AI 中的数学

邓小铁　李翰禹

2024 年 10 月 1 日

目录

第零章　引言 **1**

第一部分　AI 的逻辑 **2**

第一章　合情推理 **3**

§1.1 命题逻辑的演绎推理 4

§1.2 合情推理的数学模型 12

§1.2.1 合情推理的基本假设，似然 14

§1.2.2 似然与概率 19

§1.2.3 先验与基率谬误 21

§1.3 合情推理的归纳强论证 23

§1.3.1 归纳强论证 23

§1.3.2 有效论证和归纳强论证的比较 28

§1.4 先验模型的存在性 33

§1.5 章末注记 35

§1.6 习题 36

第二章 Markov 链与模型 37
§2.1 Markov 链 38
§2.2 Markov 奖励过程（MRP） 49
§2.3 Markov 决策过程（MDP） 55
§2.4 隐 Markov 模型（HMM） 64
§2.4.1 评估问题 67
§2.4.2 解释问题 69
§2.5 扩散模型 72
§2.5.1 采样逆向过程 77
§2.5.2 训练逆向过程 78
§2.6 章末注记 81
§2.7 习题 81

第二部分 信息与数据 82

第三章 熵与 Kullback-Leibler 散度 83
§3.1 熵 84
§3.1.1 概念的导出 84
§3.1.2 概念与性质 89
§3.2 Kullback-Leibler 散度 98
§3.2.1 定义 98
§3.2.2 两个关于信息的不等式 101
§3.3 编码理论 102
§3.3.1 熵与编码 102
§3.3.2 K-L 散度、交叉熵与编码 106

§3.4 在机器学习中的应用：语言生成模型 108
§3.5 附录：Shannon 定理的证明 110
§3.6 习题 113
§3.7 章末注记 115

第四章 高维几何，Johnson-Lindenstrauss 引理 117
§4.1 高维几何 119
§4.1.1 高维球体 119
§4.1.2 Stein 悖论 123
§4.1.3 为什么我们要正则化？远有潜龙，勿用 130
§4.2 集中不等式 131
§4.3 J-L 引理的陈述与证明 138
§4.4 J-L 引理的应用 143
§4.5 习题 146
§4.6 章末注记 147

第五章 差分隐私 148
§5.1 数据隐私问题 149
§5.2 差分隐私的定义与性质 153
§5.3 差分隐私的应用 161
§5.3.1 随机反应算法 161
§5.3.2 全局灵敏度与 Laplace 机制 164
§5.3.3 DP 版本 Llyod 算法 167
§5.4 习题 170
§5.5 章末注记 170

第三部分　决策与优化 **171**

第六章　凸分析 **172**

§6.1 决策与优化的基本原理 173
§6.1.1 统计决策理论 173
§6.1.2 优化问题 176
§6.1.3 例子：网格搜索算法 183
§6.2 凸函数 188
§6.3 凸集 194
§6.3.1 基本定义和性质 195
§6.3.2 分离超平面定理 199
§6.4 习题 202
§6.5 章末注记 202

第七章　对偶理论 **203**

§7.1 约束的几何意义 206
§7.2 条件极值与 Lagrange 乘子法 214
§7.3 Karush–Kuhn–Tucker 条件 218
§7.4 Lagrange 对偶 223
§7.4.1 原始规划与对偶规划 223
§7.4.2 对偶的几何意义 228
§7.4.3 弱对偶定理 230
§7.4.4 Slater 条件，强对偶定理 231
§7.5 应用：支持向量机（SVM） 236
§7.6 习题 239
§7.7 章末注记 239

第八章 不动点理论 **240**

§8.1 Banach 不动点定理 241

§8.2 Brouwer 不动点定理 252

§8.3 习题 259

§8.4 章末注记 259

第四部分 博弈与逻辑 **260**

第九章 博弈与逻辑 **261**

§9.1 博弈的基本语言：以井字棋为例 263

§9.2 输赢博弈 265

§9.2.1 博弈的不同维度 265

§9.2.2 Zermelo 定理与 AlphaGo Zero 267

§9.3 正则形式博弈 274

§9.3.1 定义 275

§9.3.2 理性与均衡 277

§9.3.3 生成对抗网络 280

§9.3.4 混合策略 283

§9.4 随机博弈（Markov 博弈） 289

§9.5 习题 300

§9.6 章末注记 300

第五部分 认知与逻辑 **301**

第十章 共同知识，Bayes 博弈，Aumann 知识算子 **302**

§10.1 “泥泞的孩童”谜题 305

§10.2 不完全信息博弈（Bayes 博弈） 310
§10.3 电子邮件博弈 320
§10.4 Aumann 知识算子 325
§10.5 习题 334
§10.6 章末注记 334

第十一章 模态逻辑，知识的逻辑 335
§11.1 知识逻辑的形式语言 337
§11.2 Kripke 语义 342
§11.3 模态可定义性 349
§11.4 知识逻辑的基本模型与性质 352
§11.4.1 知识逻辑的 Kripke 模型与公理 352
§11.4.2 Kripke 模型与 Aumann 结构 360
§11.4.3 “泥泞的孩童”再回顾：形式化解法 361
§11.5 对不一致达成一致 364
§11.5.1 模型 364
§11.5.2 定理及其证明 368
§11.6 习题 370
§11.7 章末注记 370

第六部分 附录：预备知识 371

附录 A 线性代数基础 371
§A.1 线性空间 371
§A.2 线性映射 378
§A.3 矩阵 384

§A.4 双线性型与二次型 393
§A.5 带内积的线性空间 399
§A.6 行列式 408
§A.7 算子范数与谱理论 412

附录 B 微分学基础 420

§B.1 点集拓扑 420
§B.1.1 度量空间，范数 420
§B.1.2 开集与闭集 425
§B.1.3 紧致性，收敛性，完备性 429
§B.1.4 连续映射 433
§B.1.5 与实数序有关的性质 438
§B.2 一元函数的微分学 441
§B.2.1 导数与微分的定义 442
§B.2.2 微分学基本定理 447
§B.3 多元函数的微分学 449
§B.3.1 微分、偏导数与导数的定义 449
§B.3.2 微分学基本定理 459
§B.3.3 隐函数定理 462

附录 C 概率论基础 468

§C.1 从朴素概率论到公理化概率论 468
§C.1.1 Kolmogorov 概率论 468
§C.1.2 条件概率，独立性 474
§C.2 随机变量，分布函数 480
§C.2.1 基本定义 480

§C.2.2 离散型随机变量 486
§C.2.3 连续型随机变量 487
§C.2.4 随机向量，条件分布，独立性 492
§C.2.5 随机变量（向量）的函数 499
§C.3 随机变量的数字特征，条件数学期望 503
§C.3.1 数学期望，Lebesgue 积分 503
§C.3.2 数学期望的性质 510
§C.3.3 随机变量的内积空间 514
§C.3.4 特征函数 517
§C.3.5 条件数学期望 519
§C.4 多元正态分布（Gauss 向量） 526
§C.5 大数定律 528

第一部分

AI 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

博弈与逻辑

第五部分

认知与逻辑

第十一章　模态逻辑，知识的逻辑

你听说过魔芋与高僧的故事吗？相传，这个有趣的故事发生在日本的一座深山之中．在那幽静的山谷中，有一座古老而荒废的寺庙，几百年来一直无人问津，只有密林与野兽相伴．

一天，一个以卖魔芋为生的流浪小贩推着他的木车，偶然来到这片偏僻的地方．他发现这座破旧的寺庙中竟然无人居住，便心想："如此偏僻之地，不如就在此安顿下来吧．"于是，他便住了下来，每天挑着魔芋，走村串户，卖给山中偶尔路过的行人．

不久之后，一位云游四方的僧人也来到了这片深山．看到这座荒废的寺庙，他心生好奇，便前去查看．僧人惊讶地发现寺庙中竟然住着一位老人，便以为这是位得道高僧．僧人心中一动，想着既然能在如此偏远之地修行，必定有过人之处，于是便恭敬地上前，请教佛法．

在佛教传统中，有一种不动声色的辩经方式，那就是只用手势来比划，而不发一言．僧人遵循这一传统，开始用手势向卖魔芋的人比划起来．然而，卖魔芋的人哪里知道这些佛教传统？他误以为僧人是在挑剔

他卖的魔芋不好，来讨说法的. 他心中不悦，但还是用手势回应僧人，心想着:“我卖的魔芋可是顶好的，若是敢挑衅，我便和你理论理论 !”

于是，两人各怀心事，展开了一场激烈的手势比划. 僧人暗自佩服卖魔芋的人的沉稳与深不可测的手势，心想自己在佛法修行上还有很大的差距；而卖魔芋的人则越比越得意，心中大呼:“这人还真有些手段，但我魔芋的品质岂是一般人能质疑的?”

比划结束后，僧人心生敬意，深感不如；卖魔芋的人则自信满满，觉得自己的魔芋赢得了这场对峙. 因此，“卖魔芋的人获胜”这件事成了他们二人的共同知识. 然而，两人心中所思所想却完全不同 ! 僧人认为自己遇到了高人，而卖魔芋的人则确信自己击退了一个挑刺的顾客.

在第十章中，我们探讨了知识的一些基本性质. 上面的魔芋故事实际上揭示了两件重要的事情:

- 一方面，知识（尤其是共同知识）要比我们想象的复杂得多. 在这个例子中，两个人知道的“事实”究竟是不是同样的呢? 这个问题并不容易回答.

- 另一方面，知识的传递与理解可以超越自然语言. 在这个例子中，两人完全没有用语言交流，而是依靠手势. 这说明知识可以有不同于自然语言的表达方式.

在本章中，第二点将是我们的讨论重点. 它意味着我们可以用形式逻辑来描述知识的概念和性质. 这种形式逻辑在现代逻辑学中被归类为*模态逻辑*. 模态逻辑是一种扩展了经典形式逻辑（如命题逻辑和一阶逻辑）的逻辑，通过引入*模态词*——例如“必然”、“可能”、“知道”等——来描述命题的性质. 最后，作为一个例子，我们将使用模态逻辑的语言表述并证明 Robert Aumann 的“对不一致达成一致”定理.

§11.1 知识逻辑的形式语言

模态词是指说话者对语句的限定. 不同的限定反映了不同的逻辑状态. 例如：必然、可能、过去、未来、知识、信念和可证明等，都是典型的模态概念. 比如，我们可以说“明天可能会下雨”、“我知道明天会下雨”、“我相信明天会下雨”、“明天可能会下雨”等等，他们要表达的意思并不是经典逻辑可以表述的.

在经典逻辑中，我们只能表达“明天下雨”或者“明天不下雨”这样的命题. 因而，模态逻辑提供了一套严密的数学工具，可以把模态的概念从自然语言中“搬到”形式语言中.

因为模态词往往都与人的认知和思维有关，因此，通过模态逻辑，我们可以算法化、自动化地模拟和推理人类的思维和认知过程. 这正是人工智能中符号主义的思想.

在研究模态逻辑时，重要的一部分是研究它的模型论，即如何定义模态逻辑的语言、然后赋予这些形式语言具体的含义与真假，即语义.

模型论的角度看，一个逻辑系统包括三个要素：语言、模型、语义. 他们分别代表，我怎么说话、我关心的对象是什么、我说的话和我关心的东西之间有什么关系. 接下来，我们以命题逻辑为例，介绍三要素.

例 11.1 (命题逻辑的逻辑三要素) 命题逻辑是由命题字母、逻辑联结词和括号组成的形式语言. 例如，$p \to q$ 这样的东西属于命题逻辑，但是 $\forall x\, P(x)$ 这样的东西不属于命题逻辑（而属于一阶逻辑）. 关于命题逻辑更系统的讨论，请参阅第一章.

命题逻辑的三要素是：

- 语言（我怎么说话）：用规则写成的字符串，例如我们可以用字符串 $p \to q$ 表示一个逻辑公式，但是不能写 $\forall x\, P(x)$、$2+2=4$ 或

者 $p \vee \vee q$. 我们可以简单理解成，我们说普通话的时候，不能说英语，也不能说语法错误的句子（如“我是是”）.

- 模型（我们关心的对象是什么）：我们关心的世界结构，在命题逻辑中就是“真”和“假”. 因此，在命题逻辑的世界里，没有实数，也没有人类，更没有诗和远方；只有冰冷（但是精确）的“真”和“假”.

- 语义（我说的话和我关心的东西之间有什么关系）：我们可以先给命题字母真假的概念，例如 p 赋值为“真”，q 赋值为“假”. 当每一个命题字母都被赋予了真假之后，我们就可以定义整个公式的真假. 例如，$p \to q$ 为真当且仅当 p 为假或者 q 为真. □

尽管在第一章中，我们也简要介绍了逻辑三要素. 但是，我们在这里才将这样的讨论展开. 这是因为，只有到了模态逻辑的世界，逻辑三要素才变得如此不平凡，我们才能真正体会到他们的意义. 接下来，我们将介绍模态逻辑的语言、模型和语义. 本节先介绍模态逻辑的语言.

首先，我们只考虑最简单的情况，基础语言是命题逻辑，然后在其中加入一个*模态算子*.

定义 11.1 (基本模态语言) 给定命题字母表 $\mathbf{P}$，我们定义它的**基本模态语言** L，按照如下方式递归生成：

- 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. □

和命题逻辑比较，我们多了一个模态算子 $\Box$，它读作“Box”，对应的自然语言可以读为“必然”（更多讨论见本部分后文）.

我们在后面将会频繁定义各种不同的模态语言，为了方便，我们引入一种更简洁的记号：

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

这种记号被称为 *Backus-Naur* 范式（*BNF*），是一种用来描述形式语言的标准记号，在编程语言设计和编译器的实现中也经常使用.

类似命题逻辑，我们有如下缩写：

- $\phi \vee \psi \iff \neg(\neg\phi \wedge \neg\psi)$.
- $\phi \to \psi \iff \neg\phi \vee \psi$.
- $\bot \iff \neg\top$.

引入模态算子 $\Box$ 之后，我们还有它对应的对偶算子 $\Diamond$，它是如下的缩写：

$$\Diamond\phi \iff \neg\Box\neg\phi.$$

$\Diamond$ 读作“diamond”，对应的自然语言可以读为“可能”. 为什么它的自然语言解释是“可能”呢？因为，“不是必然不”的意思就是“可能”. $\Box$ 和 $\Diamond$ 的对偶性，我们可以类比 $\exists$ 和 $\forall$ 的对偶性，这是因为 $\exists$ 也可以写成 $\neg\forall\neg$. 我们将在第 11.2 节中看到，这一类比其实有极其自然的模型论解释.

既然可以加入一个模态算子，我们也可以加入更多的模态算子，一个模态算子也可以修饰多个公式. 例如，我们可以引入一个新的模态算子 $\Box_a$，表示个体 a 认为的必然性. 于是，我们可以写出这样的公式：

$$\Box_a\phi \wedge \Box_b\psi \to \Box_a(\phi \wedge \psi).$$

我们也可以引入模态算子 $\nabla(\phi,\psi)$，表示 ϕ 成立的时候，ψ 必然成立. 于是，我们可以写出这样的公式：

$$\nabla(\phi \to \psi, \Box\phi) \to \Box\psi.$$

通过引入不同的模态算子来描述不同的对象，模态逻辑被赋予了这样的哲学：多视角下看同一个数学概念. 比如，我们可以把 $\Box$ 在自然语言中用不同的词来解释，由此得到不同的模态逻辑：

- 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
- 知识逻辑：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

接下来，我们具体看两个例子，这是本节最关心的两种解释：*基本模态逻辑*和*知识逻辑*.

例 11.2 (基本模态逻辑) 我们可以把模态算子 $\Box$ 读成“必然”. 于是，

- $\Box\phi$ 表示“必然有 ϕ”.
- $\Diamond\phi$ 表示“不是必然有非 ϕ”，即“可能有 ϕ”，所以 $\Diamond$ 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.

在这个读法下，我们可以用形式语言去表达一些自然语言中很拗口的句子，但更加清晰和精确. 比如：

- $\Box p \to \Diamond p$：必然的事也是可能的.
- $p \to \Box p$：真的事是必然的.
- $\Diamond p \to \Box\Diamond p$：可能的事是必然可能的. □

例 11.3 (知识逻辑) 在知识逻辑中，我们可以把模态算子 $\Box$ 读成“知道”，并写成 K（know）. K 表示某个特定的个体对世界的认知. 例如：

- $K\phi$（即 $\Box\phi$）：我知道 ϕ.
- $K\phi \to \phi$：如果我知道 ϕ，那么 ϕ 是真的.
- $\phi \to K\phi$：如果 ϕ 是真的，那么我知道 ϕ.
- $\neg K\phi$ vs. $K(\neg\phi)$：我不知道上帝存在 vs. 我知道上帝不存在. 这两句话的含义是不同的，因此，模态词使得否定的含义变得复杂.

在更一般的情况下，我们会有多个个体，于是可以用 K_a 表示“个体 a 知道”. 同样，我们可以用 B_a 表示“个体 a 相信”. 这里是一些例子：

- $K_aK_b\phi \leftrightarrow K_bK_a\phi$：我知道你知道 ϕ 当且仅当你知道我知道 ϕ.
- $K_1K_2p \wedge \neg K_2K_1K_2p$：1 知道 2 知道 p，但是 2 并不知道 1 知道 2 知道 p.
- $\neg K_ip \to K_i(\neg K_ip)$：如果我不知道 p，那么我知道我不知道 p.

- $K_i(p \wedge \neg K_i p)$：我知道如下的陈述：p 是真的，且我不知道 p. 一种类似的写法是，$K_i p \wedge K_i \neg K_i p$，即我知道 p，但是我又知道我不知道 p.
- 共同知识算子[1] C：$C\phi$ 当且仅当 $K_a(\phi \wedge C\phi)$ 对任意 a 成立. 注意，$C\phi$ 并不等价于对任意 a，$K_a\phi$ 成立. □

§11.2 Kripke 语义

接下来我们讨论模态语言对应的模型，以及模态逻辑语言在这一模型上的语义. 从本节开始，我们将模态算子限制为一元算子，即它只能修饰一个公式.

我们考虑的模型被称为 *Kripke（点）模型*. 它可以看作是一个带有标记的有向边和节点的图：

- 节点表示可能世界，上面用命题字母标记，表示这个可能世界上成立的原子命题；
- 边表示节点之间的关系，用模态算子标记，表示这两个可能世界之间的关系. 如果只有一个模态算子，我们省略模态算子标记，只写箭头.
- 我们有一个指定的节点，作为真实世界.

我们来看一个例子.

例 11.4 我们可以用下面的图 11.1 来表示一个 Kripke 模型.

[1] 这个算子的定义以及性质在第 11.4 节中会详细讨论，这里仅作为一个例子，读者不必理解具体的意义.

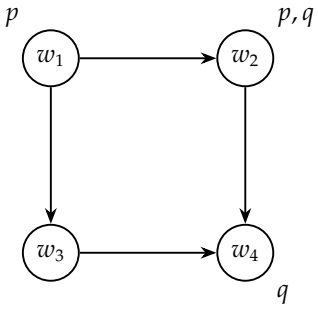

图 11.1: Kripke 模型的示例

模型中，一共有四个可能世界，用 w_1, w_2, w_3, w_4 表示. 他们之间的关系用箭头表示，例如 $w_1 \to w_2$ 表示 w_1 到 w_2 有模态算子 $\Box$ 对应的关系. 我们可以把箭头更直观地理解为，从 w_1 这个世界的视角看，他能够想象一个 w_2 的可能世界.

我们还要指定一个节点作为真实世界，这里我们指定 w_1 作为真实世界. □

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”. 接下来，我们把模态的语言和模型联系起来，定义模态逻辑的语义，完成模态逻辑三要素的定义.

设想，在真实世界 w_1，我们说 $\Box p$ 是真的，换言之，“必然有 p”. 当我们在说这句话的时候，我们其实在脑中想象了所有能想到的可能世界，然后发现在所有可能世界中，p 都是真的，于是，我们才说，“必然有 p”. 这就是模态逻辑的语义.

形式化地说，$\Box\phi$ 在世界 w 上成立当且仅当 ϕ 在 w 的所有后继上为真. 这种语义通常被称为 *Kripke 语义*或可能世界语义. 在一个世界讨论可能与必然的时候，会取决于它与其他世界的联系.

我们再来看一个例子.

例 11.5 考虑图 11.1 中的 Kripke 模型. 对哪些 i 来说，下式成立?

$$\Box(p \to \Box q).$$

我们以 w_1 为例进行讨论. 其他情况非常类似，读者可以自行验证. 如果 w_1 是真实世界，上要成立上式，必须要所有后继节点上成立

$$p \to \Box q.$$

p 的后继节点有 w_2, w_3，分别考虑这两个节点:

- 对于 w_2，p 为真，所以需要看 $\Box q$ 是否成立. w_2 的后继节点只有 w_4. 注意到，w_4 上 q 为真，因此 w_2 上 $\Box q$ 成立，所以 w_2 上 $p \to \Box q$ 也成立.
- 对于 w_3，p 为假，所以 $p \to \Box q$ 自动成立.

因此，w_1 上 $\Box(p \to \Box q)$ 成立. 同理，我们可以验证其他节点上的情况，得到在 w_2，w_3，w_4 上，$\Box(p \to \Box q)$ 都成立.

特别注意，在验证 w_4 的时候，因为 w_4 没有后继节点，所以任何命题 ϕ，$\Box\phi$ 都是成立的. □

游了对概念的自然语言描述和例子，接下来，我们形式上给出模态逻辑的模型和语义定义.

首先，定义一个 *Kripke 框架*，它单纯描述可能世界及其之间的关系，不考虑可能世界上有哪些命题成立.

定义 11.2 (Kripke 框架) 考虑基本命题模态逻辑 L. 一个 **Kripke 框架**是一个元组 $\mathcal{F} = (W, R)$，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）. □

接下来，对每一个可能世界，我们赋予它为真的原子命题，如此得到了一个 *Kripke 模型*.

定义 11.3 (Kripke 模型) 一个 **Kripke 模型** $\mathcal{M}$ 是一个元组 $(\mathcal{F}, V)$，其中 $\mathcal{F}$ 是 Kripke 框架，$V : \mathbf{W} \to 2^{\mathbf{P}}$ 是**赋值函数**，表示每个可能世界上为真的那些命题字母（即原子命题）. □

Kripke 模型并没有指定一个真实世界，接下来，我们引入一个指定的点，作为真实世界，得到 *Kripke 点模型*.

定义 11.4 (Kripke 点模型) 一个 **Kripke 点模型** $(\mathcal{M}, w)$ 是 Kripke 模型 $\mathcal{M}$ 加上一个指定的点 $w \in W$. □

现在，我们可以定义模态逻辑的语义，即 *Kripke 语义*.

定义 11.5 (Kripke 语义) 考虑基本命题模态逻辑 L. 符号 $\mathcal{M}, w \vDash \phi$ 表示 ϕ 在点模型 $\mathcal{M}, w$ 是**可满足的**. 这一概念可以递归定义如下：

- $\mathcal{M}, w \vDash \top$ 永远成立.
- $\mathcal{M}, w \vDash p$ 当且仅当 $p \in V(w)$.
- $\mathcal{M}, w \vDash (\phi \wedge \psi)$ 当且仅当 $\mathcal{M}, w \vDash \phi$ 且 $\mathcal{M}, w \vDash \psi$.
- $\mathcal{M}, w \vDash \neg\phi$ 当且仅当 $\mathcal{M}, w \nvDash \phi$.

- $\mathcal{M}, w \vDash \Box\phi$ 当且仅当对所有 v，如果 wRv，那么 $\mathcal{M}, v \vDash \phi$.

因此，$\mathcal{M}, w \vDash \Diamond\phi$ 当且仅当存在 v 满足 wRv 且 $\mathcal{M}, v \vDash \phi$. □

注意，上面的定义都是在只有一个模态算子的情况下定义的. 不过，我们可以很容易地推广到多个一元模态算子的情况. 假设模态算子是 $\Box_a$，那么 Kripke 框架中的边需要附上标签 a，表明这个边是 $\Box_a$ 对应的关系，即 $w \to_a v$. 在这种情况下，$\Box_a\phi$ 的语义定义为：对所有 v，如果 wR_av，那么 $\mathcal{M}, v \vDash \phi$.

多个一元模态算子在知识逻辑中是很常见的. 每个人都有自己对于世界的认知，因此每个人 a 在模型中会有一个对应的关系 R_a，以描述他所认为的可能世界架构，R_a 对应的 $\Box$ 模态算子写作 K_a.

现在，我们进一步讨论语义的性质. 逻辑公式的语义，无论如何定义，其实都在讨论“什么是真的”这一基本问题. 在模态逻辑中，这一概念尤其复杂.

比如，在例 11.5 中，我们讨论了 $\phi : \Box(p \to \Box q)$ 在一个 Kripke 模型中每个可能世界上的真值. 在例子的计算中，我们发现，所有可能世界上 ϕ 都是真的. 所以，我们其实可以说，ϕ 在整个 Kripke 模型上也是真的.

更进一步，我们也想知道一些关于可能和必然的真理. 为了说明这一点，先回到命题逻辑中，考虑任意命题字母 p，无论 p 是否是真的，下面这个公式永远都是真的：

$$p \to p.$$

我们在命题逻辑中称这样的公式为重言式. 重言式其实反映了命题逻辑的一些根本性质，这些性质独立于具体的命题而存在. 比如排中律：对

于任意命题 p，$p \vee \neg p$ 是重言式，再比如三段论：$((p \to q) \wedge p) \to q$ 是重言式. 他们是我们做逻辑推理的时候自动会假设的性质.

同样的问题也可以对模态逻辑提出：有没有一些关于可能和必然的性质，是独立于具体的可能世界而存在的？比如，我们是否可以说，如果 p 是真的，那么 p 必然是真的？把它写成模态逻辑的形式就是：

$$p \to \Box p.$$

如果这是一个关于必然的真理，那么，无论 p 是否为真，无论我们处于什么样的可能世界，这个公式都应该是真的.

根据上面的启发，在模态逻辑的体系中，我们可以定义各种不同粒度的“真”的概念：

定义 11.6 (模态逻辑的真值) 给定模态逻辑公式 ϕ，我们可以定义它的真值：

- ϕ 在点模型 $\mathcal{M}, w$ 可满足指的是 $\mathcal{M}, w \vDash \phi$.
- ϕ 在模型 $\mathcal{M}$ 有效，记为 $\mathcal{M} \vDash \phi$，指的是 $\mathcal{M}, w \vDash \phi$ 对所有 w 成立.
- ϕ 在点框架[2] $\mathcal{F}, w$ 有效，记为 $\mathcal{F}, w \vDash \phi$，指的是 $\mathcal{M}, w \vDash \phi$ 对所有基于 $\mathcal{F}$ 的模型 $\mathcal{M}$ 成立.
- ϕ 在框架 $\mathcal{F}$ 有效，记为 $\mathcal{F} \vDash \phi$，指的是 $\mathcal{M} \vDash \phi$ 对所有基于 $\mathcal{F}$ 的模型 $\mathcal{M}$ 成立.

[2] 我们在前文中没有定义点框架，但是点框架的定义相当直接：它是一个 Kripke 框架加上一个指定的点.

- ϕ 对框架类（即框架的一个集合）K 有效，记为 $\vDash_K \phi$，指的是 $\mathcal{F} \vDash \phi$ 对所有 $\mathcal{F} \in K$ 成立. □

我们可以看到，模态逻辑的真值有两个维度的粒度：局部-全局，模型-框架. 越靠左边的真值越具体，越靠右边的真值越一般. 我们可以用下面的表格来总结这些概念：

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

我们主要讨论高亮的两个部分.

例 11.5 其实给出了左上角的一个例子，下面，我们可以演示右下角的一个例子.

例 11.6 考虑图 11.2 中的框架，它是图 11.1 中的 Kripke 模型去掉了命题字母的结果.

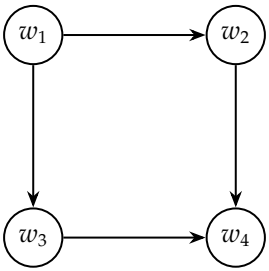

图 11.2: 框架语义的示例

我们问，对于这个框架，下面的公式是否有效?

$$\Box(p \to \Box p).$$

为此，我们需要考虑四个可能世界上的情况. 同样，我们只对 w_1 进行讨论，其他情况类似. 对于 w_1，有两个后继节点 w_2，w_3，我们需要验证 $p \to \Box p$ 在这两个节点上是否成立.

先考虑 w_2，假设 p 在 w_2 上为真，我们需要验证 $\Box p$ 是否成立. w_2 的后继节点只有 w_4，但我们可以让 p 在 w_4 上为假，这样 $\Box p$ 就不成立. 因此，w_2 上 $p \to \Box p$ 也不成立，从而 w_1 上 $\Box(p \to \Box p)$ 不成立.

于是，我们找到了一种赋值的方案，使得 $\Box(p \to \Box p)$ 在 w_1 上不成立. 因此，这个公式在整个框架上也不有效. □

最后，我们说明，上面的讨论在知识逻辑中都是自然成立的. 此时，模态公式 $K_i\phi$ 被读作"i 知道 ϕ". 从语义来说，K_i 是 $\Box$ 算子，即我知道 ϕ 意味着在我认为的所有可能世界中 ϕ 都是真的，即：

$$\mathcal{M}, w \vDash K_i\phi \iff \text{对任意 } v\text{，如果 } w \to_i v\text{，那么 } \mathcal{M}, v \vDash \phi.$$

同样，模态公式的真值有两个层面：

- 在点模型上可满足：$\mathcal{M}, w \vDash \phi$；
- 在框架上有效：$\mathcal{F} \vDash \phi$.

§11.3 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：一是通过 Kripke 模型（框架）的特殊结构，二是通过具体的模态公式. 那么，这两种方式之间有什么联系呢？本节我们研究这个问题.

先看几个例子.

例 11.7 虑知识逻辑，假设只有一个知识算子 K. 设有命题 p，如果 p 是真的，那么我不知道非 p，即 $\phi : p \to \neg K \neg p$. 这是用逻辑表述了关于知道的一种性质.

另一方面，我们也可以将对于 K 的理解反映到 Kripke 模型中. 一个合理的性质是：真实世界是我能够感知到的可能世界. 换言之，对任何 x，都有 xRx，这是一个自反关系. 在这种情况下，容易验证，对于自反点模型以及框架 $\mathcal{F}$，我们有 $\mathcal{M}, v \vDash \phi$ 以及 $\mathcal{F} \vDash \phi$. □

上面的例子说明，对于特定的公式，它会在某类模型上成立. 反过来，我们也可以讨论，成立特定公式的模型是什么样的，我们看下面的例子.

例 11.8 考虑基础模态逻辑和它的公式 $\Diamond \top$. 对于一个 Kripke 点模型 $\mathcal{M}, w$，假设 $\mathcal{M}, w \vDash \Diamond \top$，我们来研究 $\mathcal{M}$ 的特殊性质.

根据定义，这意味着存在一个点 v，使得 wRv 并且 $\mathcal{M}, v \vDash \top$，由于后一半永远成立，所以，这等价于 w 有一个后继. 同样的，对于任意框架 $\mathcal{F}$，如果 $\mathcal{F} \vDash \Diamond \top$，则基于 $\mathcal{F}$ 的每个点模型 $\mathcal{M}, v$ 都满足 $\Diamond \top$，即 $\mathcal{F}$ 的每个点都有后继. □

上面两个例子说明了模型性质和模态公式之间单向的联系，下面的例子说明，这种联系也可以是双向的.

例 11.9 考虑公式 $\phi : p \to \Diamond p$. 我们研究和这个公式完全对应的框架 $\mathcal{F}$ 的性质.

- 任给一个赋值 V 和点 w 都有 $\mathcal{M}, w \vDash p \to \Diamond p$. 考虑一个赋值 V，使得只有 w 上有 p. 因为 w 上有 p，为了使 ϕ 成立，w 必须要有

一个后继上也有 p. 在我们的这个赋值下，这意味着 w 必须以自己为后继. 因此，对任意 w，wRw 成立，即 $\mathcal{F}$ 是一个自反框架.

- 反过来，如果 $\mathcal{F}$ 是一个自反框架，那么对于任意赋值 V 和点 w，容易验证 $\mathcal{M}, w \vDash p \to \Diamond p$. 因此，$\mathcal{F} \vDash p \to \Diamond p$.

结合这两点，我们可以说，$p \to \Diamond p$ 定义了自反框架.

此外，结合这个例子与例 11.8，我们会发现，一个框架的特定性质可以由多个不同的公式定义. □

以上例子给我们启示，我们可以用模态公式去定义模型或框架的性质. 这种定义被称为*模态可定义性*，我们给出如下定义.

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 11.7 (点模型可定义性) 设 $\mathcal{K}$ 是一些点模型的集合，Σ 是一些模态公式的集合. 我们说 $\mathcal{K}$ **可由公式集 Σ 定义**，指的是对于任意点模型 $\mathcal{M}, w$，$\mathcal{M}, w \in \mathcal{K}$ 当且仅当对任意 $\phi \in \Sigma$，

$$\mathcal{M}, w \vDash \phi.$$

如果 $\Sigma = \{\phi\}$，我们就说 $\mathcal{K}$ **可以由公式 ϕ 定义**. □

我们也可以从框架的角度讨论模态可定义性，定义类似.

定义 11.8 (框架可定义性) 设 $\mathcal{K}$ 是一些框架的集合，Σ 是一些模态公式的集合. 我们说 $\mathcal{K}$ **可由公式集 Σ 定义**，指的是对于任意框架 $\mathcal{F}$，$\mathcal{F} \in \mathcal{K}$ 当且仅当对任意 $\phi \in \Sigma$，

$$\mathcal{F} \vDash \phi.$$

如果 $\Sigma = \{\phi\}$，我们就说 $\mathcal{K}$ **可以由公式 ϕ 定义**. □

§11.4 知识逻辑的基本模型与性质

接下来，我们将注意力放在知识逻辑上，将前面若干小节讨论的概念应用到知识逻辑中.

§11.4.1 知识逻辑的 Kripke 模型与公理

首先，从模型的角度看，一个自然的假设是，对每个个体 i 来说，他认为的可能世界之间应该是不可区分的. 这意味着，对于每个个体 i，他认为的可能世界之间的关系 R_i 应该是等价关系，我们记作 $\sim_i$：

- 自反：$\forall x, x \sim_i x$. 我能够想象到真实的世界.
- 传递：$\forall x, y, z(x \sim_i y \wedge y \sim_i z) \to x \sim_i z$. 如果我在 x 世界能够想象到 y 世界，在 y 世界能够想象到 z 世界，那么我在 x 世界也能够想象到 z 世界.
- 对称：$\forall x, y(x \sim_i y \leftrightarrow y \sim_i x)$. 我能够在 x 世界想象到 y 世界，那么我也能够在 y 世界想象到 x 世界.

我们将 R_i 是等价关系的框架类记为 H.

从模态可定义性的角度来说，$\sim_i$ 的特殊性质会对应 K_i 特殊的公式. 这些公式就可以被看成关于“知道”的公理或推导规则. 承认某一条公理或推导规则就必须承认可能世界具有某一种性质，反之亦然.

接下来，我们给出知识逻辑的基本性质，即公理.

命题 11.1 (分配公理)

$$\vDash (K_i(\phi \to \psi) \wedge K_i\phi) \to K_i\psi.$$

证明. 假设 $\mathcal{M}, w \vDash K_i(\phi \to \psi)$ 且 $\mathcal{M}, w \vDash K_i\phi$. 于是，对所有 R_i 后继 v 都有 $\mathcal{M}, v \vDash \phi \to \psi$ 和 $\mathcal{M}, v \vDash \phi$，因而 $\mathcal{M}, v \vDash \psi$. 根据定义，$\mathcal{M}, w \vDash K_i\psi$，因而对所有 $\mathcal{F}$，分配公理有效. □

分配公理意味着拥有知识的个体可以对自己的知识做任意的演绎推理，因而假设个体是*逻辑全知的*. 注意，分配公理并没有用到框架类 H 的性质，所以，它其实是模态逻辑的一个普适公理.

命题 11.2 (泛化规则) *对所有 $\mathcal{F}$，如果 $\mathcal{F} \vDash \phi$，那么 $\mathcal{F} \vDash K_i\phi$.*

证明. 假设 $\mathcal{F} \vDash \phi$，这意味着对所有基于 $\mathcal{F}$ 的点模型都有 $\mathcal{M}, w \vDash \phi$. 因此，对任意 w 的 R_i 后继 v，也有 $\mathcal{M}, v \vDash \phi$. 所以也有 $\mathcal{M}, w \vDash K_i\phi$ 成立，因而 $\mathcal{F} \vDash K_i\phi$. □

泛化规则可能不太容易理解，我们举一个例子. 生活在中国，我们都知道农历五月初五是端午节，“五月五，迎端午”，按照传统习俗要吃粽子. 这意味着，在我们所有对未来生活的设想中，“农历五月五要吃粽子”这一习俗是牢固存在的. 因此，对于中国人来说，我们身处的世界以及幻想的可能世界组成了框架 $\mathcal{F}$，而“农历五月五吃粽子”是这个框架中的一个普适规律.

然而，如果我们生活在冰岛，一个中国人非常少的国家，那么，“农历五月五要吃粽子”这一习俗可能就不再广为流传. 在这种情况下，我们所处的世界及想象的可能世界组成了另一个框架 $\mathcal{F}'$，在这个框架中，“农历五月五要吃粽子”不再是普适规律.

所以，泛化规则其实在说，如果 ϕ 是 $\mathcal{F}$ 普遍适用的规律，不依赖具体的命题，那么我就知道 ϕ. 当然，这个规律 ϕ 可能只适用于 $\mathcal{F}$. 但无论如何，我清晰地知道这个世界的运作规律. 特别地，我知道关于知识的这些规律.

同样，泛化规则也没有使用框架类 H 的性质，所以它也是模态逻辑的普适规则.

命题 11.3 (知识公理，真理公理)

$$\vDash_H K_i\phi \to \phi.$$

注意，从知识公理开始，我们就开始使用了框架类 H 的性质. 从这里开始的性质，和第十章中讨论的知识的性质是一致的，所以这里不再赘述. 他们的证明类似前述性质，这里不再给出，见习题[lhy: 出一下].

命题 11.4 (内省公理) 正内省公理:

$$\vDash_H K_i\phi \to K_iK_i\phi.$$

负内省公理:

$$\vDash_H \neg K_i\phi \to K_i\neg K_i\phi.$$

以上五条性质（四条公理 + 一条推导规则）加上 MP 形成的推理系统称为 *S5 公理系统*. 需要注意的是，这些公理其实都是公理模式，对每一个具体的命题 ϕ，都有一条公理，因此，它其实包含了无穷条公理.

从哲学的角度讨论，还有一些别的公理.

命题 11.5 (一致性公理)

$$\vDash_H \neg K_i\bot.$$

因此，个体不能够知道假的陈述，以此区别于信念.

我们基于框架类 H 给出了关于知识的公理. 反过来，公理对应什么样的框架结构呢？我们总结于下表:

公理	R_i 的性质
$K_i\varphi \to \varphi$	自反性
$K_i\varphi \to K_iK_i\varphi$	传递性
$\neg K_i\varphi \to K_i\neg K_i\varphi$	Euclid 性
$\neg K_i\bot$	序列性
$\varphi \to K_i\neg K_i\neg\varphi$	对称性

表中有两个性质是我们没有讨论过的，这里给一个简短的解释.

- Euclid 性：$\forall x,y,z(xR_iy \wedge xR_iz \to yR_iz)$，也就是说，关系一定形成三角形.
- 序列性：$\forall x\exists y\, xR_iy$，所有点都有后继.

以上的模态可定义性和第 11.3 节中的例子类似，这里不再给出，请见习题[lhy: 出一下].

可以看出来，以上关系其实并不是孤立的，我们有以下引理：

引理 11.1 • 如果 R_i 是自反和 *Euclid* 的，那么 R_i 是对称和传递的.

- 如果 R_i 是对称和传递的，那么 R_i 是 *Euclid* 的.
- 以下命题等价：
  - R_i 是自反、对称和传递的.
  - R_i 是对称、传递和序列的.
  - R_i 是自反和 *Euclid* 的.

这一性质的验证根据定义即可，见习题[lhy: 出一下].

下面我们向知识逻辑语言中加入共同知识算子和它的语义. 首先加入“所有人都知道”这个算子:

$$E\phi \leftrightarrow \bigwedge_i K_i\phi.$$

记 $E^k\phi$ 为 $\underbrace{E\dots E}_{k}\phi$. 于是，我们可以给共同知识算子一个语义定义:

定义 11.9 (共同知识算子) **共同知识算子** C 的语义定义为:

$$\mathcal{M}, w \vDash C\phi \iff \mathcal{M}, w \vDash E^k\phi, \quad k = 1, 2, \dots \qquad \square$$

我们可以从图结构来理解共同知识算子. 把所有边上的标记忽略掉，Kripke 模型变成一个有向图，节点上的标记是这个节点上成立的原子命题. 于是，我们有如下解读:

- $\mathcal{M}, w \vDash E^k\phi$ 的含义是，从 w 出发走恰好 k 步可到达的可能世界 v 上都有 $\mathcal{M}, v \vDash \phi$. 这一解读的示意见图 11.3，

  在这个图中，$\mathcal{M}, w_1 \vDash E^3\phi$，因为从 w_1 出发，经过三步可以到达的可能世界上都有 $\mathcal{M}, v \vDash \phi$，我们具体地标出了两条图路径: $w_1 \to_1 w_2 \to_3 w_3 \to_1 w_5$ 对应 $K_1K_3K_1\phi$，$w_1 \to_2 w_4 \to_2 w_3 \to_2 w_6$ 对应 $K_2K_2K_2\phi$.

- 因此，$\mathcal{M}, w \vDash C\phi$ 的含义便是，从 w 出发经过有限步可到达的可能世界 v 上都有 $\mathcal{M}, v \vDash \phi$.

接下来，我们给出共同知识算子的性质. 实际上，共同知识算子可以充分必要地被下面两个公理所定义（验证见习题[lhy: 出一下]）:

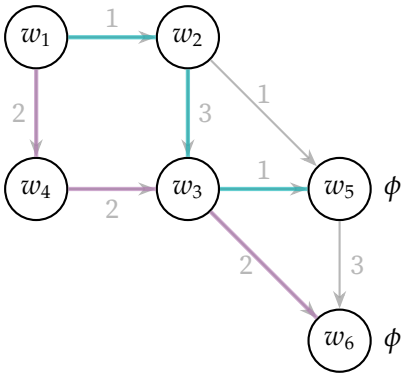

图 11.3: $\mathcal{M}, w_1 \vDash E^3\phi$ 的示意图，我们标出了其中的两条路径.

命题 11.6 (不动点公理)

$$\vDash C\phi \leftrightarrow E(\phi \wedge C\phi).$$

命题 11.7 (归纳规则) 如果

$$\mathcal{F} \vDash \phi \rightarrow E(\phi \wedge \psi),$$

那么

$$\mathcal{F} \vDash \phi \rightarrow C\psi.$$

现在我们来讲解上面两个公理的含义.

不动点公理是比较直接的. 什么是共同知识？对任意 k，$E^k\phi$ 成立. 我们把不动点公理展开，可以得到：

$$C\phi \leftrightarrow E(\phi \wedge C\phi)$$

$$\begin{aligned}
&\leftrightarrow E\phi \wedge E(C\phi)\\
&\leftrightarrow E\phi \wedge E(E\phi \wedge E(C\phi))\\
&\leftrightarrow E\phi \wedge E^2\phi \wedge E^3(C\phi)\\
&\cdots
\end{aligned}$$

因为逻辑公式不允许写无穷长的公式，所以，不动点公理就是在用一个算子的*递归方程*来定义共同知识：

$$\forall \phi, C(\phi) \leftrightarrow E(\phi \wedge C(\phi)).$$

然而，一个递归方程可能会有无穷多个解，并不是所有的解都是我们要的那个. 所以，归纳规则给出了共同知识算子的另一个性质：这是这个递归方程的*最小解*. 下面我们来说明这一点.

我们先从一个看似无关的例子讲起. 用数学归纳法证明一个关于自然数的命题 $P(n)$ 时，我们需要两个步骤：

- 证明基础情形 $P(0)$ 成立；
- 证明归纳步骤：如果 $P(n)$ 成立，那么 $P(n+1)$ 也成立.

因此，归纳的思想是，我们先找到一个基础情形，然后一点一点把成立的集合扩大，直到包含所有的情况.

注意，在真正的证明中，我们不能真的把每一个 $P(n)$ 的正确性都从更小的情况推导出来，因为这是无穷多个情况. 实际上，我们写归纳法证明的过程中，隐含地写了一个关于 P 的递归方程，左边是这个一点

一点扩大的过程，右边是我们要证明的命题：

$$(P(0) \wedge \forall n, (P(n) \to P(n+1))) \leftrightarrow \forall n, P(n).$$

所以，数学归纳法就是一个公理，它说明这样一点一点扩大的过程（左边）的确可以用来证明所有的情况（右边）. 注意，这个公理产生的 P 是递归方程的最小解，因为它是从最小情况逐步扩大得到的结果.

回到共同知识算子 C，其实 C 和数学归纳法里的证明 P 的过程是非常相似的. 它的定义也是从基础情形 ϕ 开始，然后一点一点扩大，每次扩大都是用 E 算子，直到无穷层. 因此，它也具有归纳的性质.

观察归纳规则，它的前提条件是：

$$\mathcal{F} \vDash \phi \to E(\phi \wedge \psi).$$

这一个步骤描述了，如果我们已经构造了 ϕ，我们如何用 E 算子构造下一层的命题. 这就相当于数学归纳法中从 $P(n)$ 到 $P(n+1)$ 的过程.

而归纳规则的结论是：

$$\mathcal{F} \vDash \phi \to C\psi.$$

这个结论说明，如果我们可以逐层加 E 算子，那么这个叠加可以无穷进行下去，最终得到的就是 C 算子. 这就相当于数学归纳法中证明 $P(n)$ 对整个自然数集成立.

所以，上面的类比说明了归纳规则的含义：它确保了 C 算子是递归方程的最小解，也就是那个逐层叠加 E 所定义的算子.

最后，将 S5 公理系统中加入关于 E 和 C 的公理，我们就扩展了知

识逻辑. 因为 E 和 C 是用 K_i 定义的，因此它们本身并不会带来 Kripke 模型新的结构性质.

§11.4.2 Kripke 模型与 Aumann 结构

Kripke 模型与 Aumann 结构之间存在着非常本质的对应关系.

- 在 Kripke 模型中，我们将算子 K_i 对应的关系限制为了等价关系，因为我们希望 i 想象的可能世界之间是不可区分的.
- 在 Aumann 结构中，我们将 i 的信息集定义了全集的一个划分，把每个划分作为他的原子信息.

上面的两个概念实际上是可以相互转换的. 数学上说，等价关系充分必要地给了集合一个划分；从概念上说，所谓不可区分，就是原子性. 因此，他们之间其实存在一一对应的关系.

进一步，如果回忆第一章的思想，我们可以看到逻辑与集合之间的联系如表 11.1 所示.

Kripke 模型	Aumann 结构
可能世界	样本点
公式	事件
原子命题	基本事件
模态算子	集合-集合映射
i 的等价关系	i 的划分
逻辑连接词	集合操作

表 11.1: Kripke 模型与 Aumann 结构的对应关系

实际上，在 S5 公理系统下，这种对应关系可以被严格地叙述和证明，见习题[lhy: 出一下].

Kripke 模型和 Aumann 结构的一个重要区别在于它们各自的研究偏好：Kripke 语义偏重于逻辑，而 Aumann 结构则偏重于（Bayes）概率论. 因此，用数学来研究知识论可以呈现出两种风格：一种是计算机科学、逻辑学和哲学的风格，另一种则是经济学和信息论的风格.

然而，这种对应关系完全依赖于我们对知识的基本假设：

- 对于 Kripke 模型，我们假设了知识满足 S5 公理系统；
- 对于 Aumann 结构，我们假设了知识满足(K0)–(K4).

如果这些假设被打破，那么这样的对应关系就不再成立. 例如，如果我们移除负自省公理，Kripke 模型就不再具备等价关系的性质，因此不再能够对应于 Aumann 结构中的信息集.

从对知识刻画精细程度来说，两种研究方法也有区别. 通常来说，Aumann 结构要想破坏某一条性质是非常难的，因为它是基于概率论和事件的概念. 然而，Kripke 模型具有（模态）逻辑的风格，因此，研究加入或者去除某个公理带来的影响是它最“正统”的研究方法.

§11.4.3 “泥泞的孩童”再回顾：形式化解法

作为一个具体的例子，我们现在把第十章中的“泥泞的孩童”问题用知识逻辑的语言来重新讨论. 我们终于可以用形式化的方式来严格讨论这一问题了.

我们要给出“泥泞的孩童”的逻辑三要素. 首先，语言就是知识逻辑语言. 然后，我们给出 Kripke 模型以及对应的语义.

可能世界可以表示为 $\{0,1\}^n$ 的元素 $x=(x_1,\ldots,x_n)$，其中 $x_i=1$ 表示孩子 i 的脸上有泥巴，$x_i=0$ 表示 i 的脸上没有泥巴. 我们假设每个孩童 i 的可达关系 R_i 都是一个等价关系. 在这种假设下，每个孩子唯

一不是共同知识的事情就是他们脸上泥巴的状态，而其他所有事情都被隐含在了 Kripke 模型之中.

接下来，我们给出命题字母：原子命题 p_i 表示孩子 i 脸上有泥巴，命题 p 表示至少有一个孩子脸上有泥巴.

假设现在父亲还没有宣布 p. 对于孩子 i 来说，他的认知中只有两个可能世界：一个是他的脸上有泥巴，另一个是他的脸上没有泥巴. 其他对他来说都是确定的. 因此，有 xR_iy 当且仅当 $x_j = y_j$ 对于任意 $j \neq i$ 成立. 在这种情况下，框架 $\mathcal{F}$ 对应于一个 n 维超立方体.

例如，当 $n = 3$ 时，框架 $\mathcal{F}$ 对应的立方体结构如图 11.4 所示.

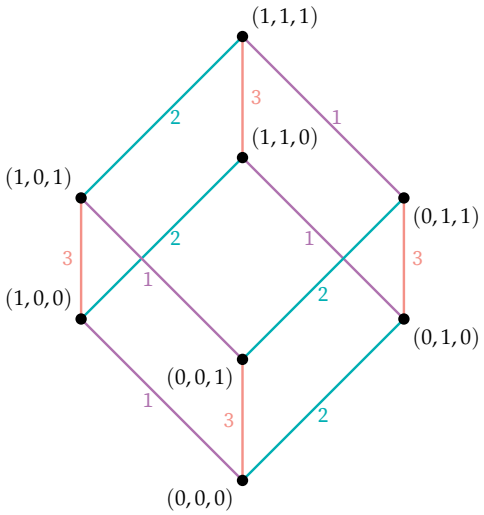

图 11.4: “泥泞的孩童”问题的 Kripke 模型

从框架 $\mathcal{F}$ 到模型 $\mathcal{M}$，我们还需要确定赋值 V.

- 对任意 p_i 和 w，有 $w \in V(p_i)$ 当且仅当 $w_i = 1$;

- $p \in V(w)$ 当且仅当 w 的所有分量 w_j 不全为零.

从模型到点模型，我们还需要确定我们所处的可能世界，从而可以讨论模态公式的可满足性. 例如：$\mathcal{M}, (1,0,1) \vDash Ep$，但是 $\mathcal{M}, (1,0,1) \vDash \neg E^2 p$.

假设父亲宣布了 p，那么框架 $\mathcal{F}$ 将会发生变化，如图 11.5 所示.

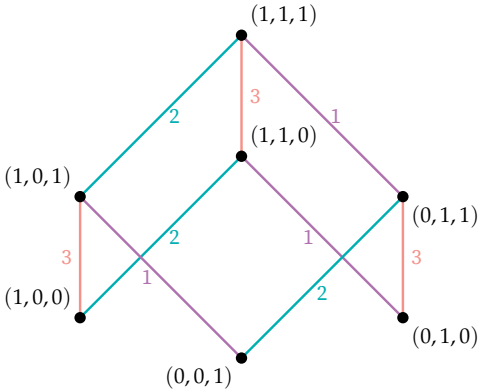

图 11.5: 父亲宣布 p 之后的 Kripke 模型

在 i 的认知中，只有两个可能世界. 因此，i 回答“知道”意味着她能够确定只有一个世界；她回答“不知道”意味着还有两个可能世界.

假如现在是第一轮问答，如果所有人都回答了“不知道”，考虑状态 $s = (1,0,0,\dots)$. 如果真实世界是 s，那么对于 1 来说，可能世界只有一个了，但她却说“不知道”，这说明真实世界不是 s. 同理，所有只有一个 1 的可能世界都会被排除. 由此可以归纳得出，第 k 轮的时候，所有只有 k 个 1 的可能世界都会被排除.

然而，如果父亲没有宣布 p，那么 $\mathcal{M}$ 是一个超立方体. 在任何一轮中，每个孩子都会认为有两个可能世界，因此不会有任何可能世界被

排除.

因此，从结构上来说，父亲宣布 p 改变了每个孩子对应的 R_i 等价的可能世界，使得一些孩子可以确定自己所处的世界. 因而，父亲宣布 p 之后，孩子们的回答会有本质不同.

实际上，这一套方法可以将类似的智力谜题用算法化的方式得到解答，这是知识逻辑的一个有趣的应用. 这是早年的人工智能研究最为热衷的几个方向之一（见第三章的章首引言）.

§11.5 对不一致达成一致

作为一个更正式的（也更更复杂 !）例子，本节将用模态逻辑的方式来探讨达成一致与共同知识的关系，这个问题最早由 Robert Aumann 提出. 现代版本的问题是：两个拥有相同决策规则的 AI 是否会进行交易？如果交易发生，这意味着买家和卖家采取了不同的决策（一个购买，一个出售）；否则，他们将不会达成交易.

我们将证明，对于有相同决策方式的两个个体，他们不可能对采取不同行动这件事具有共同知识. 因此，如果两个 AI 按照相同的规则行事，交易便不会发生. 一句话概括，

玩家不能“对不一致达成一致”（agree to disagree）.

§11.5.1 模型

下面，我们开始介绍这个模型.

首先，我们介绍一下二人博弈的模型. 这个模型是一个无随机性的 Markov 博弈，详细讨论见第 9.4 节. 设有两个玩家 1 和 2，在任意时刻，

每个玩家处于某个局部状态 s_i 之中，局部状态空间分别记为 S_1 和 S_2. 整个系统的全局状态为 $(s_1, s_2) \in S_1 \times S_2 = \mathcal{G}$.

接下来，我们描述博弈的整个过程. 用非负整数 m 来表示当前时刻，初始时刻为 0. 系统的一次运行是一个函数 $r : m \mapsto (s_1, s_2)$，描述了系统每一时刻的全局状态. $\mathcal{R}$ 是全局状态空间 $\mathcal{G}$ 上的所有可能运行的集合. 对于每个运行 $r \in \mathcal{R}$，(r, m) 被称为系统 $\mathcal{R}$ 的一个点.

玩家处于某个状态时，可以采取某种行动. 为了反映“玩家按照相同的规则行事”，我们规定两个玩家的行动集相同，记为 A，且这一行动集不依赖于全局或局部状态. 给定所有人的行动和一个全局状态，系统的转移函数 $\tau : A^2 \times \mathcal{G} \to \mathcal{G}$ 描述了如何从一个状态转移到另一个状态.

为了描述“按照规则行事”，我们引入策略的概念. 玩家 i 的策略 P_i 是从局部状态 S_i 到行动集 A 的映射，即根据状态采取行动. 两个玩家的策略组合记为 $P = (P_1, P_2)$. 这是 Markov 博弈的一种限制，我们要求玩家的策略只依赖于自己的状态，并且不允许随机行动.

策略的执行需要初始状态. 初始状态可能的集合记为 $\mathcal{G}_0$. 给定初始状态集 $\mathcal{G}_0$ 和转移函数 τ，我们就可以在系统上执行任何一种策略. 我们称元组 $\gamma = (\mathcal{G}_0, \tau)$ 为系统的上下文.

给定上下文 $\gamma = (\mathcal{G}_0, \tau)$ 和一个策略组合 P，整个系统就可以运行起来了，产生一个运行. 不难看出，给定一个运行 r，它与策略 P 相容的条件为：

$$r(0) \in \mathcal{G}_0, \quad \forall m,\ r(m+1) = \tau(P(s_1), P(s_2))(r(m)),$$

其中 $r(m) = (s_1, s_2)$. 换言之，r 是从某个可能的初始状态开始执行策略 P 产生的运行. 系统 $\mathcal{R}$ 表示上下文 γ 和策略组合 P，其包含的所有运行 r 都与 P 相容. 我们用 $\mathcal{R}^{rep}(P, \gamma)$ 表示这样的系统.

以上我们就完成了博弈部分的建模. 接下来，我们引入逻辑部分的建模，并把它与博弈的部分对应起来.

首先，我们引入 Kripke 模型. Kripke 模型的点是系统的点. 然后是原子命题集 **P**. 它的元素形如 $perf_i(a)$，表示玩家 i 采取行动 a.

我们还需要定义赋值函数 V，它将 Markov 博弈中的概念和原子命题联系起来. 具体来说，定义如下：原子命题 $perf_i(a) \in V(s)$ 当且仅当在状态 s 玩家 i 采取了行动 a.

然而，赋值函数的定义域应该是一个 Kripke 模型中的点，这个点不同于 Markov 博弈中的状态. 因此，我们需要将赋值函数扩展到系统的点集：$p \in V((r,m))$ 当且仅当 $p \in V(r(m))$.

随后，我们引入知识算子 K_i，假设 K_i 对应的等价关系是 $\sim_i$. 玩家 i 只能区分自己的局部状态 s_i，即

$$(r,m) \sim_i (r',m') \iff r(m)_i = r'(m')_i.$$

根据 Aumann 结构，局部状态 s_i 对应的信息集为

$$IS_i(s_i,\mathcal{R}) = \{(r,m) : r \in \mathcal{R}, r(m) = s_i\}.$$

如此一来，我们得到了 Kripke 点模型 $\mathcal{M},(r,m)$ 以及知识算子 K_i.

此外，除了知识算子，Markov 博弈还有时间的概念，因此我们引入时间模态算子 X，表示“下一时刻”. 其语义定义为：

$$\mathcal{M},(r,m) \vDash X\phi \iff \mathcal{M},(r,m+1) \vDash \phi.$$

利用算子 X，我们可以用公式 $act_i(a)$ 表示“i 将要采取行动 a”：

$$act_i(a) \leftrightarrow \neg perf_i(a) \wedge X perf_i(a).$$

最后一步，为了在点模型中讨论 Markov 博弈的策略，我们还需要定义 Kripke 模型的*决策函数*. 玩家 i 的策略是从局部状态 S_i 到行动集 A 的映射，因此，我们把这个定义自然地扩展到 Kripke 模型的点上，定义决策函数 D 为从 Kripke 模型点集 S 的某个子集映射到某个行动 a 的函数. 为了表示两个玩家采取相同的决策策略，决策函数没有下标.

我们还没有说“某个子集”的选法. 我们要求策略 P_i 和决策函数 D 相容，也就是说，如果玩家 i 在这个状态选择了行动 a，那么决策函数也应该对应产生 a. 我们可以用信息集来描述这个条件：决策函数在某个信息集上的行动恰好是策略在该状态的行动，即

$$P_i(s_i) = D(IS_i(s_i, \mathcal{R})), \quad \forall s_i \in S_i.$$

策略和决策函数虽然有密切联系，但易混淆. 策略描述的是在什么状态下采取什么行动，而决策函数则是基于知识做出行动决策. 在我们的背景下，

$$\text{玩家知道的信息} = \text{玩家处于的局部状态},$$

因此二者是从不同角度在描述同一概念.

我们还需要知道，给定策略组合 P 以及博弈的上下文 γ，我们可以能否找到一个相容的决策函数？下面的条件给了我们一个答案：

我们对决策函数 D 要求*并-一致*. 即对于互不相交的子集 $T_1, \ldots, T_k$，如果 $D(T_i) = a$，则 $D(\bigcup_i T_i) = a$. 如果我们把 T_i 理解成“知道的信息”，

那么这一条件并不难理解. 请看下面的例子:

- 假设我的决策函数是这样描述的: 如果今天下雨, 并且今天星期四, 那么我会去 KFC 疯狂星期四; 如果今天不下雨, 并且今天星期四, 那么我会去 KFC 疯狂星期四.
- 那么, 我的决策还应该有: 虽然我不知道今天下不下雨, 但是如果今天是星期四, 那么我会去 KFC 疯狂星期四.

命题 11.8 *给定策略组合 P 和上下文 γ, 存在一个并-一致的决策函数 D, 使得 P 和 D 相容.*

这一性质的证明并不难, 见习题[lhy: 出一下].

现在, 我们完成了逻辑部分的建模, 我们来总结一下. 在这个模型中, 两个玩家处于同一系统中. 虽然他们的局部状态和信息集可能不同, 但他们的行动集和决策函数相同. 决策函数是并-一致的, 并由某个策略组合实现. 通过上下文中的初始状态和转移函数, 系统可以产生一系列可能的运行.

§11.5.2 定理及其证明

有了上面的模型, 接下来, 我们给出达成一致定理的表述和证明.

定理 11.1 (Aumann 达成一致定理) *给定策略组合 P 和上下文 γ, 由此产生 Kripke 框架 $\mathcal{F}$. 设 $a, b \in A$ 是两个不同的行动, 如果在上下文 γ 中 P 实现了某个并-一致决策函数, 那么*

$$\mathcal{F} \vDash \neg C(act_1(a) \wedge act_2(b)).$$

如果两个玩家选择了同样的并-一致决策函数，那么他们不可能对“我们采取不同行动”这件事形成共同知识. 因此，他们不可能对不一致达成一致.

证明. 我们用反证法证明. 假设某个基于 $\mathcal{F}$ 的点模型 $\mathcal{M},(r,m)$ 使得

$$\mathcal{M},(r,m)\vDash C(act_1(a)\wedge act_2(b)).$$

接下来我们证明 $a=b$.

首先介绍一下思路，共同知识对应了从 (r,m) 出发可到达的状态集 S' 的性质. 从玩家 1 的视角来看，她在 S' 所关联的信息集上都要采取行动 a，根据并-一致性，应该有 $D(S')=a$. 从玩家 2 来看同理，因此也应该有 $D(S')=b$. 因此，$a=b$. 下面，我们把这一思路细化，得到完整的证明.

假设 S' 是从 (r,m) 出发，通过关系 $\sim_1$ 或 $\sim_2$ 可到达的点集. 取一个点 $(r',m')\in S'$，设 $r'(m')_1=s_1'$. 假设 $(r'',m'')\sim_1(r',m')$，那么 $(r'',m'')\in S'$. 因此，

$$IS_1(s_1',\mathcal{R})\subseteq S'.$$

当 s_1' 取遍 S_1，根据信息集的性质，S' 是 $IS_1(s_1',\mathcal{R})$ 的不交并.

因为 $\mathcal{M},(r,m)\vDash C(act_1(a))$，所以有 $\mathcal{M},(r',m')\vDash act_1(a)$. 这一公式意味着 $P_1(s_1')=a$. 根据 P 和 D 的关系，这等价于 $D(IS_1(s_1',\mathcal{R}))=a$. 因为这件事对任意 s_1' 都成立，根据 D 的并-一致性，$D(S')=a$. 同理，从玩家 2 的角度来说，$D(S')=b$. 因此，$a=b$. □

下面是一些关于这个定理的讨论：

- 我们的定理是对于确定性的策略证明的. 然而，一个策略可能是

非确定的，也就是在一个状态可能会有多种行动的选择，比如选择带有随机性. 此时，达成一致定理依然成立，但我们需要恰当地定义 Kripke 模型和决策函数以适应非确定性的策略.

- 当策略具有非确定性时，我们可以用这一模型来理解带有先验知识、风险或者不确定性下的达成一致定理. 只要策略能够对应一个并-一致的决策函数，结论都有效. 这实际上是 Aumann 最初考虑的版本，即共同信念，而不是共同知识. 这个版本的定理见习题[lhy: 出一下].

§11.6 习题

§11.7 章末注记

第六部分

附录：预备知识

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. (accessed 2023-07-10).

[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.

[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.