

prf.1 Introdução

fol:prf:int:
sec

As lógicas comumente têm tanto uma semântica quanto um sistema de **derivação**. A semântica trata de conceitos como verdade, satisfatibilidade, validade e consequência. O propósito dos sistemas de **derivação** é fornecer um método puramente sintático de estabelecer consequência e validade. Eles são puramente sintáticos no sentido de que uma **derivação** em tal sistema é um objeto sintático finito, geralmente uma sequência (ou outra disposição finita) de **sentenças** ou **fórmulas**. Bons sistemas de **derivação** têm a propriedade de que qualquer sequência ou disposição dada de **sentenças** ou **fórmulas** pode ser verificada mecanicamente como “correta”.

Os sistemas de **derivação** mais simples (e historicamente os primeiros) para a lógica de primeira ordem foram *axiomáticos*. Uma sequência de **fórmulas** conta como uma **derivação** em tal sistema se cada **fórmula** individual nela é ou está entre um conjunto fixo de “axiomas” ou segue de **fórmulas** anteriores na sequência por uma entre um número fixo de “regras de inferência”—e pode ser verificado mecanicamente se uma **fórmula** é um axioma e se segue corretamente de outras **fórmulas** por uma das regras de inferência. Sistemas de **derivação** axiomáticos são fáceis de descrever—e também fáceis de tratar meta-teoricamente—mas as **derivações** neles são difíceis de ler e entender, e também difíceis de produzir.

Outros sistemas de **derivação** foram desenvolvidos com o objetivo de facilitar a construção de **derivações** ou facilitar a compreensão de **derivações** uma vez completas. Exemplos são a dedução natural, árvores de verdade, também conhecidas como provas por tableaux, e o cálculo de sequentes. Alguns sistemas de **derivação** são projetados especialmente com a mecanização em mente; por exemplo, o método da resolução é fácil de implementar em software (mas suas **derivações** são essencialmente impossíveis de entender). A maioria desses outros sistemas de **derivação** representa **derivações** como árvores de **fórmulas** em vez de sequências. Isso facilita ver quais partes de uma **derivação** dependem de quais outras partes.

Assim, para uma dada lógica, como a lógica de primeira ordem, os diferentes sistemas de **derivação** darão diferentes explicações do que significa uma **sentença** ser um *teorema* e o que significa uma **sentença** ser **derivável** a partir de outras. Seja como for feito isso (via **derivações** axiomáticas, deduções naturais, **derivações** de sequentes, árvores de verdade, refutações por resolução), queremos que essas relações correspondam às noções semânticas de validade e consequência. Escrevamos $\vdash \varphi$ para “ φ é um teorema” e “ $\Gamma \vdash \varphi$ ” para “ φ é **derivável** de Γ .” Seja como for $\vdash$ definido, queremos que corresponda a $\models$, isto é:

1. $\vdash \varphi$ se, e somente se, $\models \varphi$
2. $\Gamma \vdash \varphi$ se, e somente se, $\Gamma \models \varphi$

A direção “somente se” acima é chamada de *correção*. Um sistema de **derivação** é correto se a **derivabilidade** garante consequência (ou validade). Todo sistema

de *derivação* decente tem que ser correto; sistemas de *derivação* incorretos não são úteis de modo algum. Afinal, o propósito inteiro de uma *derivação* é fornecer uma garantia sintática de validade ou consequência. Provaremos a correção para os sistemas de *derivação* que apresentamos.

A direção converso “se” também é importante: é chamada de *completude*. Um sistema de *derivação* completo é forte o suficiente para mostrar que φ é um teorema sempre que φ é válida, e que $\Gamma \vdash \varphi$ sempre que $\Gamma \models \varphi$. A completude é mais difícil de estabelecer, e algumas lógicas não têm sistemas de *derivação* completos. A lógica de primeira ordem tem. Kurt Gödel foi o primeiro a provar a completude para um sistema de *derivação* da lógica de primeira ordem em sua dissertação de 1929.

Outro conceito ligado aos sistemas de *derivação* é o de *consistência*. Um conjunto de *sentenças* é chamado inconsistente se qualquer coisa pode ser *derivada* dele, e consistente caso contrário. A inconsistência é a contrapartida sintática da insatisfatibilidade: como conjuntos insatisfatíveis, conjuntos inconsistentes de *sentenças* não fazem boas teorias; são defeituosos de maneira fundamental. Conjuntos consistentes de *sentenças* podem não ser verdadeiros ou úteis, mas pelo menos passam esse limiar mínimo de utilidade lógica. Para diferentes sistemas de *derivação*, a definição específica de consistência de conjuntos de *sentenças* pode diferir, mas como $\vdash$, queremos que a consistência coincida com sua contrapartida semântica, a satisfatibilidade. Queremos que seja sempre o caso que Γ é consistente se, e somente se, é satisfatível. Aqui, a direção “somente se” corresponde à completude (consistência garante satisfatibilidade), e a direção “se” corresponde à correção (satisfatibilidade garante consistência). De fato, para a lógica clássica de primeira ordem, as duas versões de correção e completude são equivalentes.

Créditos das Imagens

Referências Bibliográficas