Matemática Discreta

Clase 2: Pruebas por deducción natural en Lean

Federico Olmedo

Departamento de Ciencias de la Computación
Universidad de Chile

Contenido clase de hoy

1. Deducción natural

2. El asistente de pruebas Lean

¿Qué estudia la lógica?

El objetivo de la lógica es responder cuándo un argumento es válido; se la conoce como la ciencia del razonamiento.

Si llueve, baja la temperatura
No llovió

Por lo tanto, no bajó la temperatura

Forma del argumento:

$$\begin{array}{l} p \to q \\ \neg p \\ \hline \neg q \end{array}$$

¿Argumento válido?: No

Sintaxis de la lógica proposicional

Definición (sintaxis de la lógica proposicional)

Dado un conjunto $P = \{p, q, r, \ldots\}$ de proposiciones, el conjunto de fórmulas bien formadas sobre P, notado $\mathcal{L}(P)$, se define inductivamente de la siguiente manera:

Caso base: Si $p \in P$, entonces $p \in \mathcal{L}(P)$.

Caso inductivo: Si $\alpha, \beta \in \mathcal{L}(P)$, entonces $(\neg\alpha)$, $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $(\alpha \to \beta)$, $(\alpha \leftrightarrow \beta) \in \mathcal{L}(P)$.

Estableciendo la validez de un argumento

Existen dos enfoques para probar la validez de un argumento:

1. **Enfoque semántico (teoría de modelos):** verificar que cada valuación de las proposiciones que hace verdadera simultáneamente a todas las premisas, también hace verdadera a la conclusión.

$$\frac{p \vee q \rightarrow r \qquad p}{r}$$

p	q	r	$p \vee q \rightarrow r$
1	1	**1**	1
1	1	0	0
1	0	**1**	1
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	1
0	0	0	1

Estableciendo la validez de un argumento

Existen dos enfoques para probar la validez de un argumento:

2. **Enfoque sintáctico o deductivo (teoría de la demostración):** verificar que el argumento puede construirse combinando razonamientos "básicos" o "elementales".

Identificando los patrones de razonamiento elementales

$$\frac{\begin{array}{c}\alpha\\ \vdots\\ \mathit{false}\end{array}}{\neg\alpha}\ [\neg\mathsf{I}] \qquad \frac{\alpha \quad \beta}{\alpha \wedge \beta}\ [\wedge\mathsf{I}]$$

$$\frac{\alpha \wedge \beta}{\alpha}\ [\wedge\mathsf{E}_L] \qquad \frac{\alpha \wedge \beta}{\beta}\ [\wedge\mathsf{E}_R] \qquad \frac{\alpha \to \beta \quad \alpha}{\beta}\ [\to\mathsf{E}]$$

- Cada regla está asociada a un operador particular. Se nombran de acuerdo a la posición que ocupa el operador en la regla.
- Las reglas de la primera línea introducen el operador ($\neg$ y $\wedge$, resp.) en la conclusión de la regla. Se llaman reglas de introducción.
- Las reglas de la segunda línea eliminan el operador ($\wedge$ y $\to$, resp.) que aparece en las premisas. Se llaman reglas de eliminación.

Deducción natural

Deducción natural

- Sistema de prueba[1] para establecer formalmente la validez de argumentos (enfoque deductivo o sintáctico).
- Un argumento está formado de un conjunto (finito) de hipótesis o premisas, y de una conclusión. El objetivo es probar que la conclusión sigue o se deriva del conjunto de premisas (validez del argumento).
- Para ello utiliza un conjunto de reglas de inferencia que capturan patrones de razonamientos "elementales", alrededor de los distintos operadores de la lógica proposicional.

[1] Introducido por Gerhared Gentzen en 1930.

Reglas de inferencia: La conjunción

Introducción del $\wedge$:

$$\frac{\alpha \qquad \beta}{\alpha \wedge \beta}\ [\wedge \mathsf{I}]$$

Para probar la conjunción entre dos fórmulas hay que probar cada una de ellas.

Eliminación del $\wedge$:

$$\frac{\alpha \wedge \beta}{\alpha}\ [\wedge \mathsf{E}_L] \qquad \frac{\alpha \wedge \beta}{\beta}\ [\wedge \mathsf{E}_R]$$

De la conjunción entre dos fórmulas, podemos extraer cada una de ellas.

Reglas de inferencia: El implica

Introducción del $\rightarrow$:

$$\dfrac{\begin{array}{c}\alpha \\ \vdots \\ \beta\end{array}}{\alpha \rightarrow \beta}\ 1\ \ [\rightarrow \text{I}]$$

Para probar una implicancia usamos un razonamiento hipotético suponemos "temporalmente" el antecedente, y a partir de ello probamos el consecuente.

Una vez que la implicancia está probada, no podemos seguir suponiendo el antecedente. Para ello "cancelamos" el antecedente, indicándolo con una etiqueta al costado del antecedente y de la implicancia (1 en este caso). Intuitivamente, esto delimita el "alcance" de la suposición.

Reglas de inferencia: El implica

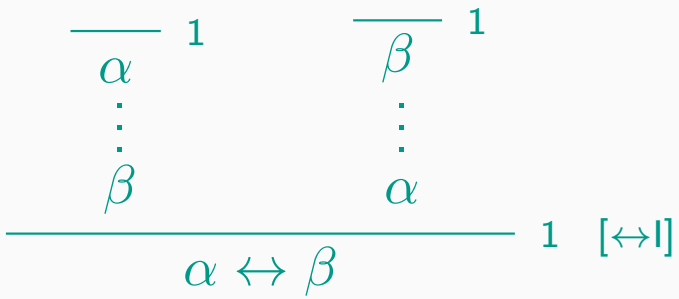
Eliminación del $\rightarrow$:

$$\frac{\alpha \rightarrow \beta \qquad \alpha}{\beta}\ [\rightarrow \text{E}]$$

Para eliminar un implica, debo probar su antecedente, y así puedo concluir su consecuente.

Reglas de inferencia: El si y sólo si

Introducción del $\leftrightarrow$:



Para probar un si-y-sólo-si, debemos probar cada uno de las dos direcciones, a través de un razonamiento hipotético.

Reglas de inferencia: El si y sólo si

Eliminación del $\leftrightarrow$: $\dfrac{\alpha \leftrightarrow \beta \quad \alpha}{\beta}\ [\leftrightarrow \mathsf{E}_L] \qquad \dfrac{\alpha \leftrightarrow \beta \quad \beta}{\alpha}\ [\leftrightarrow \mathsf{E}_R]$

Para eliminar el si-y-sólo-si entre dos fórmulas, debemos probar alguna de ellas, y así concluir la otra.

Reglas de inferencia: La disyunción

Introducción del $\vee$: $\dfrac{\alpha}{\alpha \vee \beta}\ [\vee \mathsf{I}_L] \qquad \dfrac{\beta}{\alpha \vee \beta}\ [\vee \mathsf{I}_R]$

Para probar la disyunción entre dos fórmulas hay que probar alguna de ellas.

Reglas de inferencia: La disyunción

Eliminación del $\vee$:

$$\dfrac{\alpha \vee \beta \qquad \begin{array}{c}\overline{\alpha}^{\,1}\\ \vdots \\ \gamma\end{array} \qquad \begin{array}{c}\overline{\beta}^{\,1}\\ \vdots \\ \gamma\end{array}}{\gamma}\,1\ [\vee\mathrm{E}]$$

La regla internaliza el "análisis de casos": para derivar una conclusión a partir de la disyunción entre dos fórmulas, es necesario establecer la conclusión a partir de cada una de las fórmulas. Para ello usamos un razonamiento hipotético (al igual que en $[\rightarrow \mathrm{I}]$).

Reglas de inferencia: La negación

Como

$$\neg\alpha \equiv \alpha \rightarrow \mathit{false}$$

para introducir y eliminar la negación, utilizamos las reglas del implica:

Introducción del $\neg$:

$$\frac{\begin{array}{c}\overline{\quad\alpha\quad}\ 1\\ \vdots\\ \mathit{false}\end{array}}{\neg\alpha}\ 1\ [\neg\mathrm{I}]$$

Eliminación del $\neg$:

$$\frac{\neg\alpha \qquad \alpha}{\mathit{false}}\ [\neg\mathrm{E}]$$

Reglas de inferencia: El verdadero y falso

Introducción del *true*: $\dfrac{}{true}\ [true\text{I}]$

Establecer *true* no requiere ninguna premisa (regla de introducción).

Por otro lado, a partir de *true* no podemos derivar nada (más allá del mismo *true*), por lo que no presenta regla de eliminación.

Eliminación de *false*: $\dfrac{false}{\alpha}\ [false\text{E}]$

A partir de una contradicción podemos derivar cualquier fórmula (eliminación de *false*).

Por otro lado, *false* no presenta regla de introducción porque la única manera de probarlo es a partir de hipótesis contradictorias.

Reglas de inferencia: Reducción al absurdo

Reducción al absurdo:

$$\dfrac{\dfrac{}{\neg\alpha}\,1 \quad \vdots \quad \mathit{false}}{\alpha}\,1\ \text{[RAA]}$$

Internaliza las pruebas por contradicción: para probar una fórmula, suponemos temporalmente su negación (razonamiento hipotético), y a partir de ello derivamos una contradicción (que en lógica representamos con el símbolo *false*).

Ejemplo de uso: Transitividad del implica

Vamos a demostrar que de las premisas $\alpha \to \beta$ y $\beta \to \gamma$ se deriva $\alpha \to \gamma$.

$$\dfrac{\dfrac{\beta \to \gamma \qquad \dfrac{\alpha \to \beta \quad \overline{\alpha}^{\,1}}{\beta}\,[\to\mathsf{E}]}{\gamma}\,[\to\mathsf{E}]}{\alpha \to \gamma}\;1\;[\to\mathsf{I}]$$

En deducción natural, las demostraciones (de validez de un argumento) consisten en un árbol de derivación (invertido), donde la raíz representa la conclusión y las hojas representan las premisas.

Teoremas de la deducción natural

Si Γ es un conjunto de fórmulas (premisas) y α es una fórmula (conclusión), usamos $\Gamma \vdash \alpha$ para denotar que α es demostrable (o probable) a partir de Γ, siguiendo el conjunto de reglas de inferencia antes presentado.

Ejemplo: Cualesquiera sean las fórmulas α, β y γ,

- $\{\alpha \to \beta, \beta \to \gamma\} \vdash \alpha \to \gamma$
- $\{\alpha \to \beta, \neg\beta\} \vdash \neg\alpha$

Si $\Gamma = \emptyset$, en vez de $\emptyset \vdash \alpha$ escribimos simplemente $\vdash \alpha$, y decimos que α es demostrable (o probable).

Ejemplo: Cualesquiera sean las fórmulas α y β,

- $\vdash \alpha \vee \neg\alpha$
- $\vdash \alpha \to \alpha \vee \beta$
- $\vdash (\alpha \vee \beta) \wedge \neg\alpha \to \beta$

Teoremas de la deducción natural

Dos observaciones importantes sobre uno de los ejemplos:

$$\{\alpha \to \beta, \neg\beta\} \vdash \neg\alpha$$

- Siempre es posible reducir la demostrabilidad a partir de un conjunto de premisas no-vacío, a la demostrabilidad a partir de uno vacío:

$$\{\alpha \to \beta, \neg\beta\} \vdash \neg\alpha \qquad \text{si y sólo si} \qquad \vdash (\alpha \to \beta) \wedge \neg\beta \to \neg\alpha$$

- En esa transformación, las premisas también pueden conectarse a través de $\to$'s anidados (en vez de $\wedge$'s):

$$\{\alpha \to \beta, \neg\beta\} \vdash \neg\alpha \qquad \text{si y sólo si} \qquad \vdash (\alpha \to \beta) \to (\neg\beta \to \neg\alpha)$$

Demostraciones en deducción natural

Como vimos, el siguiente árbol de derivación es la demostración de que $\{\alpha \to \beta, \beta \to \gamma\} \vdash \alpha \to \gamma$:

$$\dfrac{\dfrac{\beta \to \gamma \qquad \dfrac{\alpha \to \beta \quad \overline{\alpha}^{\,1}}{\beta}\,[\to\mathsf{E}]}{\gamma}\,[\to\mathsf{E}]}{\alpha \to \gamma}\;1\;[\to\mathsf{I}]$$

Construir estos árboles de derivación a mano es tedioso y propenso a errores. Para facilitarnos la tarea, vamos a utilizar una herramienta llamada asistente de prueba para escribir las derivaciones.

El asistente de pruebas Lean

Asistente de pruebas

- Permiten escribir teoremas usando un lenguaje lógico formal, y demostrarlos de manera interactiva.
- Las demostraciones son escritas en algo así como un "lenguaje de programación" diseñado específicamente para ese fin.
- Un vez que el usuario escribe la demostración, el asistente verifica que ésta sea correcta.

Asistente de pruebas

Demostración en el asistente de pruebas Lean de que $\sqrt{2}$ es irracional.

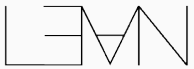
```
import data.nat.prime
open nat

theorem sqrt_two_irrational {a b : ℕ} (co : gcd a b = 1) :
  a^2 ≠ 2 * b^2 :=
assume h : a^2 = 2 * b^2,
have 2 ∣ a^2,
  by simp [h],
have 2 ∣ a,
  from prime.dvd_of_dvd_pow prime_two this,
exists.elim this $
assume (c : nat) (aeq : a = 2 * c),
have 2 * (2 * c^2) = 2 * b^2,
  by simp [eq.symm h, aeq];
    simp [nat.pow_succ, mul_comm, mul_assoc, mul_left_comm],
have 2 * c^2 = b^2,
  from eq_of_mul_eq_mul_left dec_trivial this,
have 2 ∣ b^2,
  by simp [eq.symm this],
have 2 ∣ b,
  from prime.dvd_of_dvd_pow prime_two this,
have 2 ∣ gcd a b,
  from dvd_gcd ⟨2 ∣ a⟩ ⟨2 ∣ b⟩,
have 2 ∣ (1 : ℕ),
  by simp * at *,
show false, from absurd ⟨2 ∣ 1⟩ dec_trivial
```

Asistente de pruebas

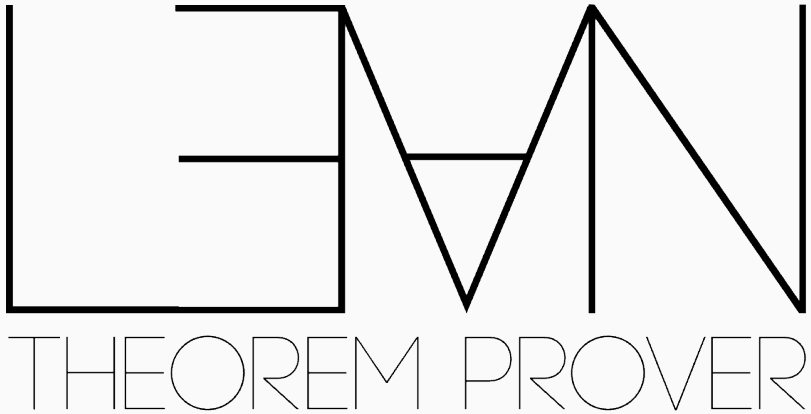
- Se han utilizado para formalizar una pletora de resultados:
  - **Matemática:** teorema de los 4 colores (teoría de grafos), teorema de la curva de Jordan (topología), teorema de los números primos (teoría de números), etc.
  - **Ciencias de la computación:** compilador de C (CompCert), protocolos criptográficos (CertiCrypt), diseño de chips, conjunto de instrucciones x86 y LLVM, metateoría sobre lenguajes de programación (artículos científicos), etc.

- Muchas opciones: Coq, Agda, F*, Isabelle, ACL2, PVS, etc.

Lean



ABOUT DOWNLOAD DOCUMENTATION PUBLICATIONS LINKS PEOPLE

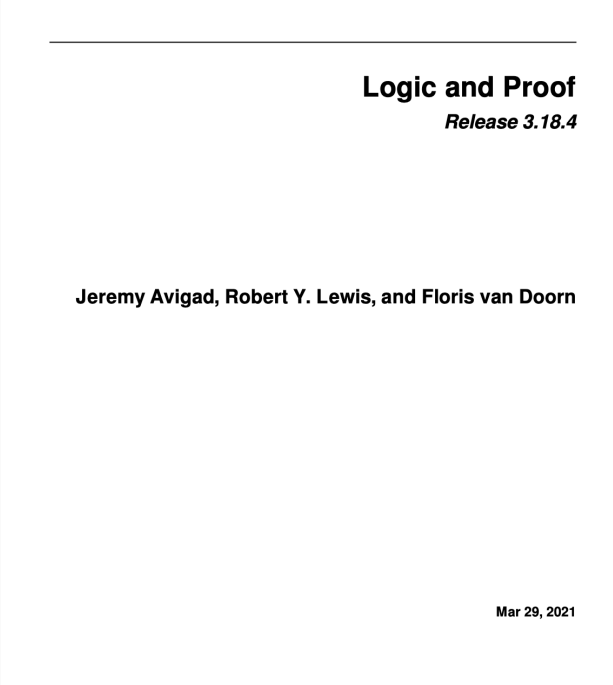


Microsoft Research

https://leanprover.github.io/

Referencias importantes

Interfaz web al asistente de pruebas (oficial, comunidad)



Material de referencia para esta unidad (link)

Demo

Material de referencia para la demo: https://leanprover.github.io/logic_and_proof/propositional_logic_in_lean.html