Contenidos

¿Qué es lógica?

Definición

Lógica es el uso y estudio del **razonamiento válido**.

Wikipedia

Introducción

- La lógica (en su sentido más amplio) estudia la **inferencia**.
    - Cómo obtener conclusiones a partir de premisas.
- Originalmente, hacía esto en el lenguaje natural.

Ejemplo

¿Es el siguiente razonamiento válido?

$$\begin{array}{l} \text{Todos los hombres son mortales.} \\ \text{Sócrates es hombre.} \\ \hline \text{Por lo tanto, Sócrates es mortal.} \end{array}$$

La lógica debería poder usarse para demostrar que sí.

Introducción

¿Qué pasa si usamos el lenguaje natural para definir cosas? Por ejemplo:

- Podemos representar los números naturales usando oraciones del lenguaje natural: “dos mil dieciocho”, “el segundo número”...
- El diccionario de la RAE tiene una cantidad finita de palabras.
- El número de oraciones con a lo más 50 palabras también es finito.
- Por lo tanto, deben existir números naturales que no pueden ser representados con oraciones de a lo más 50 palabras.
- Por PBO, debe existir un menor número que cumpla lo anterior.

Introducción

Sea n el siguiente número natural:

El primer número natural que no puede ser definido por una oración con a lo más cincuenta palabras del diccionario de la Real Academia Española.

La oración anterior define a n con 25 palabras. **¡Contradicción!**

¿Qué pasó?

¿Por qué necesitamos la lógica?

- Necesitamos un lenguaje con una **sintaxis** precisa y una **semántica** bien definida.
  - **Qué** objetos pertenecen al lenguaje y qué **significan**.
- Queremos usar este lenguaje en matemáticas para hacer cosas que hasta ahora hemos hecho de manera intuitiva.
  - Definición de objetos matemáticos (conjuntos, números naturales, números reales. . .).
  - Definición de teorías matemáticas (teoría de conjuntos, teoría de los naturales. . .).
  - Definición del concepto de demostración.
- Pero también queremos usar este lenguaje en computación.

¿Por qué necesitamos la lógica en computación?

- Bases de datos: lenguajes de consulta, lenguajes para restricciones.
- Inteligencia artificial: representación de conocimiento, razonamiento.
- Ingeniería de software: especificación de sistemas, verificación de propiedades.
- Teoría de la computación: complejidad descriptiva, algoritmos de aproximación.
- Criptografía: verificación de protocolos.
- Procesamiento de lenguaje natural.
- ...

¿Cuántas lógicas existen?

- Lógica proposicional
- Lógica de predicados
- Lógica de primer orden
- Lógica de segundo orden
- Lógicas descriptivas
- Lógicas temporales
- Lógica intuicionista
- Lógica trivalente
- etc ...

¿Qué lógicas veremos en el curso?

- **Lógica proposicional**
- **Lógica de predicados**
- Lógica de primer orden
- Lógica de segundo orden
- Lógica de segundo orden monádica
- Lógica temporal lineal
- Lógica intuicionista
- Lógica trivalente
- etc ...

Más en..

- IIC2213 - Lógica para ciencias de la computación
- IIC3263 - Teoría de modelos finitos

Sintaxis de la Lógica proposicional

Usaremos los siguientes elementos:

- Variables proposicionales: p, q, r, $\ldots$
- Conectivos lógicos: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
- Símbolos de puntuación: (,)

Las variables proposicionales representan proposiciones **completas** e **indivisibles**, que pueden ser **verdaderas** o **falsas**. En general llamaremos P al conjunto de ellas.

Ejemplo

$P = \{\text{Juan\_cursa\_IIC1253}, \text{Juan\_aprobo\_IIC1103}\}$

Sintaxis de la Lógica proposicional

Los conectivos lógicos se usan para construir expresiones más complejas que también pueden ser verdaderas o falsas.

Ejemplo

$$\text{Juan\_cursa\_IIC1253} \rightarrow \text{Juan\_aprobo\_IIC1103}$$
$$(\neg \text{Juan\_cursa\_IIC1253}) \rightarrow \text{Juan\_aprobo\_IIC1103}$$

Los símbolos de puntuación se usan para evitar ambigüedades.

Sintaxis de la Lógica proposicional

Definición

Sea P un conjunto de variables proposicionales. El conjunto de todas las **fórmulas** de lógica proposicional sobre P, denotado por $L(P)$, es el menor conjunto que cumple las siguientes reglas:

1. Si $p \in P$, entonces $p \in L(P)$.
2. Si $\varphi \in L(P)$, entonces $(\neg\varphi) \in L(P)$.
3. Si $\varphi, \psi \in L(P)$, entonces $(\varphi \wedge \psi) \in L(P)$, $(\varphi \vee \psi) \in L(P)$, $(\varphi \rightarrow \psi) \in L(P)$ y $(\varphi \leftrightarrow \psi) \in L(P)$.

¡Podemos usar Inducción estructural!

Sintaxis de la Lógica proposicional

Ejercicios

1. Defina la función $largo(\varphi)$ como el largo de una fórmula en lógica proposicional (variables, conectivos y paréntesis).
2. Defina la función $var(\varphi)$ como el número de ocurrencias de variables proposicionales en φ.
3. Demuestre que si φ no contiene el símbolo $\neg$, se cumple que $largo(\varphi) \leq 4 \cdot var(\varphi)^2$.
   - ¿Qué pasa si contiene el símbolo $\neg$?

Semántica de la Lógica proposicional

¿Cuándo una fórmula es verdadera?

- Depende del "mundo" en que la estemos interpretando.
- Un "mundo" particular le dará una interpretación a cada variable proposicional.
    - Un valor verdadero o falso.
- Con estos valores y los conectivos utilizados, podremos determinar el valor de verdad de la fórmula.

Definición

Una **valuación** o **asignación de verdad** para las variables proposicionales en un conjunto P es una función $\sigma : P \to \{0, 1\}$.

Convención: $0 =$ falso, $1 =$ verdadero.

Semántica de la Lógica proposicional

Dados un conjunto P y una asignación de verdad σ para P, necesitamos extender la noción de valuación a las fórmulas en $L(P)$. ¿Cómo lo hacemos?

Definición

La función $\hat{\sigma} : L(P) \to \{0,1\}$ se define como:

1. Si $p \in P$, entonces $\hat{\sigma}(p) = \sigma(p)$.
2. $$\hat{\sigma}((\neg\varphi)) = \begin{cases} 1 & \text{si } \hat{\sigma}(\varphi) = 0 \\ 0 & \text{si } \hat{\sigma}(\varphi) = 1 \end{cases}$$
3. $$\hat{\sigma}((\varphi \wedge \psi)) = \begin{cases} 1 & \text{si } \hat{\sigma}(\varphi) = 1 \text{ y } \hat{\sigma}(\psi) = 1 \\ 0 & \text{si } \hat{\sigma}(\varphi) = 0 \text{ o } \hat{\sigma}(\psi) = 0 \end{cases}$$
   $$\hat{\sigma}((\varphi \vee \psi)) = \begin{cases} 1 & \text{si } \hat{\sigma}(\varphi) = 1 \text{ o } \hat{\sigma}(\psi) = 1 \\ 0 & \text{si } \hat{\sigma}(\varphi) = 0 \text{ y } \hat{\sigma}(\psi) = 0 \end{cases}$$

Semántica de la Lógica proposicional

Definición

La función $\hat{\sigma} : L(P) \to \{0,1\}$ se define como:

1. Si $p \in P$, entonces $\hat{\sigma}(p) = \sigma(p)$.
2. $\hat{\sigma}((\neg\varphi)) = \begin{cases} 1 & \text{si } \hat{\sigma}(\varphi) = 0 \\ 0 & \text{si } \hat{\sigma}(\varphi) = 1 \end{cases}$
3. $\hat{\sigma}((\varphi \wedge \psi)) = \begin{cases} 1 & \text{si } \hat{\sigma}(\varphi) = 1 \text{ y } \hat{\sigma}(\psi) = 1 \\ 0 & \text{si } \hat{\sigma}(\varphi) = 0 \text{ o } \hat{\sigma}(\psi) = 0 \end{cases}$

   $\hat{\sigma}((\varphi \vee \psi)) = \begin{cases} 1 & \text{si } \hat{\sigma}(\varphi) = 1 \text{ o } \hat{\sigma}(\psi) = 1 \\ 0 & \text{si } \hat{\sigma}(\varphi) = 0 \text{ y } \hat{\sigma}(\psi) = 0 \end{cases}$

   $\hat{\sigma}((\varphi \to \psi)) = \begin{cases} 1 & \text{si } \hat{\sigma}(\varphi) = 0 \text{ o } \hat{\sigma}(\psi) = 1 \\ 0 & \text{si } \hat{\sigma}(\varphi) = 1 \text{ y } \hat{\sigma}(\psi) = 0 \end{cases}$

   $\hat{\sigma}((\varphi \leftrightarrow \psi)) = \begin{cases} 1 & \text{si } \hat{\sigma}(\varphi) = \hat{\sigma}(\psi) \\ 0 & \text{si } \hat{\sigma}(\varphi) \neq \hat{\sigma}(\psi) \end{cases}$

Semántica de la Lógica proposicional

- $\hat{\sigma}(\varphi)$ es la **evaluación** de φ dada la asignación σ y representa su valor de verdad.
- Por simplicidad, usaremos σ en lugar de $\hat{\sigma}$.

Ejemplo

Si tenemos una valuación σ tal que

$$\sigma(\text{Juan\_cursa\_IIC1253}) = 1$$
$$\sigma(\text{Juan\_aprobo\_IIC1103}) = 0$$

entonces

$$\sigma((\text{Juan\_cursa\_IIC1253} \rightarrow \text{Juan\_aprobo\_IIC1103})) = 0$$

Semántica de la Lógica proposicional

Definición

Dos fórmulas $\varphi, \psi \in L(P)$ son **lógicamente equivalentes** (denotado como $\varphi \equiv \psi$) si para toda valuación σ se tiene que $\sigma(\varphi) = \sigma(\psi)$.

Ejercicio

Demuestre que $(p \to q) \equiv ((\neg p) \vee q)$.

Tablas de verdad

Las fórmulas se pueden representar y analizar en una tabla de verdad.

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \to q$	$p \leftrightarrow q$
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

Ejercicio

Si P tiene n variables proposicionales, ¿cuántas tablas de verdad distintas hay para $L(P)$?

Tablas de verdad

Note que $\varphi \equiv \psi$ si y sólo si sus respectivas columnas en una tabla de verdad son iguales.

- Podemos usar tablas de verdad para comparar fórmulas, como en el ejercicio que ya hicimos. Otro ejemplo:

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

Tablas de verdad

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

Concluimos que $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$.

- Es decir, el conectivo $\wedge$ es asociativo.
- Podemos omitir paréntesis cuando tengamos una cadena de ellos.
    - Escribimos simplemente $p \wedge q \wedge r$.

Leyes de equivalencia

Doble negación

$\neg(\neg\varphi) \equiv \varphi$

Leyes de De Morgan

$\neg(\varphi \wedge \psi) \equiv (\neg\varphi) \vee (\neg\psi)$
$\neg(\varphi \vee \psi) \equiv (\neg\varphi) \wedge (\neg\psi)$

Conmutatividad

$\varphi \wedge \psi \equiv \psi \wedge \varphi$
$\varphi \vee \psi \equiv \psi \vee \varphi$

Leyes de equivalencia

Asociatividad

$$\varphi \wedge (\psi \wedge \theta) \equiv (\varphi \wedge \psi) \wedge \theta$$
$$\varphi \vee (\psi \vee \theta) \equiv (\varphi \vee \psi) \vee \theta$$

Distributividad

$$\varphi \wedge (\psi \vee \theta) \equiv (\varphi \wedge \psi) \vee (\varphi \wedge \theta)$$
$$\varphi \vee (\psi \wedge \theta) \equiv (\varphi \vee \psi) \wedge (\varphi \vee \theta)$$

Idempotencia

$$\varphi \wedge \varphi \equiv \varphi$$
$$\varphi \vee \varphi \equiv \varphi$$

Leyes de equivalencia

Absorción

$\varphi \wedge (\varphi \vee \psi) \equiv \varphi$
$\varphi \vee (\varphi \wedge \psi) \equiv \varphi$

Implicancia

$\varphi \to \psi \equiv (\neg\varphi) \vee \psi$

Doble implicancia

$\varphi \leftrightarrow \psi \equiv (\varphi \to \psi) \wedge (\psi \to \varphi)$

Ejercicios

1. Demuestre las leyes enunciadas.
2. ¿Es $\to$ asociativo?

Notación

- Omitiremos paréntesis externos y otros que no produzcan ambigüedad.
  - Ejemplo: escribimos $p \wedge q \wedge r$.
- Usaremos los siguientes operadores:

$$\bigvee_{i=1}^{n} \varphi_i = \varphi_1 \vee \varphi_2 \vee \ldots \vee \varphi_n$$

$$\bigwedge_{i=1}^{n} \varphi_i = \varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n$$

Conectivos y fórmulas

Las tablas de verdad también nos sirven para definir conectivos.

Ejemplo

El conectivo XOR se define con la siguiente tabla de verdad:

p	q	XOR
0	0	0
0	1	1
1	0	1
1	1	0

Ejercicio

- ¿Cuántos conectivos binarios posibles tenemos?

Conectivos y fórmulas

Tenemos una fórmula $\varphi \in L(P)$, con $P = \{p, q, r\}$, y lo único que conocemos de ella es que cumple la siguiente tabla de verdad:

p	q	r	φ
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

- ¿Podemos construir una fórmula que sea lógicamente equivalente a φ?
- Dada una tabla con n variables proposicionales, ¿podemos generalizar el argumento anterior?

Conectivos y fórmulas

Teorema

Podemos representar cualquier tabla de verdad con una fórmula.

Corolario

Podemos representar cualquier tabla de verdad con una fórmula que sólo usa $\neg$, $\wedge$ y $\vee$.

Conectivos funcionalmente completos

Definición

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

- Ya demostramos que $\{\neg, \wedge, \vee\}$ es funcionalmente completo.

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?

Notación

- Omitiremos paréntesis externos y otros que no produzcan ambigüedad.
  - Ejemplo: escribimos $p \wedge q \wedge r$.
- Usaremos los siguientes operadores:

$$\bigvee_{i=1}^{n} \varphi_i = \varphi_1 \vee \varphi_2 \vee \ldots \vee \varphi_n$$

$$\bigwedge_{i=1}^{n} \varphi_i = \varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n$$

- De ahora en adelante, el conectivo $\neg$ tendrá **precedencia** sobre los conectivos binarios.
  - Podemos eliminar paréntesis cuando hay términos negados.
  - Ejemplo: escribimos $((\neg p) \vee q) \wedge (\neg r)$ como $(\neg p \vee q) \wedge \neg r$.

Satisfacibilidad

Definición

Una fórmula φ es **satisfacible** si existe una valuación σ 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:

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

Satisfacibilidad

Dada una fórmula φ, queremos verificar si es satisfacible.

- Este es un problema fundamental tanto en computación como en ingeniería.
- Muchos problemas pueden ser resueltos usando este problema, lo que nos muestra el **poder expresivo** de la lógica proposicional.
  - Problemas en grafos (coloración, clique).
  - Problemas de optimización (vendedor viajero, mochila, programación entera).
- ¿Es difícil este problema? ¿Cómo lo resolvemos?

Una aplicación muy importante es que podemos **modelar** otros problemas mediante satisfacción lógica.

- Veamos un par de ejemplos.

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 $\varphi \in L(P)$ tal que M es 3-coloreable si y sólo si φ es satisfacible.

Ejercicio

Un **cuadrado mágico** es una matriz de $n \times n$ que sólo contiene números entre 1 y n en que la suma de cada fila y columna es la misma. Por ejemplo, la siguiente matriz de 3×3 es un cuadrado mágico:

$$\begin{bmatrix} 2 & 2 & 1 \\ 2 & 2 & 1 \\ 1 & 1 & 3 \end{bmatrix}$$

Dada una matriz C de $n \times n$ con algunas posiciones en 0, diremos que es **completable** si existe una manera de reemplazar los 0 por números entre 1 y n de manera que la matriz resultante sea un cuadrado mágico.

Satisfacibilidad

Ejercicio

Por ejemplo, la siguiente matriz es completable, pues pueden llenarse las casillas y obtener la matriz anterior:

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & 2 & 1 \\ 2 & 2 & 1 \\ 1 & 1 & 3 \end{bmatrix}$$

En cambio, la siguiente matriz no es completable:

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 3 & 0 & 0 \end{bmatrix}$$

Satisfacibilidad

Ejercicio

Dada una matriz C de 3×3, construya una fórmula $\varphi \in L(P)$ tal que C es completable si y sólo si φ es satisfacible.

Observación

Cada valuación σ que satisface a φ representa una forma de completar C.

Tautologías y contradicciones

Definición

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

Ejemplo

$p \wedge \neg p$

Definición

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

Ejemplo

$p \vee \neg p$

$p \leftrightarrow p$

Tautologías y contradicciones

Definición

Una fórmula φ es una **tautología** si para toda valuación σ 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.

Formas normales

Definición

Un literal es una variable proposicional o la negación de una variable proposicional. Por ejemplo, p y $\neg r$ son literales.

Formas normales

Definición

Decimos que una fórmula φ está en **forma normal disyuntiva (DNF)** si es una disyunción de conjunciones de literales; o sea, si es de la forma

$$B_1 \vee B_2 \vee \ldots \vee B_k$$

donde cada B_i es una conjunción de literales, $B_i = (l_{i1} \wedge \ldots \wedge l_{ik_i})$

Ejemplo

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

Teorema

Toda fórmula es equivalente a una fórmula en DNF.

Ya demostramos este teorema, ¿cierto?

Formas normales

Definición

Decimos que una fórmula φ está en **forma normal conjuntiva (CNF)** si es una conjunción de disyunciones de literales; o sea, si es de la forma

$$C_1 \wedge C_2 \wedge \ldots \wedge C_k$$

donde cada C_i es una disyunción de literales, $C_i = (l_{i1} \vee \ldots \vee l_{ik_i})$

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

Ejemplo

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

Formas normales

Teorema

Toda fórmula es equivalente a una fórmula en CNF.

Ejercicio

Demuestre el teorema.

Consecuencia lógica

Dado un conjunto de fórmulas Σ en $L(P)$, diremos que una valuación σ satisface Σ si para toda fórmula $\varphi \in \Sigma$ se tiene que $\sigma(\varphi) = 1$.

Notación: $\sigma(\Sigma) = 1$.

¿Cuándo decimos que una fórmula ψ se deduce desde Σ?

Definición

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

Notación: $\Sigma \models \psi$.

Consecuencia lógica

Ejemplos

- $\{p, p \to q\} \models q$ (Modus ponens)
- $\{\neg q, p \to q\} \models \neg p$ (Modus tollens)
- $\{p \vee q \vee r, p \to s, q \to s, r \to s\} \models s$ (Demostración por partes)

Consecuencia lógica

Definición

Un conjunto de fórmulas Σ es **satisfacible** si existe una valuación σ tal que $\sigma(\Sigma) = 1$. En caso contrario, Σ es **inconsistente**.

Teorema

$\Sigma \models \varphi$ si y sólo si $\Sigma \cup \{\neg\varphi\}$ es inconsistente.

Ejercicio

Demuestre el teorema.

Consecuencia lógica

Entonces, para chequear si $\Sigma \models \varphi$, basta ver si $\Sigma \cup \{\neg\varphi\}$ es inconsistente.

¿Cómo hacemos esto?

- El método basado en tablas de verdad es demasiado lento.
- Necesitamos un método alternativo que no construya tablas de verdad.

Cláusula vacía

Sea $\square$ una contradicción cualquiera.

- Llamaremos a $\square$ la **cláusula vacía**. ¿Por qué?

Teorema

Un conjunto de fórmulas Σ es inconsistente si y sólo si $\Sigma \models \square$.

Ejercicio

Demuestre el teorema.

Equivalencia de conjuntos

Extendemos la idea de equivalencia lógica a conjuntos de fórmulas.

Definición

Dos conjuntos de fórmulas Σ_1 y Σ_2 son **lógicamente equivalentes** ($\Sigma_1 \equiv \Sigma_2$) si para toda valuación σ se tiene que $\sigma(\Sigma_1) = \sigma(\Sigma_2)$. También diremos que Σ es lógicamente equivalente a una fórmula φ si $\Sigma \equiv \{\varphi\}$.

Observación

Todo conjunto de fórmulas Σ es equivalente a la conjunción de sus fórmulas.

$$\Sigma \equiv \bigwedge_{\varphi \in \Sigma} \varphi$$

Equivalencia de conjuntos

Teorema

Todo conjunto de fórmulas Σ es equivalente a un conjunto de cláusulas.

Ejercicio

Demuestre el teorema.

Equivalencia de conjuntos

Entonces, para convertir cualquier conjunto de fórmulas a un conjunto de cláusulas, tomamos la conjunción de sus fórmulas y la llevamos a CNF, y luego obtenemos el conjunto de sus cláusulas.

Ejemplo

$\{p, q \to (p \to r), \neg(q \to r)\} \equiv \{p, \neg q \vee \neg p \vee r, q, \neg r\}$

Conclusión

Para resolver el problema de consecuencia lógica (es decir, determinar si $\Sigma \models \varphi$) tenemos que determinar si un conjunto de cláusulas Σ' construido desde $\Sigma \cup \{\neg\varphi\}$ es tal que $\Sigma' \models \Box$.

Resolución proposicional

- El **método de resolución** nos permite determinar si un conjunto de cláusulas Σ es tal que $\Sigma \models \square$ (y por lo tanto, determinar si Σ es inconsistente).
- Usa un sistema de reglas **sintácticas** muy simples que nos permitirán demostrar "mecánicamente".

Notación

Sea l un literal.

1. Si $l = p$, entonces $\bar{l} = \neg p$.
2. Si $l = \neg p$, entonces $\bar{l} = p$.

Resolución proposicional

Regla de resolución

$$\frac{\begin{array}{c} C_1 \vee l \vee C_2 \\ C_3 \vee \bar{l} \vee C_4 \end{array}}{C_1 \vee C_2 \vee C_3 \vee C_4}$$

con C_1, C_2, C_3, C_4 cláusulas y l un literal.

Semánticamente, la regla es **correcta**:

$$\{C_1 \vee l \vee C_2, C_3 \vee \bar{l} \vee C_4\} \models C_1 \vee C_2 \vee C_3 \vee C_4$$

Resolución proposicional

Ejemplo

$$\begin{array}{c} p \vee q \\ \neg q \vee r \vee s \\ \hline p \vee r \vee s \end{array}$$

Concluimos que $\{p \vee q, \neg q \vee r \vee s\} \models p \vee r \vee s$.

Algunos casos particulares:

$$\begin{array}{c} C_1 \vee l \vee C_2 \\ \bar{l} \\ \hline C_1 \vee C_2 \end{array} \qquad \begin{array}{c} l \\ \bar{l} \\ \hline \square \end{array}$$

Resolución proposicional

Regla de factorización

$$\frac{C_1 \vee l \vee C_2 \vee l \vee C_3}{C_1 \vee l \vee C_2 \vee C_3}$$

con C_1, C_2, C_3 cláusulas y l un literal.

Resolución proposicional

Definición

Dado un conjunto Σ de cláusulas, una **demostración por resolución** de que Σ es inconsistente es una secuencia de cláusulas $C_1, \ldots, C_n$ tal que $C_n = \square$ y para cada $i = 1 \ldots n$ se tiene que:

- $C_i \in \Sigma$, o
- C_i se obtiene de dos cláusulas anteriores en la secuencia usando la regla de resolución, o
- C_i se obtiene de una cláusula anterior en la secuencia usando la regla de factorización.

Si existe tal demostración, escribimos $\Sigma \vdash \square$.

Resolución proposicional

Ejemplo

$$\Sigma = \{p \vee q \vee r, \neg p \vee s, \neg q \vee s, \neg r \vee s, \neg s\}$$

(1)	$p \vee q \vee r$	$\in \Sigma$
(2)	$\neg p \vee s$	$\in \Sigma$
(3)	$s \vee q \vee r$	resolución de $(1), (2)$
(4)	$\neg q \vee s$	$\in \Sigma$
(5)	$s \vee s \vee r$	resolución de $(3), (4)$
(6)	$s \vee r$	factorización de (5)
(7)	$\neg r \vee s$	$\in \Sigma$
(8)	$s \vee s$	resolución de $(6), (7)$
(9)	s	factorización de (8)
(10)	$\neg s$	$\in \Sigma$
(11)	$\square$	resolución de $(9), (10)$

Resolución proposicional

Teorema

Dado un conjunto de cláusulas Σ, se tiene que:

- **Correctitud:** Si $\Sigma \vdash \square$ entonces $\Sigma \models \square$.
- **Completitud:** Si $\Sigma \models \square$ entonces $\Sigma \vdash \square$.

Corolario

Si Σ es un conjunto de cláusulas, entonces $\Sigma \models \square$ si y sólo si $\Sigma \vdash \square$.

Corolario (forma alternativa)

Un conjunto de cláusulas Σ es inconsistente si y sólo si existe una demostración por resolución de que es inconsistente.

¡Resolución resuelve nuestro problema!

Resolución proposicional

¡Resolución resuelve nuestro problema! (de consecuencia lógica)

Corolario

Dados un conjunto de fórmulas Σ y una fórmula φ cualesquiera,

$$\Sigma \models \varphi \text{ si y sólo si } \Sigma' \vdash \square$$

donde Σ' es un conjunto de cláusulas tal que $\Sigma \cup \{\neg\varphi\} \equiv \Sigma'$.

Ocupando todo lo que vimos, tenemos un procedimiento para determinar si una fórmula es consecuencia lógica de un conjunto, como veremos en el siguiente ejemplo.

Ejemplo

Demuestre que $\{p, q \to (p \to r)\} \models q \to r$.

Resumen

- La lógica es el área de la matemática que estudia la inferencia.
- La sintaxis y la semántica de la lógica proposicional nos permiten definir y asignar valores de verdad a fórmulas proposicionales.
- La lógica es una herramienta fundamental tanto para la modelación de problemas en ingeniería como para formalizar el concepto de demostración.
- El método de resolución es un sistema de reglas que nos permiten demostrar de manera sistemática.

Matemáticas Discretas

Lógica proposicional

Nicolás Alvarado
nfalvarado@mat.uc.cl

Bernardo Barías
bjbarias@uc.cl

Sebastián Bugedo
bugedo@uc.cl

Gabriel Diéguez
gsdieguez@ing.puc.cl

Departamento de Ciencia de la Computación
Escuela de Ingeniería
Pontificia Universidad Católica de Chile

28 de agosto de 2023