

标题 title

作者 *author*

2023 年 7 月 28 日

前言

目录

前言	i
第一部分 科学的逻辑	1
第一章 合情推理	2
§1.1 回顾：命题逻辑的演绎推理	2
§1.2 合情推理的数学模型	5
1.2.1 似然，合情推理的原则	6
1.2.2 似然与概率	9
§1.3 合情推理的归纳强论证	11
1.3.1 先验与基率谬论	11
1.3.2 归纳强论证	12
1.3.3 有效论证和归纳强论证的比较	17
第二章 Markov 链与决策	22
§2.1 Markov 链	22
§2.2 Markov 奖励过程 (MRP)	28
§2.3 Markov 决策过程 (MDP)	31
§2.4 隐 Markov 模型 (HMM)	37
2.4.1 评估问题	38
2.4.2 解释问题	39
第二部分 信息与数据	42
第三章 信息论基础	43

§3.1 熵	43
3.1.1 概念的导出	43
3.1.2 概念与性质	46
3.1.3 熵与通信理论	51
§3.2 Kullback-Leibler 散度	54
3.2.1 定义	54
3.2.2 两个关于信息的不等式	56
3.2.3 在机器学习中的应用：语言生成模型	57
§3.3 附录：Shannon 定理的证明	58
§3.4 习题	59
§3.5 章末注记	61
第四章 Johnson-Lindenstrauss 引理	63
§4.1 机器学习中的数据	63
§4.2 矩法与集中不等式	64
§4.3 J-L 引理的陈述与证明	68
§4.4 J-L 引理的应用	72
§4.5 习题	73
§4.6 章末注记	73
第五章 差分隐私	74
§5.1 数据隐私问题	74
§5.2 差分隐私的定义与性质	76
§5.3 差分隐私的应用	80
5.3.1 随机反应算法	80
5.3.2 全局灵敏度与 Laplace 机制	81
5.3.3 DP 版本 Llyod 算法	83
§5.4 差分隐私与信息论	84
§5.5 习题	85
§5.6 章末注记	85
第三部分 决策与优化	86
第六章 凸分析	87

§6.1 决策与优化的基本原理	87
6.1.1 统计决策理论	87
6.1.2 优化问题	88
6.1.3 例子：网格搜索算法	91
§6.2 凸函数	93
§6.3 凸集	96
6.3.1 基本定义和性质	96
6.3.2 分离超平面定理	98
第七章 对偶理论	100
§7.1 条件极值与 Lagrange 乘子法	101
§7.2 Karush–Kuhn–Tucker 条件	104
§7.3 Lagrange 对偶	107
7.3.1 Lagrange 定理	107
7.3.2 弱对偶定理，强对偶定理	111
§7.4 应用：支持向量机 (SVM)	115
第八章 不动点理论	118
§8.1 Banach 不动点定理	118
§8.2 Brouwer 不动点定理	121
§8.3 不动点的一般视角	124
第四部分 逻辑与博弈	125
第九章 动态博弈	126
§9.1 输赢博弈	126
§9.2 随机博弈 (Markov 博弈)	131
第十章 静态博弈	137
§10.1 正则形式博弈	137
10.1.1 生成对抗网络	138
10.1.2 混合策略	140
§10.2 不完全信息博弈 (Bayes 博弈)	141

第五部分 认知逻辑	146
第十一章 模态逻辑基础	147
§11.1 模态逻辑的起源	147
11.1.1 三段论	147
11.1.2 非经典逻辑	148
§11.2 模态语言	149
§11.3 Kripke 语义与框架语义	152
§11.4 模态可定义性	157
第十二章 认知逻辑与共同知识	165
§12.1 “泥泞的孩童”谜题	165
§12.2 认知逻辑的基本模型与性质	170
§12.3 对不一致达成一致	180
§12.4 Rubinstein 电子邮件博弈	186

第一部分

科学的逻辑

第二部分

信息与数据

第三部分

决策与优化

第四部分

逻辑与博弈

第五部分

认知逻辑

第十一章 模态逻辑基础

人工智能的讨论不可避免要接触到很多人独有的哲学概念：认知、信念、知识、理解、情感、意识……我们需要有一套恰当的数学工具来表述这些哲学概念，从而算法化、自动化地模拟人。过去，现在，乃至未来，最为成功的数学模型就是**模态逻辑**。很多不精确的哲学讨论可以通过逻辑的方式形式化、数学化，最后算法化。模态逻辑已经在计算机科学中起到了重要的作用（模型验证、形式化方法），它势必会在人工智能中也起到根基性的作用。

§11.1 模态逻辑的起源

11.1.1 三段论

早在亚里士多德的时期，模态逻辑的概念就被提了出来。回忆：亚里士多德的强三段论是有效的：

- 大前提：所有 A 都是 B 。
- 小前提：所有 B 都是 C 。
- 结论：因此，所有 A 都是 C 。

人都会死，苏格拉底是人，所以苏格拉底会死。

三段论可以进行各种形式的扩展，例如：

- 加入量词：任何对象，如果这个对象是人，那么它会死，苏格拉底这个对象是人，所以苏格拉底会死。
- 加入性质词：肯定的、否定的
- 加入模态词：无效、可能、必然、根据情况……

实际上亚里士多德也考虑过模态三段论. 亚里士多德认为如下模态三段论是有效的:

- 大前提: 所有 A 都必然是 B .
- 小前提: 所有 B 都是 C .
- 结论: 因此, 所有 A 都必然是 C .

然而, 他认为如下的模态三段论不是有效的:

- 所有 A 都是 B .
- 所有 B 都必然是 C .
- 因此, 所有 A 都必然是 C .

所有通班同学都是单身汉, 所有单身汉都必然是男性. 那么是否有: 所有通班同学都必然是男性? 实际上, 通过现代模态逻辑的表述可以证明, XLL 型三段论也是有效的。这足以说明, 亚里士多德对于模态的理解还有很多缺陷, 或者反过来说, 模态三段论比经典的三段论更加复杂.

类似的例子是, 从物和从言, 考虑如下句子:

我觉得有人作弊.

这句话有两种解读方式. 从物的角度, 应该读作 $\exists x$ (我觉得: x 作弊). 从言的角度, 应该读作我觉得 ($\exists x: x$ 作弊).

以上例子说明, 关于模态的逻辑, 并没有非常干净漂亮的、符合直观的定义, 所以模态三段论的讨论并没有流行起来。

11.1.2 非经典逻辑

经典逻辑并不能够非常准确刻画我们对于“逻辑”的认识, 我们将给出以下几个例子。

回忆经典逻辑中的语义等值

$$p \rightarrow q \iff \neg p \vee q.$$

然而, 在哲学上, 这两者是不一样的含义. 考虑命题 p : $1+1 > 2$, q : 太阳从东边升起. “如果 $1+1 > 2$, 那么太阳从东边升起”是毫无道理的. 然而, “或者 $1+1 \not> 2$, 或者太阳从东边升起”是含义清晰的.

再看另一个例子, 考虑命题 p_n : π 的小数位包含连续的 n 个 1. $p_{100} \rightarrow p_{99}$ 是显然的, 然而 $\neg p_{100} \vee p_{99}$ 并不直观!

以上的缺陷都源自我们对蕴含的理解过于简单粗暴。上面的那种蕴含被称为**实质蕴含**，它最重要的特性是承认 $p \rightarrow q \iff \neg p \vee q$ ，特别地允许有 p 假但是 q 真。

为了解决上面的问题，C. I. Lewis 提出了**严格蕴含**的概念，其符号为 $p \rightarrow q$ 。从语义上说，这个符号的意思是**必然有**当 p 是真的时候， q 是真的。也可以说，**不可能**有 p 是真且 q 是假。严格蕴含其实是对实质蕴含的一种改造，我们后面可以证明， p 严格蕴含 q 当且仅当**必然有** p 实质蕴含 q 。

实质蕴含还会导出还有很多不合乎常理的重言式。比如 $p \rightarrow (q \rightarrow p)$ 或者 $(p \rightarrow q) \vee (q \rightarrow r)$ 。

另一种解决此问题的方式是从否定入手，这就是 Brouwer 的直觉主义逻辑，它不承认反证法，因而否定和蕴含的含义发生了变化，例如 $\neg\neg p$ 不再等价于 p 。

从本质上说，非经典逻辑都是尝试将元语言中的概念拿到对象语言中。例如经典逻辑的元语言就是自然语言，而对象语言就是经典逻辑形式系统。在经典逻辑中，必然、可能、过去、未来、知识、信念、可证明等概念都没有办法表示，因此模态逻辑的解决方案是：将这些元语言的概念拿到对象语言中，并进行形式化。

§11.2 模态语言

现在我们正式定义模态语言。我们只考虑最简单的情况，没有量词，只有一个模态算子。基本模态语言 L 可以按照如下定义递归生成：

- 命题字母 $p \in \mathbf{P}$ 属于 L ， \top 属于 L 。
- 如果 ϕ 属于 L ，那么 $\neg\phi$ 和 $\Box\phi$ 也属于 L 。
- 如果 ϕ_1, ϕ_2 属于 L ，那么 $(\phi_1 \wedge \phi_2)$ 也属于 L 。

\Box ：读作“Box”。定义 L 更便捷的记号是使用 Backus-Naur 范式：

$$\phi ::= p \mid \top \mid \neg\phi \mid (\phi \wedge \phi) \mid \Box\phi.$$

类似命题逻辑，我们有如下缩写：

- $\phi \vee \psi \iff \neg(\neg\phi \wedge \neg\psi)$ 。
- $\phi \rightarrow \psi \iff \neg\phi \vee \psi$ 。
- $\perp \iff \neg\top$ 。

这些缩写意味着，我们对 Boole 连接词，依然保持经典逻辑的含义。非经典性只体现在模态算子 \Box 。对偶模态算子 \Diamond 定义为 $\neg\Box\neg$ ，读作“diamond”。类比： $\exists = \neg\forall\neg$ 。

模态逻辑的哲学是多视角下看同一个数学概念。下面是一些模态逻辑的例子，他们对于模态算子的解读都不一样：

- 基本模态逻辑：可能/必然是
- 时序逻辑：将会是
- 道义逻辑：被允许是
- 认知逻辑：被知道是
- 可证性逻辑：可以被证明是
- 动态逻辑：（在经过某些程序步骤之后）会是
- 联盟逻辑：被（她的父母）确保是
- 特征逻辑和描述逻辑：具有的属性是

下面我们看一些不同的逻辑下模态算子的解读：

例 11.1 (基础语义) 如果我们把模态算子按照基础语义来解读的话，我们可以把模态算子 \Box 读成“必然”。 $\Box\phi$ 读作“必然有 ϕ ”。 $\Diamond\phi$ 读作“不是必然有非 ϕ ”，即可能有 ϕ 。因此， \Diamond 读作“可能”。反之， $\Box\phi$ 也可以读作“不可能有非 ϕ ”，即必然有 ϕ 。因此 \Diamond 和 \Box 确实是对偶的。在这样的语义下，我们还有这些例子：

- $\Box p \rightarrow \Diamond p$ ：必然的是可能的。
- $p \rightarrow \Box p$ ：真的是必然的。
- $\Diamond p \rightarrow \Box\Diamond p$ ：可能的是必然可能的。

例 11.2 (认知语义) 如果把模态算子按照认知逻辑来解读解读，我们可以把模态算子 \Box 读成“知道”，并写成 K 。于是， K 表示某个特定的个体对世界的认知。有如下例子：

- $K\phi$ （即 $\Box\phi$ ）表示我知道 ϕ 。
- $K\phi \rightarrow \phi$ 表示如果我知道 ϕ ，那么 ϕ 是真的。
- $\phi \rightarrow K\phi$ 表示如果 ϕ 是真的，那么我知道 ϕ 。

- 最后, $\neg K\phi$ 与 $K(\neg\phi)$ 的含义是不同的, 比如 ϕ 表示“上帝存在”, 那么前者是“我不知道上帝存在”, 而后者是“我知道上帝不存在”, 明显后者的判断要更强一些.

例 11.3 (可证性语义) 如果把模态算子按照可证性逻辑解读, 我们可以把模态算子 \Box 读成“可证明”. 例如, $\Box\phi$ 读作 ϕ 是可证明的. 考虑 *Peano* 算术系统 **PA**, 即一阶逻辑加上 *Peano* 公理. 符号 $\mathbf{PA} \vdash \phi$ 表示 ϕ 可以由 **PA** 演绎出, 即 ϕ 可以被证明. 算术公理系统一个重要的定理叫 Löb 定理: 如果 $\mathbf{PA} \vdash \text{Prov}(\ulcorner \phi \urcorner) \rightarrow \phi$, 那么 $\mathbf{PA} \vdash \phi$. 用自然语言来读, 如果可以证明“如果 ϕ 是可证明的, 那么 ϕ 是真的”, 那么就可以证明 ϕ . 因此, 在可证逻辑中, 它对应 Löb 公式: $\Box(\Box\phi \rightarrow \phi) \rightarrow \Box\phi$.

一般地, 我们可以考虑多个模态算子、一个模态算子涉及多个公式的情形. 模态语言类型是一个元组 (O, ρ) , 其中 O 是一个模态算子 ∇ 的非空集合, $\rho: O \rightarrow \mathbb{N}$ 表示每一个模态算子的元数. 多元模态语言的 BNF 为:

$$\phi ::= p \mid \top \mid \neg\phi \mid (\phi \wedge \phi) \mid \nabla(\underbrace{\phi, \dots, \phi}_{\rho(\nabla)}),$$

其中 $p \in \mathbf{P}$, $\nabla \in O$.

类似地, 定义对偶模态算子 $\Delta(\phi_1, \dots, \phi_k)$ 为 $\neg\nabla(\neg\phi_1, \dots, \neg\phi_k)$.

例 11.4 (时序逻辑) 基础时序逻辑有两个一元模态算子: G 和 H . $G\phi$ 表示未来总会有 ϕ (*always Going to be*). $H\phi$ 表示过去总有 ϕ (*always Has been*).

他们的对偶算子是: F 和 P . $F\phi$ 表示在未来某个时刻会有 ϕ (*be true at some Future time*). $P\phi$ 表示在过去某个时刻有 ϕ (*was true at some Past time*).

还可以加入一个“直到”(Until) 算子 $U(\phi, \psi)$, 它表示直到 ϕ 发生都有 ψ .

下面是一些时序逻辑的例子:

- $P\phi \rightarrow GP\phi$: 如果过去发生过 ϕ , 那么 ϕ 在未来总会发生过.
- $F\phi \rightarrow FF\phi$: 如果未来某个时刻会有 ϕ , 那么在未来的某个时刻会发生: 未来的某个时刻会有 ϕ .
- McKinsey 公式 $GFp \rightarrow FGp$: 如果原子的信息总会在某个未来时刻为真, 那么他会在未来某个时刻之后变得总为真.

例 11.5 (认知逻辑) 基本模态算子为 K_a 和 B_a , K_a 表示个体 a 知道, B_a 表示个体 a 相信. 例如 $K_a K_b \phi \leftrightarrow K_b K_a \phi$ 表示我知道你知道 ϕ 当且仅当你知道我知道 ϕ .

此外, 我们也可以加入共同知识算子 C , $C\phi$ 当且仅当 $K_a(\phi \wedge C\phi)$ 对任意 a 成立.

我们也可以加入二元的相对算子。如相对共同知识 $C^r(\phi, \psi)$ ，表示当所有人都知道 ψ 时，所有人具有共同知识 ϕ 。以及条件信念 $B_a(\phi, \psi)$ ，表示当 ψ 为真时，个体 a 相信 ϕ 。

例 11.6 (命题动态逻辑) 命题动态逻辑 (PDL) 有无穷多个模态算子，记为 $[\pi]$ ，其中 π 按照程序来理解。 $[\pi]\phi$ 解释为：从当前状态开始运行程序 π ，任何一种终止状态， ϕ 都成立。它的对偶算子记为 $\langle \pi \rangle$ ， $\langle \pi \rangle \phi$ 解释为：从当前状态开始运行程序 π ，存在一种终止状态， ϕ 成立。

PDL 的重要区别在于我们可以用模态算子来构造新的模态算子。假设基本程序为 a, b, \dots ，我们可以用三种操作构造新的程序（模态算子）：

- 选择：如果 π_1, π_2 是程序，那么 $\pi_1 \cup \pi_2$ 也是程序，它（非确定性地）执行 π_1 或 π_2 。模态算子为 $[\pi_1 \cup \pi_2]$ 和 $\langle \pi_1 \cup \pi_2 \rangle$ 。
- 复合：如果 π_1, π_2 是程序，那么 $\pi_1; \pi_2$ 也是程序，它先执行 π_1 再执行 π_2 。模态算子为 $[\pi_1; \pi_2]$ 和 $\langle \pi_1; \pi_2 \rangle$ 。
- 迭代：如果 π 是程序，那么 π^* 也是程序，它执行 π 有限次（可能是零次）。模态算子为 $[\pi^*]$ 和 $\langle \pi^* \rangle$ 。

以上构造得到的 PDL 被称为正则 **PDL**。我们还可以引入交 $\pi_1 \cap \pi_2$ ，表示并行计算；也可以引入条件程序 $\phi?$ ，其中 ϕ 是公式。

下面是一些命题动态逻辑的例子：

- $\langle \pi^* \rangle \phi \leftrightarrow \phi \vee \langle \pi; \pi^* \rangle \phi$ 。在执行 π 有限次数后到达一个带有信息 ϕ 的状态当且仅当要么我们已经在当前状态中拥有信息 ϕ ，要么我们可以执行一次 π ，然后在有限次数的 π 迭代后找到一个带有信息 ϕ 的状态。
- Segerberg 公理（归纳公理）： $[\pi^*](\phi \rightarrow [\pi]\phi) \rightarrow (\phi \rightarrow [\pi^*]\phi)$ 。这个公式的含义是什么？
- 用模态算子表示 *if ϕ then a else b* ： $(\phi?; a) \cup (\neg\phi?; b)$ 。

§11.3 Kripke 语义与框架语义

从模型论的角度来说，一个逻辑框架是一个三元组：(语言，模型，语义) $(\mathbf{L}, \mathbf{C}, \models)$ 。例如在命题逻辑中，这三元组分别是：

- 语言：命题公式的集合 $\{\top, \perp, p, p \vee q, \dots\}$ 。

- 模型：常值和命题字母的真假： $\top : \top$, $p : \top$, $q : \perp$, 等等.
- 语义：Boole 函数的真值表递归定义.

对于基本模态逻辑，我们有如下要素：

- 语言：基本模态语言 $\mathbf{ML}(\mathbf{P}, \Box)$.
- 模型：Kripke 模型.
- 语义：Kripke 语义.

一个 Kripke 模型（关系模型）可以看作是一个带有标记的有向边和节点的图，节点表示可能的世界，状态或对象等，用命题字母标记；边表示节点之间的关系，用模态算子标记. 一个框架是一个没有节点标记的模型.

在数学上，Kripke 模型是唯一的定义，然而在哲学上，我们可以对这一数学结构做不同的解读.

在可能世界语义（或 Kripke 语义）中，我们将节点解读为可能世界. 此时 \Box 被理解为“必然”， \Diamond 被理解为“可能”. $\Box\phi$ 在当前世界为真当且仅当 ϕ 在当前世界的所有可能的替代世界上为真. 用形式化的表述就是， $\Box\phi$ 在世界 w 上成立当且仅当 ϕ 在 w 的所有后继上为真. 在这种语义中，一个世界的意义取决于它与其他世界的联系.

在状态语义中，我们将节点解读为状态. 于是边就被解读为状态的转移. $\Box\phi$ 在当前状态为真当且仅当 ϕ 在所有可能转移到的状态上为真. PDL 可以用以上语义来理解. 在哲学中，状态往往是不完全可观测的，此时模态逻辑可以被理解为不完全信息中可以确定的性质.

在对象语义中，我们将节点解读为对象. w 有边指向 v 意味着 w 是 v 包含的一个整体， v 是 w 的一个部分. 在哲学上，模态逻辑可以讨论整体论与还原论. $\Box\phi$ 对一个对象为真当且仅当 ϕ 在它的所有部分都为真.

在基本情形下，Kripke 模型的正式定义如下：

定义 11.1 (Kripke 框架，模型，点模型，二元情形) 一个 **Kripke** 框架是元组 $\mathcal{F} = (W, R)$ ，其中

- W 是非空集合（可能世界集）；
- $R \subseteq W \times W$ 是一个 W 上的二元关系（边）.

一个 **Kripke** 模型 \mathcal{M} 是一个元组 (\mathcal{F}, V) ，其中 \mathcal{F} 是 Kripke 框架， $V : \mathbf{P} \rightarrow 2^W$ 是赋值函数.

一个 **Kripke** 点模型 (\mathcal{M}, w) 是一个 Kripke 模型 \mathcal{M} 加上一个指定的点 $w \in W$.

一个典型的 Kripke 模型如图 11.1 所示.

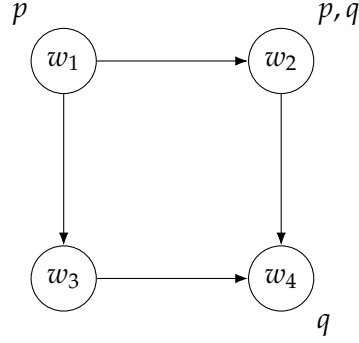


图 11.1: Kripke 模型的例子

对于多元情形, 我们有如下定义:

定义 11.2 (Kripke 框架, 模型, 点模型: 一般情形) 考虑 $\mathbf{ML}(\mathbf{P}, (O, \rho))$, 一个 **Kripke** 框架指的是元组 $\mathcal{F} = (W, \{R_\nabla : \nabla \in O\})$, 其中 W 是非空集合 (可能世界集), R_∇ 是一个 W 上的 $\rho(\nabla) + 1$ 元关系.

一个 **Kripke** 模型 \mathcal{M} 指的是元组 (\mathcal{F}, V) , 其中 \mathcal{F} 是框架, $V : \mathbf{P} \rightarrow 2^W$ 是赋值函数.

一个 **Kripke** 点模型 (\mathcal{M}, w) 是 Kripke 模型 \mathcal{M} 加上一个指定的点 $w \in W$.

下面我们再给出 Kripke 语义的形式化定义.

定义 11.3 (Kripke 语义: 基本情形) 考虑 $\mathbf{ML}(\mathbf{P}, \Box)$. 符号 $\mathcal{M}, w \models \phi$ 表示 ϕ 在点模型 \mathcal{M}, w 是可满足的. 这一个概念可以递归定义如下

- $\mathcal{M}, w \models \top \iff$ 总是.
- $\mathcal{M}, w \models p \iff p \in V(w)$.
- $\mathcal{M}, w \models (\phi \wedge \psi) \iff \mathcal{M}, w \models \phi$ 且 $\mathcal{M}, w \models \psi$.
- $\mathcal{M}, w \models \neg \phi \iff \mathcal{M}, w \not\models \phi$.
- $\mathcal{M}, w \models \Box \phi \iff$ 对所有 v , 如果 wRv , 那么 $\mathcal{M}, v \models \phi$.

因此, $\mathcal{M}, w \models \Diamond \phi \iff$ 存在 v 满足 wRv 使得 $\mathcal{M}, v \models \phi$.

思考: 在图 11.1 中, 有对哪些 i 来说, 成立 $\mathcal{M}, w_i \models \Box(p \rightarrow \Diamond q)$?

定义 11.4 (Kripke 语义：一般情形) 考虑 $\text{ML}(\mathbf{P}, (O, \rho))$. 符号 $\mathcal{M}, w \models \phi$ 表示 ϕ 在点模型 \mathcal{M}, w 是可满足的. 这一个概念可以递归定义如下

- $\mathcal{M}, w \models \top \iff$ 总是.
- $\mathcal{M}, w \models p \iff p \in V(w)$.
- $\mathcal{M}, w \models (\phi \wedge \psi) \iff \mathcal{M}, w \models \phi$ 且 $\mathcal{M}, w \models \psi$.
- $\mathcal{M}, w \models \neg \phi \iff \mathcal{M}, w \not\models \phi$.
- $\mathcal{M}, w \models \nabla(\phi_1, \dots, \phi_{\rho(\nabla)}) \iff$ 对任意 $w_1, w_2, \dots, w_{\rho(\nabla)}$, 如果 $R(w, w_1, \dots, w_{\rho(\nabla)})$, 那么存在 w_i 使得 $\mathcal{M}, w_i \models \phi_1$.

思考：为什么要这么定义 ∇ 的语义？

如果一个模态算子对应的关系是二元关系，我们就称这个模态算子是一元的. 此时，关系 wRv 可以记为 $w \rightarrow_a v$. 模态算子一般写作 \Box_a .

注. (模型验证) 我们考虑如下两个模型验证问题：局部模型验证，即测试 $\mathcal{M}, w \models \phi$ 是否成立；全局模型验证，即计算集合 $\{w \in W_{\mathcal{M}} : \mathcal{M}, w \models \phi\}$.

设 $l_R(X) = \{w \in W_{\mathcal{M}} : \forall v : w \rightarrow_{\mathcal{M}} v \implies v \in X\}$ ，我们可以递归定义 \mathcal{M} 中公式的扩张：

$$\begin{aligned} \llbracket \top \rrbracket^{\mathcal{M}} &= W_{\mathcal{M}}, & \llbracket p \rrbracket^{\mathcal{M}} &= \{w : p \in V(w)\}, \\ \llbracket \neg \phi \rrbracket^{\mathcal{M}} &= W_{\mathcal{M}} \setminus \llbracket \phi \rrbracket^{\mathcal{M}}, & \llbracket (\phi \wedge \psi) \rrbracket^{\mathcal{M}} &= \llbracket \phi \rrbracket^{\mathcal{M}} \cap \llbracket \psi \rrbracket^{\mathcal{M}}, \\ \llbracket \Box \phi \rrbracket^{\mathcal{M}} &= l_R(\llbracket \phi \rrbracket^{\mathcal{M}}). \end{aligned}$$

全局模型验证的有一个很容易想到的算法：按照公式的复杂程度，用 ϕ 的子公式标记 \mathcal{M} 中每个状态的真值. 然而，这个问题在实践中并不平凡，因为状态的数量可能是指数多的！

模态公式的（语义）真值可以从两个维度来讨论：全局还是局部，模型还是框架. 我们有如下四种语义真值的定义：

- ϕ 在点模型 \mathcal{M}, w 可满足指的是 $\mathcal{M}, w \models \phi$.
- ϕ 在模型 \mathcal{M} 有效，记为 $\mathcal{M} \models \phi$ 指的是 $\mathcal{M}, w \models \phi$ 对所有 w 成立.
- ϕ 在点框架 \mathcal{F}, w 有效，记为 $\mathcal{F}, w \models \phi$ 指的是 $\mathcal{M}, w \models \phi$ 对所有基于点框架 \mathcal{F}, w 的点模型 \mathcal{M}, w 上可满足.
- ϕ 在框架 \mathcal{F} 有效，记为 $\mathcal{F} \models \phi$ 指的是 $\mathcal{M} \models \phi$ 对所有基于框架 \mathcal{F} 的模型 \mathcal{M} 上有效.

- ϕ 对框架类 K 有效, 记为 $\models_K \phi$ 指的是 $\mathcal{F} \models \phi$ 对所有 $\mathcal{F} \in K$ 成立.

模态逻辑的真值总结为表 11.1, 我们主要关心高亮的两个部分.

	模型	框架
局部	$\mathcal{M}, w \models \phi$	$\mathcal{F}, w \models \phi$
全局	$\mathcal{M} \models \phi$	$\mathcal{F} \models \phi$

表 11.1: 模态逻辑的真值

我们来看一个框架语义的例子。如图 11.2 所示, 是否成立 $\mathcal{F} \models \Box(p \rightarrow \Diamond p)$?

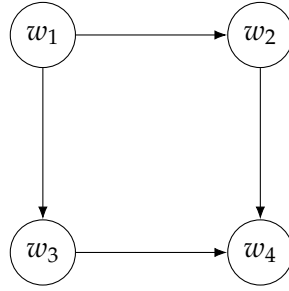


图 11.2: 框架语义的例子

例 11.7 (基本模态逻辑) 考虑基本模态逻辑, 因此他只有模态算子 \Box 和 \Diamond , 分别表示必然和可能. *Kripke* 模型的点应该被解读为可能世界. 我们可以将对于可能和必然的理解写成模态公式. 如果一个东西是真的, 那么他也是可能的: $\phi : p \rightarrow \Diamond p$.

如果一个东西是真的, 那么他也是可能的: $\phi : p \rightarrow \Diamond p$. 我们可以将对于可能和必然的理解反映到 *Kripke* 模型中. 真实世界是一个可能的世界: xRx , 这是一个自反关系. 对于自反点模型以及框架, $\mathcal{M}, v \models \phi$ 以及 $\mathcal{F} \models \phi$.

例 11.8 (时序逻辑) 考虑基本的时序逻辑, 因此他只有模态算子 G 和 H , 以及对偶 F 和 P , 他们分别对应未来和过去. *Kripke* 模型的点应该被解读为时刻, 时刻之间有两个关系: $t_1 R_F t_2$ 表示时刻 t_2 是时刻 t_1 的未来, $t_1 R_P t_2$ 表示时刻 t_2 是时刻 t_1 的过去. 我们对时间的理解将会反映在 *Kripke* 模型上.

关于时间的的一个自然的假设是过去是未来的倒转: 对任意时刻 t_1, t_2 , $t_1 R_F t_2 \iff t_2 R_P t_1$. 如果我们承认时间具有这样的性质, 那么 R_F 和 R_P 实际上就是箭头倒转一下. 记 $R_F = R$, 我们有:

$$\mathcal{M}, w \models F\phi \iff \exists v(wRv \wedge \mathcal{M}, v \models \phi).$$

$$\mathcal{M}, w \models P\phi \iff \exists v(vRw \wedge \mathcal{M}, v \models \phi).$$

进一步假设时间是线性的，也就是关系 R 是一个严格全序：

- 反自反： $\forall x \neg xRx$.
- 传递： $\forall x, y, z (xRy \wedge yRz \rightarrow xRz)$.
- 完全： $\forall x, y (xRy \vee yRx \vee x = y)$.

设 \mathcal{F} 是时间框架，是否有： $\mathcal{F} \models Pp \rightarrow GPp$ 以及 $\mathcal{F} \models Fp \rightarrow HFp$?

§11.4 模态可定义性

逻辑的意义在于把对事物的抽象认知用形式化的语言表述出来. 我们已经看到，我们对事物的认知可以被两种方式描述出来：Kripke 模型（框架）的特殊结构，或者具体的模态公式. 本节探讨这两种方式之间的联系。

我们先看几个简单的例子：

例 11.9 • $\mathcal{M}, w \models \Diamond \top$.

存在一个 v ， $w \rightarrow v$ 并且 $\mathcal{M}, v \models \top$ ，也就是 w 有一个后继。

- $\mathcal{F} \models \Diamond \top$.

每个基于 \mathcal{F} 的点模型 \mathcal{M}, v 都有 $\Diamond \top$ ，即 \mathcal{F} 每个点都有后继。

- $\mathcal{F} \models p \rightarrow \Diamond p$.

任意赋值 V 和任意点 w ，都有 $\mathcal{M}, w \models p \rightarrow \Diamond p$. 因此，如果 w 上有 p ，那么 w 必须有一个后继上面也有 p . 取 V 使得只有 w 上有 p ，因为对任意赋值都要成立，所以在这个赋值下， w 必须要以自己为后继. 因此 \mathcal{F} 充分必要地是一个自反框架。

以上例子启发我们，模态公式可以定义 Kripke 模型的特殊结构，特殊的 Kripke 模型划定了具体的模态公式，这种联系可以被形式化为模态可定义性。

从点模型的角度，我们可以讨论模态公式定义了什么样的点模型。

定义 11.5 (点模型可定义性) 设 K 是一些点模型的集合， Σ 是一些模态公式的集合. 我们说 K 可由公式集 Σ 定义，指的是对任意点模型 \mathcal{M}, w ， $\mathcal{M}, w \in K$ 当且仅当任何 Σ 中的公式在 \mathcal{M}, w 都是可满足的. 如果 $\Sigma = \phi$ ，我们就说 K 可以由公式 ϕ 定义。

对于框架可定义性，我们有类似的定义。

模态可定义性：如果定义 K 的公式集 Σ 有限，那么 K 也可以由单个公式 $\bigwedge_{\phi \in \Sigma} \phi$ 定义。如果 K 由公式 ϕ 定义，那么它的补 \bar{K} 就可以由 $\neg\phi$ 定义。然而，如果 K 由无穷个公式定义，它的补 \bar{K} 不一定可以用无穷个公式定义。这是因为形式上来说， \bar{K} 可以被 $\bigvee_{\phi \in \Sigma} \neg\phi$ 定义，然而这是一个无穷析取，不一定能等价于某一个集合的公式。

我们来看框架类可定义性的更多例子：

例 11.10 • $\Box p \rightarrow \Diamond p$.

定义了每一个点都有后继的框架，与 $\Diamond \top$ 定义了一样的框架类。

• $\Box p \rightarrow \Box \Box p$.

定义了传递的框架类。

• $\Diamond p \rightarrow \Diamond \Diamond p$.

定义了稠密的框架类，即如果 $x \rightarrow y$ ，那么存在 z 满足 $x \rightarrow z$ 且 $z \rightarrow y$ 。

注意，如果将这些模态算子放在时序逻辑中理解，我们实际上已经得到了关于时间的公理！

注. 我们注意到，所有以上的例子，模型的结构都是可以用一阶公式描述的。

- 每个点都有后继： $\forall x \exists y (xRy)$.
- 传递： $\forall x, y, z (xRy \wedge yRz \rightarrow xRz)$.
- 稠密： $\forall x, y (xRy \rightarrow \exists z (xRz \wedge zRy))$.

将一阶公式中的变元 x, y, \dots 看成模型的点，类似模态公式，我们可以讨论一阶公式可定义模型类/框架类。

更一般地，我们可以问，给定一个点模型类 K ，是否存在模态公式（集）可以定义 K ？类似地，给定一个框架类 K ，是否存在模态公式（集）可以定义 K ？对于点模型来说，可以被模态公式集定义以及可以被模态公式定义，都有充分必要的刻画定理。对于框架来说，如果限制框架是一阶可定义的框架，我们有 *Goldblatt-Thomason* 定理，这是一个充分必要条件。

参考文献

- [Bre57] Leo Breiman. The Individual Ergodic Theorem of Information Theory. *The Annals of Mathematical Statistics*, 28(3):809–811, 1957.
- [CT12] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 2012.
- [Huf52] David A. Huffman. A Method for the Construction of Minimum-Redundancy Codes. *Proceedings of the IRE*, 40(9):1098–1101, September 1952.
- [Inf] Information | Etymology, origin and meaning of information by etymonline. <https://www.etymonline.com/word/information>.
- [Jay02] Edwin T. Jaynes. *Probability Theory: The Logic of Science*. Cambridge University Press, 2002.
- [KL51] S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.
- [LLG⁺19] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, October 2019.
- [McM53] Brockway McMillan. The Basic Theorems of Information Theory. *The Annals of Mathematical Statistics*, 24(2):196–219, June 1953.
- [RHW86] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*, pages 318–362. MIT Press, Cambridge, MA, USA, January 1986.

- [Rob49] Robert M. Fano. *The Transmission of Information*. March 1949.
- [Sha48] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, July 1948.
- [Shi96] A. N. Shiryaev. *Probability*, volume 95 of *Graduate Texts in Mathematics*. Springer, New York, NY, 1996.
- [Tin62] Hu Kuo Ting. On the Amount of Information. *Theory of Probability & Its Applications*, 7(4):439–447, January 1962.
- [Uff22] Jos Uffink. Boltzmann’s Work in Statistical Physics. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, summer 2022 edition, 2022.
- [李 10] 李贤平. 概率论基础. 高等教育出版社, 2010.

索引

Kripke 框架, 153, 154
Kripke 模型, 153, 154
Kripke 点模型, 153, 154
Kripke 语义, 154, 155

Löb 公式, 151
Löb 定理, 151

McKinsey 公式, 151

PDL, 152
 正则~, 152

三段论, 147
 强~, 147
 模态~, 148

严格蕴含, 149

元语言, 149
关系模型, 153
可满足, 154, 155
实质蕴含, 149
对象语言, 149

有效, 155, 156
框架, 153
模型验证, 155
模态可定义性, 157
 点模型可定义性, 157
模态算子, 149

模态词, 147
模态语言, 149
 基本~, 149
 多元~, 151
模态语言类型, 151
模态逻辑, 147

语义
 Kripke~, 153
 可能世界~, 153
 对象~, 153
 状态~, 153

逻辑
 动态~, 150
 可证性~, 150, 151
 命题动态~, 152
 基本模态~, 150, 156
 描述~, 150
 时序~, 150, 151, 156
 特征~, 150
 直觉主义~, 149
 联盟~, 150
 认知~, 150, 151
 道义~, 150