Matemática inversa: axiomas y teoremas y viceversa

Ignacio Mas Mesa

25 de junio de 2019

Universidad de Granada

Introducción

¿Qué es la matemática inversa?

> *When the theorem is proved from the right axioms, the axioms can be proved from the theorem.*
>
> — Harvey Friedman

Algunos teoremas son equivalentes a ciertos axiomas.

Consideraciones

Cualesquiera dos fórmulas verdaderas son **equivalentes** desde un punto de vista lógico.

Necesitamos un sistema axiomático más débil para formalizar esta idea.

Aritmética y aritmetización

El marco de trabajo

El sistema Z_2 de la aritmética de segundo orden.

- Axiomas básicos de la aritmética ($n+1 \neq 0$, $n+0=n$, etc).
- Axioma de inducción

$$(0 \in X \wedge (n \in X \implies n+1 \in X)) \implies \forall n(n \in X).$$

- Esquema de axioma de comprensión

$$\exists X \forall n(n \in X \iff \varphi(n)).$$

Forma prenexa y jerarquía aritmética

Toda fórmula es equivalente a una fórmula en forma prenexa, i.e., con todos los cuantificadores apareciendo al inicio.

La jerarquía aritmética permite asignar una medida de complejidad a una fórmula en función de cuántas veces alterna entre cuantificadores existenciales y universales.

$$\exists m(n = 2 \cdot m) \in \Sigma_1^0,$$

$$\forall n \exists m(n < m) \in \Pi_2^0,$$

$$(n + 2) \cdot k > m - 3 \in \Sigma_0^0 = \Delta_0^0 = \Pi_0^0,$$

Computabilidad

Funciones primitivas recursivas y parciales recursivas

- Funciones iniciales: función cero, función sucesor y proyecciones.
- Composición.
- Recursión primitiva (codifica `reduce` y los bucles `for`).

La función de Ackermann es computable pero no primitiva recursiva

$$A(m,n) := \begin{cases} n+1 & \text{si } m = 0, \\ A(m-1,1) & \text{si } m > 0 \text{ y } n = 0, \\ A(m-1,A(m,n-1)) & \text{en caso contrario.} \end{cases}$$

Hace falta incluir el operador μ de minimización o búsqueda no acotada.

Máquinas de Turing

Tienen su propio formalismo matemático, pero conceptualmente representan una máquina, con un programa pregrabado, que puede leer y escribir sobre una cinta infinita, para lo cual se ayuda de estados.

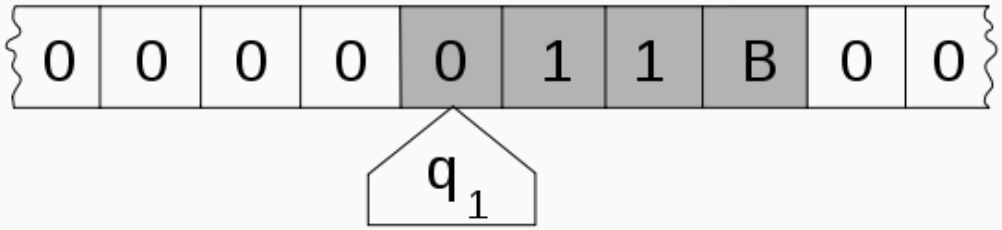

Figura 1: Diagrama de una máquina de Turing

Son equivalentes en *poder* a las funciones parciales recursivas, pero más *intuitivas* que estas o que el cálculo λ. Constituyeron la base de los primeros ordenadores y aún hoy se identifican semejanzas.

Tesis de Church-Turing

Toda función para la cual existe un procedimiento intuitivamente efectivo para calcular sus valores es computable.

Enumerando máquinas de Turing

Dada una máquina de Turing, podemos asignarle un único número identificador de forma computable. Dado el número, podemos recuperar la máquina de Turing también de forma computable.

Turing advirtió la importancia de esto: existe una máquina universal de Turing. Este descubrimiento ayudó a establecer el paradigma del programa almacenado en disco.

Existen problemas que no pueden ser resueltos por una máquina de Turing, como el problema de la parada.

Reducibilidad, oráculos y grados de Turing (I)

La reducibilidad nos permite comparar la dificultad relativa de dos conjuntos o problemas. Estudiamos las reducciones muchos-a-uno y las reducciones de Turing.

Una máquina de Turing con oráculo puede hacer preguntas de problemas que no se pueden resolver y usar las respuestas para resolver problemas aún más difíciles.

Se puede generalizar el problema de la parada para obtener la operación salto de Turing.

Reducibilidad, oráculos y grados de Turing (II)

La reducibilidad induce una relación de equivalencia sobre los conjuntos que da lugar a los grados de Turing.

El teorema de Post relaciona íntimamente la estructura de los grados de Turing con la de la jerarquía aritmética. En particular:

- Σ_1^0 es lo mismo que recursivamente enumerable,
- Π_1^0 es lo mismo que co-recursivamente enumerable,
- Δ_1^0 es lo mismo que computable.

Subsistemas de Z_2

El subsistema base: RCA_0

- Axiomas básicos de la aritmética ($n + 1 \neq 0$, $n + 0 = n$, etc).
- Restringe el axioma de inducción a fórmulas Σ^0_1.
- Restringe el esquema de axioma de comprensión a fórmulas Δ^0_1 (computables).

Permite definir $\mathbb{N}$, así como $\mathbb{Z}$ y $\mathbb{Q}$ por clases de equivalencia.

Los números reales son sucesiones de racionales de Cauchy con ciertas propiedades adicionales.

$\mathbf{WKL_0}$

El lema débil de König asegura que todo árbol binario infinito debe tener una rama infinita.

El sistema WKL_0 añade el lema débil de König a los axiomas de RCA_0.

Es propiamente más fuerte que RCA_0.

$\mathbf{ACA}_0$

El sistema ACA_0 añade a los axiomas de RCA_0

1. el axioma de inducción de segundo orden,
2. comprensión aritmética.

Es propiamente más fuerte que WKL_0.

Resultados

Equivalencias en RCA_0

- RCA_0 demuestra el teorema de Bolzano,
- En RCA_0 son equivalentes el lema débil de König y el teorema de Heine-Borel,
- En RCA_0 equivalen
  - (I) comprensión aritmética,
  - (II) teorema de Bolzano-Weierstraß,
  - (III) teorema de la convergencia monótona,
  - (IV) criterio de convergencia de Cauchy y
  - (V) existencia de supremo para sucesiones acotadas.

Conclusiones y trabajo futuro

La matemática inversa es una potente herramienta para el estudio fundacional de las matemáticas, que permite comparar la *fuerza relativa* de dos teoremas formalizando esta noción.

En caso de continuar con el proyecto podríamos considerar estudiar los sistemas ATR_0 y Π^1_1-CA_0, propiedades de los modelos de los sistemas que hemos escogido o profundizar en la estructura de los grados de Turing.

Ejemplo de matemática inversa: WKL_0 y el teorema de Heine-Borel

$\mathbf{WKL_0}$ implica el teorema de Heine-Borel (I)

Sea $\{(a_i, b_i) : i \in \mathbb{N}\}$ una sucesión de intervalos abiertos que recubre el intervalo $[0, 1]$.

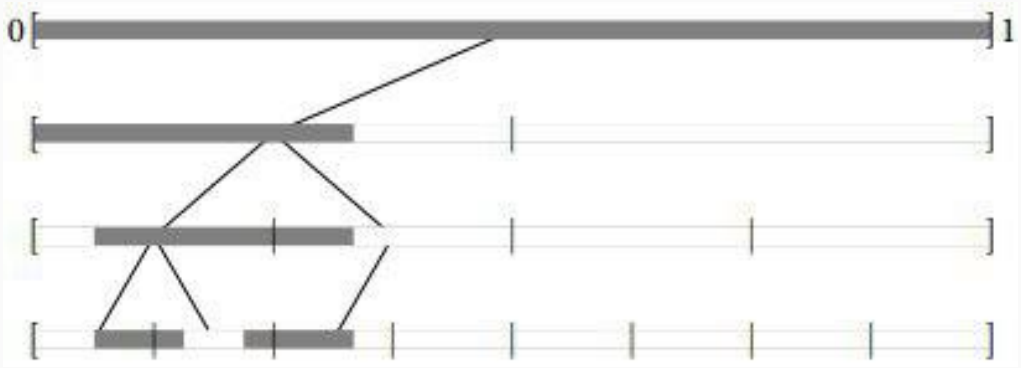

Figura 2: Construcción de T

Construimos un árbol binario T en el que incluimos los intervalos $[\frac{m}{2^n}, \frac{m+1}{2^n}]$ si dicho intervalo no está completamente recubierto $(a_1, b_1), \ldots, (a_n, b_n)$.

$\mathbf{WKL_0}$ implica el teorema de Heine-Borel (II)

Para cada $x \in [0,1]$, existe i tal que $x \in (a_i, b_i)$ y n suficientemente grande tal que

$$x \in \left[\frac{m}{2^n}, \frac{m+1}{2^n}\right] \subset (a_i, b_i).$$

Por tanto, T no puede tener ramas infinitas, luego es finito. Entonces podemos encontrar un subrecubrimiento de $[0,1]$

$$(a_1, b_1), \ldots, (a_r, b_r)$$

para un r apropiado.

El conjunto de Cantor

Para la demostración del recíproco haremos uso del conjunto de Cantor, C.

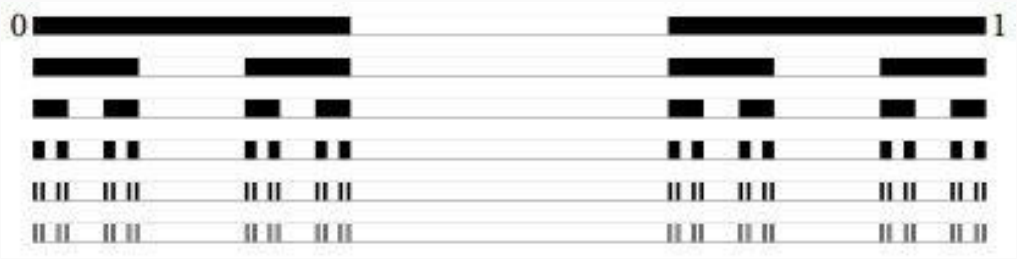

Figura 3: Conjunto de Cantor

Cada punto del mismo se puede identificar con una rama infinita del árbol binario completo. Además, la sucesión de intervalos $(\frac{1}{3}, \frac{2}{3}), (\frac{1}{9}, \frac{2}{9}), (\frac{7}{9}, \frac{8}{9}), (\frac{1}{27}, \frac{2}{27}), \ldots$ recubre $[0, 1] \setminus C$.

El teorema de Heine-Borel implica WKL_0 (I)

Asociaremos a cada vértice del árbol binario completo un intervalo abierto de forma que recubran C.

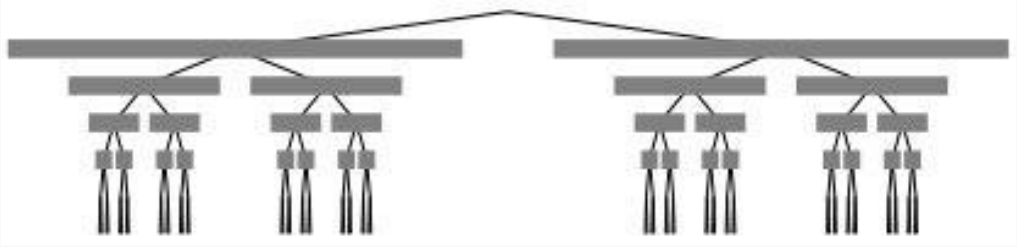

Figura 4: Intervalos recubridores de C

El teorema de Heine-Borel implica WKL_0 (II)

Sea T un árbol binario sin ramas infinitas. Consideramos los intervalos asociados a sus hojas caídas, que recubren C.

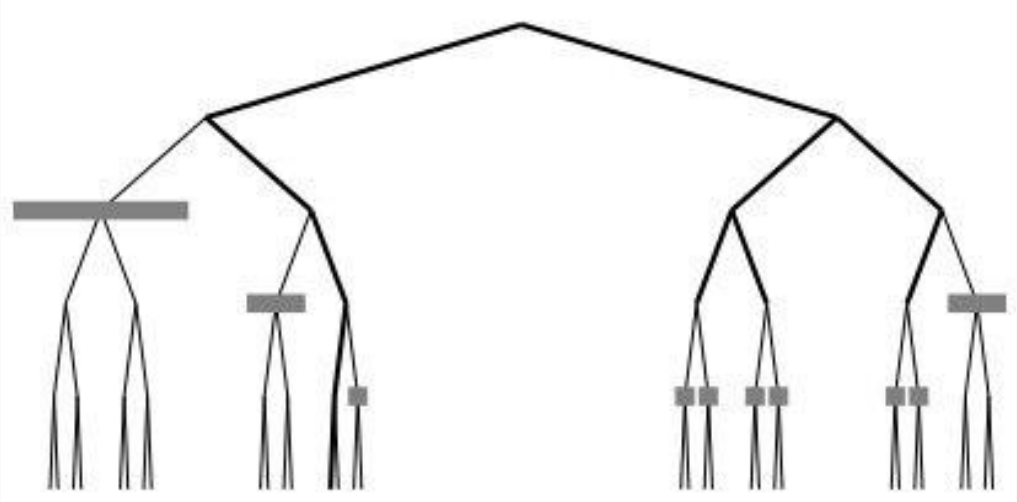

Figura 5: Hojas caídas de T

Si unimos esta sucesión de intervalos a los que recubren $[0,1] \setminus C$, obtenemos un recubrimiento de $[0,1]$ por abiertos.

El teorema de Heine-Borel implica WKL_0 (III)

Por el teorema de Heine-Borel, podemos quedarnos con un subrecubrimiento finito. Sin embargo, ninguno de los intervalos que recubren $[0,1] \setminus C$ contiene ningún elemento de C. Por tanto, existe un número finito de intervalos asociados a hojas caídas que recubre C.

Por la construcción de los intervalos, esto significa que T tiene un número finito de hojas caídas y por tanto es finito.

Preguntas

https://github.com/cronos2/reverse-mathematics

Referencias

Figura 1: *Turing machine* por Nynexman4464. `https://commons.wikimedia.org/wiki/File:Turing_machine_2b.svg`

Figura 2, Figura 3, Figura 4, Figura 5: Stillwell, J. (2019) *Reverse Mathematics: Proofs from the Inside Out*. Princeton University Press (pp. 137, 138, 141), fig.

Bibliografía i


Harvey Friedman. "Some systems of second order arithmetic and their use". En: (1975), págs. 235-242.


Stephen G. Simpson. *Subsystems of Second Order Arithmetic*. 2.ª ed. Perspectives in Logic. Cambridge University Press, 2009. DOI: 10.1017/CBO9780511581007.


John Stillwell. *Reverse Mathematics: Proofs from the Inside Out*. Princeton University Press, 2018. ISBN: 9780691177175.


Alan M. Turing. "On Computable Numbers, with an Application to the Entscheidungsproblem". En: *Proceedings of the London Mathematical Society* 2.42 (1936), págs. 230-265. URL: http://www.cs.helsinki.fi/u/gionis/cc05/OnComputableNumbers.pdf.

Bibliografía ii

- Wilhelm Ackermann. “Zum hilbertschen aufbau der reellen zahlen”. En: *Mathematische Annalen* 99.1 (1928), págs. 118-133.
- Stephen Cole Kleene. “Introduction to metamathematics”. En: (1968).
- S Barry Cooper. *Computability theory*. Chapman y Hall/CRC, 2017.
- John E Hopcroft y Jeffrey D Ullman. “Introduction to Automata Theory, Languages and Computation. Adison-Wesley”. En: *Reading, Mass* (1979).