标题 **title**

作者 *author*

2023 年 7 月 27 日

前言

目录

第一部分

科学的逻辑

第一章　合情推理

§1.1　回顾：命题逻辑的演绎推理

命题逻辑

- *命题逻辑*（propositional logic）的命题公式由如下定义递归形成
  - 命题变元 $p, q, r, \ldots$ 是命题公式.
  - 如果 ϕ 和 ψ 是命题公式，那么 $(\neg\phi)$，$(\phi \vee \psi)$，$(\phi \wedge \psi)$，$(\phi \leftrightarrow \psi)$ 和 $(\phi \to \psi)$ 都是命题公式.
- 例：$(p \vee (q \to r))$ 是命题公式，但是 $p \vee \vee q$ 不是.
- $\neg, \vee, \wedge, \leftrightarrow, \to$ 被称为*连接词*（connectives）.
- 在不产生混淆的时候也会省略括号.
- $A \wedge B$ 也常写作 AB，$\neg A$ 也常写作 $\overline{A}$.

语义和语法

- 命题逻辑最重要的问题是

  什么样的公式是真的？

- 给定一个公式集 Γ，对一个公式 ϕ，我们有两种真的概念：
  - 语义（semantic）：$\Gamma \models \phi$.
  - 语法（syntactic）：$\Gamma \vdash \phi$.

语义

- 每一个命题变元可以赋值真假：$\top$（真）或 $\bot$（假）.
- 对于一般公式，可以利用真值表递归地定义公式的真假赋值.
  - 例如，$p \to q$ 的真值表为

p	q	$p \to q$
$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$
$\bot$	$\top$	$\top$
$\bot$	$\bot$	$\top$

- $\Gamma \models \phi$：对任意赋值，只要 Γ 全为真，ϕ 就为真.

语法

- 推导规则（即语法）描述了从一些公式出发如何得到另外一些公式.
- 语法推导的形式是

$$\frac{\phi \quad \phi \to \psi}{\psi}$$

  横线上方的称为前提（premise），横线下方的称为结论（conclusion）.
- 上面的推导法则被称为肯定前件（Modus Ponens，MP），是三段论的基础.

语法

- 语法推导可以引入新的连接词，例如

$$\frac{\phi \quad \psi}{\phi \wedge \psi}$$

- 也可以消除连接词，例如

$$\frac{\phi \wedge \psi}{\phi}$$

- 在给定一组推导法则下，我们就可以根据规则来推演命题.

演绎推理

- 例如，将*归谬法*（reductio ad absurdum rule, RAA）加入推导法则：将 $\neg\phi$ 作为前提，得到了结论 $\bot$，那么可以推出 ϕ 才是结论. 写作

$$\dfrac{\begin{array}{c}[\neg\phi]\\ \vdots\\ \bot\end{array}}{\phi}$$

  方括号表示假设 $[\neg\phi]$ 是前提，省略号表示推导的中间步骤.

- 我们就可以得到反证法.

演绎推理

- $\Gamma \vdash \phi$：以 Γ 作为前提，依据推导法则，可以得到 ϕ 作为结论.
- 从前提出发，按照法则，得出结论的过程，即称谓*演绎推理*（deductive reasoning）.
- 命题逻辑的主要定理是

  定理 1.1 (完备性定理，Completeness theorem) *对任意公式集 Γ 和任意公式 ϕ，*

$$\Gamma \models \phi \iff \Gamma \vdash \phi.$$

- 推论：检验一个演绎推理的正确性可以用真值表完成.

重言式

- 如果 $\models \phi$，那么称 ϕ *为重言式*（tautology）.
- 重言式即是不需要加任何假设也一定成立的公式，表明了这一推理逻辑所包含的“正确的废话”.
- 如果 $\psi \leftrightarrow \phi$ 是重言式，我们就说 ψ, ϕ *是等值的*（equivalent）.
- 等值的公式在演绎推理中起着相同的作用.
- 例如：$p \to q$ 与 $\neg q \to \neg p$ 是等值的，所以证明一个定理也可以去证明它的逆否命题.

§1.2 合情推理的数学模型

合情推理

- 生活中的推理并不总是演绎推理.
- 走在北大校园里，看到有人突然倒地不起，你的第一反应是?
  - 他突发疾病了.
- 在常识的意义下，这是一种合理的推理.

强三段论

- 演绎推理包含两个强三段论.

$$\frac{\begin{array}{c}A \to B \\ A \text{ 是真的}\end{array}}{B \text{ 是真的}} \qquad \frac{\begin{array}{c}A \to B \\ B \text{ 是假的}\end{array}}{\text{A 是假的}}$$

- 具体到上面的例子，A 是“突发疾病”，B 是“突然倒地不起”.
- 因此演绎推理应该是：因为突发疾病，所以突然倒地不起.

弱三段论

- 然而，我们实际上作出的推理形如：因为突然倒地不起，所以更可能突发疾病.
- 这是一个弱三段论，数学上写作

$$\frac{\begin{array}{c}A \to B \\ B \text{ 是真的}\end{array}}{A \text{ 变得更合理}}$$

- 虽然我们不能得知 A 的真假，但是根据已有的事实 B，我们可以得到 A 的合理性.
- 这样没有严格真假的推理模式就是合情推理（plausible reasoning）.

推理的合理程度

- 再次考虑同样的问题，走在北大校园里，看到有人突然倒地不起，你的第一反应是?
  - 选项 1：他突发疾病了.

- 选项 2：他接到情报有一枚来自俄罗斯的导弹意外偏航，马上要落在北京，他正在躲避.

- 虽然选项 1 和选项 2 都是可能的，但是他们的合理程度并不同.

- 问题：如何刻画不同推理的合理程度?

1.2.1 似然，合情推理的原则

似然的定义

- 用符号 $A|B$ 表示当 B 已知（或假设）为真时，命题 A 合理程度的非负实数度量.

- 如果 $A|B$ 是一个度量，那么任意非负函数 $f(A|B)$ 也会是一个度量，因此 $A|B$ 还不是一个唯一的数学概念.

- 因此，我们现在称 $A|B$ 为以 B 为假设时命题 A 的似然（likelihood）.

- 似然不能定义在任意两个命题之间，只有有合理的关联的命题才能定义似然.

合情推理的原则

- 合情推理的原则实际表明了似然之间的关系.

- 给定相同的假设，似然应该保持命题逻辑中的等值命题. 例如：

  - $(A \vee B)|C = (B \vee A)|C$.
  - $(AB)C|D = A(BC)|D$.
  - $\neg\neg A|B = A|B$.

与规则

- 考虑从 C 出发推出 AB 为真.

- 一种可能的推理方法是先推断出 A 为真，然后在 A 为真的前提下，推断出 B 为真.

- 所以似然 $AB|C$ 应该取决于这两段推理的似然，即 $A|C$ 和 $B|AC$，因此存在函数 F 满足
$$AB|C = F(A|C, B|AC).$$

- 思考：能否写成 $AB|C = F(B|C, A|C)$，如果不能，是否有违反常识的例子?

– 考虑 A 是指小明的左眼是蓝色的，B 是指小明的右眼是棕色的. $A|C$ 和 $B|C$ 都可以很合理，但 $AB|C$ 则几乎不可能发生.

与规则

定理 1.2 如果 F 是连续函数，那么 F 充分必要地满足

$$cf(F(p,q)) = f(p)f(q),$$

这里，c 是某个常数，f 是某个函数. 因此，

$$cf(AB|C) = f(A|C)f(B|AC).$$

- 我们之前说过，如果 $A|B$ 是一个合理性度量，那么 $f(A|B)$ 也是，因此，遵循惯例，不妨就考虑似然 $A|B$ 本身，并且取 $c = 1$. 因此

$$AB|C = (A|C)(B|AC).$$

与规则

- 合情推理的与规则表述如下：

与规则 对命题 A, B, C，成立

$$AB|C = (A|C)(B|AC).$$

否定规则

- 第二个考虑的是似然 $A|B$ 和 $\neg A|B$ 的关系.
- 两个似然应该存在某种函数关系，即存在函数 S 使得

$$\neg A|B = S(A|B).$$

定理 1.3

$$S(x) = (1 - x^m)^{1/m},$$

这里的 m 是一个正常数.

否定规则

- 由上面的定理，我们可以得到以下性质

$$(A|B)^m + (\neg A|B)^m = 1.$$

- 证明：$(\neg A|B) = S(A|B) = (1 - (A|B)^m)^{1/m}$.

- 注意到 $(A|B)^m$ 也是一个满足与规则的似然，所以我们不妨就取似然为 $(A|B)^m$，于是有

否定规则 对命题 A, B，成立

$$(A|B) + (\neg A|B) = 1.$$

合情推理的完备集

- 连接词 $\{\wedge, \neg\}$ 组成了演绎逻辑的完备集：所有真值函数都可以用这个集合的词来表示.

- 回忆：我们要求合情推理保持演绎逻辑的等值公式.

- 与规则对应 $\wedge$，否定规则对应 $\neg$.

- 因此合情推理的两个规则是完备的：任何命题都可以通过这两个公式计算出似然.

合情推理的完备集

- 例如，计算 $A \vee B|C$,

$$\begin{aligned}
A \vee B|C &= 1 - (\overline{A}\overline{B}|C) \\
&= 1 - (\overline{A}|C)(\overline{B}|\overline{A}C) \\
&= 1 - (\overline{A}|C)(1 - (B|\overline{A}C)) \\
&= (A|C) + (\overline{A}B|C) \\
&= (A|C) + (B|C)(\overline{A}|BC) \\
&= (A|C) + (B|C)(1 - (A|BC)) \\
&= (A|C) + (B|C) - (AB|C).
\end{aligned}$$

1.2.2 似然与概率

Kolmogorov 公理化概率论

- Kolmogorov 的概率论研究事件空间上的概率测度.
- 其研究对象的总体被称为*样本空间*，记为 Ω.
- 我们只能观测到某种可观测的特性 P，而不能直接观测样本点，即我们只能观察*事件*，或者说集合
$$\{\omega \in \Omega : P(\omega)\}.$$
- 我们可以观测的所有事件的集合称为*事件域*，记为 $\mathscr{F}$.

事件域

- 事件域 $\mathscr{F}$ 中的事件之间互相有关联.
- 我们自然可以观测到 Ω，因此 $\Omega \in \mathscr{F}$.
- 如果我们可以观测到事件 A，那么我们也可以通过没有观测到 A 来判断观测到了 $\Omega \setminus A$. 因此，
$$A \in \mathscr{F} \implies \Omega \setminus A \in \mathscr{F}.$$
- 如果我们观测到了 A 或者 B，我们其实也观测到了 $A \cup B$，即
$$A, B \in \mathscr{F} \implies A \cup B \in \mathscr{F}.$$

公理化概率论

- *概率* Pr 是一个事件域 $\mathscr{F}$ 到实数的映射，并且满足：
  - 规范性：$\Pr(\Omega) = 1$.
  - 非负性：$\forall A \in \mathscr{F}\ \Pr(A) \in [0, 1]$.
  - 可列可加性：对 $A_1, A_2, \ldots$ 满足 $A_i \cap A_j = \varnothing, i \neq j$，有
$$\Pr\left(\bigcup_i A_i\right) = \sum_i \Pr(A_i).$$

事件与命题

公理化概率论	合情推理
事件	命题
（条件）概率	似然
链式法则	与规则
补事件公式	否定规则

- 事件是命题的集合论描述.
- 具体来说，有如下对应

事件	命题
Ω	$\top$
$\varnothing$	$\bot$
$A \cap B$	$A \wedge B$
$A \cup B$	$A \vee B$
$A \subseteq B$	$A \to B$
$A = B$	$A \leftrightarrow B$

条件概率

- 回忆条件概率的定义，我们可以得到链式法则：

$$\Pr(AB|C) = \frac{\Pr(ABC)}{\Pr(C)} = \frac{\Pr(B|AC)\Pr(AC)}{\Pr(C)} = \Pr(B|AC)\Pr(A|C).$$

- 补事件公式：$\Pr(A|C) + \Pr(\overline{A}|C) = \Pr(\Omega|C) = 1$.
- 这两个公式恰好对应了合情推理的与规则以及否定规则.

似然与概率

- 这并不是巧合. 可以证明，合情推理与事件域的公理化概率具有一一对应的关系：

（条件）概率是似然唯一的数学模型！

- 从此，我们将似然 $A|B$ 定义为概率 $\Pr(A|B)$.

§1.3 合情推理的归纳强论证

1.3.1 先验与基率谬论

条件概率 vs. 概率

- 在前面，我们有意模糊了条件概率和概率在合情推理中的区别.
- 然而，这样的区别是非常重要的.
- 在合情推理中，非条件的概率被称为*先验概率*（prior probability），它表示了对这个命题合理程度的一种无条件的信念.
- 对应地，条件概率就是*后验概率*（posterior probability）或似然，它表示了对合情推理合理程度的一种信念.
- 先验概率和后验概率有若干相互转化的公式.

回忆：全概率公式

定理 1.4 (全概率公式) *设 A_i 是彼此互斥的事件，$\cup_i A_i = \Omega$，那么*

$$\Pr(B) = \sum_i \Pr(B|A_i)\Pr(A_i).$$

- 全概率公式表明了如何使用似然建立起不同先验概率之间的联系.

回忆：Bayes 定理

定理 1.5 (Bayes 定理)

$$\Pr(A|B) = \Pr(B|A)\frac{\Pr(A)}{\Pr(B)}.$$

- Bayes 定理表明了两个不同的后验概率如何基于先验概率相互转化.
- 在合情推理中，这表明了前提推结果的强三段论和结果推前提合理性的弱三段论之间的关系.

基率谬论

- 一辆出租车在夜间发生了一起肇事逃逸事故. 这座城市有两家出租车公司，绿色和蓝色.

- 这个城市历史上肇事逃逸车辆 85% 是绿色的，15% 是蓝色的.
- 一名目击者指认出租车是蓝色的，这里的指认并不一定正确.
- 考虑一种理想的假设，法庭知道这位证人 80% 的概率能正确识别颜色，20% 的概率会把颜色识别错.
- 问：事故车辆是那种颜色的可能性更大?
- 忽略先验概率会产生答案是蓝车的结论.
- 基率谬论（base-rate fallacy）指因为忽略先验概率（即基率）而产生的错误判断.

基率谬论

- 下面我们考虑先验概率再次做计算.
- 记 B 为肇事逃逸的出租车为蓝色，G 为肇事逃逸的出租车为绿色，R 为目击者指认出租车是蓝色.
- 先验概率：$\Pr(B) = 0.15$，$\Pr(G) = 0.85$.
- 似然：$\Pr(R|B) = 0.8$，$\Pr(R|G) = 0.2$.
- $\Pr(R) = \Pr(B)\Pr(R|B) + \Pr(G)\Pr(R|G) = 0.29$.
- $\Pr(B|R) = \Pr(R|B)\Pr(B)/\Pr(R) \approx 0.41$.
- 然而，$\Pr(G|R) = \Pr(R|G)\Pr(G)/\Pr(R) \approx 0.59$.
- 因此我们更应该倾向于认为肇事逃逸的出租车是绿色的！

1.3.2 归纳强论证

合情推理的论证方式

- 在基率谬论的例子中，我们看到合情推理必须要完整地考虑先验的影响.
- 另一方面，我们看到最终做出决策的方式是最大似然（maximum likelihood），即似然更高的那个命题更有可能是对的.
- 这说明合情推理中有很不同于演绎推理的论证方式.

强三段论

- 我们首先指出合情推理包含了命题逻辑中的两个强三段论：

$$\frac{A \to B \quad A \text{ 是真的}}{B \text{ 是真的}} \qquad \frac{A \to B \quad B \text{ 是假的}}{A \text{ 是假的}}$$

- 我们以第一个三段论为例，记 $C \equiv A \to B$.
  - 由链式法则，$\Pr(B|AC) = \Pr(AB|C) / \Pr(A|C)$.
  - $A \to B$ 意味着作为事件 $A \subseteq B$，即 $AB = A$，$\Pr(AB|C) = \Pr(A|C)$.
  - 代入上式得到 $\Pr(B|AC) = 1$，这就是说，当 A 为真时，B 也为真.

弱三段论

- 除了演绎逻辑中的强三段论，合情推理还包含了弱三段论的定量形式.

$$\frac{A \to B \quad B \text{ 是真的}}{A \text{ 变得更合理}}$$

- $\Pr(A|C)$ 是 A 的似然，而 $\Pr(A|BC)$ 是假设 B 为真时，A 的似然.
- 由链式法则，$\Pr(A|BC) = \Pr(A|C) \Pr(B|AC) / \Pr(B|C)$.
- 因为 $\Pr(B|AC) = 1$ 且 $\Pr(B|C) \leq 1$，所以 $\Pr(A|BC) \geq \Pr(A|C)$. 也就是当 B 为真时，A 的合理程度会变大.

有效论证的等价定义

- 回忆记号 $X \vdash Y$：从前提为 X 出发可以跟据推导规则得到结论 Y.
- 此时，我们说从 X 到 Y 的过程是一个*有效论证*（valid argument）. 它与以下三个表述等价：
  - $X \models Y$.
  - $X \to Y$ 是重言式.
  - $X \wedge \neg Y$ 是矛盾式.

- 等价性证明：蕴含的推导法则（或演绎定理）+ 完备性定理.
- 思考：如何在合情推理中定义类似的概念?

随机真值表

- 回忆：事件是命题的集合论描述.
- 合情推理中，每个事件被赋予一个概率（似然），对应的命题也会被赋予同样的概率.
- 于是对应于演绎推理中的语义真值表，合情推理中有*随机真值表*（stochastic truth table）.

Pr	A	B	$A \vee B$
0.4	$\top$	$\top$	$\top$
0.2	$\top$	$\bot$	$\top$
0.25	$\bot$	$\top$	$\top$
0.15	$\bot$	$\bot$	$\bot$

归纳强论证

- 在合情推理中，我们也有和有效论证对应的归纳强论证.
- 考虑如下推理：

$$\frac{X}{Y}$$

利用随机真值表，我们可以尝试定义*归纳强论证*为 $X \wedge \neg Y$ 的**不太可能**为真（或 $\neg X \vee Y$ **很可能**为真）.

- 然而我们会看到，仅仅用随机真值表得到的概念是不符合合情推理的直觉的.
- 我们通过两个例子来引入归纳强论证的限制条件.

奇怪的例子一

- 记 X 为一个北京大学的同学今年 2000 岁, Y 为一个北京大学的同学今年 2000 岁，并且有三条胳膊.
- 直观来讲, 如上 $X \models Y$ 不是归纳强论证.

- 但是，其等效的公式 $\neg X \vee Y$，成立的概率足够大.
- 这个悖论却为这个应该不成立的结论给了一个归纳强论证.
- 所以，从这个角度来看，判断是否为归纳强论证不能只关注结论成立的概率.

限制条件一：证据支持（Evidential Support）

- 假设 X 和 Y 是公式，$t \in [0.5, 1]$.
- 如果 $\Pr(Y|X) > t$, 我们称：X 支持 Y 的证据强度大于 t.
- 证据支持是比最大似然准则的更进一步的要求.

奇怪的例子二

- 记 X 为小明是一位北京大学的学生, Y 为小明不会飞.
- 表面看来, $\Pr(Y|X) = 1$，但我们知道 $X \models Y$ 并不应该是归纳强论证.
- 问题出在了 $\Pr(Y)$ 本身就等于 1，所以 $\Pr(Y|X) = 1$ 并没有什么实际意义，由此引出第二条必要条件.

必要条件二：正相关性

- 我们称 X 与 Y 正相关，如果 $\Pr(Y|X) > \Pr(Y)$.
- 等价地，示性函数 $I(X)$ 和 $I(Y)$ 相关系数大于 0.
- 类似地，
  - 如果 $\Pr(Y|X) < \Pr(Y)$（或 $I(X)$ 和 $I(Y)$ 的相关系数小于 0），那么 X 和 Y 负相关.
  - 如果 $\Pr(Y|X) = \Pr(Y)$（或 $I(X)$ 和 $I(Y)$ 的相关系数等于 0），那么 X 和 Y 不相关.
- 在归纳强论证中，我们要求 X 和 Y 正相关.

归纳强论证的严格定义

- 如果 $X \models Y$ 满足以下三个条件，我们称之为归纳强论证（inductively strong argument）：

  - X 证据支持 Y：$\Pr(Y|X) > 0.5$.
  - X 与 Y 正相关：$\Pr(Y|X) > \Pr(Y)$.
  - $X \to Y$ 不是有效论证.

- 不将有效论证定义为归纳强论证得原因之一是：一个论证可以在前提 X 是矛盾式时成为有效.
  - 例如：$P \wedge \neg P \models Q$ 是有效论证.
  - 但是，由于 $\Pr(P \wedge \neg P) = 0$，$\Pr(Q|P \wedge \neg P)$ 是无定义的，所以 $P \wedge \neg P$ 并不证据支持 Q，也不和 Q 正相关.

认可度（Confirmation Measure）

- 进一步，我们还希望能够衡量前提 X 在多大程度上确认结论 Y 成立.
- 认可概率增量：衡量事件 X 发生后给事件 Y 的发生增加了多大的概率.

$$d(X,Y) = \Pr(Y|X) - \Pr(Y).$$

- 认可度似然比（Likelihood Ratios of Confirmation Measure）. 衡量假设 Y 发生时 X 的似然会比假设 Y 没发生时 X 的似然增加多少. 该差值越大表示观测到 X 的话越应该发生了 Y. 分母归一化使得 $\ell(X,Y) \in [-1,1]$.

$$\ell(X,Y) = \frac{\Pr(X|Y) - \Pr(X|\neg Y)}{\Pr(X|Y) + \Pr(X|\neg Y)}$$

认可度和相关性的关系

- 设 $0 < \Pr(X), \Pr(Y) < 1$.
- X 和 Y 正相关 $\iff d(X,Y) > 0 \iff \ell(X,Y) > 0$.
- X 和 Y 不相关 $\iff d(X,Y) = 0 \iff \ell(X,Y) = 0$.
- X 和 Y 负相关 $\iff d(X,Y) < 0 \iff \ell(X,Y) < 0$.

认可度与有效论证的关系

- 设 $0 < \Pr(X), \Pr(Y) < 1$.

- $d(X,Y) = \begin{cases} \Pr(\neg Y), & \text{如果} X \models Y, \\ -\Pr(Y), & \text{如果} X \models \neg Y. \end{cases}$
- $\ell(X,Y) = \begin{cases} 1, & \text{如果} X \models Y, \\ -1, & \text{如果} X \models \neg Y. \end{cases}$

1.3.3 有效论证和归纳强论证的比较

回顾

- 考虑一个论证 $X \Rightarrow Y$，我们已经有三种方式评估 X 如何支持 Y：
  1. $X \Rightarrow Y$ 是一个演绎推理：$X \models Y$.
  2. 基于随机真值表，X 证据支持 Y：$\Pr(Y|X) > 0.5$.
  3. 基于随机真值表，X 正相关于 Y：$\Pr(Y|X) > \Pr(Y)$.
- 其中，
  - 1 对应有效论证，
  - 2 和 3 都是合情推理中的归纳强论证的必要条件.
- 我们将进一步讨论有效论证和归纳强论证的一些不同之处.

合情推理的非单调性

- 有效论证具有单调性：论证的有效性随着前提的增加不会下降. 即：
- 对于任意 X, Y, Z, 若 $X \models Y$, 则 $X, Z \models Y$.
- 然而合情推理中，单调性不再存在.
- 存在这样的例子：X, Y, Z 和对应的随机真值表，使得 X 证据支持 Y, 但 $Z \wedge X$ 并不证据支持 Y.
- 增加新的前提反而可能降低结论发生概率.

非单调性：例子

Pr	X	Y	Z	$X \wedge Z$
0.1	$\top$	$\top$	$\top$	$\top$
0.2	$\top$	$\top$	$\bot$	$\bot$
0.2	$\top$	$\bot$	$\top$	$\top$
0	$\top$	$\bot$	$\bot$	$\bot$
0.1	$\bot$	$\top$	$\top$	$\bot$
0.1	$\bot$	$\top$	$\bot$	$\bot$
0.1	$\bot$	$\bot$	$\top$	$\bot$
0.2	$\bot$	$\bot$	$\bot$	$\bot$

- $\Pr(Y|X) = (0.1 + 0.2)/(0.1 + 0.2 + 0.2 + 0) = 0.6 > 0.5$.
- $\Pr(Y|X \wedge Z) = (0.1)/(0.1 + 0.2) = 1/3 < 0.5$.

确凿性原则

- 我们试图用 Z 论证 Y，将 X 看作某种附加的条件，我们考虑 X 对 Y 这一论据的影响.
- 在演绎推理中，若 $Z, X \models Y$ 和 $Z, \neg X \models Y$ 都满足，则 $Z \models Y$. 如果类比到合情推理中呢?
- *确凿性原则*（Sure-thing Principle）：如果不论条件在 X 还是 $\neg X$，Z 都是 Y 的一个 “好的论据”，那么 Z 就是 Y 的一个 “好的论据”.
- *无条件确凿性原则*（Unconditional Sure-thing Principle）：如果 $Z \wedge \neg X$ 和 $Z \wedge X$ 都是 Y 的一个 “好的论据”，那么 Z 就是 Y 的一个 “好的论据”.
- “好的论据” 可以从证据支持和正相关性两方面考虑.

确凿性原则

- 在任何随机真值表中，如果 $\Pr(Y|Z \wedge X) > 0.5$ 且 $\Pr(Y|Z \wedge \neg X) > 0.5$，那么 $\Pr(Y|Z) > 0.5$.
  - 因此，从证据支持角度，确凿性原则是成立的.
  - 练习：从证据支持角度，无条件确凿性原则也是成立的.

- 在任何随机真值表中，如果 $\Pr(Y|Z \wedge X) > \Pr(Y)$ 且 $\Pr(Y|Z \wedge \neg X) > \Pr(Y)$，那么 $\Pr(Y|Z) > \Pr(Y)$.
  - 因此，从正相关性角度，无条件确凿性原则是成立的.
  - 思考：从正相关性角度，确凿性原则成立吗?

Simpson 悖论

- 实际上，并不一定成立，有反直觉的例子：存在 X, Y, Z 和对应的随机真值表，使得
  - $\Pr(Y|Z \wedge X) > \Pr(Y|X)$.
  - $\Pr(Y|Z \wedge \neg X) > \Pr(Y|\neg X)$.
  - 然而，$\Pr(Y|Z) \leq \Pr(Y)$.
- 这样的现象和 Simpson 悖论有关.
- 举个例子，球员甲的两分球和三分球命中率均高于球员乙，但是球员甲的总投篮命中率却可能低于乙.

Simpson 悖论

- 考虑如下具体的例子，有一个班中一半同学来自北京大学，另一半来自清华大学，我们抽出一名同学 Bob，估计 Bob 投篮命中的概率.
- 记 Y 为 Bob 投篮命中，记 X 为 Bob 投出一个两分球，则 $\neg X$ 为 Bob 投出一个三分球（我们这里只考虑有两分和三分球），记 Z 为 Bob 来自清华大学.
- $\Pr(Y)$ 表示全班学生的投篮命中率.
- $\Pr(Y|Z)$ 表示全班来自清华大学的学生的投篮命中率.
- $\Pr(Y|X)$ 表示全班学生的两分命中率，
  $\Pr(Y|\neg X)$ 表示全班学生的三分命中率.
- $\Pr(Y|Z \wedge X)$ 表示这个班来自清华大学的学生的两分命中率，$\Pr(Y|Z \wedge \neg X)$ 表示这个班来自清华大学的学生的三分命中率.

Simpson 悖论

- 考虑这样一个投篮数据的实例：

	全班同学	来自清华大学
两分球	50/100	6/10
三分球	1/101	1/100
总命中率	51/201	7/110

- $\Pr(Y) = 51/201$，$\Pr(Y|Z) = 7/110$ 为投篮命中率.
- $\Pr(Y|X) = 50/100 = 1/2$，$\Pr(Y|Z \wedge X) = 6/10 = 3/5$ 为两分命中率.
- $\Pr(Y|\neg X) = 1/101$，$\Pr(Y|Z \wedge \neg X) = 1/100$ 为三分命中率.
- Simpson 悖论在这一实例下的解释：这个班里来自清华大学的学生两分命中率和三分命中率分别都比全班平均水平高，但总体投篮命中率反倒比全班水平低.

Simpson 悖论

- 我们将这两个概率用全概公式展开来寻找原因：

$$\Pr(Y|Z) = \Pr(Y|Z \wedge X)\Pr(X|Z) + \Pr(Y|Z \wedge \neg X)\Pr(\neg X|Z),$$

$$\Pr(Y) = \Pr(Y|X)\Pr(X) + \Pr(Y|\neg X)\Pr(\neg X).$$

- $\Pr > \Pr$.
- 然而，关键是有可能发生 $\Pr(X|Z) \neq \Pr(X)$.
- 在上面篮球的例子中表现为清华大学的同学选择投两分球和三分球的比例和全班同学不同.

合取谬误

- 合取谬误（Conjunction Fallacy）是一种认知偏差.
- 经典例子：Linda 是一位单身、外向且年龄为 31 岁的女性. 在大学期间，她主修哲学，十分关注种族歧视和社会公正问题，而且曾参加过反核游行.（记为 E）请问以下哪一件事情更可能发生?

  1. Linda 是一名银行出纳员（记为 B）
  2. Linda 是一名银行出纳员，同时她还是一名女权主义者（记为 $B \wedge F$）

- 在调查实验中多数被试选择了 2.
- 但是，我们可以肯定 $\Pr(B \wedge F|E) \leq \Pr(B|E)$. 如何理解?

合取原则

- 为了理解这种谬误产生的原因，考虑合取原则（Conjunction Principle）：如果 E 是 $P \wedge Q$ 的 “好论据”，那么 E 也是 P 的 “好论据”.
- 在演绎推理中，因为 $E \to (P \wedge Q) \models E \to P$，所以合取原则成立.
- 类似确凿性原则，我们在合情推理中也可以从证据支持和正相关性两方面考虑.
- 从证据支持的角度，合取原则成立.
  - 如果 $\Pr(P \wedge Q|E) > 0.5$，那么 $\Pr(P|E) \geq \Pr(P \wedge Q|E) > 0.5$.

合取原则

- 然而，从正相关性的角度，合取原则未必成立.
  - 也就是说，假设 $\Pr(P \wedge Q|E) > \Pr(P \wedge Q)$，不一定能推出 $\Pr(P|E) > \Pr(P)$.
- 当人们给定对 Linda 的描述 E 的时候，很容易建立起 E 和 $B \wedge F$ 的正相关性.
- 然而这并不意味着 E 和 B 是正相关的！因此发生了合取谬误.
- 从 Simpson 悖论和合取谬误可以看出，只依靠正相关性进行推理很容易犯错误，因此证据支持（极大似然）是归纳强论证不可缺少的要素.

第二章　Markov 链与决策

§2.1　Markov 链

概率的解释

- 回顾：我们在第一次课说过，似然，或者说合情推理的合理程度，是一种概率的解释模型.
- 尽管概率论在数学上通常被形式化为 Kolmogorov 公理体系，但是公理体系并没有回答“概率”是什么.
- 概率的解释是一个哲学课题. 两个主要的例子：
  - 频率解释：概率是无穷次独立重复试验的频率（大数定律）.
  - 主观解释（Bayes 解释）：概率是对命题合理程度的信念（似然）.

主观解释的缺陷

- 主观解释对推理的假设是逻辑的、静态的，时间的概念并不出现在似然里面.
- 例：考虑一个罐子，里面有除颜色之外不可区分的 N 个球，有 n 个白球，剩下的是黑球. 顺序从中拿出 N 个球，第 k 次拿出的球颜色是 W_k 或 B_k.
  - $\Pr(W_iW_j) = \Pr(W_i|W_j)\Pr(W_j) = \Pr(W_j|W_i)\Pr(W_i)$.（$i < j$）
  - $\Pr(W_i) = \Pr(W_j) = n/N \implies \Pr(W_i|W_j) = \Pr(W_j|W_i)$.
- 从似然的角度，$\Pr(W_i|W_j)$ 和 $\Pr(W_j|W_i)$ 不仅是可计算的，而且是相等的. 概率的计算告诉了我们，更早状态的信息依赖于未来的状态！
- 逻辑上蕴含关系并不意味着实际上的因果关系，但是似然完全没有考虑这一点. 因此，我们需要引入一个带有时间的模型，这就是 Markov 链.

Markov 链

- *Markov 链*（马氏链，Markov chain）是一个随机变量序列 $\{X_t\}_{t=0}^{\infty}$. 包含如下概念：
  - 状态空间 $\mathcal{S}$：X_t 所有可能值构成的集合，有限或者可数.
  - 转移矩阵 $\mathcal{P}$：下一时刻系统状态之间转移的概率.
    * $\mathcal{P} = (p_{ij})_{i,j\in\mathcal{S}}$，$p_{ij}$ 是从 i 状态转移到 j 状态的概率.
  - *Markov 性*(Markov property)：对任意时刻 $t = 1, \ldots, n$ 和任意状态 $j, k, j_0, \ldots, j_{t-1} \in \mathcal{S}$，如下等式成立

$$\begin{aligned}&\Pr(X_{t+1} = j | X_t = k, X_{t-1} = j_{t-1}, \ldots, X_0 = j_0)\\ =&\Pr(X_{t+1} = j | X_t = k) = p_{kj}.\end{aligned}$$

- 注：我们给出的定义是简化的 Markov 链，每个时刻之间的转移都是一样的转移矩阵，这样的 Markov 链被称为时齐的（time-homogeneous）.

Markov 链

- Markov 链是一种简化的带时间的概率模型.
- Markov 性：在固定现在的情况下，过去与未来相互独立.
  - 条件在 $X_n = i$ 下，$\{Y_m\}_{m=0}^{\infty} := \{X_{m+n}\}_{m=0}^{\infty}$ 是一个转移矩阵为 P 的 Markov 链，并且与 $(X_0, \ldots, X_{n-1})$ 相互独立.
- 时齐性：状态的转移不依赖当前时间，只和当前的状态有关.
  - $\Pr(X_{m+n} = j | X_n = k) = \Pr(X_m = j | X_0 = k)$.
- 有时候也会考虑带初态的 Markov 链，X_0 服从分布 $\lambda = (\lambda_s)_{s\in\mathcal{S}}$.

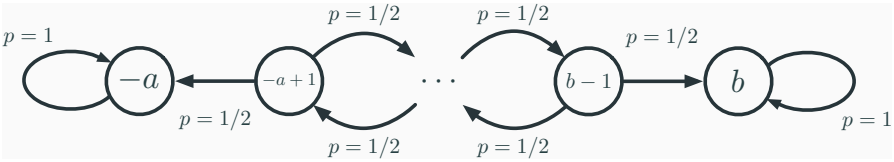
Markov 链的例子

- 公平对赌. 玩家 A 和 B 抛硬币来赌钱，A 赌正面，B 赌反面.
- 每一轮独立地抛硬币，正面朝上的概率和反面朝上的概率相等，都是 $1/2$. 赢的一方给输的一方一块钱. A 输 a 块钱破产，B 输 b 块钱破产，
- Z_i 是第 i 轮 A 的收入. $Z_0 = X_0 = 0$ 是 A 初始的收入. $X_n = Z_0 + \cdots + Z_n$ 是 A 的累计收入.

- 那么，$\{X_n\}_{n\geq 0}$ 是一个 Markov 链.
  - 状态空间：$\mathcal{S} = \{-a, -a+1, \ldots, 0, 1, \ldots, b\}$.
  - 转移概率：对 $-a < i < b-1$，$p_{i,i+1} = p_{i+1,i} = 1/2$；$p_{-a+1,-a} = p_{b-1,b} = 1/2$，$p_{-a,-a} = p_{b,b} = 1$；其他值为 0.

Markov 链的例子

- $\{X_n\}_{n\geq 0}$ 是一个 Markov 链.
  - 状态空间：$\mathcal{S} = \{-a, -a+1, \ldots, 0, 1, \ldots, b\}$.
  - 转移概率：对 $-a < i < b$，$p_{i,i+1} = p_{i+1,i} = 1/2$；$p_{-a+1,-a} = p_{b-1,b} = 1/2$，$p_{-a,-a} = p_{b,b} = 1$；其他值为 0.



赌徒谬误

- A 的累计收入 $\{X_n\}_{n\geq 0}$ 形成了 Markov 链.
- 根据 Markov 性，未来双方的收入变化只取决于现在，而和过去运气无关.
- 赌徒谬误（gambler's fallacy）：深陷赌局中的人会按照自己历史上的运气来评估自己未来的运气，认为过去运气差未来运气就会变好.
- “风水轮流转” 在一场公平对赌中是不正确的认知.
- 思考：如何评估赌局的公平性?

多步转移概率

- 如果对赌是公平的，那么我们应该认为两个人每一轮的累计收入分布都是一样的，即

$$\Pr(X_n = i | X_0 = 0) = \Pr(X_n = -i | X_0 = 0).$$

- 因此，我们需要能够计算多步转移的概率.
- 设 $p_{ij}^{(k)}$ 表示从状态 i 用 k 步转移到状态 j 的概率.

- k 步转移概率形成了一个矩阵 $\mathcal{P}^{(k)}$.
- Kolmogorov-Chapman 方程：

$$\mathcal{P}^{(k+l)} = \mathcal{P}^{(k)}\mathcal{P}^{(l)}.$$

多步转移概率

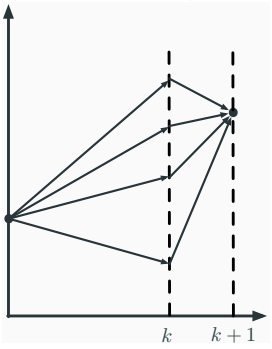
- Kolmogorov-Chapman 方程：

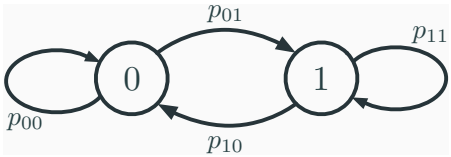
$$\mathcal{P}^{(k+l)} = \mathcal{P}^{(k)}\mathcal{P}^{(l)}.$$

- 证明：由 Markov 性、时齐性和全概率公式，$p_{ij}^{(k+l)} = \Pr(X_{k+l} = j | X_0 = i) = \sum_\alpha \Pr(X_{k+l} = j, X_k = \alpha | X_0 = i) = \sum_\alpha \Pr(X_k = \alpha | X_0 = i) \Pr(X_{k+l} = j | X_k = \alpha) = \sum_\alpha p_{i\alpha}^{(k)} p_{\alpha j}^{(l)}$.
- 特例：前向方程（forward equation）$\mathcal{P}^{(k+1)} = \mathcal{P}^{(k)}\mathcal{P}$，后向方程（backward equation）$\mathcal{P}^{(l+1)} = \mathcal{P}\mathcal{P}^{(l)}$.
- 推论：$\mathcal{P}^{(k)} = \mathcal{P}^k$.
- 若已知初始分布向量为 λ，我们可以计算它随时间的演化：

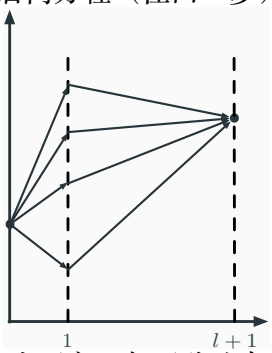
$$\lambda^\mathsf{T}, \lambda^\mathsf{T}\mathcal{P}, \ldots, \lambda^\mathsf{T}\mathcal{P}^n, \ldots$$

多步转移概率　　前向方程（往前一步）：





后向方程（往回一步）：



遍历定理与平稳分布

- 思考：如何计算公平对赌中 X_n 的概率分布？
- 我们先来看一个简单的例子. 假设 $|p_{00}+p_{11}-1|<1$，考虑只有两个状态 $0,1$，转移矩阵为

$$\mathcal{P}=\begin{pmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{pmatrix}.$$

遍历定理

- 可以归纳证明：

$$\begin{aligned}\mathcal{P}^n =& \frac{1}{2-p_{00}-p_{11}}\begin{pmatrix} 1-p_{11} & 1-p_{00} \\ 1-p_{11} & 1-p_{00} \end{pmatrix} \\ &+\frac{(p_{00}+p_{11}-1)^n}{2-p_{00}-p_{11}}\begin{pmatrix} 1-p_{00} & -(1-p_{00}) \\ -(1-p_{11}) & 1-p_{11} \end{pmatrix}.\end{aligned}$$

- $\lim_{n\to\infty} p_{i0}^{(n)} = (1-p_{11})/(2-p_{00}-p_{11}), \lim_{n\to\infty} p_{i1}^{(n)} = (1-p_{00})/(2-p_{00}-p_{11})$.
- 随着时间的推移，Markov 链初始状态对概率分布的影响逐渐消失. 这个规律具有普遍性，这就是遍历定理.

遍历定理

定理 2.1 (遍历定理，Ergodic theorem) *Markov 链的状态空间为 $\mathcal{S} = \{1, \dots, N\}$，转移矩阵为 $\mathcal{P} = (p_{ij})$.*

- *如果对于某一个 n_0 有*

$$\min_{ij} p_{ij}^{(n_0)} > 0, \tag{2.1}$$

那么存在分布 $\lambda = (\lambda_1, \dots, \lambda_N)$ 使得

$$\lambda_i > 0, \quad \sum_i \lambda_i = 1, \tag{2.2}$$

并且对于每一个 $j \in \mathcal{S}$ 和任意 $i \in \mathcal{S}$ 都有

$$p_{ij}^{(n)} \to \lambda_j, n \to \infty. \tag{2.3}$$

遍历定理

定理 2.2 (遍历定理，续)

- *反之，如果存在满足 (2.2) 和 (2.3) 的 λ，则存在满足 (2.1) 的 n_0.*
- *式 (2.2) 的 λ 满足*

$$\lambda^{\mathsf{T}} = \lambda^{\mathsf{T}} \mathcal{P}. \tag{2.4}$$

- 条件 (2.1) 表明超过某个步数 n_0 之后，从 i 出发到达 j 的概率总是正的，这个条件被称为遍历（ergodic）.
- 条件 (2.2) 表明每一个状态被访问到的概率都是正的，没有 “死状态”.
- 遍历定理表明遍历的 Markov 链从任何状态出发都是不可逆的，最终会把每个状态都走过一遍（遍历），变成一个混合均匀的状态.
  - 这可以用来解释物理学中的扩散现象.

平稳分布

- 满足条件 (2.4) 的分布被称为平稳分布（stationary distribution）.
- 平稳分布为初始状态时，Markov 链的演化与时间无关：$(X_k, \dots, X_{k+l})$ 的联合分布不依赖于 k.
- 如果 Markov 链是遍历的，那么平稳分布是唯一的.

- 假设 μ 是另外一个平稳分布，那么 $\mu_j = \sum_\alpha \mu_\alpha p_{\alpha j} = \cdots = \sum_\alpha \mu_\alpha p_{\alpha j}^{(n)}$.
- 因为 $p_{\alpha j}^{(n)} \to \lambda_j$，所以 $\mu_j = \sum_\alpha (\mu_\alpha \lambda_j) = \lambda_j$.

- 非遍历 Markov 链也可能存在(唯一)平稳分布，考虑 $\mathcal{P} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ 和 $\lambda = (1/2, 1/2)^\mathsf{T}$.

§2.2 Markov 奖励过程（MRP）

决策理论

- 我们接下来的目标就是在 Markov 链上建立决策理论.
- 每一阶段我们可以选择某个行动，这个行动在 Markov 链会产生一些奖励.
- 我们的目标是选择恰当的行动方式是的我们的总奖励最大.
- 首先我们定义奖励的过程.

Markov 奖励过程

- 一个 *Markov 奖励过程*（Markov reward process，MRP）是四元组 $\langle \mathcal{S}, \mathcal{P}, \mathcal{R}, \gamma \rangle$：
  - $\mathcal{S}$ 是一个有穷的状态集合.
  - $\mathcal{P}$ 是一个状态转移矩阵，从 i 转移到 j 的概率记为 $\mathcal{P}_{i,j}$.
  - $\mathcal{R}$ 是一个奖励函数，$\mathcal{R}_s = \mathbb{E}[R_{t+1} | S_t = s]$：当 t 时刻位于状态 s 时下一时刻（离开）获得的奖励的期望，R_{t+1} 是下一阶段所处状态的奖励.
  - γ 是一个折扣系数，$\gamma \in [0, 1]$.
- 注：条件数学期望 $\mathbb{E}[\cdot | S_t = s]$ 可以理解为条件在 $\{S_t = s\}$ 下定义的概率所求的期望.

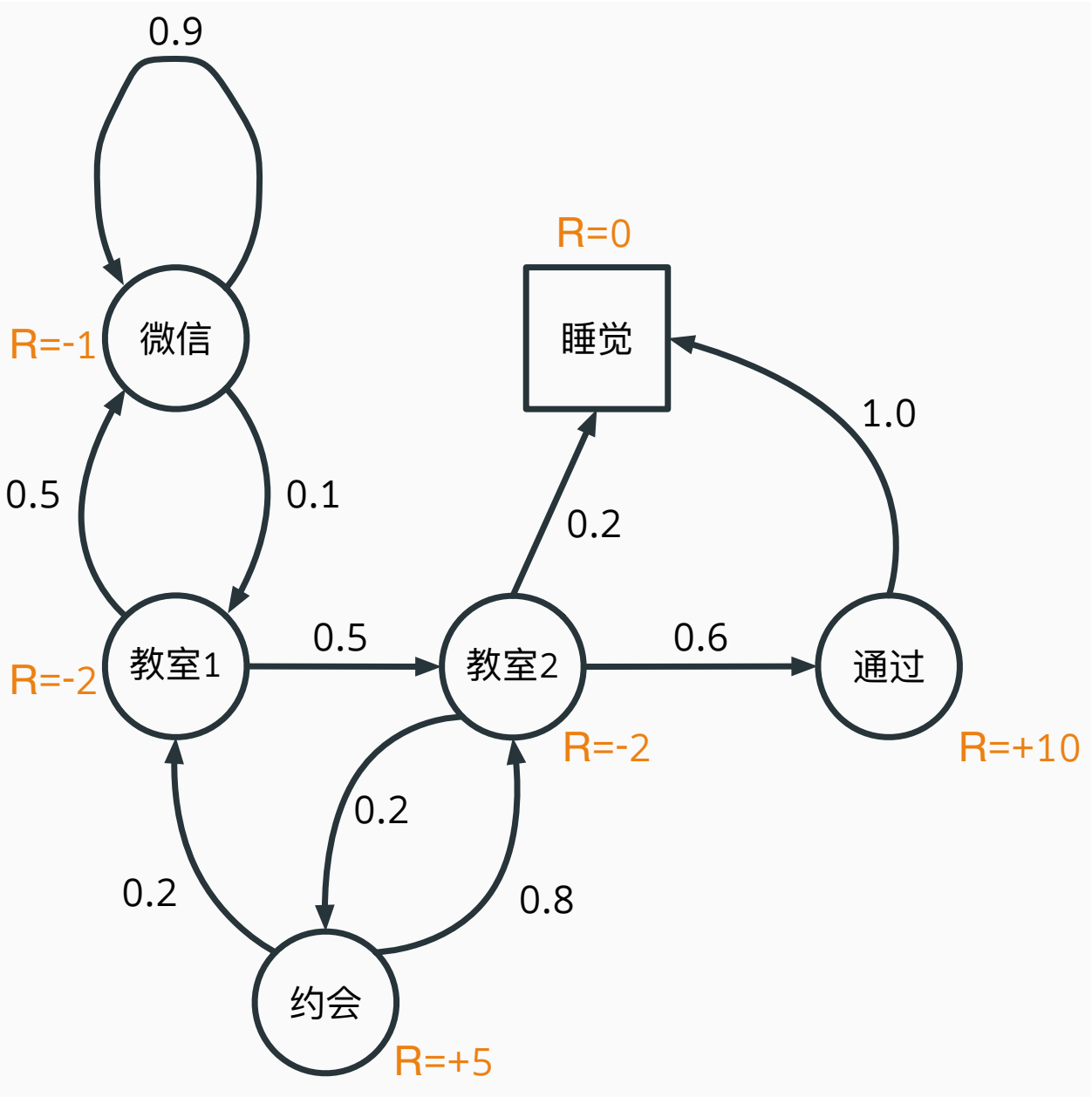
例子：学生 MRP

回报

- MRP 中，t 时刻以后的总*回报*（Return）G_t 定义为

$$G_t = R_{t+1} + \gamma R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}.$$

- $\gamma \in [0, 1]$ 衡量了未来下一时段 1 的奖励在当前时刻的价值.


0.9
R=0
睡觉
R=-1
微信
1.0
0.5
0.1
0.2
教室1
0.5
教室2
0.6
通过
R=-2
R=-2
R=+10
0.2
0.2
0.8
约会
R=+5

- 未来 $k+1$ 时刻的奖励对当前时刻 t 的作用是 $\gamma^k R_{t+k+1}$.
- 若 $\gamma \to 0$，表示对奖励进行“短视”的评估；反之更“远见”.

折扣系数的意义

- 许多 MRP 和后面学习的 MDP 都有与时间无关的折扣系数 $\gamma < 1$，原因：
  - 起始于对未来不确定性对冲：直接对应于利润率.
  - 动物和人类对即时回报具有偏好.
- 有时也使用非折扣化的 MRP（即 $\gamma = 1$），例如当所有的转移序列都会有固定的终止时间.

价值函数

- 在 MRP 中，状态*价值函数*（value function）$v(s)$ 表示从状态 s 出发的期望回报

$$v(s) = \mathbb{E}(G_t | S_t = s).$$

- 价值函数 $v(s)$ 衡量了状态 s 的长期效益.
- Markov 性：只从当前起考虑未来收益，不考虑历史收益（沉没成本）的影响.
- 时齐性：价值函数的定义不依赖于时刻 t (无穷阶段情形).

MRP 的 Bellman 方程

- 价值函数可以被分解为两部分：
  - 即时回报 R_{t+1}
  - 下一个状态开始的折扣价值 $\gamma v(S_{t+1})$

$$\begin{aligned} v(s) &= \mathbb{E}(G_t | S_t = s) \\ &= \mathbb{E}(R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots | S_t = s) \\ &= \mathbb{E}(R_{t+1} + \gamma(R_{t+2} + \gamma R_{t+3} + \dots) | S_t = s) \\ &= \mathbb{E}(R_{t+1} + \gamma G_{t+1} | S_t = s) \\ &= \mathbb{E}(R_{t+1} + \gamma v(S_{t+1}) | S_t = s) \\ &= \mathcal{R}_s + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{s,s'} v(s'). \end{aligned}$$

矩阵形式的 Bellman 方程

- Bellman 方程可以用矩阵形式表达：

$$v = \mathcal{R} + \gamma \mathcal{P} v.$$

这里 v 是列向量 $v = (v(s))_{s \in \mathcal{S}}$.

Bellman 方程的解

- Bellman 方程是一个线性方程，可以被直接解:

$$v = \mathcal{R} + \gamma \mathcal{P} v \implies (I - \gamma \mathcal{P}) v = \mathcal{R} \implies v = (I - \gamma \mathcal{P})^{-1} \mathcal{R}.$$

- 对于 n 个状态的 Markov 链，计算复杂度为 $\mathcal{O}(n^3)$.
- 对于较小的 MRP 可以直接解，太大的 MRP 开销太大.
- 对于大型 MRP，可以采用迭代算法，例如：
  - 动态规划（dynamic programming）
  - Monte-Carlo 评估（Monte-Carlo evaluation)
  - 时序差分学习（temporal-difference learning）

§2.3 Markov 决策过程（MDP）

Markov 决策过程

- *Markov 决策过程*（Markov decision process）是一个定义了决策的 MRP. 它可以看做一个任意状态都具有 Markov 性的环境.
- 一个 MDP 是五元组 $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$.
  - $\mathcal{S}$ 是一个有限的状态集合.
  - $\mathcal{A}$ 是一个有限的行动（action）集合.
  - $\mathcal{P}$ 是状态转移概率矩阵，

$$\mathcal{P}^{a}_{ss'} = \Pr(S_{t+1} = s' | S_t = s, A_t = a).$$

- $\mathcal{R}$ 是一个奖励函数，$\mathcal{R}_s^a = \mathbb{E}(R_{t+1}|S_t = s, A_t = a)$，$R_{t+1}$ 是进行某一行动到达某一状态后的奖励.
- γ 是一个折扣系数 $\gamma \in [0,1]$.

例子：学生 MDP

策略

- 一个策略（policy）π 是给定状态下行动的分布，

$$\pi(a|s) = \Pr(A_t = a|S_t = s).$$

- 一个策略完全决定了一个智能体在 MDP 环境中的行为.
- Markov 性：MDP 的策略取决于当前状态，而非历史状态.
- 时齐性：MDP 的策略不依赖于时刻 t.

策略

- 给定一个 MDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ 和一个策略 π.
- $\langle \mathcal{S}, \mathcal{P}^\pi \rangle$ 是一个 Markov 链.
- $\langle \mathcal{S}, \mathcal{P}^\pi, \mathcal{R}^\pi, \gamma \rangle$ 是一个 MRP.
- 其中

$$\mathcal{P}_{s,s'}^\pi = \mathbb{E}_{a\sim\pi(\cdot|s)}(\mathcal{P}_{s,s'}^a) = \sum_{a\in\mathcal{A}} \pi(a|s)\mathcal{P}_{s,s'}^a,$$

$$\mathcal{R}_s^\pi = \mathbb{E}_{a\sim\pi(\cdot|s)}(\mathcal{R}_s^a) = \sum_{a\in\mathcal{A}} \pi(a|s)\mathcal{R}_s^a.$$

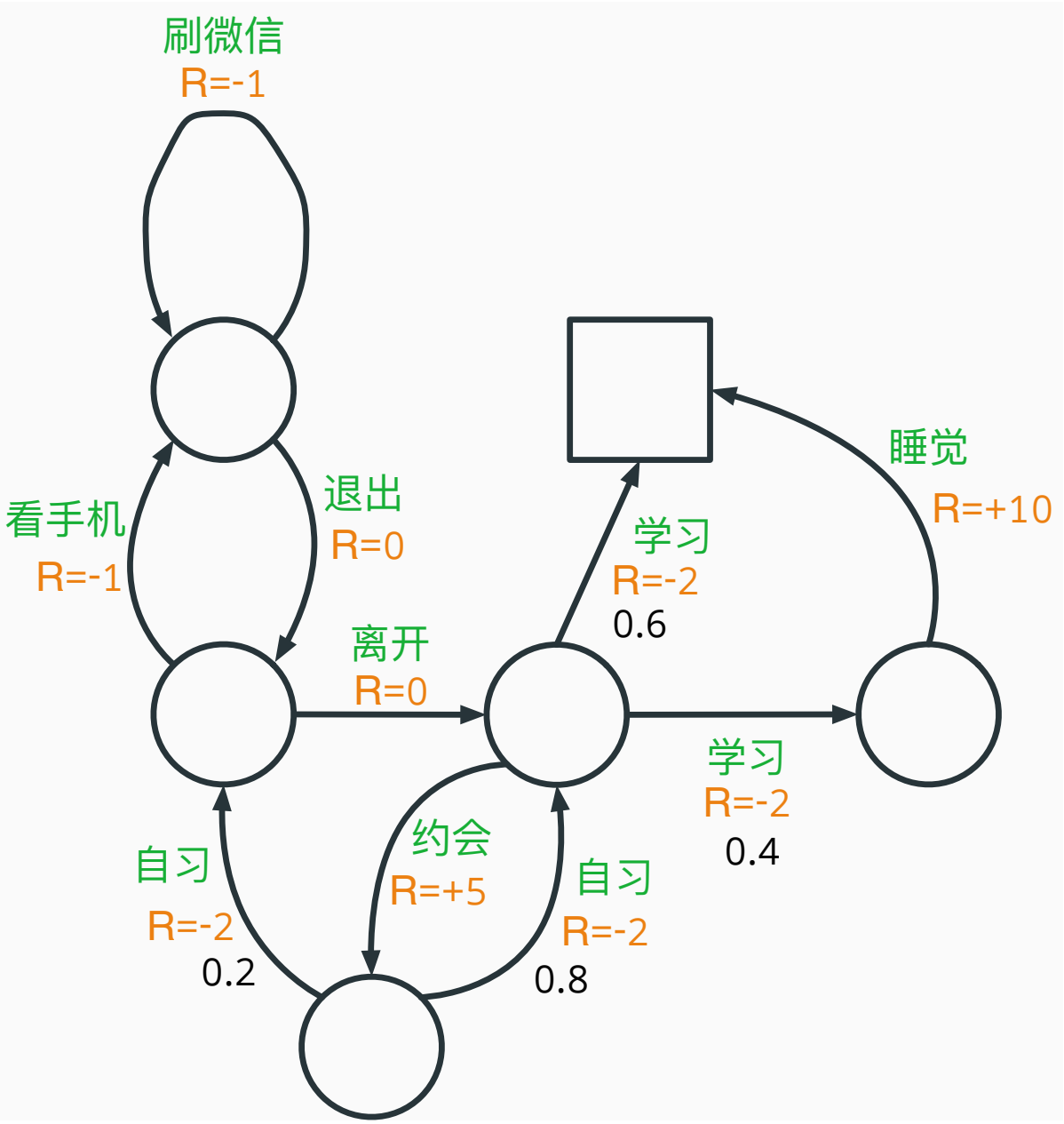
价值函数

- 在 MDP 中，状态-价值函数 $v_\pi(s)$ 是从状态 s 出发，遵从策略 π 的期望回报

$$v_\pi(s) = \mathbb{E}_\pi(G_t|S_t = s).$$

- 行动-价值函数 $q_\pi(s,a)$ 是从状态 s 出发，采取行动 a，遵从策略 π 的期望回报

$$q_\pi(s,a) = \mathbb{E}_\pi(G_t|S_t = s, A_t = a).$$


刷微信
R=-1
看手机
R=-1
退出
R=0
离开
R=0
学习
R=-2
0.6
睡觉
R=+10
学习
R=-2
0.4
自习
R=-2
0.2
约会
R=+5
自习
R=-2
0.8

- 注意，以上定义都具有 Markov 性和时齐性.

Bellman 期望方程

- 状态-价值函数可以被分解为：即时回报加后续状态的折扣价值，

$$v_\pi(s) = \mathbb{E}_\pi(R_{t+1} + \gamma v_\pi(S_{t+1})|S_t = s).$$

- 行动-价值函数可以被类似地分解，

$$q_\pi(s,a) = \mathbb{E}_\pi(R_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1})|S_t = s, A_t = a).$$

- 二者之间的关系（全概率公式、一步转移概率）：

$$q_\pi(s,a) = \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} P_{s,s'}^a v_\pi(s').$$

$$v_\pi(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}(q_\pi(s,a)) = \sum_{a\in\mathcal{A}} \pi(a|s) q_\pi(s,a),$$

Bellman 期望方程

- 因此，我们得到 MDP 的 Bellman 期望方程：

$$v_\pi(s) = \sum_{a\in\mathcal{A}} \pi(a|s) \left(\mathcal{R}_s^a + \gamma \sum_{s'\in\mathcal{S}} \mathcal{P}_{s,s'}^a v_\pi(s') \right),$$

$$q_\pi(s,a) = \mathcal{R}_s^a + \gamma \sum_{s'\in\mathcal{S}} \mathcal{P}_{s,s'}^a \sum_{a'\in\mathcal{A}} \pi(a'|s') q_\pi(s',a').$$

- 矩阵形式：

$$v_\pi = \mathcal{R}^\pi + \gamma \mathcal{P}^\pi v_\pi = (I - \gamma \mathcal{P}^\pi)^{-1} \mathcal{R}^\pi.$$

最优价值函数

- 最优状态-价值函数 $v_\star(s)$ 是所有决策中最大的状态-价值函数

$$v_\star(s) = \max_\pi v_\pi(s).$$

- 最优行动-价值函数 $q_\star(s,a)$ 是所有决策中最大的行动-价值函数

$$q_\star(s,a) = \max_\pi q_\pi(s,a).$$

- 最优价值函数确定了 MDP 中的最佳收益.
- 解 MDP 即确定达到最优价值函数的策略.

最优策略

- 然而，每个状态取到最大价值的策略 π 可能并不是同一个.
- 幸运的是，确实存在一个这样的最优策略. 定义一个策略的偏序:

$$\pi \geq \pi' \iff \forall s \in \mathcal{S}\ v_\pi(s) \geq v_{\pi'}(s).$$

定理 2.3 (MDP 解的存在性) *对任意 MDP,*

- *存在一个最优策略 $\pi_\star$ 使得 $\forall \pi\ \pi_\star \geq \pi$.*
- *最优策略取得最优状态-价值函数: $v_{\pi_\star}(s) = v_\star(s)$.*
- *最优策略取得最优行动-价值函数: $q_{\pi_\star}(s,a) = q_\star(s,a)$.*

寻找最优决策

- 可以通过最大化 $q_\star(s,a)$ 来寻找:
  - 固定 s.
  - 找到一个 $a_\star$ 使得 $q_\star(s,a_\star) = \max_a q_\star(s,a)$，令 $\pi_\star(a_\star|s) = 1$.
  - 对 $\forall a \neq a_\star$，$\pi_\star(a|s) = 0$.
- 证明: 根据选法，$\pi_\star$ 取得最优行动-价值函数.
- 由 $v_\pi(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}(q_\pi(s,a)) \leq \mathbb{E}_{a\sim\pi(\cdot|s)}(q_\star(s,a)) \leq q_\star(s,a_\star) = v_{\pi_\star}(s)$ 知 $\pi_\star$ 取得最优状态-价值函数.
- 推论: 对任意 MDP，总存在一个非随机的最优决策.
- 如果我们知道 $q_\star(s,a)$，我们就能获得最优决策.

Bellman 最优性方程

- 最优价值函数由 Bellman 最优性方程联系：

$$v_\star(s) = \max_a q_\star(s,a),$$

$$q_\star(s,a) = \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{s,s'}^a v_\star(s'),$$

$$v_\star(s) = \max_a \left\{ \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{s,s'}^a v_\star(s') \right\},$$

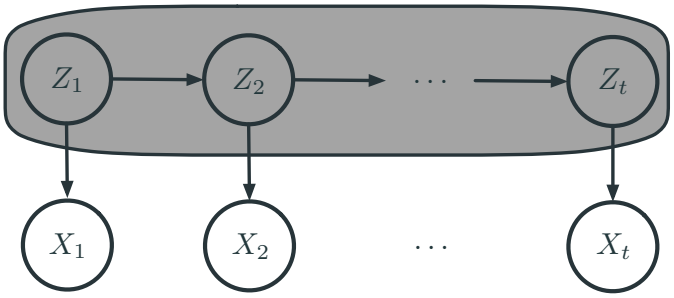
$$q_\star(s,a) = \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{s,s'}^a \max_{a'} q_\star(s',a').$$

解 Bellman 最优性方程

- Bellman 最优性方程不是线性的. 因此没有解析形式的（closed form）解.
- 但是 MDP 的数值解是可以多项式时间求出来的.
- 我们一般采用迭代算法求解：
  - 价值迭代（value iteration）
  - 策略迭代（policy iteration）
  - Q-learning
  - Sarsa

关于 Bellman 方程

- Bellman 方程是强化学习（reinforcement learning）、经济学动态优化（dynamic optimization）的核心.
- Bellman 方程的推导是 Markov 链中最为常用的技巧：考虑从当前状态转移到下一状态，利用全概率公式，一步转移会将两个状态之间的概率（期望）用递推公式联系起来.
  - 随机过程中的例子：前向方程、Wald 等式、调和函数（harmonic function）.
  - 后面的 HMM 也是类似的例子.



§2.4 隐 Markov 模型（HMM）

问题的引入

- 我们考虑 Markov 链上的另一种应用.
- 在统计学和机器学习中，我们有时候要处理一类含时间的数据.
- 最简单的情况是回归，即数据完全由所处时刻决定.
- 但是通常，现在的数据依赖于过去的数据.
- 因此，一种最简单的考虑就是数据依赖于 Markov 链，这就是隐 Markov 模型.

隐 Markov 模型

- 一个*隐 Markov 模型*(hidden Markov model，HMM)是一列随机变量 $X_1, X_2, \ldots, X_t$，满足：
  - X_t 的分布仅依赖于隐状态 Z_t，即 $\Pr(X_1, \ldots, X_t | Z_1, Z_2, \ldots, Z_t) = \prod_i \Pr(X_i | Z_i)$.
  - $\{Z_t\}$ 构成一条 Markov 链.

有限观测 HMM

- 一个 HMM 包含：
  - $\mathcal{Z}$：有限的状态集合.
  - $\mathcal{X}$: 有限的观测集合.
  - $T : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}_{\geq 0}$，$\mathcal{Z}$ 的转移概率.
  - $M : \mathcal{Z} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$，给定状态时的观测概率（条件概率）.
  - $\lambda : \mathcal{Z} \to \mathbb{R}_{\geq 0}$，初始状态的先验概率分布列.

- 如果随机过程 $\{X_t\}$ 的值域是有限集，我们则可以用矩阵表达 HMM.
  - T 是 $\{Z_t\}$ 的转移矩阵.
  - M 是观测矩阵：$M_{i,k} = \Pr(X_t = k | Z_t = i)$.
  - λ 是一个概率向量.

2.4.1 评估问题

HMM 的评估

- 给定一个特定的 HMM，它对实际观测序列的拟合程度有多好?
- 记号：随机向量 $X = (X_1, \ldots, X_t)$，$Z = (Z_1, \ldots, Z_t)$.
- HMM 的*评估*（evaluation）问题：给定一个 HMM $\mathcal{M}$，以及它的观测历史 $x = (x_1, x_2, \ldots, x_t)$，计算 $\Pr(X = x|\mathcal{M})$.
- 关键困难：我们不知道状态历史 $Z = (z_1, z_2, \ldots, z_t)$.

朴素方法

- 直接使用条件概率进行推导:

$$\Pr(X = x|\mathcal{M}) = \sum_{Z=(z_1,\ldots,z_t)\in\mathcal{Z}} \Pr(X = x|Z = z, \mathcal{M}) \Pr(Z = z|\mathcal{M}),$$

$$\Pr(X = x|Z = z, \mathcal{M}) = \prod_{i=1}^{t} \Pr(X_i = x_i|Z_i = z_i) = M_{z_1,x_1} \cdot M_{z_2,x_2} \ldots M_{z_t,x_t},$$

$$\begin{aligned}\Pr(Z = z|\mathcal{M}) &= \Pr(Z_1 = z_1) \prod_{i=2}^{t} \Pr(Z_i = z_i|Z_{i-1} = z_{i-1}) \\ &= \lambda_{z_1} \cdot T_{z_1,z_2} \cdot T_{z_2,z_3} \ldots T_{z_{t-1},z_t}.\end{aligned}$$

- 时间复杂度: $\mathcal{O}(t|\mathcal{Z}|^t)$.

前向算法

- 思路：类似前向方程，我们可以从前 k 步的结果推出前 $k+1$ 步的结果. 因此可以列出递推方程.
- 记号：$X_{i:j} = (X_i, \ldots, X_j)$.

- 具体地，定义 $\alpha_k(z) := \Pr(X_{1:k} = x_{1:k}, Z_k = z|\mathcal{M})$，我们有
  - $\alpha_1(z) = \lambda(z)M_{z,x_1}$.
  - $\alpha_{k+1}(z) = \sum_{z' \in \mathcal{Z}} \alpha_k(z')T_{z',z}M_{z,x_{k+1}}$.
- $\Pr(X = x|\mathcal{M}) = \sum_{z \in \mathcal{Z}} \alpha_t(z)$.
- 时间复杂度 $\mathcal{O}(t|\mathcal{Z}|^2)$.

后向算法

- 类似后向方程，从前 $k+1$ 步的结果推出前 k 步的结果. 同样可以列出递推方程.
- 定义 $\beta_k(z) := \Pr(X_{k+1:t} = x_{k+1:t}|Z_k = z, \mathcal{M})$，我们有
  - 当 $k = t$，$\beta_k(z) = 1$.
  - 当 $1 \le k < t$，$\beta_k(z) = \sum_{z' \in \mathcal{Z}} T_{z,z'}M_{z',x_{k+1}}\beta_{k+1}(z')$.
- $\Pr(X = x|\mathcal{M}) = \sum_{z \in \mathcal{Z}} \lambda(z)M_{z,x_1}\beta_1(z)$.
- 时间复杂度 $O(t|\mathcal{Z}|^2)$.

2.4.2 解释问题

HMM 的解释问题

- HMM 的解释（explanation）问题：给定一个 HMM $\mathcal{M} = (\mathcal{Z}, \mathcal{X}, T, M, \lambda)$, 一列观测历史 $x = (x_1, x_2, \ldots, x_t)$, 寻找一个状态序列，能最好地解释这些历史观察.
- 具体地，我们考虑如下四个问题
  1. 过滤（filtering）：计算 $\Pr(Z_k = s|X_{1:k} = x_{1:k}, \mathcal{M})$.
  2. 平滑（smoothing）：计算 $\Pr(Z_k = s|X = x, \mathcal{M})$，$k < t$.
  3. 预测（prediction）：计算 $\Pr(Z_k = s|X = x, \mathcal{M})$，$k > t$.
  4. 解码（decoding）：找到最有可能的状态序列 $z = (z_1, z_2, \ldots, z_t)$.

过滤：$\Pr(Z_k = s|X_{1:k} = x_{1:k}, \mathcal{M})$

- 回顾：$\alpha_k(s) = \Pr(X_{1:k} = x_{1:k}, Z_k = s|\mathcal{M})$.

- 我们有

$$\begin{aligned}\Pr(Z_k = s|X_{1:k} = x_{1:k}, \mathcal{M}) &= \frac{\Pr(X_{1:k} = x_{1:k}, Z_k = s|\mathcal{M})}{\Pr(X_{1:k} = x_{1:k}|\mathcal{M})} \\ &= \frac{\alpha_k(s)}{\sum_{z\in\mathcal{Z}} \alpha_k(z)}.\end{aligned}$$

平滑：$\Pr(Z_k = s|X = x, \mathcal{M})$，$k < t$

- 回顾：$\alpha_k(s) = \Pr(X_{1:k} = x_{1:k}, Z_k = s|\mathcal{M})$.

- 回顾：$\beta_k(s) = \Pr(X_{k+1:t} = x_{k+1:t}|Z_k = s, \mathcal{M})$.

- 可以证明：

$$\Pr(z_k = s|X = x, \mathcal{M}) = \frac{\beta_k(s)\alpha_k(s)}{\sum_{z\in\mathcal{Z}} \alpha_t(z)}.$$

预测：$\Pr(Z_k = s|X = x, \mathcal{M})$，$k > t$.

- 首先用过滤计算 $\lambda = \Pr(Z_t = s|X = x, \mathcal{M})$.

- 然后用 λ 作为 Markov 的初始状态，向前计算 $k - t$ 步.

* 解码：Viterbi 算法

- 定义

$$\delta_k(s) = \max_{Z_{1:k-1}} \Pr(Z_{1:k} = (z_{1:k-1}, s), X_{1:k} = x_{1:k}|\mathcal{M}).$$

- 根据一步转移，我们有

$$\delta_{k+1}(s) = \max_{q\in\mathcal{Z}}\{\delta_k(q)T_{q,s}\}M_{s,x_{k+1}}.$$

- 问题转化为: 记录最高概率的路径，这是一个动态规划问题.

* 解码: Viterbi 算法

- 初始化:
  - $\delta_1(s) = \lambda(s)M_{s,z_1}$.
  - $\mathrm{Pre}_1(s) = \varnothing$.
- 对 $k = 1, 2, \ldots, t-1$，$s \in \mathcal{Z}$:

- $\delta_{k+1}(s) = \max_{q \in \mathcal{Z}} \{\delta_k(q) T_{q,s}\} M_{s,x_{k+1}}$.
- $\mathrm{Pre}_{k+1}(s) = \mathrm{argmax}_{q \in \mathcal{Z}} \{\delta_k(q) T_{q,s}\}$.

- $z_t = \mathrm{argmax}_{s \in \mathcal{Z}} \delta_t(s)$.
- 对 $1 \leq k < t$，$z_k = \mathrm{Pre}_{k+1}(z_{k+1})$.
- 时间复杂度: $\mathcal{O}(t|\mathcal{Z}|^2)$.

第二部分

信息与数据

第三章　信息论基础

信息是什么？不同于真实的物理世界，信息仿佛看不见，摸不着. 然而，任何人都可以体会到信息的存在，信息是我们认识世界的基础. 信息的存在正如同物理世界中的能量、动量一般，抽象而具有一般性. 信息论已经在计算机、AI、认知理论等诸多领域中得到了广泛的应用. 本章探讨信息论的基础，并给出他们在 AI 中的一些应用.

在第 3.1 节，我们讨论熵的概念与性质. 在第 3.2 节，我们讨论 Kullback-Leibler 散度的概念与性质. 在第 3.3 节，我们给出 Shannon 定理证明.

§3.1　熵

3.1.1　概念的导出

我们常说“恐惧来源于未知”，信息似乎代表着某种确定的东西，某种知识，因而和不确定性有相反的关系. 更精确地说，消除不确定性的东西被称为信息. 当然，这句话本身似乎是一种循环论证，它并没有真正回答信息或者不确定性到底是什么. 所以我们进一步的问题是，给定一个“对象”，如何定量衡量它不确定性（或信息量）?

然而，单个对象的信息是一个非常难以划定的概念. 同样的内容，对于不同的人来说，信息量是完全不同的. 比如说，已经学过信息论的读者再看这一部分内容，他获得的信息一定比没有学过的读者要少得多. 因而实际上，一种更加容易的办法是我们将世界视为不确定的，因而有多种可能的对象，然后考虑这一堆对象的信息量. 比如说，这本书的读者的背景是不确定的，可能学过信息论，也可能没学过，但是我们可以综合考虑不同读者的背景，然后给出一个信息的概率分析.

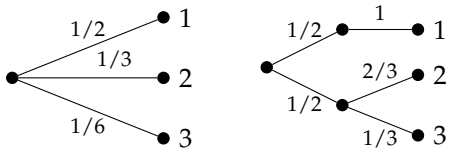
我们可以用数学来表述上面的考虑，假如我们进行一次试验，一共有 n 种可能的结果，第 i 种发生的概率为 p_i. 我们预测试验的结果，如果越能正确地预测，那么就说明我们对这个试验中包含的信息知道的越多. 假如 $p_1 = 1$，那么我们完全确定试验一定会产生结果 1. 如果 $p_i = 1/n$，那么我们完全无法预计试验的结果. 我们对试验结果的预期与

试验结果的概率分布有密切联系. 因此概率分布给我们带来了**信息**，使得我们能够产生不同的判断. 另一方面，概率分布带来了**不确定性**，使我们不能总是确信预言会成真.

我们遵循“信息论之父”Shannon 的思路，为信息提供一个严格的数学模型：熵. 假设随机变量 X 表示了所有可能的结果（编号为 1 到 n），$\Pr(X=i)=p_i$，$p=(p_1,\dots,p_n)$，有时候也把 p_i 写作 $p(i)$. 我们把不确定性度量记为 $H(p)$. Shannon 假设 H 满足以下三个性质：

1. H 是一个连续函数.
2. 事件结局可能数变多则不确定性增大：$p_i=1/n$ 时，$H(p)$ 随 n 单调递增，n 是正整数.
3. 如果一个试验被分成了两个相继的试验，那么原来的 H 应该等于分开之后的 H 的加权和.

注. 第三个假设可以用下图来理解.



假设我们有一个试验，有三种可能的结果，1,2,3，概率分别为 1/2,1/3,1/6. 该试验的不确定性是 $H(1/2,1/3,1/6)$. 我们把试验分成两步相继的试验，第一步试验有两种可能的结果，概率分别都是 1/2. 当第一步试验出现上面的结果时，第二步试验以概率 1 产生结果 1；当第二步试验出现下面的结果时，第二步试验以概率 2/3 产生结果 2，以概率 1/3 产生结果 3. 我们可以看到，分成两步之后，第一步试验的不确定性是 $H(1/2,1/2)$，第二步试验的不确定性有一半概率是 $H(1)$（上面的分支），有一半概率是 $H(2/3,1/3)$（下面的分支），因而加权的不确定性是 $1/2\cdot 0+1/2\cdot H(2/3,1/3)$. 因此第三个假设可以具体表述为

$$H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)=H\left(\frac{1}{2},\frac{1}{2}\right)+\left[\frac{1}{2}\cdot H(1)+\frac{1}{2}\cdot H\left(\frac{2}{3},\frac{1}{3}\right)\right].$$

这里，我们可以看出 Shannon 的哲学思想：不确定性只来自于概率分布而不是具体对象. 他的考虑具有浓厚的工程意味，正如他自己针对通信的数学理论所说：“消息是具有含义的……然而，通信的语义层面并不是工程问题所关心的.“正是因为抽象掉了具体考虑的对象，信息论的应用才变得如此广泛.

基于上面三个假设，Shannon 证明了如下定理，这一定理直接给出了熵的概念.

定理 3.1 (Shannon 定理) *H 满足三个假设当且仅当*

$$H(p) = -C\sum_i p_i \log p_i,$$

其中 C 是正常数，$0\log 0 = 0$.

这一定理的证明较长并且和后面的讨论关联较小，所以我们在第 3.3 节 中给出证明.

根据对数的的换底公式，可以将 $C\log p_i$ 写为 $\log_b p_i$，这里 $C = 1/\log b$. 于是，Shannon 定理直接给出了熵的如下定义：

定义 3.1 (熵) *分布列 $p = (p_1, \ldots, p_n)$ 的熵定义为*

$$H(p) = -\sum_{i=1}^{n} p_i \log_b p_i.$$

其中 $b = \mathrm{e}$（自然对数底数），$0\log 0 = 0$. 当 $b = 2$ 时，我们记熵为 $H_2(p)$.

通常来说，使用 e 作为底数会使得数学推导简洁，而用 2 为底数则常常是讨论信息量时的习惯. 在后面通信理论中，我们将讨论熵在通信中的含义，以 2 为底的时候熵的实际意义会更清楚些. 如果没有特别强调，我们在讨论时总是假设 $b = \mathrm{e}$.

熵的定义还可以用数学期望的形式写出. 假设 X 的分布列是 p，$p(i) = \Pr(X = i)$，那么我们也可以把熵写成期望的形式：

$$H(p) = -\mathbb{E}[\log p(X)].$$

每一个（离散）随机变量 X 会确定一个分布列 p_X，因此我们也可以定义随机变量的熵：

定义 3.2 (随机变量的熵) *随机变量 X 的熵定义为*

$$H(X) = -\mathbb{E}[\log p_X(X)].$$

其中 p_X 是 X 的分布列，$0\log 0 = 0$.

尽管从信息论的角度我们可以唯一确定熵的定义，但是熵的概念在物理学上早就已经存在. 下面我们给出统计力学中熵的推导过程. 在经典力学中，物理系统的状态由粒子的位置和动量（速度）完全确定，将粒子位置和动量可能的值集合称为*相空间*，于是物理系统的演化就是相空间中的粒子状态的变化. 将相空间等分成 m 个单元，编号 1 到 m. 假设相空间中有 N 个可区分的粒子，相互独立，没有相互作用，每个粒子等可能出现在每一个单元中. 如果单元 i 中有 N_i 个粒子，那么按照粒子在单元中的分布来看，系统处于某个特定状态的概率为

$$P = \frac{N!}{N_1! \ldots N_m!} \left(\frac{1}{m}\right)^N.$$

这是一个多项分布. 两边取对数，得

$$\log P = \log(N!) - \sum_i \log(N_i!) - N \log m.$$

考虑充分大的 N_i，由 Stirling 公式，有

$$\log(N_i!) \sim \log\left(\sqrt{2\pi N_i}\left(\frac{N_i}{\mathrm{e}}\right)^{N_i}\right) \sim N_i \log N_i.$$

因此，

$$\log P \sim N \log N - \sum_i N_i \log N_i - N \log m \sim N \log N - \sum_i N_i \log N_i. \tag{3.1}$$

假设 N_i 充分大的时候，N_i/N 呈现固定的比例 p_i，那么

$$\begin{aligned} N \log N - \sum_i N_i \log N_i &\sim N \log N - \sum_i N p_i \log(N p_i) \\ &= -N \sum_i p_i \log p_i. \end{aligned}$$

$\log P \sim -N \sum_i p_i \log p_i$. 于是我们证明了：

$$\frac{1}{N} \log P \to H(p_1, \ldots, p_m), \quad N \to \infty.$$

因此，熵刻画了充分多粒子的物理系统某种特定状态出现概率！熵越大的系统越有可能达到. 更进一步，在统计力学中有 Boltzmann H-定理：孤立的粒子系统会向着熵（H）增加的方向演化，并最终达到熵最大的状态. H-定理是热力学第二定律的微观解释，熵越大的系统出现概率越大、越混乱、越接近均衡.

3.1.2 概念与性质

现在，我们将进一步探讨熵的若干拓展定义，并讨论他们的性质.

首先，我们考虑最简单的情形，即分布列为 (p_1, p_2)，此时，我们不妨设 $p_1 = p$，$p_2 = 1 - p$，那么熵就是

$$H(p_1, p_2) = H(p, 1-p) = -p \log p - (1-p) \log(1-p).$$

H 是关于 p 的函数，作图如图 3.1 所示.

利用导数的方法，很容易证明：

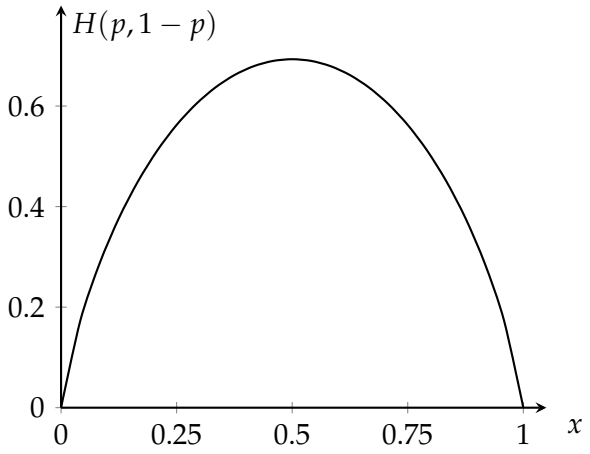


图 3.1: 熵 $H(p)$ 的图像.

命题 3.1 *$H(p)$ 在 $p \in (0, 1/2)$ 严格单调递增，在 $p \in (1/2, 1)$ 严格单调递减. 它的最小值是 0，在 $p \in \{0, 1\}$ 取得；它的最大值是 $\log 2$，在 $p = 1/2$ 取得.*

这与我们对于"不确定性"的直觉是相一致的：当 p 接近 0 或 1 时，我们对于 X 的取值几乎是确定的，因此熵接近 0；当 p 接近 $1/2$ 时，我们对于 X 的取值几乎是完全不确定的，因此熵接近最大值 $\log 2$. 实际上，这样的性质对于一般的分布也是成立的.

考虑一般分布的熵 $H(p) = H(p_1, \ldots, p_n)$. 我们有如下性质：

命题 3.2 *$H(p) \geq 0$，等号成立当且仅当某个 $p_i = 1$.*

证明 这是一个典型的证明，主要的技巧是使用熵的期望形式. 考虑随机变量 X，其分布列为 p. 回忆 Jensen 不等式：如果 f 是一个严格凸函数，那么

$$\mathbb{E}[f(X)] \geq f(\mathbb{E}[X]).$$

等号成立当且仅当 X 是常数.

因为 $-\log(\cdot)$ 是严格凸函数，所以根据 Jensen 不等式

$$H(X) = \mathbb{E}[-\log p(X)] \geq -\log \mathbb{E}[p(X)] \geq -\log 1 = 0.$$

等号成立当且仅当 X 是常数，即对某个 i，$p(i) = 1$. □

命题 3.3 *p_i 朝着相等方向改变的时候 H 增加. 也就是说，假设 $p_i < p_j$，再假设 $p_i' > p_i, p_j' < p_j$ 且 $p_i + p_j = p_i' + p + j'$. 用 p_i' 和 p_j' 代替原来的 p_i 和 p_j，那么 H 变大.*

证明 为简化符号，考虑 $i=1$ 和 $j=2$. 利用假设三，第一步试验中，将试验的结果 1 和结果 2 合并，第二步试验再按照 $p_1/(p_1+p_2)$ 和 $p_2/(p_1+p_2)$ 的概率产生结果 1 和结果 2. 于是，

$$\begin{aligned}
&H(p_1,p_2,\dots)\\
=&H(p_1+p_2,p_3,\dots)+(p_1+p_2)H\left(\frac{p_1}{p_1+p_2},\frac{p_2}{p_1+p_2}\right) \quad \text{（假设三）}\\
\leq&H(p_1+p_2,p_3,\dots)+(p_1+p_2)<H\left(\frac{p_1'}{p_1'+p_2'},\frac{p_2'}{p_1'+p_2'}\right) \quad \text{（命题 3.1）}\\
=&H(p_1',p_2',p_3,\dots). \quad \text{（假设三）}
\end{aligned}$$

□

命题 3.4 *当 $p_1=\cdots=p_n=1/n$ 时 H 取得最大值 $\log n$.*

证明 若存在 $p_i\neq p_j$，因为 $\sum_i p_i/n=1/n$，根据鸽巢原理，则必有 i,j 满足 $p_i<1/n<p_j$. 根据命题 3.3，我们可以将 p_i 和 p_j 替换为 $1/n$ 和 p_i+p_j-1/n，而 H 增大. 只要还有两个 p_i 不相等，这一过程就可以重复，每一次都会增大 H，直到所有 p_i 都等于 $1/n$. □

至此，命题 3.2 和命题 3.4 证明了一般情形的命题 3.1. 在等可能的时候不确定性最大，熵最大；在确定事件的时候不确定性最小，熵最小. 所以熵是符合直观的定义.

接下来，我们讨论熵的拓展形式.

在一次试验中，我们可以观察多个变量，比如说 X 和 Y. 我们也可以说，我们观察到了一个结果 (X,Y)，服从分布 $p(i,j)$. 因此有对应的熵，这就是**联合分布的熵**：

$$H(X,Y)=-\mathbb{E}[\log p(X,Y)].$$

对应地，我们也可以写成和的形式：

$$H(p)=-\sum_{i,j}p(i,j)\log p(i,j).$$

自然，联合分布也可以引出**边缘分布的熵**：

$$H(X)=-\mathbb{E}[\log p_X(X)]=-\sum_i\sum_j p(i,j)\log\sum_j p(i,j).$$

$$H(Y)=-\mathbb{E}[\log p_Y(Y)]=-\sum_j\sum_i p(i,j)\log\sum_i p(i,j).$$

有了两个随机变量，我们就可以讨论“条件”的概念. 具体来说，我们可以把试验分为两步，第一步观测 X，第二步观测 Y，那么，第二步所产生的熵就是已经知道第一步结果之后的熵，即：

$$H(Y|X=x) = -\mathbb{E}[\log p_{Y|X=x}(Y)|X=x] = -\sum_j p_{Y|X=x}(j)\log p_{Y|X=x}(j),$$

其中 $p_{Y|X=x}(j) = p(x,j)/p_X(x)$. 当我们知道了 $X=x$ 之后，对 Y 的观测就消除了部分的不确定性，因此根据我们对于不确定性和信息关系的讨论，从 $X=x$ 中获得的关于 Y 的信息是

$$I(X=x:Y) = H(Y) - H(Y|X=x).$$

考虑一个特殊情况，$Y=X$，那么刚刚的讨论就变成了自己从自己身上获得的信息，或者说知道 $X=x$ 带来的信息量. 首先有

$$p_{X|X=x}(i) = \begin{cases} 1, & i=x \\ 0, & i\neq x. \end{cases}$$

因此，

$$H(X|X=x) = -\sum_j p_{X|X=x}(j)\log p_{X|X=x}(j) = -1\log 1 = 0.$$

于是，

$$I(X=x:X) = H(X) - H(X|X=x) = H(X).$$

这正是定量版本的“消除不确定性的东西被称之为信息”! 此外，我们之前说过，熵刻画的是一族可能对象的信息，这一点也反映在了这一公式中：只要知道了 X 的值，无论它具体是多少，我们得到的信息量是一样的！

再回到一般情况，还是同样的两步试验，我们定义给定 X 时 Y 的**条件熵**为

$$\begin{aligned} H(Y|X) =& \mathbb{E}[H(Y|X=x)] \\ =& -\mathbb{E}[\log p_{Y|X}(Y)] \\ =& -\sum_x p_X(x)\sum_j p_{Y|X=x}(j)\log p_{Y|X=x}(j) \\ =& -\sum_{x,j} p(x,j)\log p_{Y|X=x}(j). \end{aligned}$$

换言之，我们现在进一步假定 X 也是不知道的，于是 $H(Y|X)$ 就是平均上来说第二步中 Y 的不确定性. 条件熵和熵有着类似的性质：

命题 3.5 *$H(Y|X)\geq 0$，等号成立当且仅当 Y 是退化的，即 Y 概率 1 只取一个值.*

证明 仿照命题 3.2 的证明即可. □

类似地，我们可以考虑平均上 Y 中包含的关于 X 的信息量：

$$\mathbb{E}[I(X = x : Y)] = H(Y) - H(Y|X).$$

与之相对应地，平均上 X 中包含的关于 Y 的信息量为

$$\mathbb{E}[I(Y = y : X)] = H(X) - H(X|Y).$$

一个自然的问题是，二者相互包含的信息量是什么关系？根据概率的链式法则，$p(x,y) = p_{X|Y}(x|y)p_Y(y)$，带入 $H(X,Y)$ 的定义得熵的链式法则：

命题 3.6 *对任意离散随机变量 X,Y，$H(X,Y) = H(Y) + H(X|Y)$.*

利用链式法则，我们注意到，$H(X) - H(X|Y) = H(X) - (H(X,Y) - H(Y)) = H(X) + H(Y) - H(X,Y) = H(Y) - H(Y|X)$. 所以，$X$ 中包含的 Y 的信息和 Y 中包含的 X 的信息是一样多的！此外，直观上我们还应该觉得，信息量不能是负的，实际上的确如此：

命题 3.7 *$H(X) - H(X|Y) \geq 0$，等号成立当且仅当 X 和 Y 相互独立.*

我们将在第 3.2 节看到，命题 3.7 就是 K-L 散度信息不等式的一个特例，所以我们就不在这里给出证明了. 命题 3.7 表明知道任何信息都不会增加不确定性，这个原理被称为“Information doesn't hurt.”根据以上讨论，我们可以自然地定义 X 和 Y 的**互信息**为 $I(X;Y) = I(Y;X) = \mathbb{E}[I(X = x : Y)] = \mathbb{E}[I(Y = y : X)]$.

类似联合分布的熵，条件熵和互信息的概念也可以推广到多元情形. 对于三个随机变量 X,Y,Z，我们可以定义条件熵为

$$H(X,Y|Z) = H(X,Y,Z) - H(Z).$$

类似地，我们可以定义互信息为

$$I(X,Y;Z) = H(X,Y) - H(X,Y;Z).$$

他们的含义以及性质和二元情形类似.

同样，我们可以定义**条件互信息**为 $I(X;Y|Z) = H(X|Z) - H(X|Y,Z)$，表明 Z 已知时候 Y 给 X 带来的平均信息增益. 类似互信息，我们如下性质：

命题 3.8 *条件互信息满足以下性质：*

1. 非负性：$I(X;Y|Z) \geq 0$，等号成立当且仅当 X 和 Y 在给定 Z 的条件下相互独立.

2. 对称性：$I(X;Y|Z) = I(Y;X|Z)$.

3. 链式法则：$I(X,Y;Z) = I(X;Z|Y) + I(Y;Z)$.

4. 条件信息量：$I(X:X|Y) = H(X|Y) - H(X|X,Y) = H(X|Y)$.

最后一条性质说的其实是，在平均的意义下，给定 Y 的时候，知道 X 所能够得到的额外信息量就是 $H(X|Y)$. 这一命题的证明和前面都非常相似，我们留做习题.

最后，我们将各种熵以及信息量的关系总结为图 3.2. 在集合论中，这样的图被称为 Venn 图，所以我们可以用集合论来理解信息与熵. 对应关系可以总结为表 3.1.

信息论	集合论
$H(X)$	A
$H(Y)$	B
$H(X\|Y)$	$A \setminus B$
$H(X,Y)$	$A \cup B$
$I(X;Y)$	$A \cap B$

表 3.1: 信息论和集合论的对应关系.

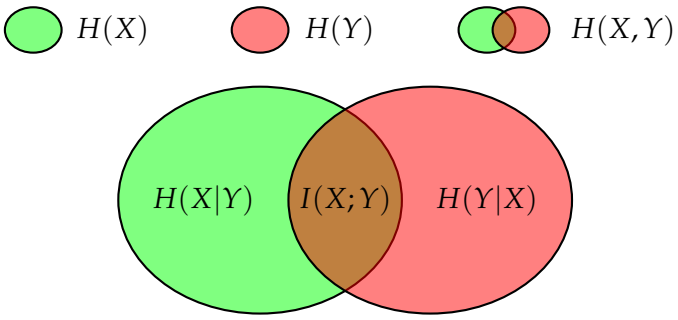


图 3.2: 熵和信息量的关系.

3.1.3 熵与通信理论

最早的时候，Shannon 建立信息论，就是为了给通信理论一个数学基础. 从通信的角度出发，我们可以更本质地理解信息和熵.

通信就是一个发射端和一个接收端，中间有信道传递消息. 将所有可能要传递的消息集合记为 Ω（一个有限集），我们现在考虑 Ω 所蕴含的信息量是多少. 注意到，根据 Shannon 的思想，Ω 里面具体是什么并不重要，重要的是有多少个. 我们可以用自然数 $1, 2, \dots$ 表示集合 Ω 里的元素. 那么，使用二进制编码，我们至少需要 $\log_2 |\Omega|$ 个比特来表示 Ω 里的元素. 于是，假如说随机变量 X 表示收到的消息，那么 X 的熵就定义为 $H(X) = \log_2 |\Omega|$，它衡量了接收端收到的消息的不确定性. 当我们选定了具体的消息 $m \in \Omega$，X 的不确定性被消除了，于是 $X = a$ 的过程产生了（或者说传递了）$\log_2 |\Omega|$ 比特的信息. 比如说，我们发送一个长为 n 的二进制序列，消息的集合大小就是 2^n，发送任何一条具体的消息，我们就传递了 n 比特的信息.

有时候，我们会把消息看成一个序列. 具体来说，我们可以发送独立的 k 条消息，其中第 i 条 X_i 来自消息集合 Ω_i，$|\Omega_i| = n_i$，那么 $(X_1, \dots, X_k)$ 的熵就是

$$H(X_1, \dots, X_k) = \log_2 n_1 + \cdots + \log_2 n_k,$$

它衡量了 k 条消息的不确定性. 在更常见的情况下，每次发送的其实不是一条消息，而是一个字母，所有的字母组成了一个字母表，我们用 $\Sigma = \{x_1, \dots, x_s\}$ 来表示. 于是，X_i 就是消息的第 i 个字母，于是，一条消息可以写作 $X_1 \dots X_k$，其中每一个 X_i 都来自 Σ.

我们现在考虑更加简单的情形，即每一个字母 X_i 其实是同一个随机变量 X 的独立采样. 如果我们具体知道某一个 x_i 出现的次数，那么我们其实可以有更高效的传递信息的方式. 譬如说，在极端情况下，如果只有 x_1 和 x_2 会出现，那么我们其实只需要 $\log_2 2 = 1$ 比特就足够传递所有消息了. 在一般情况下，考虑 Ω 中只包含长为 k 的消息，并且 x_i 在消息中出现 k_i 次，那么所有可能的消息数量为

$$N(k) = \frac{k!}{k_1! \cdot \dots \cdot k_s!}.$$

假定我们需要 $h(\omega)$ 比特来具体确定发的消息是 ω. 首先，无序集合本身需要 $\log_2 |\Omega|$ 比特来编码，其次，我们还需要确定 $(k_1, \dots, k_s)$，确定它的一种方式是按照顺序给出每一个 k_i. 每个 k_i 最多需要 $\log_2 k$ 比特来表示，所以按顺序表示所有的 k_i 至多需要 $s \log_2 k$ 比特. 于是，我们需要的比特数为

$$\log_2 \frac{k!}{k_1! \cdot \dots \cdot k_s!} \le h(m) \le s \log_2 k + \log \frac{k!}{k_1! \cdot \dots \cdot k_s!}.$$

这刚好和我们在统计力学中推导熵的过程是一致的！假设消息足够的长，x_i 出现的频率逐渐接近 p_i，那么同样的推理我们可以知道，

$$h(m) \sim -k \sum_i p_i \log_2 p_i = k H_2(p_1, \dots, p_s).$$

因此，如果知道字母的出现频率，我们传递单位长度的消息至少需要 $H(p_1,\dots,p_s)$ 比特，这完全给出了熵的具体含义，而且，我们现在也不难理解熵的形式为何会出现 log 了：熵就是期望上编码一个字母需要的比特数（即 $\log(1/p(X))$）.

那么，是否有一种编码确实达到了这个理论上的编码长度下界呢？答案是肯定的，它被称为 *Huffmann 编码*. 它的核心思想在于把出现频率高的字母用更短的编码表示. 类似的思想被用在了机器学习的决策树中，作为选择节点非常常用的一种依据.

> **注.** 决策树是一种常用的机器学习分类模型. 假设数据有很多属性 $P_1,\dots,P_k$，这些属性共同决定了某一条数据的类别. 比如，在银行的信用系统中，给定了一个人的性别、是否已婚、是否负债等信息，我们希望给他评估一个信用评级. 决策树的做法是，将决策过程写成一棵树，然后叶节点是决策类别的结果. 比如说，我们会先看这个人是否负债，如果不负债，那么看是否已婚，如果已婚，那么我们信用评级就给 *A*. 那么，如何选择每个节点需要去判断的属性呢？树本身其实就是一种广义的消息，从根节点沿着树走到叶节点得到的就是一条消息. 直观上，如果先选择带来信息增益比较高的属性，那么我们就可以用更少的比特来表示这条消息，或者说，我们决策树结构更加简单. 这样的选择方式叫做 ID3 策略.

我们进一步的问题是，为什么我们知道了每个字母的频次就可以压缩编码？我们接下来将要说明，其实长为 k 的消息中的“典型的消息”数量是远远少于所有 k 长消息的数目，因此我们实际上相当于只是针对一个子集进行编码. 注意到，当 k 充分大的时候，

$$\log_2 N(k) \sim h(m) \sim kH_2(p_1,\dots,p_s).$$

因此，

$$N(k) \approx 2^{kH_2(p_1,\dots,p_s)} = \mathrm{e}^{kH(p_1,\dots,p_s)}.$$

然而，长为 k 的所有消息数目为

$$s^k = \mathrm{e}^{k\log s}.$$

根据命题 3.4，只有当所有 p_i 相等的时候 $N(k)$ 才会达到这一量级. 从这个意义上说，熵所刻画的信息量定量刻画了数据压缩可能的极限.

以上关于信息编码下界以及数据压缩的讨论，再更一般的情况下也成立，此时这样的性质被称为*渐近等分性*. 而这一性质成立对应的结果被称为 *Shannon–McMillan–Breiman* 定理，它的陈述以及证明都需要用到更多随机过程的知识，这里就不再给出了.

> **注.** 现代的主流信息论都是从 *Shannon* 发展起来的. 然而，这一信息论也有很多问题. 首先，信息论使用了概率论进行建模. 但我们已经看到，概率要么是作为频率的近似理论（频率学派），要么反映了人们对未知的信念（主观学派）. 无论哪种解释，都将问题简化了. 正如 *Kolmogorov* 所说：“如果事情没有按照我们的预期发展，那么问题一定出在我们对于概率和真实世界的随机之间关系不清晰的认识上.”其次，这一信息论考虑的是一族对象的信息. 我

们是否能够用这样的方式来衡量单个对象的信息量呢？比如，我们要考虑这本书中包含的信息量，是它放在所有可能的书的集合中去考虑呢，还是把它的每一个章节分开考虑成一个随机序列呢？因此，信息论并不能很好地回答“单个对象”的信息量的问题.

现代概率论的奠基人 *Kolmogorov* 也非常严肃地考虑了这一问题. 他提出了被后世称为 ***Kolmogorov* 复杂度**的概念，旨在刻画一个随机字符串的随机程度. 简单来说，一个字符串的 *Kolmogorov* 复杂度就是描述它所需要的最短的代码长度，越随机的字符串就越需要更复杂的程序去描述它的产生方式. 利用这一概念，我们可以将信息的概念变成一个对象自己的属性，而不再需要把对象放在可能的一堆对象中去考虑. 这是信息论的另一种构建思路.

§3.2 Kullback-Leibler 散度

3.2.1 定义

为了引入 K-L 散度，我们从互信息出发. 它的定义是：

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= -\sum_x p_X(x)\log p_X(x) + \sum_{x,y} p(x,y)\log\frac{p(x,y)}{p_Y(y)} \\ &= -\sum_{x,y} p(x,y)\log p_X(x) + \sum_{x,y} p(x,y)\log\frac{p(x,y)}{p_Y(y)} \\ &= \sum_{x,y} p(x,y)\log\frac{p(x,y)}{p_X(x)p_Y(y)}. \end{aligned}$$

根据命题 3.7，$I(X;Y)\geq 0$，等号成立当且仅当 X,Y 相互独立，即 $p(x,y)=p_X(x)p_Y(y)$. X,Y 之间的互信息越大，说明他们之间的关联越强，分布越不独立，$p(x,y)$ 越不接近 $p_X(x)p_Y(y)$. 实际上，这样的想法可以被推广到一般分布上.

我们从数理统计的视角出发，考虑两个概率分布的似然函数p_1 和 p_2（也就是他们的分布列）. 抽取一个样本 X，考虑假设检验问题：

$$H_1\text{：样本 } X \text{ 来自 } p_1 \text{ 的分布} \quad \text{vs.} \quad H_2\text{：样本 } X \text{ 来自 } p_2 \text{ 的分布}$$

假设检验中有一种很常用的技巧，称为*似然比检验法*，即考虑两个假设分布的似然比 p_1/p_2. 如果这个比值越大，就越说明 p_1 的值更大，因而更有可能，倾向于接受 H_1，反之则越倾向于接受 H_2. 于是，可以自然定义区分 H_1 和 H_2 的检验量为对数似然比：

$$\log(p_1(x)/p_2(x)).$$

假设 H_1 是真的，那么在 H_1 的世界里，这个检验量的期望为

$$\mathbb{E}_{X\sim p_1}(\log(p_1(X)/p_2(X))) = \sum_i p_1(i)\log\frac{p_1(i)}{p_2(i)}.$$

实际上，上面的期望就是 K-L 散度的定义.

定义 3.3 (Kullback-Leibler 散度) *对于两个概率分布列 p_1, p_2，他们的* ***Kullback-Leibler 散度****或***相对熵***被定义为*

$$D(p_1\|p_2) = \mathbb{E}_{X\sim p_1}(\log(p_1(X)/p_2(X))) = \sum_i p_1(i)\log\frac{p_1(i)}{p_2(i)}.$$

其中规定 $0\log(0/0) = 0$，$0\log(0/a) = 0$，$a\log(a/0) = +\infty$.

我们马上知道，互信息是 K-L 散度的一种特殊情况：

命题 3.9 *对于两个随机变量 X, Y，成立 $I(X;Y) = D(p_{X,Y}\|p_X p_Y)$，其中 $p_{X,Y}$ 是 X, Y 的联合分布列，p_X, p_Y 分别是 X, Y 的边缘分布列.*

K-L 散度可以看成两个分布之间的区分衡量标准，但他不是度量. 一般来说，甚至连对称性都不成立. 例如，设 p_1 和 p_2 都是定义在 $0, 1$ 上的 Bernoulli 分布，参数分别为 $1/2$ 和 $1/4$. 于是

$$D(p_1\|p_2) = \frac{1}{2}\log\frac{1/2}{3/4} + \frac{1}{2}\log\frac{1/2}{1/4} = \frac{1}{2}\log\frac{4}{3}.$$

$$D(p_2\|p_1) = \frac{3}{4}\log\frac{3/4}{1/2} + \frac{1}{4}\log\frac{1/4}{1/2} = \frac{1}{2}\log\frac{3\sqrt{3}}{4}.$$

这两个值是不相等的.

我们在定义中还提到了 K-L 散度的另一个名字——相对熵. 实际上，这可以从编码中看出来. 假设事实上消息中字母的分布是 p_1，那么期望上编码单位长度消息需要的比特数是 $H(p_1) = \mathbb{E}_{X\sim p_1}[\log p_1(X)]$. 如果我们错误地认为消息中字母的分布是 p_2 并使用最优编码，那么实际上期望编码单位长度消息需要的比特数是 $\mathbb{E}_{X\sim p_1}[\log p_2(X)]$. 由于错误的认识所产生的额外编码长度是

$$\mathbb{E}_{X\sim p_1}[\log p_1(X) - \log p_2(X)] = D(p_1\|p_2).$$

根据在第 3.1.3 节 中的讨论，我们知道，额外的编码长度代表的是额外的不确定性，因而这一概念是某种“熵”的概念. 这正是“相对熵”的由来，$D(p_1\|p_2)$ 表示了当我们错误地把 p_1 当成 p_2 时带来的额外的不确定性，或者说额外的信息损失.

在机器学习中，比起讨论 K-L 散度，更加常用的是直接讨论量 $\mathbb{E}_{X\sim p_1}[\log p_2(X)]$. 从机器学习的观点来说，$p_1$ 是真实的分布，而 p_2 是我们所学习到的分布. 根据刚刚的讨论，这个量越小越说明 p_2 接近真实的 p_1，因此这又是一种衡量两个分布之间关系的量，我们称之为*交叉熵*：

定义 3.4 (交叉熵) 给两个随机变量 X, Y，X 的分布为 p_X，Y 的分布为 p_Y，则 X 的分布 p_X 和 Y 的分布 p_Y 的**交叉熵**[1] 为

$$CH(p_X, p_Y) = \mathbb{E}_{X\sim p_X}[\log p_Y(X)] = -\sum_i p_X(i) \log p_Y(i).$$

在机器学习的分类问题中，我们希望学习到的分布 p_Y 尽可能地接近真实的分布 p_X，所以我们训练的目标经常是最小化交叉熵 $CH(p_X, p_Y)$. 有趣的是，从数理统计的角度来看，最小化交叉熵等价于进行最大似然估计，因此这为最大似然估计提供了一种信息论意义下的理解. 相关讨论留作练习.

3.2.2 两个关于信息的不等式

利用 K-L 散度，我们可以给出两个关于信息的不等式，它们分别是信息不等式和数据处理不等式.

定理 3.2 (信息不等式) 对于两个概率分布列 p, q，成立 $D(p\|q) \geq 0$，当且仅当 $p = q$ 时取等号.

证明 由于 $\log x$ 是凸函数，所以由 Jensen 不等式，我们有

$$D(p\|q) = -\mathbb{E}_{X\sim p}\left[\log \frac{q(X)}{p(X)}\right] \geq -\log \mathbb{E}_{X\sim p}\left[\frac{q(X)}{p(X)}\right] = -\log \sum_i p(i) \cdot \frac{q(i)}{p(i)} = 0.$$

因此，$D(p\|q) \geq 0$，当且仅当 $p = q$ 时取等号. □

信息不等式表明，K-L 散度虽然不是度量，但却是非负的，因而确实可以被作为熵，用来衡量“额外的不确定性”. 此外，命题 3.7 是信息不等式的直接推论. 利用类似的方法，我们可以证明条件互信息的非负性（即命题 3.8 中的第一条）.

接下来我们叙述并证明数据处理不等式.

定理 3.3 (数据处理不等式) 假设随机变量 X, Y, Z 形成了 *Markov* 链，那么 $I(X;Y) \geq I(X;Z)$. 特别地，对任意函数 f，成立 $I(X;Y) \geq I(X;f(Y))$.

证明 根据互信息链式法则，

$$\begin{aligned} I(X;Y,Z) &= I(X;Z) + I(X;Y|Z) \\ &= I(X;Y) + I(X;Z|Y). \end{aligned}$$

[1] 文献中，经常会直接写为 $H(p_X, p_Y)$，但是在本书中为了区分熵，我们使用了符号 CH.

根据 Markov 性，条件在 Y 上，X 和 Z 相互独立. 因此，$I(X;Z|Y)=0$，根据条件互信息的非负性，$I(X;Y|Z)\geq 0$，所以 $I(X;Y)\geq I(X;Z)$.

显然，$X,Y,f(Y)$ 也形成了 Markov 链，所以 $I(X;Y)\geq I(X;f(Y))$. □

数据处理不等式表明，无论我们对随机变量 Y 进行了何种处理，甚至是允许带随机的处理，它的信息量都不会增加.

3.2.3 在机器学习中的应用：语言生成模型

现如今，机器学习中最为瞩目的成果之一就是*大语言模型*，它通过学习人类海量的高质量语料库来形成一个生成式的模型，其中最为典型的例子是 ChatGPT. 从思路上来说，大语言模型的核心思想非常简单：给一段话，将其中一些词掩盖掉，让模型填出这些词来. 例如，给出“我在 [mask] 面条，真好吃“，模型应该能够填出“我在吃面条，真好吃“. 这样的思想，对于更一般的数据也是成立的：用（修改改过的）数据本身作为输入，训练一个*编码器*，然后将编码器的输出送入*解码器*，而解码器的输出具有原始数据的格式，我们希望这一输出能够尽量匹配原始的输入. 在自然语言处理中，一个*生成模型*往往同时有编码器和解码器. 比如说，图 3.3 展示的就是 BART [LLG+19] 的结构.

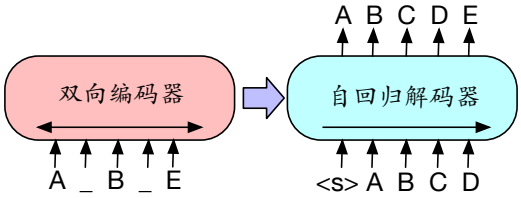


图 3.3: 生成式语言模型 BART 的示意图.

我们已经指出，熵和编码有着密切的联系. 从这个角度出发，我们很容易理解生成模型背后的思想：我们希望通过训练的方式得到一个由神经网络所表示的编码和解码规则，他要尽可能符合真实数据的分布.

我们可以用一种非常简单的模型去理解这一过程. 假设所有的单词的集合为 Σ，单词数为 k 的文本集合为 Ω. 我们希望训练一个生成模型 M，给它输入 $k-1$ 个单词，它可以给出第 k 个单词的概率分布，我们选择出现概率最大的那个词作为预测. 在训练的时候，对于一个句子 ω，我们只保留前 $k-1$ 个词，得到 $\omega[1:k]$，然后将它输入到生成模型 M 中，让它去预测第 k 个词.

对于这一个具体的句子来说，理想的分布应该是一个 *Dirac* 分布[2] $\delta_{\omega[k]}$，即以概率 1

[2]Dirac 分布是一个数学物理中更加常用的名字. 在概率论中，这也被称为*退化分布*；而在机器学习中，分

取到 $\omega[k]$. 假如说生成模型的输出是一个概率分布 $M(\omega[1:k-1]) = p$，那么，我们可以用 K-L 散度去衡量这两个分布的差异，因为 $H(\delta_{\omega[k]})$ 是固定的，所以我们只考虑交叉熵 $CH(\delta_{\omega[k]}, p)$. 因为一次训练会给多个样本，所以我们的目标是同时最小化这些交叉熵的和. 假如训练集是 T，我们的目标就是

$$\min_M \sum_{\omega \in T} CH(\delta_{\omega[k]}, M(\omega[1:k-1])).$$

实际上，这个例子是有普适性的，所有的监督训练的分类问题都可以用这种方式来建模. 而在 **??** 我们也会看到，此时交叉熵实际上被作为了一种*损失函数*.

§3.3 附录：Shannon 定理的证明

我们在这一部分给出 Shannon 定理（定理 3.1）的证明. 整体上的思路是：

1. 证明如果 f 是单调函数，对正整数 m, n 成立 $f(mn) = f(m) + f(n)$，那么 $f(n) = C \log n$.
2. 求出 $H(1/n, \ldots, 1/n)$ 的表达式.
3. 假设 p_i 是有理数，设 $p_i = n_i / \sum_j n_j$，考虑 $\sum_j n_j$ 个等可能试验结果，利用假设 3 推出 H 的表达式.
4. 利用有理数的稠密性和 H 的连续性推出一般情形.

最后一步是显然的，我们只需要证明前三步即可.

对第一步，我们需要证明的是，如果 f 是单调函数，对正整数 m, n 成立 $f(mn) = f(m) + f(n)$，那么 $f(n) = C \log n$. 首先，利用数学归纳法容易看出，对正整数 n, k，成立

$$f(n^k) = kf(n). \tag{3.2}$$

设 m, n 是任意两个大于 1 的整数，再选任意大的正整数 k，从 m 进制数的性质可以看出，总存在正整数 l 使得

$$m^l \le n^k < m^{l+1}. \tag{3.3}$$

根据 f 的单调性，我们有

$$f(m^l) \le f(n^k) < f(m^{l+1}).$$

利用式 (3.2)，我们有

$$lf(m) \le kf(n) < (l+1)f(m) \iff \frac{l}{k} \le \frac{f(n)}{f(m)} < \frac{l+1}{k}.$$

将式 (3.3) 取对数，得到

$$l \log m \le k \log n < (l+1) \log m \iff \frac{l}{k} \le \frac{\log n}{\log m} < \frac{l+1}{k}.$$

所以

$$\left| \frac{\log n}{\log m} - \frac{f(n)}{f(m)} \right| \le \frac{1}{k}.$$

布经常会表示为一个概率向量，文献中称为*独热向量*.

因为 k 可以是任意大的正整数，取 $k \to \infty$，我们就得到了

$$\frac{\log n}{\log m} = \frac{f(n)}{f(m)}.$$

由 m,n 的任意性，取 $m = 2$，我们就得到了 $f(n) = (f(2)/\log 2) \cdot \log n = C \log n$. 容易检验，$f(1) = 0 = C \log 1$，因此这一等式对所有正整数 n 都成立.

对第二步，我们需要求出 $f(n) = H(1/n, \ldots, 1/n)$ 的表达式. 我们要利用第一步的结果，首先，根据假设二，$f(n)$ 是单调递增的函数. 然后，考虑 mn 个等可能试验，我们可以将它分成两步试验，第一步有 m 中等可能的结果，而在每一种结果之下，第二步有 n 种等可能结果. 根据假设三，

$$f(mn) = f(m) + \frac{1}{n} \cdot nf(n) = f(m) + f(n).$$

所以 $f(n)$ 符合第一步的假设. 第二步就可以直接从第一步推出.

最后，我们证明第三步. 设 $p_1, \ldots, p_n$ 都是有理数，那么，他们可以被写为

$$p_i = \frac{n_i}{\sum_{j=1}^{n} n_j}.$$

其中 n_i 是非负整数. 我们考虑 $\sum_j n_j$ 个等可能试验，这个试验可以被看成两步的试验，第一步有 n 种可能的结果，第 i 种结果出现的概率是 p_i，而在第 i 种结果之下，第二步有 n_i 种等可能的结果. 根据假设三，和证明的第三步，我们有

$$C \log \sum_{j=1}^{n} n_j = H(p_1 + \cdots + p_n) + \sum_{i=1}^{n} p_i \cdot C \log n_i.$$

因此，

$$\begin{aligned}
H(p_1, \ldots, p_n) &= C \left(\log \sum_{j=1}^{n} n_j - \sum_{i=1}^{n} p_i \log n_i \right) \\
&= C \left(\log \sum_{j=1}^{n} n_j - \sum_{i=1}^{n} p_i \log \left(p_i \sum_{j=1}^{n} n_j \right) \right) \\
&= C \left(\log \sum_{j=1}^{n} n_j - \sum_{i=1}^{n} p_i \log p_i - \sum_{i=1}^{n} p_i \log \sum_{j=1}^{n} n_j \right) \\
&= -C \sum_{i=1}^{n} p_i \log p_i.
\end{aligned}$$

这正是我们要证明的. 于是，我们证明了 Shannon 定理.

§3.4 习题

1. 我们在熵以及 K-L 散度的定义中，都规定了一些无定义的量的值，这些值并不是随便规定的，他们实际上反映了熵或者 K-L 散度定义中的连续性.

(1) 证明：对给定的 $a > 0$，$\lim_{x \to 0+} x \log(x/a) = 0$，因此我们规定了 $0 \log 0 = 0$ 以及 $0 \log(0/a) = 0$.

(2) 证明：对给定的 $a > 0$，$\lim_{x\to 0+} x\log(a/x) = +\infty$，因此我们规定了 $0\log(a/0) = +\infty$.

2. 考虑关于 n 的正实数序列 $a_1(n),\dots,a_k(n)$ 以及 $b_1(n),\dots,b_k(n)$，假设对所有 i，都成立 $\lim_{n\to\infty} a_i(n)/b_i(n) = 1$，证明：

$$\lim_{n\to\infty} \frac{a_1(n)+\cdots+a_k(n)}{b_1(n)+\cdots+b_k(n)} = 1.$$

由此证明式 (3.1).

3. 证明命题 3.1.

4. 用 Lagrange 乘子法重新证明命题 3.4.

提示：如果你不知道 *Lagrange* 乘子法，可以参考 **??**.

5. 证明命题 3.8.

6. [Tin62] 仿照集合论的思路，我们可以定义三个随机变量的互信息为：

$$I(X;Y;Z) = I(X;Y) - I(X;Y|Z).$$

(1) 证明对称性：$I(X;Y;Z) = I(Y;X;Z) = I(X;Z;Y)$.

(2) 举一个例子说明，可能会有 $I(X;Y;Z) < 0$，所以这样定义的互信息并不一定真的代表“信息量”.

7. 举一个例子说明，即便 $D(p_1\|p_2)$ 很接近 0，$D(p_2\|p_1)$ 也可能会很大.

8. （单变量数据处理不等式）对任意离散随机变量 X 和函数 f，证明：$H(X) \geq H(f(X))$.

9. 考虑二分类的学习问题，此时对单个样本我们观察到的结果要么是 0 或 1，假设在真实世界中样本总体服从参数为 θ 的 Bernoulli 分布，即 $\Pr(X=1) = 1-\Pr(X=0) = \theta$. 假设我们的数据集是 $(x_1,y_1),\dots,(x_N,y_N)$，他们是从总体中独立采样得到的.

(1) 将问题考虑成一个数理统计问题，估计 θ. 写出似然函数 $L(\theta;y_1,\dots,y_N)$.

(2) 再将问题考虑为一个信息论问题，写出每个样本的真实分布与估计分布之间的交叉熵之和 $CH(\theta;y_1,\dots,y_N)$.

(3) 证明：$\max_\theta L(\theta;y_1,\dots,y_N) = \min_\theta CH(\theta;y_1,\dots,y_N)$，也就是说，最大似然估计等价于最小化交叉熵.

10. 请查找文献回答以下问题：

(1) Fisher 信息量是什么？它与 K-L 散度有什么样的关系？

(2) 列举其他概率分布之间散度的概念，他们是否是度量？

(3) 列举概率分布之间的度量，他们之间是否有关联？

§3.5 章末注记

信息一词的英文是“information”，从动词“inform”来，意思是告知、通知. 早在 15 世纪中叶，“information”一词的出现了义项“在通信中针对特定主题的知识”.[Inf] 这说明在那个时候人类就已经意识到，通信会产生新的东西，被称为知识或信息. 然而，人类对信息的严谨探索起步晚得多. 关于信息的物理学讨论源自统计力学，Boltzmann 提出了著名的熵，证明了 H 定理，以此给出了热力学第二定律的微观解释. 关于 Boltzmann 的工作，参见 [Uff22].

一般认为，现代信息论的起源是 Shannon 的论文 [Sha48]，他在论文中提出了信息的数学定义，以及信息的基本性质. 但是，Shannon 的工作并不是孤立的，他的工作是在统计力学的基础上发展起来的. 事实上，Shannon 在论文中也提到了 Boltzmann 的熵. 这篇工作也被视为通信理论以及编码理论的奠基性工作. Shannon 在这篇论文中还给出了渐近意义下达到理论下界的最优编码，并且独立地被 Fano [Rob49] 以一种不同的形式发现，因此后世称为 Shannon-Fano 编码. 但是 Shannon-Fano 编码并不是精确地达到下界，实际上，最优编码是 Huffman [Huf52] 给出的. Shannon 在这篇论文中还讨论了渐近等分性，后来 MciMillan 的工作 [McM53] 和 Breiman 的工作 [Bre57] 拓展了这一结果，因此后世称为 Shannon-McMillan-Breiman 定理.

关于信息论与集合论的关系工作，可以参见 Hu Kuo Ting 的工作 [Tin62]. 他的工作还给出了多个随机变量互信息的定义，在这一章习题中有涉及.

相对熵的概念依然是从 Shannon 的奠基性论文 [Sha48] 中提出的，但他只局限于通信的问题. 更加一般的讨论是由 Kullback 和 Leibler 在 [KL51] 给出，他们的是一种数理统计的思路，但是他们也具体地讨论了这一概念与信息的关系. 他们的论文中也讨论了交叉熵这一概念.

机器学习中编码器和解码器的思路，最早是由 Rumelhart，Hinton 和 Williams 在 [RHW86] 中提出，他们将编码器和解码器的整体称作*自编码器*. 这篇工作几乎可以被视为深度学习的开山之作，它还提出了训练神经网络最常用的反向传播算法.

关于信息论的经典教科书，可以参见 [CT12] ，此外，概率论的教材中也有很多很好的讨论，比如 [Jay02]，[Shi96] 以及 [李 10].

关于 Kolmogorov 复杂度的讨论，可以参见专著 [?]，这本书对于随机、信息、编码、复杂度，乃至归纳推理等概念都有非常独到的见解，值得一读.

第四章　Johnson-Lindenstrauss 引理

我们已经在上一章看到，使用概率分布建模的信息论在机器学习中起到了举足轻重的作用. 基于概率论的信息论总是考虑一个集合的对象的信息量，因此**数据**成为了这种方法论的核心前提：数据表征了一个集合的对象的某一特征. 在这一章，我们将探讨机器学习中数据的特性，以及一种重要的数据压缩的原理：Johnson-Lindenstrauss 引理. 证明这一引理所用到的概率论技术是**矩法**，这是*机器学习理论*中最为核心的几个技术之一. 因此本章也可以看做机器学习理论的一个引论.

§4.1　机器学习中的数据

从编码的角度来说，数据最简单的表示方法是使用固定长度的字符串. 比如说，人的生理性别有男或者女两种，于是我们可以用字符串 0 表示男，1 表示女. 这样，我们就可以用一个长度为 1 的字符串来表示人的生理性别. 人的属性还有很多，比如说年龄、身高、体重、学历、职业等等，这些属性都可以分别用固定长度的字符串来表示. 于是，一个人就被抽象为了一个固定长度的字符串.

然而，这种表示方式必须要假定数据只取有限个值. 有时候，为了简化建模和计算，我们还会考虑可以取无限个值的数据. 我们看一个具体的例子，人的身高. 从现代物理的角度来说，身高的变化是离散的，它有一个最小变化的单位. 从生物学的角度来说，身高是有上界的，比如说所有人的身高都不会超过十米. 因此，从理论上说，身高也只能取有限个值，所以也可以用字符串来表示. 然而，更加方便的方式是假定身高是一个*非负实数*，因此用一个数而不是一个字符串来表示.

因此，更加常见的情况下，我们会用*实数*或者*整数*来编码数据. 此时，将对象的多种属性按顺序排在一起，我们就得到了一个*向量*. 总而言之，在机器学习的框架，*数据被表示成数值向量*. 例如，要表示一个人的年龄、身高、体重、学历、职业，我们需要

1. 身高使用厘米作为单位；体重用千克作为单位；给学历编一个编号，比如 0 是高中，1 是本科，2 是硕士，3 是博士，-1 是其他；给职业也编号，例如 1 表示提示词工程师.

2. 用一个五维向量来表示年龄、身高、体重、学历、职业. 例如，$(20, 180, 70, 2, 1)$ 表示一个年龄为 20 岁，身高 180 厘米，体重 70 千克，学历为硕士，职业为提示词工程师的人.

注. *[lhy: 介绍一下计算机中对数值的编码].*

机器学习中，如此表示数据具备了独特的性质，一言以蔽之：**维数高，但是稀疏.** [lhy: 介绍一下高维高斯分布的特点，以及图像处理中数据稀疏的特点.]

§4.2 矩法与集中不等式

我们先引入示性函数的概念.

定义 4.1 (示性函数) 对事件 A，定义 A 的示性函数为一个从样本空间 Ω 到 $\mathbb{R}$ 的随机变量：

$$I(A)(\omega) := \begin{cases} 1, & \omega \in A. \\ 0, & \omega \notin A. \end{cases}$$

从定义就可以得到如下基本性质：

命题 4.1 设 A, B 是两个事件，则

1. $I(AB) = I(A)I(B)$.
2. $I(A)^2 = I(A)$.
3. $I(A \cup B) = I(A) + I(B) - I(AB)$.

证明 这里只作为一个示意，证明第三点，其他都类似. 我们需要证明，对任意样本点 $\omega \in \Omega$，我们有

$$I(A \cup B)(\omega) = I(A)(\omega) + I(B)(\omega) - I(AB)(\omega).$$

假设 $\omega \in A \cup B$，那么左边等于 1. 我们分类讨论：

- 如果 $\omega \in A$，那么右边第一项为 1.
  - 如果 $\omega \in B$，那么右边第二项为 1. 此时自然也有 $\omega \in AB$，所以右边第三项为 1，因此右边等于 1，等于左边.
  - 如果 $\omega \notin B$，那么右边第二项为 0. 此时自然也有 $\omega \notin AB$，所以右边第三项为 0，因此右边等于 1，等于左边.
- 如果 $\omega \notin A$，那么右边第一项为 0. 此时必须有 $\omega \in B$，所以右边第二项为 1. 但是此时自然也有 $\omega \notin AB$，所以右边第三项为 0，因此右边等于 1，等于左边.

如果 $\omega \notin A \cup B$，讨论类似，这里不再赘述. □

示性函数之所以重要，是因为它联系了期望与概率. 我们先来看一个显然的命题：

命题 4.2 *设 A 是一个事件，则*

$$\mathbb{E}[I(A)] = \Pr(A).$$

示性函数可以把对概率的计算变成对期望的计算. 回忆期望的线性性：设 $a, b \in \mathbb{R}$，X, Y 是有期望的随机变量，那么成立

$$\mathbb{E}(aX + bY) = a\mathbb{E}(X) + b\mathbb{E}(Y).$$

利用期望的线性性，示性函数可以导出很多概率恒等式与不等式. 例如：容斥公式

$$\begin{aligned}\Pr(A \cup B) &= \mathbb{E}[I(A \cup B)] = \mathbb{E}[I(A) + I(B) - I(AB)] \\ &= \mathbb{E}[I(A)] + \mathbb{E}[I(B)] - \mathbb{E}[I(AB)] \\ &= \Pr(A) + \Pr(B) - \Pr(AB).\end{aligned}$$

对于概率论以及机器学习理论来说，下面的这个不等式非常重要：

定理 4.1 (Markov 不等式) *如果 X 是非负有期望的随机变量，$a > 0$，那么*

$$\Pr(X \geq a) \leq \frac{\mathbb{E}[X]}{a}.$$

证明 直接利用示性函数，我们有：

$$\begin{aligned}\mathbb{E}[X] =& \mathbb{E}[XI(X \geq a) + XI(X < a)] \\ =& \mathbb{E}[\underbrace{XI(X \geq a)}_{\geq aI(X \geq a)}] + \mathbb{E}[\underbrace{XI(X < a)}_{\geq 0}] \\ \geq& a\mathbb{E}[I(X \geq a)] = a\Pr(X \geq a).\end{aligned}$$

□

注. 为了使得证明有效，我们必须要假设上面的推导中出现的期望都是存在的，当然这实际上很容易验证. 为了避免不必要的技术细节，在后面的所有证明以及推导中，我们都会默认写出来的期望是存在的，不再赘述.

我们利用 Markov 不等式可以直接得到以下结果.

推论 4.1 (Chebyshev 不等式) 设 X 是任意有方差的随机变量，那么对任意 $a > 0$，成立

$$\Pr(|X - \mathbb{E}[X]| \geq a) \leq \frac{\mathsf{Var}(X)}{a^2}.$$

证明 设 $Y = (X - \mathbb{E}[X])^2$，$t = a^2$，那么 Y 是非负随机变量，且 $\mathbb{E}[Y] = \mathsf{Var}(X)$，于是由 Markov 不等式，我们有

$$\begin{aligned}\Pr(|X - \mathbb{E}[X]| \geq a) &= \Pr(|X - \mathbb{E}[X]|^2 \geq a^2) \\ &= \Pr(Y \geq t) \\ &\leq \frac{\mathbb{E}[Y]}{t} = \frac{\mathsf{Var}(X)}{a^2}.\end{aligned}$$

□

Chebyshev 不等式告诉我们采样到偏离其期望的概率有一个上界. 像这样利用矩（即 $\mathbb{E}[f(X)]$）来估计概率上界的方法被称为矩法.

实际上，很多情况下，偏离期望是非常小概率的事件，远小于上面的估计值. 为了得到更精确的上界，我们需要一些技巧. 考虑任意随机变量 X，对 $\lambda > 0$，

$$X \geq a \iff \lambda X \geq \lambda a \iff e^{\lambda X} \geq e^{\lambda a}.$$

由 Markov 不等式（如何得到?），

$$\Pr(X \geq a) = \Pr\left(e^{\lambda X} \geq e^{\lambda a}\right) \leq e^{-\lambda a} \cdot \mathbb{E}\left[e^{\lambda X}\right].$$

注意到这个不等式应该对任意 $\lambda > 0$ 成立，所以

$$\Pr(X \geq a) \leq \inf_{\lambda > 0} e^{-\lambda a} \cdot \mathbb{E}\left[e^{\lambda X}\right].$$

以上方法可以得到概率更精确的上界. 这样用指数进行推导的方法称为指数矩或 *Cramér-Chernoff 方法*.

利用指数矩，我们可以更加精确地研究 Chebyshev 不等式中随机变量所表现出来的性质，这种性质被称为概率的集中性. 我们可以用集中不等式来刻画这样的性质. 这样的不等式描述随机变量 X 有多大概率偏离某个值 μ 多少值（t），它表现为

$$\Pr(|X - \mu| \geq t) \leq \text{小量}.$$

通常来说，μ 是随机变量的期望或者中位数，在这本书中，只会讨论关于期望的集中性. 我们可以看到 Chebyshev 不等式就是一种特殊的集中不等式，但是它的界太松. 利用指数矩，我们将证明界更紧的 Hoeffding 不等式和 Chernoff 不等式.

定理 4.2 (Hoeffding 不等式) *设 $X_1, \dots, X_n$ 相互独立且服从对称 Bernoulli 分布，即 X_i 满足 $\Pr(X_i = 1) = 1 - \Pr(X_i = -1) = 1/2$. 考虑向量 $a = (a_1, \dots, a_n) \in \mathbb{R}^n$，对任意 $t \geq 0$，我们有*

$$\Pr\left(\sum_{i=1}^{n} a_i X_i \geq t\right) \leq \exp\left(-\frac{t^2}{2\|a\|_2^2}\right).$$

证明 由指数矩，我们有

$$\begin{aligned}
\Pr\left(\sum_{i=1}^{n} a_i X_i \geq t\right) &= \Pr\left(\exp\left(\lambda \sum_{i=1}^{n} a_i X_i\right) \geq \exp(\lambda t)\right) \\
&\leq e^{-\lambda t} \mathbb{E}\left[\exp\left(\lambda \sum_{i=1}^{n} a_i X_i\right)\right] \\
&= e^{-\lambda t} \prod_{i=1}^{n} \mathbb{E}\left[\exp(\lambda a_i X_i)\right].
\end{aligned}$$

这个不等式对任意 $\lambda > 0$ 都成立. 利用 $X_1, \dots, X_n$ 服从对称 Bernoulli 分布，得到（习题）[lhy: 习题]

$$e^{-\lambda t} \prod_i \mathbb{E}\left[\exp(\lambda a_i X_i)\right] \leq \exp\left(-\lambda t + \frac{\lambda^2}{2} \sum_i a_i^2\right). \tag{4.1}$$

由于这一不等式对任意 $\lambda > 0$ 都成立，根据二次函数的性质，取 $\lambda = t / \sum_i a_i^2$，可得

$$\begin{aligned}
\inf_{\lambda > 0} \exp\left(-\lambda t + \frac{\lambda^2}{2} \sum_i a_i^2\right) &= \exp\left(-\frac{t}{\sum_i a_i^2} t + \frac{1}{2}\left(\frac{t}{\sum_i a_i^2}\right)^2 \sum_i a_i^2\right) \\
&= \exp\left(-\frac{t^2}{2\|a\|_2^2}\right).
\end{aligned}$$

□

利用相同的证明技巧，我们可以证明一般形式的 Hoeffding 不等式，我们把证明留作习题.[lhy: 习题]

定理 4.3 (Hoeffding 不等式，一般情形) *设 $X_1, \dots, X_n$ 是相互独立的随机变量，对任意 i 都成立 $X_i \in [m_i, M_i]$. 那么对任意 $t \geq 0$，我们有*

$$\Pr\left(\sum_{i=1}^{n} (X_i - \mathbb{E}[X_i]) \geq t\right) \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{n} (M_i - m_i)^2}\right).$$

下面我们介绍 Chernoff 不等式.

定理 4.4 (Chernoff 不等式) 设 $X_1, \ldots, X_n$ 是相互独立的随机变量，分别服从于参数为 $p_1, \ldots, p_n$ 的 *Bernoulli* 分布. 记 $\sum_{i=1}^{n} X_i$ 的期望为 $\mu = \sum_{i=1}^{n} p_i$，对于任意 $t > \mu$，我们有

$$\Pr\left(\sum_{i=1}^{n} X_i \geq t\right) \leq e^{-\mu}\left(\frac{e\mu}{t}\right)^t.$$

这里 e 是自然对数的底数.

证明 和证明 Hoeffding 不等式的第一步相同，我们先利用指数矩，对任意 $\lambda > 0$ 有

$$\Pr\left(\sum_{i=1}^{n} X_i \geq t\right) \leq e^{-\lambda t}\prod_{i=1}^{n}\mathbb{E}\left[\exp(\lambda X_i)\right].$$

然后，将 $\prod_{i=1}^{n}\mathbb{E}[\exp(\lambda X_i)]$ 进一步放缩：

$$\begin{aligned}\prod_{i=1}^{n}\mathbb{E}\left[\exp(\lambda X_i)\right] &= \prod_{i=1}^{n}\left(e^{\lambda}p_i + (1-p_i)\right)\\ &\leq \prod_{i=1}^{n}\exp\left((e^{\lambda}-1)p_i\right).\end{aligned}$$

因此

$$\begin{aligned}\Pr\left(\sum_{i=1}^{n} X_i \geq t\right) &\leq e^{-\lambda t}\prod_{i=1}^{n}\exp\left((e^{\lambda}-1)p_i\right)\\ &= e^{-\lambda t}\exp\left((e^{\lambda}-1)\sum_{i=1}^{n}p_i\right)\\ &= \exp\left(\mu e^{\lambda} - t\lambda - \mu\right).\end{aligned}$$

右边的最小值在 $\lambda = \log(t/\mu)$ 取得，代入得到：

$$\Pr\left(\sum_{i=1}^{n} X_i \geq t\right) \leq e^{-\mu}\left(\frac{e\mu}{t}\right)^t.$$

□

§4.3 J-L 引理的陈述与证明

有了上面矩法的准备，我们可以陈述并证明 J-L 引理了.

定理 4.5 (Johnson-Lindenstrauss 引理) 给定 N 个单位向量 $v_1,\dots,v_N \in \mathbb{R}^m$ 和 $n > 24\log N/\epsilon^2$，随机矩阵 $A \in \mathbb{R}^{n\times m}$ 每个元素独立重复采样自 $\mathcal{N}(0,1/n)$，$\epsilon \in (0,1)$ 是给定常数，那么至少有 $(N-1)/N$ 的概率，使得对所有的 $i \neq j$，都成立

$$(1-\epsilon)\left\|v_i - v_j\right\|_2^2 < \left\|Av_i - Av_j\right\|_2^2 < (1+\epsilon)\left\|v_i - v_j\right\|_2^2.$$

我们可以把 n 理解成降维后的维度，Av_i 是降维后的向量. 这个引理告诉我们只要 $n > 24\log N/\epsilon^2$，我们就可以用变换 A 把原本 m 维的向量映射到 n 维空间，并且保证它们相对距离的偏离不超过 ϵ，因此我们可以把 A 看成一个损失率很低的压缩变换. 不严格地说，塞下 N 个向量，只需要 $\mathcal{O}(\log N)$ 维空间.

下面我们开始证明 J-L 引理. 为了看出来证明的思路，我们第一个任务是算出压缩后 Av_i 的分布. 我们首先回忆一些正态向量的基本性质. 关于正态向量的讨论，可以参考**??**.

命题 4.3 假设 $u \sim \mathcal{N}(\mu,\Sigma)$ 是一个 n 维正态向量，M 是一个 $m\times n$ 矩阵，那么 Mu 是一个 m 维正态向量，并且 $Au \sim \mathcal{N}(M\mu, M\Sigma M^T)$.

利用这一个命题，很容易可以得到 Av_i 的分布：

引理 4.1 假设 $u \in \mathbb{R}^m$ 是一个单位向量，那么 $Au \sim \mathcal{N}(0, n^{-1}I_n)$.

证明 将 A 视作一个 mn 维的正态向量，注意到，$(Au)_i = \sum_{j=1}^m A_{ij}u_j$，所以 Au 是一个从向量 A 线性变换得到的向量. 根据命题 4.3，Au 是一个正态向量，只需计算它的期望和协方差矩阵.

注意到，对不同的 i，向量 $(A_{ij})_j$ 相互是独立的，所以分量 $(Au)_i$ 相互也是独立的，因此只需要计算正态变量 $(Au)_i$ 的期望与方差. 其期望为 $\sum_{j=1}^m 0\cdot u_j = 0$，方差为

$$\sum_{j=1}^m \left(\frac{1}{n}\cdot u_j^2\right) = \frac{1}{n}.$$

所以 Au 的期望是 0，协方差矩阵是 $n^{-1}I_n$. □

然而，我们关心的其实不单单是 Av_i 的分布，更重要的其实是 $Av_i - Av_j$ 的分布，即压缩后的向量之间的相对距离，幸运的是，我们并不需要做额外的什么计算，我们直接有如下结果：

引理 4.2 向量 $u = \frac{v_i - v_j}{\left\|v_i - v_j\right\|_2}$ 是一个单位向量，因此 $Au \sim \mathcal{N}(0, n^{-1}I_n)$.

J-L 引理实际上在说，$\|Au\|_2$ 偏离 1 的一定程度的概率是非常小的. 于是，为了证明 J-L 引理，我们最重要的任务是给出 Au 这样向量模长的集中不等式：

引理 4.3 (单位模引理) 设 $u \sim \mathcal{N}(0, n^{-1}I_n)$，$\epsilon \in (0,1)$ 是给定的常数，那么我们有

$$\Pr(|\,\|u\|_2^2 - 1| \geq \epsilon) \leq 2\exp\left(-\frac{\epsilon^2 n}{8}\right).$$

注意到 $\mathbb{E}[\|u\|_2^2] = n \cdot (1/n) = 1$，所以这个引理在说高维空间中，如果正态向量具有单位模长平方期望，那么它的模长就会集中在单位长度附近，因此称为单位模引理.

证明 $|\,\|u\|_2^2 - 1| \geq \epsilon$ 发生有两种可能，$\|u\|_2^2 - 1 \geq \epsilon$ 和 $1 - \|u\|_2^2 \geq \epsilon$. 我们先来计算 $\|u\|_2^2 - 1 \geq \epsilon$ 的概率，根据指数矩，

$$\Pr\left(\|u\|_2^2 - 1 \geq \epsilon\right) \leq \inf_{\lambda>0}\left\{e^{-\lambda(\epsilon+1)}\mathbb{E}\left[e^{\lambda\|u\|_2^2}\right]\right\}.$$

因为 u 的各个分量是相互独立的，所以我们可以把 $\|u\|_2^2$ 展开

$$\mathbb{E}\left[e^{\lambda\|u\|_2^2}\right] = \mathbb{E}\left[e^{\lambda\sum_i u_i^2}\right] = \mathbb{E}\left[\prod_i e^{\lambda u_i^2}\right] = \prod_i \mathbb{E}\left[e^{\lambda u_i^2}\right].$$

可以算得 $\mathbb{E}\left[e^{\lambda u_i^2}\right] = \sqrt{n/(n-2\lambda)}$[lhy: 习题]，所以

$$\Pr\left(\|u\|_2^2 - 1 \geq \epsilon\right) \leq \inf_{\lambda>0}\left\{e^{-\lambda(\epsilon+1)}\left(\frac{n}{n-2\lambda}\right)^{n/2}\right\}.$$

可以验证最小值在 $\lambda = n\epsilon/(2(1+\epsilon))$ 处取到，代入可得

$$\Pr\left(\|u\|_2^2 - 1 \geq \epsilon\right) \leq e^{n(\log(1+\epsilon)-\epsilon)/2} \leq e^{-n\epsilon^2/8}.$$

这里最后一个不等号使用了不等式 $\log(1+\epsilon) \leq \epsilon - \epsilon^2/4$.

计算 $1 - \|u\|_2^2 \geq \epsilon$ 的概率的过程和 $\|u\|_2^2 - 1 \geq \epsilon$ 几乎完全相同的，可以得到

$$\Pr\left(1 - \|u\|_2^2 \geq \epsilon\right) \leq e^{n(\log(1-\epsilon)+\epsilon)/2} \leq e^{-n\epsilon^2/8}.$$

$$\begin{aligned}\Pr\left(|\,\|u\|_2^2 - 1| \geq \epsilon\right) &\leq \Pr\left(\|u\|_2^2 - 1 \geq \epsilon\right) + \Pr\left(1 - \|u\|_2^2 \geq \epsilon\right) \\ &\leq 2e^{-n\epsilon^2/8}.\end{aligned}$$

□

有了单位模引理，我们就可以很容易证明 J-L 引理了. 将引理 4.2 中的 u 带入单位模引理，得到

$$\Pr\left(\left|\left\|\frac{A(v_i - v_j)}{\|v_i - v_j\|_2}\right\|_2^2 - 1\right| \geq \epsilon\right) \leq 2\exp\left(-\frac{\epsilon^2 n}{8}\right).$$

这个结论对任意 $i \neq j$ 成立，因此遍历所有 i, j 对，可得

$$\begin{aligned}\Pr\left(\exists(i,j): \left|\left\|\frac{A(v_i - v_j)}{\|v_i - v_j\|_2}\right\|_2^2 - 1\right| \geq \epsilon\right) &\leq 2\sum_{i\neq j}\exp\left(-\frac{\epsilon^2 n}{8}\right) \\ &= 2\binom{N}{2}\exp\left(-\frac{\epsilon^2 n}{8}\right).\end{aligned}$$

换言之，对任意 i, j，$\left|\left\|\frac{A(v_i-v_j)}{\|v_i-v_j\|_2}\right\|_2^2 - 1\right| < \epsilon$ 都成立的概率不小于

$$1 - 2\binom{N}{2}\exp\left(-\frac{\epsilon^2 n}{8}\right) = 1 - N(N-1)\exp\left(-\frac{\epsilon^2 n}{8}\right).$$

代入 $n > \frac{24\log N}{\epsilon^2}$，可得这一概率

$$1 - N(N-1)\exp\left(-\frac{\epsilon^2 n}{8}\right) \geq 1 - N(N-1)N^{-3} \geq 1 - N^{-1} = \frac{N-1}{N}.$$

很多时候，我们关心的并不是向量间的距离，而是向量的内积（比如使用余弦度量的时候），这时候我们可以使用内积版本的 J-L 的引理：

定理 4.6 (J-L 引理，内积形式) 给定 N 个单位向量 $v_1, \ldots, v_N \in \mathbb{R}^m$ 和 $n > 24\log N/\epsilon^2$，随机矩阵 $A \in \mathbb{R}^{n\times m}$ 每一个元素都独立重复采样自 $\mathcal{N}(0, 1/n)$，$\epsilon \in (0,1)$ 是给定常数，那么至少有 $(N-1)/N$ 的概率，使得对所有的 $i \neq j$，都成立

$$|\langle Av_i, Av_j\rangle - \langle v_i, v_j\rangle| < \epsilon.$$

证明 由原始 J-L 引理可知，至少有 $\frac{N-1}{N}$ 的概率满足对于任意 $i \neq j$ 有：

$$\begin{aligned}(1-\epsilon)\|v_i - v_j\|_2^2 < \|Av_i - Av_j\|_2^2 < (1+\epsilon)\|v_i - v_j\|_2^2, \\ (1-\epsilon)\|v_i + v_j\|_2^2 < \|Av_i + Av_j\|_2^2 < (1+\epsilon)\|v_i + v_j\|_2^2.\end{aligned}$$

我们将第一行乘 -1 加到第二行可以得到

$$4\langle v_i, v_j\rangle - 2\epsilon(\|v_i\|_2^2 + \|v_j\|_2^2) < 4\langle Av_i, Av_j\rangle < 4\langle v_i, v_j\rangle + 2\epsilon(\|v_i\|_2^2 + \|v_j\|_2^2).$$

因为 v_i, v_j 是单位向量，所以上式等价于 $|\langle Av_i, Av_j\rangle - \langle v_i, v_j\rangle| < \epsilon$. □

注. *[lhy: 讨论渐近等分性、指数矩、大偏差理论之间的关系.]*

§4.4 J-L 引理的应用

回顾：J-L 引理描述的是对于 N 个向量，我们可以将它们降到 $\mathcal{O}(\log N)$ 维空间，并将相对距离的误差控制在一定范围内. 它的内容本身就和降维相关，所以最基本的应用就是直接作为降维方法. 许多其它算法例如局部敏感哈希（LSH）、随机 SVD，本质上也都依赖 J-L 引理. 除此之外，J-L 引理对机器学习模型中维度的选择提供了一些理论解释. 下面我们将介绍两个具体的应用案例.

例 4.1 (词向量维度) *在 NLP 的发展中产生了像 Word2Vec、GloVe 这样经典的词向量模型和基于注意力机制的各种大语言模型. 这里一个问题自然的问题是，当我们对 N 个单词进行建模，词向量的维度选择多少比较合适？如果维度过高，会使得后续的计算变得更加复杂. 如果维度过低，会无法完全表达出这些单词本身的信息. 对于这一问题，J-L 引理给出了一个比较直接的结论，$\mathcal{O}(\log N)$ 空间足以容纳下 N 个单词. 但是要注意，这一结论成立的前提是正态随机矩阵，然而单词的空间是否符合正态分布是不知道的，所以这一结果只是从理论上给了一个直观，选择什么样的 n 还是由具体的实验效果来决定.*

例 4.2 (多头注意力) *在注意力机制中，我们往往会先把 `head_size` 降低到 64 再做内积. 那么一个很自然的问题是，`head_size` 为 64 的注意力机制是否足以拟合任何概率分布？具体来说，注意力的计算公式为*

$$a_{ij} = \frac{e^{\langle q_i, k_j \rangle}}{\sum_{j=1}^{L} e^{\langle q_i, k_j \rangle}}.$$

其中 $q_i, k_j \in \mathbb{R}^d$.

我们希望能够做到：给定任意的概率矩阵 (p_{ij})，上述 (a_{ij}) 都能够很好的逼近 (p_{ij}). 换言之，给定 (p_{ij}) 和维度 d，我们是否能找到一组 $q_1, \ldots, q_L, k_1, \ldots, k_L \in \mathbb{R}^d$，使得对应项 a_{ij} 与 p_{ij} 足够接近. 其实这就和词向量模型的维度选择问题是等价的. 词向量的维度变成了`head_size`，词表大小变成了序列长度. J-L 引理告诉我们的答案依然是只需要 $\mathcal{O}(\log N)$ 的空间就足以容纳下 N 个单词，一个很粗糙的计算，

[lhy: 这两个例子写得更详细一些.]

§4.5 习题

§4.6 章末注记

第五章　差分隐私

在本章我们关心数据的另一个维度：社会属性. 机器学习需要大量的数据，这些数据从何而来？大部分时候，是通过收集个体的数据得到的. 于是这里就涉及到了隐私的问题，如何保护个体的隐私，同时又能够让机器学习得到足够的数据？**差分隐私**就是解决这个问题的一个方法，本章将更详细地介绍差分隐私的概念和应用.

§5.1　数据隐私问题

许多科研工作的开展和推进都需要有大量真实有效的数据作为支撑. 以医学为例，我们需要大量真实的病人提供病情数据. 但这些数据可能都涉及到病人的隐私信息，例如一些*敏感数据*. 因此，我们必须找到一种方法，既可以收集到这些数据，又保护病人的隐私信息.

保护病人的隐私信息的一种理解是不会让数据获得者将数据和人对应起来. 那么最直观的想法就是将每条*数据匿名化*. 比如，每条数据只包含患者的生日、性别、邮政编号（代表位置）、病情. 这样做依然有问题：同一天生日、同一性别、相同邮政编号的人很少，而这些信息很容易被找到，比如公开的选民名册. 于是通过一条数据的各种属性可以轻易定位到这个人，这样的匿名化是不安全的.

我们再看一个例子。2006 年，Netflix 举办了关于电影推荐系统的算法设计比赛. 只公开了匿名代号和对应用户的观看电影名称、打分的数据集（这次连生日、性别都没有）. 但这一数据集很快被破解，问题出在只要对某个用户稍微熟悉一些，就很容易对应出这个用户和观影数据. 这种破解让部分用户感到焦虑，例如性少数群体害怕其他人可以从自己的观影记录中判断出自己的性取向. 第二届 Netflix Prize 竞赛也因此停办.

这两个例子展现的是一种*去匿名化*的现象，也就是说匿名的数据实际上揭示了数据对应的那个个体。这去匿名化的出现都是因为数据具有*独特性*. 比如我们看表 5.1，这是医院甲的病人数据表。56 岁的病人只有 Rebecca，所以假如我们知道 Rebecca 的年龄并了解到她去过这家医院，便立即得知她患有 HIV.

Name	Age	Gender	Zip Code	Smoker	Diagnosis
Richard	64	Male	19146	Y	Heart disease
Susan	61	Female	19118	N	Arthritis
Matthew	67	Male	19104	Y	Lung cancer
Alice	63	Female	19146	N	Crohn's disease
Thomas	69	Male	19115	Y	Lung cancer
Rebecca	**56**	**Female**	19103	N	HIV
Tony	52	Male	19146	Y	Lyme disease
Mohammed	59	Male	19130	Y	Seasonal allergies
Lisa	55	Female	19146	N	Ulcerative colitis

[lhy: 改成中文版]

表 5.1: 医院甲的病人数据表。

一种减少独特性的思想是k-匿名性：任何一个人的信息都不能和其他至少 $(k-1)$ 人区分开. 比如，可以不明确写出姓名、年龄和邮编，只给出模糊的范围，于是数据变成了下面的表 5.2.

Name	Age	Gender	Zip Code	Smoker	Diagnosis
*	60-70	Male	191**	Y	Heart disease
*	60-70	Female	191**	N	Arthritis
*	60-70	Male	191**	Y	Lung cancer
*	60-70	Female	191**	N	Crohn's disease
*	60-70	Male	191**	Y	Lung cancer
*	**50-60**	**Female**	191**	N	HIV
*	50-60	Male	191**	Y	Lyme disease
*	50-60	Male	191**	Y	Seasonal allergies
*	**50-60**	**Female**	191**	N	Ulcerative colitis

[lhy: 改成中文]

表 5.2: 医院甲的病人数据表，模糊了姓名、年龄和邮编。

这种方法仍然存在问题，因为我们不能把关键信息（病症信息）也模糊化. 如果我们还拿到了另一家医院乙的模糊之后的病人数据表（表 5.3），那么依然有办法定位到

Rebecca：这两张表上 50-60 岁的女性只有 HIV 是重合的，如果我们知道 Rebecca 的年龄并知道她同时去过两家医院，便立即得知她患有 HIV.

Name	Age	Gender	Zip Code	Diagnosis
*	**50-60**	**Female**	191**	HIV
*	**50-60**	**Female**	191**	Lupus
*	**50-60**	**Female**	191**	Hip fracture
*	60-70	Male	191**	Pancreatic cancer
*	60-70	Male	191**	Ulcerative colitis
*	60-70	Male	191**	Flu-like symptoms

[lhy: 改成中文]

表 5.3: 医院乙的病人数据表，模糊了姓名、年龄和邮编。

除了使用匿名化的手段，还有一种方法可以保护隐私：不再提供单人的数据，而是直接公布将数据集的总体信息，比如平均值. 但这种方法也不一定能保证不泄露单人数据. 例如：我们只公布一个机器学习模型（这算是一直非常抽象的总体信息）. 很多研究表明可以通过尝试不同的测试数据来判断出这个机器学习模型的训练集，这是因为机器学习模型总是会偏向于过拟合训练集数据. 所以如果对某个测试数据的结果很有自信，往往说明这一数据存在于训练集中. [lhy: 给一个 ref]

注. 以上这些内容都说明**匿名化**很难保护个人隐私. 那么是否可以使用密码学的手段进行**加密**？其实加密和隐私在出发点上完全不同. 加密的目的是为了不让别人获取到真实数据. 而隐私是一个比简单地锁定数据更微妙的问题——我们希望我们的算法的结果能够释放有用的信息，而不是泄露私人信息. 因此我们这里讨论的其实是隐私保护问题而不是加密问题。

§5.2 差分隐私的定义与性质

我们上面探讨了隐私保护的必要性以及它的微妙之处，现在我们要给出一种合理的方案解决隐私保护的问题，这个方案就是*差分隐私*。要给出一个数学模型，不仅要知道什么情况下算是隐私泄漏，也需要知道什么情况下不算，所以我们再来看一个反面的例子。

Broky 是一位长期吸烟的男子，他参加了一项有关“吸烟与健康”的调查. 这项调查在不久后发布了一项结果，表明长期吸烟的人患上肺癌的几率更大. 伴随着这一结果的公

布，保险公司在出售相同保险时会对长期吸烟者索要更高的价格. Broky 当然也受到了这一政策的影响. 那我们是否可以认为这项研究泄露了 Broky（更有可能患病）的隐私呢?

我们的直观应该是不算泄露了隐私. 因为“长期吸烟的人患上肺癌的几率更大”这项结论并不依赖于 Broky 是否参加了调查. 考虑这样的对照，x 代表原来参加调查的人的集合，x' 代表其他人不变，只是 Broky 换成了另外一个人的集合. 如果是 x' 这些人参与了调查，结论是否会发生变化? 大概率不会！

Broky 的例子告诉我们对于隐私的一种合理的衡量应该有以下性质：当数据集中包含 *Broky* 的信息，相比数据集中不包含 *Broky* 的信息，并不会**明显**增加损害 *Broky* 的利益的概率. 这一思想引出了差分隐私的概念，我们将在下面给出数学形式的定义.

考虑数据的空间 $\mathcal{X}$，其中的每一个元素都包含了个体的所有可能数据例如姓名、性别、年龄、国籍等. 考虑 n 个人的数据，形成了有序的数据集 $x = (x_1, \cdots, x_n) \in \mathcal{X}^n$. 设 A 是一种随机算法：在固定输入数据集 $x \in \mathcal{X}^n$ 下，$A(x)$ 是结果空间 $\mathcal{Y}$ 上的一个随机变量. 当我们改变（增加或删除）一个人的数据时，我们希望结果分布的变化可以控制在一定范围内. 为此，我们引入相邻数据集的概念：

定义 5.1 (k-相邻数据集) 设 $x, x' \in \mathcal{X}^n$，如果 x 和 x' 最多有 k 条数据不同，即至多存在 k 个不同的 $i_1, \ldots, i_k \in [n]$ 使得 $x_{i_j} = x'_{i_j}$ 对 $j \in [k]$ 成立，那么称 x 和 x' 是 k-相邻的.

在之前的例子中，含有 Broky 的被调查者的数据集和把 Broky 换为任意一个其他人的数据集时 1-相邻数据集.

现在我们给出差分隐私的定义。

定义 5.2 (ϵ-DP) 考虑随机算法 $A : \mathcal{X}^n \to \mathcal{Y}$，如果对于任一对 1-相邻数据集 x, x'，对任意（可测）值集 $E \subseteq \mathcal{Y}$，有

$$\Pr(A(x) \in E) \leq e^{\epsilon} \cdot \Pr(A(x') \in E),$$

那么我们称 A 为数据集大小为 n 的 ϵ-DP 算法.

需要注意的是，这一定义是针对随机算法的，并且是对称的，也就是 x 和 x' 的地位是平等的。直观上一个，ϵ-DP 算法的输出分布在相邻数据集上的变化不会太大. ϵ 衡量的是信息的泄漏量；ϵ 越大，算法泄漏的信息就越多，隐私保护效果当然也变差.

以上定义需要对所有的（可测）值集 E 都成立，这给验证带来了极大的困难，如果随机算法的输出分布是更加常规的，我们可以简化验证的过程.

对于离散型的输出，我们有如下等价定义：

命题 5.1 如果 $A(x)$ 对于任意 $x \in \mathcal{X}^n$ 都是离散型随机变量，那么随机算法 A 是数据集大小为 n 的 ϵ-DP 算法，当且仅当对于任意一对 1-相邻数据集 x, x' 和所有的 $y \in \mathcal{Y}$，有

$$\Pr(A(x) = y) \leq e^{\epsilon} \cdot \Pr(A(x') = y).$$

证明 $\Longrightarrow$：取 $E = \{y\}$ 即可证明.

$\Longleftarrow$：假设 $E = \{y_1, \ldots, y_k, \ldots\}$. 对每一个 y_i，都有

$$\Pr(A(x) = y_i) \leq e^{\epsilon} \cdot \Pr(A(x') = y_i).$$

因为 $A(\cdot) = y_i$ 对于不同的 i 是互斥事件，所以概率可以直接相加，于是：

$$\Pr(A(x) \in E) = \sum_i \Pr(A(x) = y_i) \leq e^{\epsilon} \cdot \sum_i \Pr(A(x') = y_i) = e^{\epsilon} \cdot \Pr(A(x') \in E).$$

□

对连续型的输出，我们有如下等价定义：

命题 5.2 如果 $A(x)$ 对于任意 $x \in \mathcal{X}^n$ 都是连续型随机变量，那么它存在概率密度函数，记为 h_x. 此时随机算法 A 是数据集大小为 n 的 ϵ-DP 算法，当且仅当对于任意一对 1-相邻数据集 x, x' 和几乎所有的 $y \in \mathcal{Y}$，有

$$h_x(y) \leq e^{\epsilon} \cdot h_{x'}(y).$$

证明 [1] $\Longrightarrow$：根据概率密度函数的定义（实际上是 Lebesgue 微分定理），对 $x \in \mathcal{X}^n$，取 $E_\delta = (y - \delta, y + \delta)$，对几乎所有的 $y \in \mathcal{Y}$，有

$$\frac{\mathrm{d}}{\mathrm{d}\delta} \Pr(A(x) \in E_\delta) = h_x(y).$$

因此，对几乎所有的 $y \in \mathcal{Y}$，

$$\forall \delta > 0 \ \Pr(A(x) \in E_\delta) \leq e^{\epsilon} \cdot \Pr(A(x') \in E_\delta) \implies h_x(y) \leq e^{\epsilon} \cdot h_{x'}(y).$$

$\Longleftarrow$：依然根据概率密度函数的定义，考虑 $x, x' \in \mathcal{X}^n$，对任意可测 $E \subseteq \mathcal{Y}$，有

$$\Pr(A(x) \in E) = \int_E h_x(y)\mathrm{d}y \leq e^{\epsilon} \cdot \int_E h_{x'}(y)\mathrm{d}y = e^{\epsilon} \cdot \Pr(A(x') \in E).$$

□

接下来我们给出差分隐私的基本性质。

[1] 这一部分的严格表述需要测度论的基础，所以这一证明从直观上理解即可，不需要考虑严格的定义。

命题 5.3 (复合性，两个算法的情形) *A_1 和 A_2 是相互独立的随机算法，其中 $A_1 : X^n \to \mathcal{Y}_1$，$A_2 : \mathcal{Y}_1 \times \mathcal{X}^n \to \mathcal{Y}_2$. 假设 A_1 是 ϵ_1-DP 算法，A_2 是 ϵ_2-DP 算法.*

令 $A : \mathcal{X}^n \to \mathcal{Y}_1 \times \mathcal{Y}_2$ 是随机算法，输出为 $A(x) = (y_1, y_2)$，其中 $y_1 = A_1(x)$，$y_2 = A_2(y_1, x)$，那么 A 是 $(\epsilon_1 + \epsilon_2)$-DP 算法.

证明 为了简化记号，这里我们只证明离散的情况. 令 x, x' 是 $\mathcal{X}^n$ 中的两个 1-相邻数据集，A 输出为 $y = (y_1, y_2) \in \mathcal{Y}_1 \times \mathcal{Y}_2$，那么根据定义和独立性，

$$\Pr(A(x) = (y_1, y_2)) = \Pr(A_1(x) = y_1) \cdot \Pr(A_2(y_1, x) = y_2).$$

由于 A_1 是 ϵ_1-DP 算法，A_2 是 ϵ_2-DP 算法，得到

$$\begin{aligned}\Pr(A(x) = (y_1, y_2)) &= \Pr(A_1(x) = y_1) \cdot \Pr(A_2(y_1, x) = y_2) \\ &\le e^{\epsilon_1} \Pr(A_1(x') = y_1) \cdot e^{\epsilon_2} \Pr(A_2(y_1, x') = y_2) \\ &= e^{\epsilon_1 + \epsilon_2} \cdot \Pr(A(x') = (y_1, y_2)).\end{aligned}$$

□

利用数学归纳法，很容易推广到多个随机算法的复合性：

命题 5.4 (复合性，多个算法的情形) *设 $A_1, A_2, \cdots, A_k$ 为一列相互独立的随机算法，*

$$\begin{aligned}&A_1 : \mathcal{X}^n \to \mathcal{Y}_1, \\ &A_i : \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_{i-1} \times \mathcal{X}^n \to \mathcal{Y}_i, \quad i = 2, 3, \cdots, k.\end{aligned}$$

也就是 A_i 将 $A_1, \cdots, A_{i-1}$ 的输出和 $\mathcal{X}^n$ 中的一个数据集作为输入元素. 对 $i = 1, \ldots, k$, A_i 是 ϵ_i-DP 算法.

依次运行算法 A_i 得到算法 $A : \mathcal{X}^n \to \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_k$，那么 A 是 ϵ-DP，其中 $\epsilon = \sum_{i=1}^n \epsilon_i$.

接下来的性质是说明操纵随机算法的输出不会影响隐私保护的效果：

命题 5.5 (后处理) *令 $A : \mathcal{X}^n \to \mathcal{Y}$, $B : \mathcal{Y} \to \mathcal{Z}$ 为相互独立的随机算法，其中 $\mathcal{X}$，$\mathcal{Y}$，$\mathcal{Z}$ 是任意集合. 如果 A 是 ϵ-DP 算法，那么组合算法 $B(A(\cdot))$ 也是 ϵ-DP 算法.*

证明 我们仍然只考虑离散情形，采用定义的方法证明

$$\begin{aligned}\Pr(B(A(x)) = b) &= \sum_{y \in \mathcal{Y}} \Pr(A(x) = y) \Pr(B(y) = b) \\ &\le e^{\epsilon} \sum_{y \in \mathcal{Y}} \Pr(A(x') = y) \Pr(B(y) = b) \\ &= e^{\epsilon} \Pr(B(A(x')) = b).\end{aligned}$$

□

最后，我们讨论如果有多个人的数据都发生变化的时候，隐私保护的性质会发生什么变化。

命题 5.6 (群体隐私) *令 $x, x' \in \mathcal{X}^n$ 是 k-相邻数据集，$1 \le k \le n$. 如果 A 是 ϵ-DP 算法，那么对所有的值集 E，我们有*

$$\Pr(A(x) \in E) \le e^{k\epsilon} \Pr(A(x') \in E).$$

证明 考虑数据集 $x_0, x_1, \cdots, x_k$，其中 $x_0 = x$, $x_k = x'$，且 x_i 和 x_{i+1} 是 1-相邻数据集，$i = 0, \cdots, k-1$. 那么

$$\begin{aligned}\Pr(A(x) \in E) &\le e^{\epsilon} \Pr(A(x_1) \in E) \le e^{2\epsilon} \Pr(A(x_2) \in E) \\ &\le \cdots \le e^{k\epsilon} \Pr(A(x') \in E).\end{aligned}$$ □

换言之，k-相邻数据集上 ϵ-DP 算法的表现仿佛一个 $k\epsilon$-DP 算法。

这一性质还可以推出 ϵ 的含义。我们知道数据集 $x, x' \in \mathcal{X}^n$ 最多在 n 个位置不同. 所以对于一个 ϵ-DP 算法 A，一定有

$$\Pr(A(x) \in E) \le e^{n\epsilon} \Pr(A(x') \in E).$$

如果这里的 ϵ 太小，意味着这一算法对任何输入都有相似的输出. 换句话说，算法压根没有输出任何有意义的内容. 于是，我们更定量说明了，ϵ 还代表信息的泄露量. 因此，一个实用的 DP 算法不能让 ϵ 太小，否则输出没有意义；也不能让 ϵ 太大，否则隐私保护效果不好.

§5.3 差分隐私的应用

在这一部分，我们将会具体讨论三个差分隐私的算法。

5.3.1 随机反应算法

我们从一个具体场景开始。假设有一名老师想要调查班上的同学有多少人曾经在考试中作弊. 设班上一共有 n 名同学，每个人回答一个数字 $x_i \in \{0, 1\}$. 对于每个 i，独立地按照以下规则根据 x_i 得到对应的 y_i：

$$y_i = \begin{cases} x_i, & \text{以 } 2/3 \text{ 的概率,} \\ 1 - x_i, & \text{以 } 1/3 \text{ 的概率.} \end{cases}$$

并输出 $\sum_{i=1}^{n} y_i$.

我们称这一算法为*随机反应*（RR）算法. 当 $y_i = 1$ 时，学生 i 可以声称这是由于算法的随机机制造成的，而并非自己真的作弊过.

RR 算法在隐私保护上的表现由以下定理给出：

定理 5.1 *RR 算法是* $\log 2$*-DP 算法.*

证明 记 Y_i 是 y_i 对应的随机变量. 我们知道 y_i 之间相互独立，所以

$$\Pr(A(x) = y) = \prod_{i=1}^{n} \Pr(Y_i = y_i \mid x_i).$$

对于 y_i，我们有

$$\frac{\Pr(Y_i = y_i \mid x_i = y_i)}{\Pr(Y_i = y_i \mid x_i = 1 - y_i)} = \frac{2/3}{1/3} = 2.$$

所以对于一对 1-相邻数据集 x, x' 和任意的 $y \in \mathcal{Y}$，假设 $x_j \neq x'_j$，有

$$\frac{1}{2} \leq \frac{\Pr(A(x) = y)}{\Pr(A(x') = y)} = \frac{\prod_{i=1}^{n} \Pr(Y_i = y_i \mid x_i)}{\prod_{i=1}^{n} \Pr(Y_i = y_i \mid x'_i)} = \frac{\Pr(Y_j = y_j \mid x_j)}{\Pr(Y_j = y_j \mid x'_j)} \leq 2.$$

由定义，RR 算法是 $\log 2$-DP 算法. □

另一方面，我们还关注 RR 算法得到的 $\sum_{i=1}^{n} y_i$ 是否能很好的估计出 $\sum_{i=1}^{n} x_i$. 为此，假设随机变量 $X_1, \cdots, X_n$ 独立服从参数为 p 的 Bernoulli 分布，即 $\Pr(X_i = 1) = p$ 而 $\Pr(X_i = 0) = 1 - p$. 于是，

$$q = \Pr(Y_i = 1) = p \cdot \frac{2}{3} + (1 - p) \cdot \frac{1}{3}$$
$$\implies p = 3q - 1.$$

我们得到的 $\sum_{i=1}^{n} y_i$ 相当于是 q，真正的参数 p 和 q 存在上述关系. 设 $\hat{X} = \sum_{i=1}^{n} (3Y_i - 1)$，可得 $\mathbb{E}[\hat{X}] = \mathbb{E}[\sum_{i=1}^{n} X_i]$.

5.3.2 全局灵敏度与 Laplace 机制

从 RR 算法中获得灵感，我们可以在算法中添加随机性，比如向算法 f 的输出添加噪声. 那么就引出了另一个问题，需要添加多大的噪声？这和算法本身的性质有关，比如，当输入只改变一点时，算法的输出会改变多大？我们定义全局灵敏度来衡量这一性质.

定义 5.3 (全局灵敏度) *给定算法* $f : \mathcal{X}^n \to \mathbb{R}$，*定义* f *的全局灵敏度为*

$$\mathsf{GS}_f = \sup_{x, x' \text{ 在 } \mathcal{X}^n \text{ 1-相邻}} |f(x) - f(x')|.$$

定义中的 1-相邻，可能会随着情景不同而改变. 全局灵敏度的定义是很直观的，就是改变一条数据会对算法输出带来的最大可能变化。

我们来计算一个简单的例子.

例 5.1 设 $f(x)=\frac{1}{n}\sum_{i=1}^{n}\phi(x_i)$，$\phi:\mathcal{X}\to[0,1]$ 是满射. 那么，

$$\begin{aligned}\mathsf{GS}_f &= \sup_{x,x'\text{ 在}\mathcal{X}^n\text{ 1-相邻}} |f(x)-f(x')| \\ &= \sup_{x,x'\text{ 在}\mathcal{X}^n\text{ 1-相邻}} \frac{1}{n}\left|\sum_{i=1}^{n}\phi(x_i)-\sum_{i=1}^{n}\phi(x_i')\right| \\ &= \sup_{x,x'\text{ 在}\mathcal{X}^n\text{ 只在}j\text{ 不同}} \frac{1}{n}\left|\phi(x_j)-\phi(x_j')\right| = \frac{1}{n}.\end{aligned}$$

利用全局灵敏度的概念，我们实际上有一个一般的方法来构造差分隐私算法. 我们称这一方法为 *Laplace* 机制. 首先引入 Laplace 分布的概念.

定义 5.4 (Laplace 分布) 给定参数 $\mu\in\mathbb{R}$ 和 $\lambda>0$，定义概率密度函数

$$h(x\mid\mu,\lambda)=\frac{1}{2\lambda}\exp\left(-\frac{|x-\mu|}{\lambda}\right).$$

我们称具有这一密度的分布为 Laplace 分布，记为 $\mathsf{Lap}(\mu,\lambda)$.

Laplace 分布是服从双边指数分布的随机变量进行线性变换后服从的分布. 具体来说，设 $X\sim\mathsf{DExp}(1)$，那么 $\lambda X+\mu\sim\mathsf{Lap}(\mu,\lambda)$. 在图 5.1 中，我们展示了不同参数的 Laplace 分布密度函数图像。更多关于 Laplace 分布的性质，我们留做习题. [lhy: 习题]

下面我们来陈述 Laplace 机制。我们想利用 Laplace 分布来创造一些随机性. 给定一个数据集 $x\in\mathcal{X}^n$ 和参数 ϵ. 对于一个算法 $f:\mathcal{X}^n\to\mathbb{R}$，先计算出 f 的全局灵敏度 GS_f。输出 $A_{\mathsf{Lap}}(\epsilon,x)=f(x)+Z$，其中 $Z\sim\mathsf{Lap}(0,\mathsf{GS}_f/\epsilon)$. 将随机算法 $A_{\mathsf{Lap}}(\epsilon,\cdot)$ 称为 *Laplace* 机制。我们有以下定理：

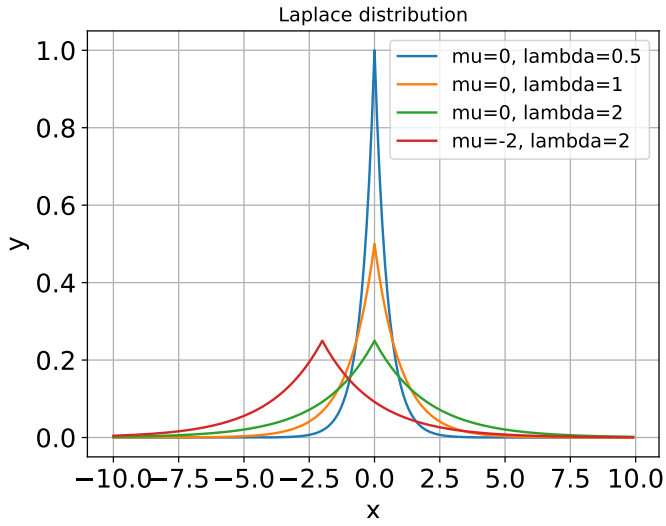
定理 5.2 对于任意的 $\epsilon>0$，*Laplace* 机制是 ϵ-*DP* 算法.

证明 设 x，x' 是两个 1-相邻数据集，记 $\mu=f(x)$，$\mu'=f(x')$. 由 Laplace 分布的性质可知，$A_{\mathsf{Lap}}(\epsilon,x)\sim\mathsf{Lap}(\mu,GS_f/\epsilon)$，$A_{\mathsf{Lap}}(\epsilon,x')\sim\mathsf{Lap}(\mu',GS_f/\epsilon)$.

因此，对于任意的 $y\in\mathcal{Y}$，有

$$\begin{aligned}\frac{h_x(y)}{h_{x'}(y)} &= \exp\left(-\epsilon\frac{|\mu-y|-|\mu'-y|}{GS_f}\right) \\ &\le \exp\left(\epsilon\frac{|\mu-\mu'|}{GS_f}\right)\le\exp(\epsilon).\end{aligned}$$

根据命题 5.2，命题得证。 □



[lhy: 重画]

图 5.1: Laplace 分布的密度函数图像

5.3.3 DP 版本 Llyod 算法

作为一个 Laplace 机制的具体实例，我们将 k-均值聚类问题的经典算法 Llyod 算法改造成一个差分隐私算法.

k-均值聚类问题指的是给定一个数据集 x，找到 k 个点（中心）$\{c_i\} \subseteq \mathbb{R}^d$，使得 $\sum_{i\in[n]} \min_{j\in[k]} \|x_i - c_j\|^2$ 最小. 通俗来说，就是找到 k 个中心，使得数据集中每个点到最近的中心的距离之和最小. k-均值问题最常见的解决方法是使用迭代的启发式的 *Lloyd 算法*，其表述为下：[lhy: 改成算法模板]

- 输入：数据集 $x \in \mathcal{X}^n$，这里 $\mathcal{X} = \{x \in \mathbb{R}^d : \|x\|_1 \leq 1\}$，参数 k.
- 随机初始化 $c_1^{(0)}, c_2^{(0)}, \cdots, c_k^{(0)} \in \mathcal{X}$.
- for $t = 1$ to T
  - for $j = 1$ to k
    - 计算 $S_j = \{i : c_j^{(t-1)}$ 是 x_i 最近的中心$\}$.
    - 更新 $c_j^{(t)} = \frac{1}{|S_j|} \sum_{i\in S_j} x_i$.
- 输出：$c_1^{(T)}, c_2^{(T)}, \cdots, c_k^{(T)}$.

这个算法可以达到很好的效果，但它并不能保证 DP 性质. 我们希望对这一算法进行小规模的修改，让它具有 ϵ-DP 的性质. 我们给出如下的 DP 版本 Llyod 算法. [lhy: 改成算法模板]

- 输入：数据集 $x \in \mathcal{X}^n$，这里 $\mathcal{X} = \{x \in \mathbb{R}^d : ||x||_1 \leq 1\}$，参数 k，参数 ϵ.
- $\epsilon' = \frac{\epsilon}{2T}$，随机初始化 $c_1^{(0)}, c_2^{(0)}, \cdots, c_k^{(0)} \in \mathcal{X}$.
- for $t = 1$ to T
  - for $j = 1$ to k
    - 计算 $S_j = \{i : c_j^{(t-1)}$ 是x_i 最近的中心$\}$.
    - $n_j = |S_j|$.
    - $a_j = \sum_{i \in S_j} x_i$.
    - 计算 $\hat{n}_j = n_j + Y$，$Y \sim \mathsf{Lap}(0, 2/\epsilon')$.
    - 计算 $\hat{a}_j = a_j + (Z_1, \cdots, Z_d)$，$Z_i$ i.i.d. $\sim \mathsf{Lap}(0, 2/\epsilon')$.
    - 更新 $c_j^{(t)} = \begin{cases} \frac{\hat{a}_j}{\hat{n}_j}, & \hat{n}_j \geq 1, \\ \mathcal{X} \text{ 上的一个随机均匀采样}, & \hat{n}_j < 1. \end{cases}$
- 输出：$c_1^{(T)}, c_2^{(T)}, \cdots, c_k^{(T)}$.

以下定理表明上面的算法确实是一个 ϵ-DP 算法。

定理 5.3 *DP 版本的 Lloyd 算法是 ϵ-DP 算法.*

证明 (证明概要) 我们只在这里陈述证明的大致想法，将细节留到习题[lhy: 习题].

在第一步中，我们设置了 $\epsilon' = \epsilon/2T$. 我们把整体的算法拆成 T 个阶段，其中第 t 轮迭代 A_t 以 $c_1^{(t-1)}, c_2^{(t-1)}, \ldots, c_k^{(t-1)}$ 作为输入，输出 $c_1^{(t)}, c_2^{(t)}, \ldots, c_k^{(t)}$. 如果可以得到 A_t 是 $2\epsilon'$-DP 算法，那么由 DP 算法的复合性，就可以得到整个算法是 $T \cdot 2\epsilon'$-DP，也就是 ϵ-DP.

进一步，考虑证明每一个 A_t 是 $2\epsilon'$-DP 算法. 将 A_t 内循环的每一轮（以 j 为变量）视为输入为 n_j, a_j，输出为 $\hat{n}_j$，$\hat{a}_j$ 的算法. 分别证明这些算法符合 $2\epsilon'$-DP，然后再次借助 DP 算法的复合性. □

§5.4 差分隐私与信息论

[lhy: 讨论信息论约束下的差分隐私问题]

§5.5 习题

§5.6 章末注记

第三部分

决策与优化

第六章　凸分析

本章将会建立关于决策与优化的基本理论，这些方法论都是数据驱动的机器学习的基础，他们涉及从数据到建立模型的步骤（即训练）. 优化与分析有着密不可分的联系，所以本章我们会立足优化问题的一些基本事实，建立凸分析理论。凸分析是优化理论的基础.

§6.1　决策与优化的基本原理

6.1.1　统计决策理论

[lhy: 这部分要细化]

我们在前一部分讨论过，数据（或者说信息）的意义总是体现在集合的对象中，我们把我们所关心的集合对象称为（随机）*总体* P。从概率论角度看，总体就是一个概率分布。现在我们从总体 P 中抽取一个*样本* X。这件事情在概率论上意味着我们得到了一个随机变量 X 服从分布 P。拿到样本之后，我们的任务是做出*好的决策*，因此，决策 T 是一个依赖 X 的函数. 比如说，P 是所有大学生的身高，X 是随机抽选一个人测量的身高，我们的决策 T 是估计大学生的平均身高。

“好的决策”指的是函数 T 能够具备某些量化指标. 其中非常常用的一个方法是通过*损失函数*来衡量，它是总体 P 和决策 $T(X)$ 的函数，即 $L(P,T(X))$. 损失函数在不同语境下有不同称呼。*损失函数*是机器学习和数理统计语境下常用的称呼。在控制理论中以及机器学习中，它被称为*代价函数*。在经济学和金融学的风险理论中，损失函数被称为*风险函数*，它意味着个体在面对不确定的环境下所需要面对的风险。而在优化理论中，损失函数往往被称为*目标函数*，表明所要优化的对象。

决策 T 的一种量化指标是最小化期望意义下的损失函数：

$$\min_T \mathbb{E}_{X\sim P}(L(P,T(X))).$$

在经济学中，这一量化指标实际上是 von Neumann 和 Morgenstern *期望效用理论*的具体体现。这一理论认为，个体在面对不确定的环境时，会选择最大化期望效用的决策；在风险理论的语境下，则是最小化期望风险的的决策。

现在我们考虑一个非常一般的决策任务。假设我们的任务是估计函数 f，但是我们只知道观测到的自变量 X（来自总体 P）以及它的函数值 $Y = f(X)$，我们的决策是函数的估计值 $\hat{f}$. 在机器学习中，f 通常是需要训练的模型. 我们可以写出若干种损失函数：

- 平方（L^2）损失函数：$L(P, T(X)) = (Y - \hat{f}(X))^2$. 使用此损失函数的时候，我们要假定 f 在实数范围取值。
- L^1 损失函数：$L(P, T(X)) = |Y - \hat{f}(X)|$. 使用此损失函数的时候，我们要假定 f 在实数范围取值。
- SVM 损失函数（hinge 损失函数）：$L(P, T(X)) = \max\{0, 1 - Y \cdot \hat{f}(X)\}$. 使用此损失函数的时候，我们一般要假定 $f(X) \in [-1, 1]$。
- 交叉熵损失函数：$L(P, T(X)) = CH(\hat{f}(X), Y)$.

这些损失函数会用在不同的场景之中。通常来说，机器学习中有两类问题：*回归问题*和*分类问题*和。他们两个的区别主要在于回归问题中 f 取值为实数，而且通常随自变量连续变化；而分类问题中 f 只取有限多个值，他们通常被作为标签（比如这张图片是人还是青蛙）使用。在回归问题中，我们通常使用平方损失函数或者 L^1 损失函数；在分类问题中，我们通常使用 SVM 损失函数或者交叉熵损失函数。

6.1.2 优化问题

现在我们从决策过渡到优化。在最简单的决策问题中，我们的目标就是找到某个 x 使得（期望）损失函数 f 最小. 此时，问题的一般形式为：

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & f_i(x) = 0, \quad i = 1, \ldots, m, \\ & g_j(x) \leq 0, \quad j = 1, \ldots, n, \\ & x \in \Omega. \end{aligned}$$

这里，s.t（subject to）之后的内容表明了 x 取值的限制，因此被称为*约束*. 其中 $f_i(x) = 0$ 和 $g_j(x) \leq 0$ 被称为*函数约束*，而 $x \in \Omega$ 被称为*集合约束*.

优化的基本任务就是找到 x 最小化损失函数.

根据损失函数 f、约束条件 f_i 和 g_j 的不同性质，我们可以对优化问题进行分类：

- 无约束优化：约束条件 f_i 和 g_j 实际上不存在，即 $m = n = 0$，并且 Ω 是全空间，比如 $\mathbb{R}^n$.
- 有约束优化：至少存在一个约束条件，即 $\min\{m, n\} \geq 1$，或者 Ω 不是全空间。
- 光滑优化：损失函数和约束条件都是可微函数.[1]
- 线性优化：损失函数和约束条件都是线性函数（形如 $a^\top x + b$）.

注. *[lhy: 介绍一下控制理论与优化理论的异同，特别是连续控制和随机优化。]*

下面我们看几个经典的优化例子。

例 6.1 (最小二乘法) 给定矩阵 $A \in \mathbb{R}^{m\times n}$ 和向量 $b \in \mathbb{R}^m$，考虑如下优化问题：

$$\begin{aligned} \min_x \quad & \|Ax - b\|_2^2 \\ s.t. \quad & x \in \mathbb{R}^n. \end{aligned}$$

这个问题被称为**最小二乘法**. 目标函数可以被写为 $(Ax-b)^\top(Ax-b)$，因此最小二乘法是一种典型的无约束光滑优化问题。

最小二乘法的解 x^* 实际上是**投影解**：b 的行向量投影到 A 的列向量形成的线性空间，正好是 Ax^*. *[lhy: 加个图，补全细节]* 因此，求投影也可以被写作一个优化问题.

例 6.2 (线性规划) 给定矩阵 $A \in \mathbb{R}^{m\times n}$ 和向量 $b \in \mathbb{R}^m$，考虑如下优化问题：

$$\begin{aligned} \min_x \quad & c^\top x \\ s.t. \quad & Ax \leq b, \\ & x \geq 0. \end{aligned}$$

这个问题被称为**线性规划**. 目标函数和约束条件都是线性的，因此线性规划是一种典型的线性优化问题. *[lhy: 加个图，补全细节]*

上面两个例子远远不能覆盖所有的优化问题，实际上，相当多的运筹学、机器学习和计算机科学中的问题都可以被视作（非线性）优化问题。

- 运筹学：线性规划、二次规划、整数规划、网络流问题、组合优化问题等.

[1] 光滑这一词的含义在不同的文献中大相径庭，它可以指（连续）可微、连续可微、二次（连续）可微或者无穷次可微。

- 金融学：投资组合优化、风险控制等.
- 机器学习：模型的训练.
- 计算机科学：图论中的极值问题，例如最短路径问题、最小生成树问题等.

因此，如果有一个能够解决通用优化问题的灵丹妙药，那么将会有极其重大的意义. 然而我们后面将会看到，一般的优化是一个难解的问题，更严谨一点说，不存在通用高效算法。

我们先需要明确解优化问题的算法到底是什么. 我们通过给出算法的一些特征来最终明确这一点。大部分优化算法都用了**迭代法**的思想：算法 A 接受一个自变量 x，输出一个自变量 $A(x)$，并把它作为下一轮的输入. 此外，一个算法还应该具有**通用性**，即它必须要能解决一类优化问题 F. 然后，算法具备通用性就意味着它在进行**黑箱优化**：F 必须要给算法提供必要的信息来完成求解，我们将这样的提供机制抽象为**先知**，记为 $\mathcal{O}$. 具体来说算法输入 x 给 $\mathcal{O}$，$\mathcal{O}$ 返回一些信息给算法（例如 x 处的函数值、导数值、Hessian 矩阵）.

接下来的问题是衡量优化算法的性能好坏。我们关注的是最坏情况，也就是说假如我们关注的是问题类 $P \subseteq F$，那么，我们要看的是优化算法在 P 中最差的表现如何。衡量优化算法性能的指标有以下几个：

- 近似程度：我们需要求在允许误差 ϵ 的情况下的近似解. 例如，函数值不大于最优值的 ϵ，或者离最优点距离不超过 ϵ. 考虑近似解是优化问题非常重要的一个想法，因为计算机的表示精度是有限的，我们不可能在所有情况下都求出精确解，所以求近似解是合理的要求。
- 运行时间（收敛速度，复杂度）：找到目标近似解需要调用先知的次数. 通常来说，运行时间会随近似度要求变高而变长，因此运行时间是一个关于近似程度的函数.

> **注.** 通常来说，优化算法的执行过程中还会进行除了调用先知之外的操作，例如进行加减乘除。然而，如果我们把所有这些操作都算入复杂度之中，算法的分析会变得非常困难，因此我们通常只考虑调用先知的次数。这样做的合理性在于，每一次的加减乘除等额外操作，几乎都是因为调用一次先知所以才进行的，因此我们可以把这些额外操作的时间都算入先知调用的时间之中。

有了上面这些准备，我们就可以将“没有万能算法”这一陈述写成定理了。

定理 6.1 (没有免费午餐定理) *[lhy: 给一个证明，以及更加严格的表述]* 设 F 是有限个优化问题的集合，F 上有一个任意的概率分布。考虑一个 F 上的优化算法，记号 d_t 表示 t 轮迭代之后算法产生的点列

$$(x_t(1), y_t(1)), \ldots, (x_t(t), y_t(t)).$$

给定迭代轮数 t，优化问题 f，算法 A，优化过程所产生的点列概率分布为 $P(d_t|f,t,A)$。那么，对任意优化算法 A_1, A_2，

$$\sum_{f\in F} P(d_t|f,t,A_1) = \sum_{f\in F} P(d_t|f,t,A_2).$$

这一定理意味着，对特定的点列，任何算法在所有实例上产生它的概率总和是一样的。

那么，点列和“没有万能算法”有什么样的关系呢？实际上，衡量算法性能的指标和点列有非常密切的联系。比如说，算法花了 k 步找到一个 ϵ-近似解，用点列的语言来说就是算法迭代产生的点列，长度至多是 k 并且最后一个点距离最优解距离不大于 ϵ. 粗略地说，任意点列成立的性质意味着任意指标成立的性质。因此，对于任何一类优化问题来说，不论以何种指标来衡量性能，优化算法在某些问题上表现出来的突出性能一定会在另一些问题上被抵消. 没有一个万能的算法可以高效解决所有优化问题！

6.1.3 例子：网格搜索算法

前面对于概念的讨论依然非常抽象，所以下面我们看一个具体的例子，这个例子将会展示从算法分析的角度，优化所关注的主要问题。考虑如下优化问题：

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & x \in [0,1]^n. \end{aligned} \tag{6.1}$$

其中 $f(x)$ 是 Lipschitz 连续函数，即它满足

$$|f(x) - f(y)| \le L \|x - y\|_\infty, \quad \forall x, y \in [0,1]^n.$$

关于优化算法的假设如下。首先，我们可以访问**零阶先知**，即 $\mathcal{O}(x) = f(x)$。其次，优化算法需要去找到 **ϵ-近似解**，即函数值至多比最小值大 ϵ 的解.

> **注.** 在优化中，我们会经常使用词语“零阶”“一阶”等等，所谓的“阶”指的是函数导数阶数，零阶先知指的是我们可以访问函数值，一阶先知指的是我们可以访问一阶导数，以此类推. 后面还会有零阶条件、一阶条件等等，他们的含义类似。

我们考虑一个非常简单的算法，他被称为网格搜索：

- 将 $[0,1]$ 等分成 p 份，$[0,1] = [0,1/p] \cup \ldots [(p-1)/p, 1]$.
- 遍历 $(p+1)^n$ 个格点：

$$x_{(i_1,\ldots,i_n)} = \left(\frac{i_1}{p}, \ldots, \frac{i_n}{p}\right)^{\mathsf{T}},$$

  $i_k \in \{0,1,\ldots,p\}$.
- 对每个格点询问先知得到其函数值，输出函数值最小的一个（记为 $(\bar{x}, f(\bar{x}))$）.

我们对于网格搜索算法问的问题是，它的复杂度如何。也就是说，它需要调用先知多少次才能找到一个 ϵ-近似解？我们从一个引理开始。

引理 6.1 *设 (6.1) 的最优值为 f^*，那么*

$$f(\bar{x}) - f^* \leq \frac{L}{2p}.$$

证明 设 x^* 是最优点，存在一个方格包含 x^*：

$$x_{(i_1,\ldots,i_n)} \leq x^* \leq x_{(i_1+1,\ldots,i_n+1)}.$$

这个方格的长为 $1/p$，所以我们可以选取方格的某个顶点 $\hat{x}$，使得它的每一个轴离 x^* 的距离都不超过 $1/(2p)$.[lhy: 画个图]

于是根据 Lipschitz 条件，

$$f(\bar{x}) - f^* \leq f(\hat{x}) - f(x^*) \leq L \|\hat{x} - x^*\|_\infty \leq \frac{L}{2p}.$$ □

利用这个引理，我们可以证明网格搜索算法的复杂度.

定理 6.2 *网格搜索算法可以找到找到一个 ϵ-近似解，其调用 $\mathcal{O}$ 的次数至多为*

$$\left(\left\lfloor \frac{L}{2\epsilon} \right\rfloor + 2\right)^n.$$

- 证明：取 $p = \lfloor L/(2\epsilon) \rfloor + 1$，代入引理 6.1 即可.

网格搜索法的运行时间给了优化问题 (6.1) 一个求解时间的**上界**. 然而这个上界维数呈指数关系，通常来说都是不可接受的复杂度. (6.1) 会有更好的算法呢？这就是**下界问题**. 令人惊讶的是，对于这一个问题，我们可以证明网格搜索法是渐近意义下最优的！

定理 6.3 设 $\epsilon < L/2$，任何访问 $\mathcal{O}$ 的算法（零阶算法）找到 (6.1) 的 ϵ-近似解至少需要调用 $\mathcal{O}$

$$\left\lfloor \frac{L}{2\epsilon} \right\rfloor^n$$

次.*[lhy: 改一下表述，看不懂]*

证明 设 $p = \lfloor L/(2\epsilon) \rfloor$，对任意算法 A，我们尝试构造一个函数，使得 A 调用 $\mathcal{O}$ p^n 次时最多找到一个 ϵ-近似解.

构造思路：对任何测试点，使得 $\mathcal{O}$ 总是返回 0，于是，算法 A 只能找到 $f = 0$ 的解 $\bar{x}$. 注意到算法只能根据先知的返回来进行操作，因此我们先假定这样的函数存在。[lhy: 改一下，读不懂]。

根据鸽巢原理，网格中至少有一个长为 $1/p$ 的小方格 B 内部没有包含任何测试点. 假设这个小方格的中心是 x^*，构造 $\bar{f}(x) = \min\{0, L\|x - x^*\|_\infty - \epsilon\}$。容易看出，$\bar{f}$ 是 L-Lipschitz 函数，并且最小值为 $-\epsilon$.

函数 $\bar{f}$ 非零的点只在方格 $B' = \{x \in [0,1]^n : \|x - x^*\|_\infty \le \epsilon/L\}$ 内部. 因为 $1/(2p) \ge \epsilon/L$，所以 $B' \subseteq B$. 所以所有测试点上 $\mathcal{O}$ 都会返回 0，这是一个 ϵ-近似解. 因此 A 通过小于 p^n 次对 $\mathcal{O}$ 的调用最多只能找到 ϵ-近似解. □

以上两个结论分别给出了 (6.1) 问题的上下界，对比他们：

问题的上界：	问题的下界：
$\left(\left\lfloor \frac{L}{2\epsilon} \right\rfloor + 2\right)^n$	$\left\lfloor \frac{L}{2\epsilon} \right\rfloor^n$

尽管网格搜索是一个很慢的算法，但是我们证明了，在渐近意义下，优化问题 (6.1) 的最优算法就是网格搜索！因此，我们可以说，一般的优化问题是难解的。

当我们聚焦在特定的问题类上，优化问题并不一定是难解的。比如，线性规划可以在关于约束个数和变量个数的多项式时间内解出精确解。然而，现实中大部分重要的问题并不是线性的，因此，我们接下来的关键问题是识别出一类可以快速求解的非线性优化问题，这就是凸函数的意义。

§6.2 凸函数

我们首先看无约束优化，看看什么样的损失函数可以快速求最小值. **梯度下降方法**是最古老也最常用的方法. 梯度下降每步计算函数的导数（梯度），然后朝着负梯度方向移动到下一个点. 与梯度下降算法相关的最小值必要条件是一阶条件。

定理 6.4 (一阶条件) *如果 x^* 是可微函数 f 的局部最小值，那么*

$$f'(x^*) = 0.$$

证明 根据局部最小值的定义，存在 $r > 0$，对于任意 $\|y - x^*\| < r$，$f(y) \geq f(x^*)$. 因此 $f(y) = f(x^*) + \langle f'(x^*), y - x^* \rangle + o(\|y - x^*\|) \geq f(x^*)$. 因此，对任意 $s \in \mathbb{R}^n$，$\langle f'(x^*), s \rangle \geq 0$. 考虑方向 s 和 $-s$ 可得 $\langle f'(x^*), s \rangle = 0$. 由 s 的任意性，$f'(x^*) = 0$. □

现在，从一阶条件出发，我们考虑如下优化函数类 $\mathcal{F}$，满足如下三个假设：

- 假设 1：对任意 $f \in \mathcal{F}$，如果 x 满足一阶条件，那么 x 是 f 的全局最小值点.
- 假设 2：对任意 $f, g \in \mathcal{F}$，$\alpha, \beta \geq 0$，$\alpha f + \beta g \in \mathcal{F}$.
- 假设 3：线性函数 $f(x) = \langle \alpha, x \rangle + b \in \mathcal{F}$.

假设 1 使得利用一阶条件的算法可以找到全局最优解. 假设 2 描述了对 $\mathcal{F}$ 封闭的操作，这样的操作实际上就是要求函数对线性组合封闭。要求系数 α 和 β 非负是为了保证一阶条件得到的确实是最小值而不是最大值。一个例子是，如果 $x^2 \in \mathcal{F}$，并且线性组合不限制非负系数，那么 $-x^2 \in \mathcal{F}$，但是后者一阶条件对应的是最大值而非最小值，这就会与假设 1 矛盾。假设 3 提供了 $\mathcal{F}$ 的基本函数，即线性函数。我们之前说过，线性规划是易解的，所以 $\mathcal{F}$ 至少要包含线性函数。

从这三个假设出发，我们可以给出函数类 $\mathcal{F}$ 的刻画。

固定一个函数 $f \in \mathcal{F}$，一个点 $x \in \mathbb{R}^n$，定义 $\phi(y) = f(y) - \langle f'(x), y \rangle$。根据假设 2 和假设 3，$\phi(y) \in \mathcal{F}$. $\phi'(y)|_{y=x} = f'(x) - f'(x) = 0$，根据假设 1，$x$ 是 ϕ 的全局最小值. 因此，$\phi(y) \geq \phi(x)$，即

$$f(y) \geq f(x) + \langle f'(x), y - x \rangle . \tag{6.2}$$

这一不等式给出了**可微凸函数**的定义：任意 x, y 都满足 (6.2) 的函数. 这一不等式有很强的几何直观，从 x 处做函数 f 的切线，那么切线上的点都在函数下方. 从这个角度来看，凸函数的定义是向下凸的函数. [lhy: 画个图]

非常有趣的是，$\mathcal{F}$ 完全由可微凸函数组成，这一点可以通过下面的定理得到证明。

定理 6.5 *函数 $f \in \mathcal{F}$ 当且仅当 f 是可微凸函数.*

证明 只需验证满足 (6.2) 的函数属于 $\mathcal{F}$.

- 假设 1 令 $f'(x) = 0$ 即得任意 y 都有 $f(y) \geq f(x)$.

- 假设 2 利用内积的双线性性和导数加法公式.
- 假设 3 是平凡的. □

[lhy: 习题：如果 f 是二次可微的，那么他的二阶导数（Hessian 矩阵）$f''(x)$ 和凸函数有何关系？]

[lhy: 给一些凸函数的例子]

从数学的角度来说，给了凸性的定义，下一步任务就是给出保持凸性不变的操作，这样我们可以用基本函数构造出更多的函数。

假设 2 实际上已经给出了一种凸性不变的操作，我们将它写成以下命题：

命题 6.1 *对任意 $f,g \in \mathcal{F}$，$\alpha,\beta \geq 0$，$\alpha f + \beta g \in \mathcal{F}$.*

另一个可以保持凸性的操作是仿射变换可以保持凸性。所谓仿射变换，指的是向量空间 $\mathbb{R}^n$ 到 $\mathbb{R}^m$ 的映射 $x \mapsto Ax + b$，其中 A 是 $m \times n$ 矩阵，$b \in \mathbb{R}^m$. 仿射变换实际上就是线性函数，只是我们用变换的方式来表示它。

命题 6.2 *假设函数 $f : \mathbb{R}^n \to \mathbb{R}$ 属于 $\mathcal{F}$，那么对任意仿射变换 $x \mapsto Ax + b$，$g(x) = f(Ax + b) \in \mathcal{F}$.*

证明 $g'(x) = A^\mathsf{T} f'(Ax + b)$，因此

$$\begin{aligned} g(y) = f(Ay + b) &\geq f(Ax + b) + \langle f'(Ax + b), (Ay + b) - (Ax + b) \rangle \\ &= f(Ax + b) + \langle f'(Ax + b), A(y - x) \rangle \\ &= g(x) + \left\langle A^\mathsf{T} f'(Ax + b), y - x \right\rangle \\ &= g(x) + \langle g'(x), y - x \rangle . \end{aligned}$$

□

更多保持凸性不变的操作，见习题。[lhy: 习题：给出更多保持凸性不变的操作]

凸函数的一个重要性质是 Jensen 不等式：

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha) f(y). \tag{6.3}$$

Jensen 不等式具有很强的几何解释：画一条 f 的割线，那么 f 的函数图像位于割线上方。实际上，Jensen 不等式给了凸函数一种等价的定义：

定理 6.6 *设 f 是连续可微的函数，那么 f 满足 (6.2) 当且仅当 f 满足 (6.3).*

证明 $\Longrightarrow$：在 (6.2) 中，取 x 为 $\alpha x+(1-\alpha)y$，y 分别取为 x 和 y，如此得到两个不等式，加权求和即得 (6.3).

$\Longleftarrow$：$f(y) \geq (1-\alpha)^{-1}(f(\alpha x+(1-\alpha)y)-\alpha f(x))$

$$= f(x)+(1-\alpha)^{-1}(f(x+(1-\alpha)(y-x))-f(x)).$$

令 $\alpha \to 1$ 即得 (6.2). □

如果函数 f 不是可微的，那么定理 6.6 给了一个凸函数更加本质的定义：

定义 6.1 (凸函数) *函数 f 满足对任意 x,y 成立(6.3)，那么称 f 是*凸函数.

扩展定义之后的凸函数包括了我们之前讲的 L^p（$p=1,2$）损失和 SVM 损失，以及机器学习中用到的大部分损失函数. 在实际情况中，凸函数是一类存在快速收敛算法的函数，例如梯度下降和 Netwon 迭代法. 因此，我们可以说，凸函数类划定了非线性优化中可以快速求解的函数类. 自此，凸性成为了优化中的核心概念，正如 R.T.Rockafellar [?] 所说：

> In fact the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity.

§6.3 凸集

接下来我们考虑约束优化问题：

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & x \in \Omega. \end{aligned}$$

一个自然的问题是，什么样 Ω 会存在快速收敛的算法？我们将看到，凸集将会是这个问题的答案.

6.3.1 基本定义和性质

回忆凸函数的一般定义：任意 $\alpha \in [0,1]$ 和 $x,y \in \mathbb{R}^n$，

$$f(\alpha x+(1-\alpha)y) \leq \alpha f(x)+(1-\alpha)f(y).$$

这里，我们隐含的要求是线段 xy 上的每一点都可以求函数值. 因此，如果我们希望凸函数能够包含在带约束的优化中，一个自然的要求就是对任意 $x,y \in \Omega$，线段 $xy \subseteq \Omega$. 这就是凸集的定义：

定义 6.2 (凸集) 集合 C 被称为**凸集**当且仅当对任意 $x,y \in C$，线段 $\{\alpha x + (1-\alpha)y : \alpha \in [0,1]\} \subseteq C$.

我们来看一些凸集的例子：

例 6.3

- 超平面：$\{x \in \mathbb{R}^n : a^\top x = b\}$，$a \in \mathbb{R}^n$，$b \in \mathbb{R}$.
- 半空间：$\{x \in \mathbb{R}^n : a^\top x \geq b\}$，$a \in \mathbb{R}^n$，$b \in \mathbb{R}$.
- 球：$\{x \in \mathbb{R}^n : \|x - x_0\| \leq r\}$，其中 $\|\cdot\|$ 是任意一种范数.
- 锥：C 是一个锥指的是任意 $x,y \in C$ 和任意 $\alpha,\beta \geq 0$，$\alpha x + \beta y \in C$.

另外一些重要的例子是凸函数诱导的凸集。首先是上图。

定义 6.3 (上图) 函数 f 的**上图**是指集合 $\mathrm{epi}(f) = \{(x,y) \in \mathbb{R}^n \times \mathbb{R} : y \geq f(x)\}$. 直观上说，$\mathrm{epi}(f)$ 是位于函数 f 的图像上方的区域.

[lhy: 画图]

上图揭示了凸集与凸函数的关系：

定理 6.7 上图 $\mathrm{epi}(f)$ 是凸集当且仅当 f 是凸函数.

证明 $\Longrightarrow$：$(x,f(x)),(y,f(y)) \in \mathrm{epi}(f)$，因此 $(\alpha x + (1-\alpha)y, \alpha f(x) + (1-\alpha)f(y)) \in \mathrm{epi}(f)$，所以 $\alpha f(x) + (1-\alpha)f(y) \geq f(\alpha x + (1-\alpha)y)$.

$\Longleftarrow$：取 $(x_1,y_1),(x_2,y_2) \in \mathrm{epi}(f)$，得到 $f(\alpha x_1 + (1-\alpha)x_2) \leq \alpha f(x_1) + (1-\alpha)f(x_2) \leq \alpha y_1 + (1-\alpha)y_2$，所以 $(\alpha x_1 + (1-\alpha)x_2, \alpha y_1 + (1-\alpha)y_2) \in \mathrm{epi}(f)$. □

然后是下水平集。

定义 6.4 (下水平集) 给定 $t \in \mathbb{R}$，函数 f 的**下水平集**是指集合 $C_t(f) = \{x \in \mathbb{R}^n : f(x) \leq t\}$. 直观上说，下水平集是函数值小于 t 的区域.

命题 6.3 如果函数 f 是凸函数，那么对任意 $t \in \mathbb{R}$，下水平集 $C_t(f)$ 是凸集.

这个命题的证明是直接的，我们留做习题。值得注意的是，这一命题的逆命题是不成立的，我们也在习题中讨论。[lhy: 习题：证明命题 6.3]

接下来，我们研究凸集的性质。根据定义，直接有：

命题 6.4 凸集的任意交依然是凸集.

我们可以利用这个性质来构造新的凸集.

例 6.4
- 仿射空间：有限个超平面的交，等价地写作 $\{x \in \mathbb{R}^n : Ax = b\}$，$A \in \mathbb{R}^{m\times n}$，$b \in \mathbb{R}^m$.
- 多面体：有限个半空间的交，等价地写作 $\{x \in \mathbb{R}^n : Ax \le b\}$，$A \in \mathbb{R}^{m\times n}$，$b \in \mathbb{R}^m$.
- 单纯形：$\Delta_n = \{x \in \mathbb{R}^n : x_1 + \cdots + x_n = 1, x_i \ge 0, \forall i\}$，是一种特殊的多面体.
- 凸包：给定任意集合 S，可以定义包含它的最小凸集：

$$\bigcap_{S\subseteq C \text{ 是凸的}} C.$$

从优化的角度来看，凸集本身具有最优近似性质. 我们之前在例 6.1 讨论过，求点到线性空间的投影是一个优化问题. 任何一个点都可以唯一地投影到线性空间的某个点上，因此整个空间通过投影就被近似到了一个线性子空间中.

现在我们来推广这一考虑。给定任意非空集合 $C \subseteq \mathbb{R}^n$，我们尝试将整个空间近似到集合 C 中. 定义点 x 到 C 的距离为：$d(x, C) = \inf_{p\in C} \|x - p\|_2$. 如果存在 $p \in C$ 达到了距离 $d(x, C)$，我们就说 p 是 x 在 C 上的一个投影. 到当 C 就是线性空间的时候，这个定义恰好也是原来投影的定义.

如果 $\mathbb{R}^n$ 中的每个点都在 C 中有唯一的投影，那么就称 C 是 **Chebyshev 集**. C 是 Chebyshev 集意味着 C 是整个空间的一个好的近似. 我们有如下定理：

定理 6.8 *在 $\mathbb{R}^n$ 中，C 是 Chebyshev 集当且仅当 C 是闭凸集.*

这一定理的证明非常复杂，我们留做习题。[lhy: 习题：证明上述定理]

因此，闭凸集是唯一具有良好近似性质的集合类，这又一次从优化角度说明了凸性的重要性.

6.3.2 分离超平面定理

[lhy: 扩展这部分内容，把 Banach-Hahn 定理还有画图的事情处理好。]

凸集还有一个不平凡且重要的性质：

定理 6.9 (分离超平面定理) *设 C, D 是两个非空不交凸集，也就是 $C \cap D = \varnothing$. 那么，存在 $a \neq 0$ 和 $b \in \mathbb{R}$ 使得*

- *任意 $x \in C$，$a^\top x \le b$.*

- 任意 $x \in D$，$a^{\mathsf{T}} x \geq b$.

由 $a^{\mathsf{T}} x = b$ 定义的超平面被称为**分离超平面**.

如果两个凸集只有一个公共点，并且其中一个凸集有内点，分离超平面定理依然成立，证明留做习题。[lhy: 习题：证明分离超平面定理]

下面我们来证明定理 6.9.

证明 定义两个集合间的距离为：

$$d(C, D) = \inf_{x \in C, y \in D} \|x - y\|_2 .$$

我们只证明 C 和 D 都是有界闭集的情况. 此时，存在 $c \in C, d \in D$ 使得 $\|c - d\|_2 = d(C, D)$. 令 $a = d - c$，$b = (\|d\|_2^2 - \|c\|_2^2)/2$. 只需证明 $f(x) = a^{\mathsf{T}} x - b$ 在 C 上非正在 D 上非负. 对称地，只证明在 D 上非负.

注意到 $f(x) = a^{\mathsf{T}} x - b = (d - c)^{\mathsf{T}}(x - (d + c)/2)$. 假设对某个 $u \in D$，$f(u) < 0$，于是

$$f(u) = (d - c)^{\mathsf{T}}(u - d) + \frac{1}{2}\|d - c\|_2^2 < 0 \implies (d - c)^{\mathsf{T}}(u - d) < 0.$$

因此，对充分小的 $t > 0$，$\|d + t(u - d) - c\|_2 < \|d + 0 \cdot (u - d) - c\|_2 = \|d - c\|_2$. 同时，因为 D 是凸集，$d + t(u - d) \in D$. 这与 d 和 c 的假设矛盾！ □

第七章　对偶理论

在本章中，我们考虑带约束的规划问题。它的一般形式是

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & h_i(x) = 0, \qquad i = 1, \dots, m, \\ & g_j(x) \le 0, \qquad j = 1, \dots, p, \\ & x \in \Omega \subseteq \mathbb{R}^n. \end{aligned}$$

其中，$m \le n$，函数 f, h_i, g_j 都是连续的，且通常假设它们拥有连续的二阶导.

为简化记号，们用向量形式的函数，即 $h = (h_1, h_2, \dots, h_m)$ 和 $g = (g_1, g_2, \dots, g_p)$，把问题的形式重写为：

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & h(x) = 0, \\ & g(x) \le 0, \\ & x \in \Omega. \end{aligned}$$

约束 $h(x) = 0, g(x) \le 0$ 被称作**函数约束**. $x \in \Omega$ 是**集合约束**. 我们并不强调集合约束，因此假设在大部分情况下 Ω 就是整个 $\mathbb{R}^n$ 的空间，或者问题的解就在 Ω 的内部.

一个满足所有函数约束的点 $x \in \Omega$ 被称作**可行解**，而使得 f 取得最小值的可行解叫做**最优解**. 有时候优化问题的目标可能是最大化 f，此时相应的最优解就是使得 f 取得最大值的可行解。本章的任务是讨论各种情况下最优值的必要条件，这些必要条件最终形成了所谓的**对偶理论**。

§7.1 条件极值与 Lagrange 乘子法

我们现在先只考虑等式约束

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & h(x) = 0, \\ & x \in \Omega. \end{aligned} \tag{7.1}$$

这些约束定义了一个 $\mathbb{R}^n$ 的子集，可以被看作一个曲面. 在恰当的条件下，这个曲面是 $n-m$ 维的（类比线性空间）. 如果函数 $h_i, i = 1, 2, \ldots, m$ 有一阶连续导数（记为属于 C^1），那么他们定义的曲面就是光滑的. 曲面上可以定义切空间.

为了引入切空间，我们先介绍曲线，然后曲线的定义可以导出切空间的定义.

定义 7.1 (曲线与切空间) • 超平面 S 上的一条**曲线**是一系列点的集合：$x(t) \in S$，它们以 t 为参数，$a \le t \le b$ 且在该区间上连续.

- 称曲线是**可微的**，如果 $\dot{x} = \mathrm{d}(x(t))/\mathrm{d}t$ 存在.
- 称曲线 $x(t)$ **经过点** x^*，如果存在 $t^* \in [a, b]$ 使得 $x^* = x(t^*)$.
- 曲线在 x^* 的**导数**被定义为 $\dot{x}(t^*)$，该导数是 $\mathbb{R}^n$ 内的一个向量，这个向量可以看作沿着曲线 t 在 x^* 处的切向量.
- 考虑所有 S 内经过点 x^* 的可微曲线. 点 x^* 处的**切空间** $T_{x^*}(S)$ 被定义为这些曲线在点 x^* 处的导数的集合.

切空间的重要特点是，它是一个线性空间.

引理 7.1 切空间是一个线性空间。

既然切空间是一个线性空间，我们的一个主要目标就是给出切空间的显示表达，比如给出它的一组基向量。考虑一条曲线 $x(t)$，如果它在 $h_i(x) = 0$ 形成的曲面上，那么应该有

$$\frac{\mathrm{d}}{\mathrm{d}t} h_i(x(t)) = 0 \iff \nabla_x h_i(x(t)) \dot{x}(t) = 0.$$

因此 $x(t)$ 的切向量和该点处函数 $h_i(x(t))$ 的导数正交. 于是，如果 $x(t)$ 在 $h(x) = 0$ 形成的曲面上，那么 $x(t)$ 处的导数 $\nabla h(x(t))$ 是切平面的法向量. 这一数学推导的示意图见图 7.1.

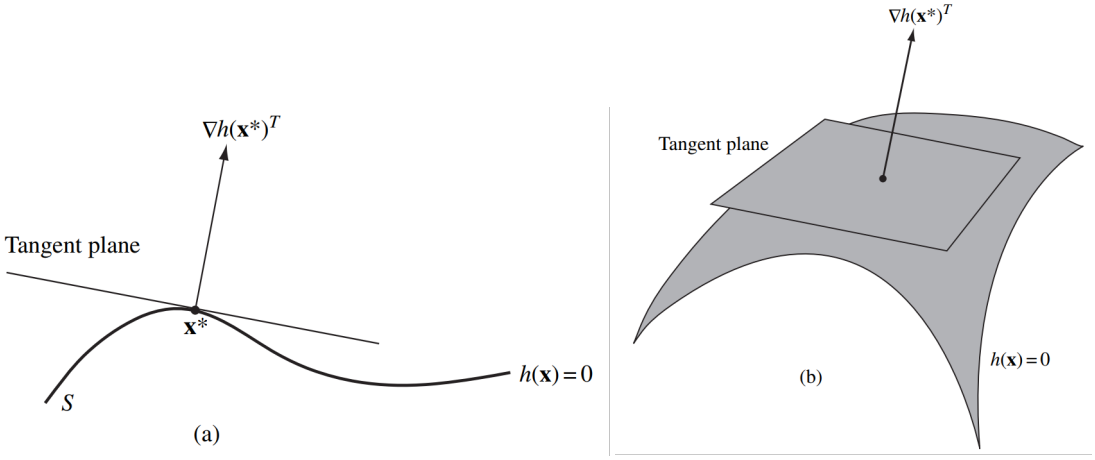


图 7.1: 切空间的示意图.

我们把刚刚得到的垂直于 $\nabla h(x^*)y$ 的子空间（即正交补空间）记作

$$M = \{y \in \mathbb{R}^n : \nabla h(x^*)y = 0\}.$$

我们已经证明 $T_{x^*}(S) \subseteq M$。反过来，在什么条件下会有 $M = T_{x^*}(S)$? 为此，我们引入正规点的概念。

定义 7.2 (正规点) *考虑优化问题 (7.1)，当一个点 $x^* \in \Omega$ 满足约束 $h(x^*) = 0$，且梯度向量 $\nabla h_1(x^*), \nabla h_2(x^*), \ldots, \nabla h_m(x^*)$ 线性无关时，它被称作该约束的*正规点.

直观上来说，正规点上每一条约束都起到了实际的作用，因此梯度向量 $\nabla h_i(x^*)$ 形成了一个线性无关的集合，张成了空间 $M^\perp$. 此时，切空间恰好完全垂直于 $M^\perp$，即 $T_{x^*}(S) = M$. 这一几何直观见图 7.2，点 x^* 处的两个等式约束共同确定了该点的切空间. 因此，在正规点，用约束函数的梯度来描述切空间是可行的.

定理 7.1 (正规点切空间刻画定理) *设曲面 $S \subseteq \mathbb{R}^n$ 由约束 $h(x) = 0$ 定义，$x^* \in S$ 是正规点，那么，*

$$T_{x^*}(S) = M = \{y : \nabla h(x^*)y = 0\}.$$

该定理的证明需要隐函数定理，对微积分要求较高，我们这里略去.

有了切空间的准备，现在我们要对正规点推导带约束的优化问题的极值条件. 考虑优化(7.1)，设 x^* 是一个约束 $h(x) = 0$ 一个正规点，同时也是函数 f 的一个在可行域中的极值点.

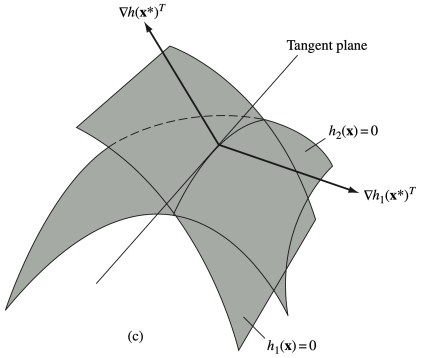


图 7.2: 正规点示意图。

引理 7.2 *对 $y \in \mathbb{R}^n$，如果 $\nabla h(x^*)y = 0$，那么 $\nabla f(x^*)y = 0$.*

证明 因为 x^* 是正规点，根据正规点切空间刻画定理，$\nabla h(x^*)y = 0$ 等价于 y 是 x^* 处的切空间中的向量. 根据定义，存在该约束曲面内的某个光滑曲线 $x(t)$，经过点 x^*，并且以 y 为切向量. 那么，$x(0) = x^*$，$\dot{x}(0) = y$，且 $h(x(t)) = 0$ 在区间 $-a \le t \le a$ 上成立（对某个正数 a）. 因为点 x^* 是一个函数 f 的受等式约束的极值点，我们有

$$\frac{\mathrm{d}}{\mathrm{d}t} f(x(t))\bigg|_{t=0} = 0 \iff \nabla f(x^*)y = 0. \qquad \square$$

引理 7.2 对任意 $y \in \mathbb{R}^n$ 都成立，根据线性代数零空间的性质，这等价于 $\nabla f(x^*)$ 是 $\nabla h_i(x^*)$ 的线性组合，即

$$\nabla f(x^*) = \sum_i \lambda_i \nabla h_i(x^*).$$

据此，我们得到条件极值的一阶必要条件：

定理 7.2 (条件极值的一阶必要条件) *令 x^* 是一个 f 的满足约束 $h(x) = 0$ 的正规极值点. 那么存在一个 $\lambda \in \mathbb{R}^m$ 使得*

$$\nabla f(x^*) + \lambda^{\mathsf{T}} \nabla h(x^*) = 0.$$

一阶必要条件 $\nabla f(x^*) + \lambda^{\mathsf{T}} \nabla h(x^*) = 0$ 以及约束 $h(x^*) = 0$ 给出了 $n + m$ 个等式，包含 x^*, λ 在内的 $n + m$ 个变量. 因此在非退化的情况下，他们给出了一个唯一解.

引入与这个约束问题对应的 Lagrange 函数：

$$l(x, \lambda) = f(x) + \lambda^{\mathsf{T}} h(x).$$

λ 被称为 *Lagrange* 乘子. 必要条件可以被写作:

$$\nabla_x l(x,\lambda) = 0,$$
$$\nabla_\lambda l(x,\lambda) = 0.$$

例 7.1 (最大熵) 考虑一个离散的概率分布，其分布列为 $p_i = \Pr(X = x_i), i = 1,\dots,n$. 该分布的熵为

$$\epsilon = -\sum_{i=1}^{n} p_i \log p_i.$$

该分布的均值为 $\sum_{i=1}^{n} x_i p_i$.

如果均值固定为 m，求解使熵最大化的参数可以被转化成以下问题:

$$\begin{aligned} \max \quad & -\sum_{i=1}^{n} p_i \log p_i \\ s.t. \quad & \sum_{i=1}^{n} p_i = 1, \\ & \sum_{i=1}^{n} x_i p_i = m, \\ & p_i \geq 0, \qquad i = 1,2,\dots,n. \end{aligned}$$

我们先忽略非负约束，假设这些约束不会被触发. 引入两个 *Lagrange* 乘子，λ 和 μ，则 *Lagrange* 函数为

$$l = \sum_{i=1}^{n} (-p_i \log p_i + \lambda p_i + \mu x_i p_i) - \lambda - \mu m.$$

由一阶必要条件，$-\log p_i - 1 + \lambda + \mu x_i = 0$，$i = 1,2,\dots,n$. 因此，

$$p_i = \exp((\lambda - 1) + \mu x_i), \quad i = 1,2,\dots,n.$$

注意 $p_i > 0$，所以非负约束确实没有被触发. *Lagrange* 乘子 λ 和 μ 是两个用来保证等式约束被满足的参数.

§7.2 Karush–Kuhn–Tucker 条件

现在加入不等式约束，考虑以下形式的问题:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & h(x) = 0, \\ & g(x) \leq 0. \end{aligned} \tag{7.2}$$

假设 f 和 h 和前面一样，g 是一个 p 维的函数，$f,h,g \in C^1$.

我们推广正规点 x^* 的定义为：

定义 7.3 (正规点) *考虑优化问题 (7.2)，点 x^* 被称为*正规点*，如果*

- *它满足约束：$h(x^*) = 0, g(x^*) \leq 0$.*
- *令 J 为满足 $g_j(x^*) = 0$ 的下标 j 的集合（激活的约束）. 那么，梯度向量 $\nabla h_i(x^*)$，$\nabla g_j(x^*)$，$1 \leq i \leq m$，$j \in J$ 是线性无关的.*

换言之，此时的正规点不仅考虑等式约束，还要考虑起作用的或者说被激活的不等式约束，这些不等式约束相当于等式约束。类似 Lagrange 乘子法，此时的一阶必要条件为：

定理 7.3 (Karush-Kuhn-Tucker 条件) *令 x^* 为优化问题 (7.2) 的正规极小值点，那么，存在向量 $\lambda \in \mathbb{R}^m$ 和向量 $\mu \in \mathbb{R}^p$ 且 $\mu \geq 0$ 使得*

$$\nabla f(x^*) + \lambda^{\mathsf{T}} \nabla h(x^*) + \mu^{\mathsf{T}} \nabla g(x^*) = 0, \tag{7.3}$$

$$\mu^{\mathsf{T}} g(x^*) = 0. \tag{7.4}$$

证明 首先，因为 $\mu \geq 0$ 且 $g(x^*) \leq 0$，(7.4) 等价于：μ 的一个分量非零仅当对应的约束被激活（即取到等号）. 这是一个互补松弛条件，即 $g(x^*)_i < 0$ 可得出 $\mu_i = 0$，以及 $\mu_i > 0$ 可得出 $g(x^*)_i = 0$.

设被激活的下标为 J. 因为 x^* 是约束集合上的一个极小点，它也是满足等式约束 $h(x) = 0, g_i(x) = 0, i \in J$ 的极小点. 因此，在新的等式约束问题中，x^* 的邻域中存在 Lagrange 乘子，满足一阶必要条件. 我们得出结论：一阶必要条件 (7.3) 成立，且若 $g_j(x^*) \neq 0$，则 $\mu_j = 0$.（于是也有 (7.4) 成立）

现在还需要证明 $\mu \geq 0$. 用反证法，假设 $\mu_k < 0$ 对某个 $k \in J$ 成立. 设 S 为其他所有被激活的约束在 x^* 处定义的曲面，$M = T_{x^*}(S)$. 因为 x^* 是正规的，存在 $y \in M$ 且 $\nabla g_k(x^*) y < 0$. 令 $x(t)$ 为一条在 S 内且经过 x^*（此处 $t = 0$）的曲线，且有 $\dot{x}(0) = y$. 则对于充分小的 $t \geq 0$，$x(t)$ 是可行的，由 (7.3) 以及 $y \in M$，

$$\begin{aligned} \left.\frac{\mathrm{d} f(x(t))}{\mathrm{d} t}\right|_{t=0} &= \nabla f(x^*) y \\ &= -\lambda^{\mathsf{T}} \nabla h(x^*) y - \mu^{\mathsf{T}} \nabla g(x^*) y \\ &= -\mu_k \nabla g_k(x^*) y < 0. \end{aligned}$$

这与 x^* 是极小点矛盾. □

注. 这一证明具有很强的几何直观，关键在于找一个可行的方向使得函数值下降. 非常需要注意的是，这一证明并不能用于否证 $\mu_k > 0$. 此时需要取 $y \in M$ 使得 $\nabla g_k(x^*)y > 0$. 然而此时对应的 $x(t)$ 不再可行，因为对充分小的 $t > 0$，$g_k(x(t)) > 0$，违背了约束的条件.

下面我们来看一个运用 KKT 条件的例子:

例 7.2 考虑问题

$$\begin{aligned} \min \quad & 2x_1^2 + 2x_1x_2 + x_2^2 - 10x_1 - 10x_2 \\ s.t. \quad & x_1^2 + x_2^2 \leq 5, \\ & 3x_1 + x_2 \leq 6. \end{aligned}$$

KKT 条件为（注意，一阶必要条件还需要加入问题中的约束条件）

$$\begin{aligned} 4x_1 + 2x_2 - 10 + 2\mu_1 x_1 + 3\mu_2 &= 0, \\ 2x_1 + 2x_2 - 10 + 2\mu_1 x_2 + \mu_2 &= 0, \\ \mu_1(x_1^2 + x_2^2 - 5) &= 0, \\ \mu_2(3x_1 + x_2 - 6) &= 0, \\ \mu_i \geq 0, \quad i &= 1, 2. \end{aligned}$$

为了求解此类问题，我们假设一些约束被激活，然后检查所得出的 *Lagrange* 乘子的符号正负. 在这个问题中，我们可以尝试假设有 *0*，*1*，*2* 个约束被激活.

假设第一个约束被激活，第二个约束没有被激活，得出等式

$$\begin{aligned} 4x_1 + 2x_2 - 10 + 2\mu_1 x_1 &= 0, \\ 2x_1 + 2x_2 - 10 + 2\mu_1 x_2 &= 0, \\ x_1^2 + x_2^2 &= 5. \end{aligned}$$

可得解 $x_1 = 1, x_2 = 2, \mu_1 = 1$.

由于 $3x_1 + x_2 = 5$，因此第二个约束也被满足了. 因此，因为 $\mu_1 > 0$，我们得出结论，这个解满足一阶必要条件.

§7.3 Lagrange 对偶

7.3.1 Lagrange 定理

现在，我们不再假设函数可微，我们考虑极值点的零阶必要条件，首先考虑只有等式约束的情形：

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & h(x) = 0, \\ & x \in \Omega. \end{aligned} \tag{7.5}$$

如果函数 f 是凸函数，m 维函数 h 是仿射的，并且集合 $\Omega \subset \mathbb{R}^n$ 是是凸的，那么这个规划问题是一个*凸规划问题*.

为了给这样的问题一个一阶必要条件，我们依然需要引入正规性条件. 此时正规性不再仅仅只对一个点，而是对仿射函数 h.

定义 7.4 (正规性条件) 一个仿射函数 h 关于集合 Ω 是**正规的**，指的是像集 $h(\Omega) = \{y : \exists x \in \Omega\, h(x) = y\}$ 包含 0 处的一个开球邻域. 也就是说，$h(\Omega)$ 包含一个形如 $\{y : \|y\| < \epsilon\}$（对某个 $\epsilon > 0$）的集合.

> **注.** 这个条件是一阶正规点定义的推广. 如果 h 在点 x^* 有连续的导数，那么一阶正规性条件意味着 $\nabla h(x^*)$ 是满秩的，并且由隐函数定理可知存在一个 $\epsilon > 0$ 使得对于任意满足 $\|y - h(x^*)\| < \epsilon$ 的 y，都有一个 x 使得 $h(x) = y$. 换言之，存在一个 $y^* = h(x^*)$ 周围的开球.

我们可以用 Lagrange 乘子来表述零阶必要条件：

定理 7.4 (零阶必要条件，等式约束情形) 假设 $\Omega \subset \mathbb{R}^n$ 是凸的，f 是 Ω 上的凸函数，h 是一个 Ω 上的 m 维仿射函数. 假设 h 是对于 Ω 正规的. 如果 x^* 是 (7.5) 的解，那么存在 $\lambda \in \mathbb{R}^m$ 使得 x^* 是以下 *Lagrange* 问题的解：

$$\begin{aligned} \min \quad & f(x) + \lambda^{\mathsf{T}} h(x) \\ \textit{s.t.} \quad & x \in \Omega. \end{aligned}$$

这一定理证明的关键在于引入原始函数. 对应于问题(7.5) 的**原始函数**是：

$$\omega(y) = \inf\{f(x) : h(x) = y, x \in \Omega\}, \quad y \in h(\Omega).$$

证明 (零阶必要条件的证明) 令 $f^* = f(x^*)$. 定义 $\mathbb{R}^m \times \mathbb{R}$ 内的集合 A 和 B 为:

$$A = \{(y,r) : r \geq \omega(y), y \in h(\Omega)\},$$
$$B = \{(y,r) : r \leq f^*, y = 0\}.$$

A 是 ω 的上图，B 是 f^* 向下延申并与原点对齐的垂线. A 和 B 都是凸集. 他们唯一的公共点是 $(0, f^*)$. 由超平面分离定理可知，存在一个超平面分离 A 和 B. 这个超平面可以被表示成一个在 $\mathbb{R}^m \times \mathbb{R}$ 内的形如 $(\lambda, s), \lambda \in \mathbb{R}^m$ 的非零向量，还有一个分离常数 c. 分离条件是

$$sr + \lambda^\mathsf{T} y \geq c, \quad \forall (y,r) \in A, \quad sr + \lambda^\mathsf{T} y \leq c, \quad \forall (y,r) \in B.$$

这一过程的示意图见图 7.3.

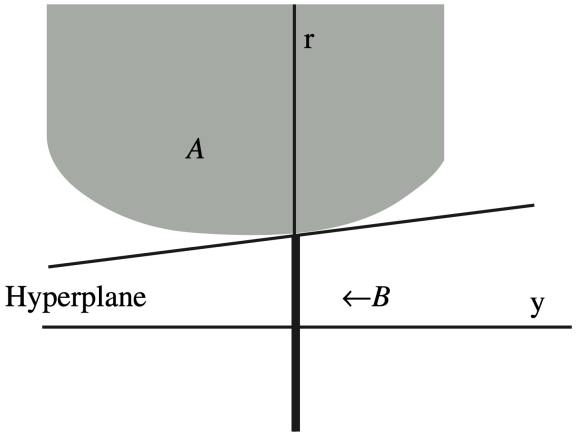


图 7.3: 证明示意图。

注意到 $s \geq 0$，否则取 $|r|$ 非常大的负数 r，点 $(r,0) \in B$ 违反第二个分离不等式. 几何上看，若 $s = 0$，超平面将垂直. 我们来证明 $s \neq 0$. 假设 $s = 0$，因为 s 和 λ 不能都是 0，$\lambda \neq 0$. 因为分离超平面必须包含点 $(f^*, 0)$，从第二个分离不等式得 $c = 0$. 由 h 的正规性，以 $0 \in h(\Omega)$ 为中心的某个球包含在 $h(\Omega)$ 中，任取 y 属于这个开球. 第一个分离不等式左侧为 $\lambda^\mathsf{T} y$，它对于某些 y 来说是负的. 这违背第一个分离不等式. 因此 $s \neq 0$，继而 $s > 0$.

不失一般性，可以假设 $s = 1$. 假设 $x \in \Omega$. 那么 $(h(x), f(x)) \in A$ 且 $(0, f(x^*)) \in B$. 因此，由分离不等式可知，我们有

$$f(x) + \lambda^\mathsf{T} h(x) \geq f(x^*) = f(x^*) + \lambda^\mathsf{T} h(x^*).$$

因此 x^* 是优化问题 (7.5) 解. □

我们再考虑只有不等式约束的模型

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) \leq 0, \\ & x \in \Omega. \end{aligned} \tag{7.6}$$

其中，g 是一个 p 维的函数.

然后我们引入正规性条件。对于不等式约束来说，正规性条件也被称为做 *Slater* 条件.

定义 7.5 (Slater 条件) *考虑优化问题 (7.6)，令*

$$D = \{z \in \mathbb{R}^p : \exists x \in \Omega\, g(x) \leq z\}.$$

正规性条件（Slater 条件）为：存在一个 $z' \in D$ 使得 $z' < 0$.

直观来说，Slater 条件指的是存在满足约束的内点.

类似地，我们可以用 Lagrange 乘子来表述零阶必要条件：

定理 7.5 (零阶必要条件，不等式情形) *假设 Ω 是一个 $\mathbb{R}^n$ 的凸子集，且 f 和 g 是凸函数. 假设优化问题 (7.6) 满足正规性条件，x^* 是该问题的解，那么存在一个向量 $\mu \in \mathbb{R}^p$ 满足 $\mu \geq 0$ 使得 x^* 是下述 Lagrange 问题的解：*

$$\begin{aligned} \min \quad & f(x^*) + \mu^{\mathsf{T}} g(x) \\ s.t. \quad & x \in \Omega. \end{aligned}$$

此外，$\mu^{\mathsf{T}} g(x^*) = 0$.

这一定理的证明类似于定理 7.4 的证明。首先还是引入原始函数。问题 (7.6) 对应的原始函数为：

$$\omega(z) = \inf\{f(x) : g(x) \leq z, x \in \Omega\}, z \in D.$$

证明 (证明概要) 令 $f^* = f(x^*)$. 在 $\mathbb{R}^p \times \mathbb{R}$ 内定义两个集合

$$\begin{aligned} A &= \{(z, r) : r \geq \omega(z), z \in D\}, \\ B &= \{(z, r) : r \leq f^*, z \leq 0\}. \end{aligned}$$

A 和 B 都是凸的. 证明依然是构造 A, B 的分离超平面，正规性条件保证了超平面不会是垂直的. 这个过程的示意见图图 7.4.

条件 $\mu^{\mathsf{T}} g(x^*) = 0$ 是互补松弛条件，这一讨论类似 KKT 条件. □

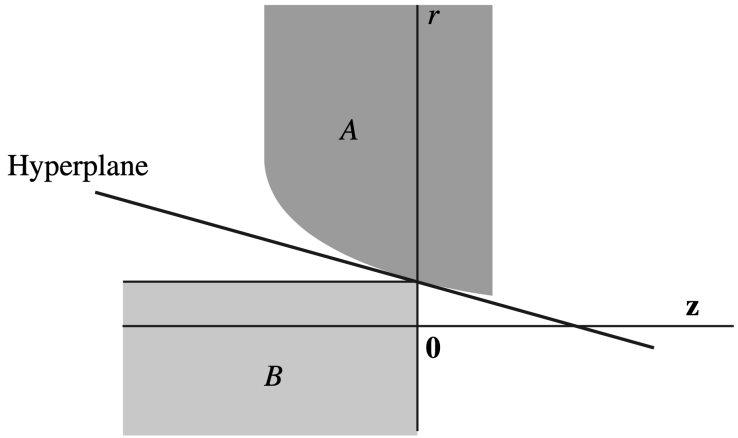


图 7.4: 证明示意图。

现在，我们考虑一般情形，

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & h(x) = 0, \\ & g(x) \leq 0, \\ & x \in \Omega. \end{aligned} \tag{7.7}$$

组合以上两个零阶必要条件，我们得到一般情形的 Lagrange 定理.

定理 7.6 (Lagrange，零阶必要条件，混合情形) *假设 $\Omega \subset \mathbb{R}^n$ 是凸集. f 和 g 是一维和 p 维的凸函数，h 是维数为 m 的仿射函数. 假设 h 满足对于 Ω 的正规性条件，且 g 在 (7.7) 的可行域上满足正规性条件. 假设 x^* 是问题 (7.7) 的解. 那么存在向量 $\lambda \in \mathbb{R}^m$ 和 $\mu \in \mathbb{R}^p$ 满足 $\mu \geq 0$ 使得 x^* 是以下 Lagrange 问题的解：*

$$\begin{aligned} \min \quad & f(x) + \lambda^{\mathsf{T}} h(x) + \mu^{\mathsf{T}} g(x) \\ s.t. \quad & x \in \Omega. \end{aligned}$$

此外，$\mu^{\mathsf{T}} g(x^*) = 0$.

[lhy: 举个例子]

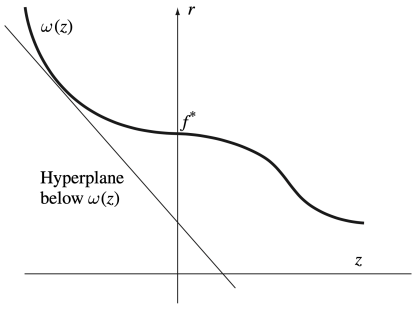


图 7.5: 纵截距的示意图.

7.3.2 弱对偶定理，强对偶定理

Lagrange 定理有非常强的几何直观，这一直观最终导致了优化中的对偶理论. 先考虑不等式约束的情形：

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) \leq 0, \\ & x \in \Omega. \end{aligned} \tag{7.8}$$

$\Omega \subset \mathbb{R}^n$ 是凸集，函数 f 和 g 定义在 Ω 上. 函数 g 是 p 维的.

回忆原始函数的定义：

$$\omega(z) = \inf\{f(x) : g(x) \leq z, x \in \Omega\}.$$

设 x^* 是 (7.8) 的解，$f^* = f(x^*)$，那么函数 $\omega(z)$ 与纵轴的交点是 f^*. 如果 (7.8) 没有解，那么 $f^* = \inf\{f(x) : g(x) \leq 0, x \in \Omega\}$ 就是纵轴与 $\omega(z)$ 的交点. 考虑在 $\omega(z)$ 以下的超平面，关注其纵截距（见图图 7.5），我们用它产生对偶原理.

为了刻画超平面以及其纵截距，我们引入对偶函数. 在 $\mathbb{R}^p_{\geq 0}$ 上定义**对偶函数**为：

$$\varphi(\mu) = \inf\{f(x) + \mu^{\mathsf{T}} g(x) : x \in \Omega\}.$$

定义其最大值为

$$\varphi^* = \sup\{\varphi(\mu), \mu \geq 0\}.$$

我们很容易可以证明以下定理：

定理 7.7 (弱对偶定理) $\varphi^* \leq f^*$.

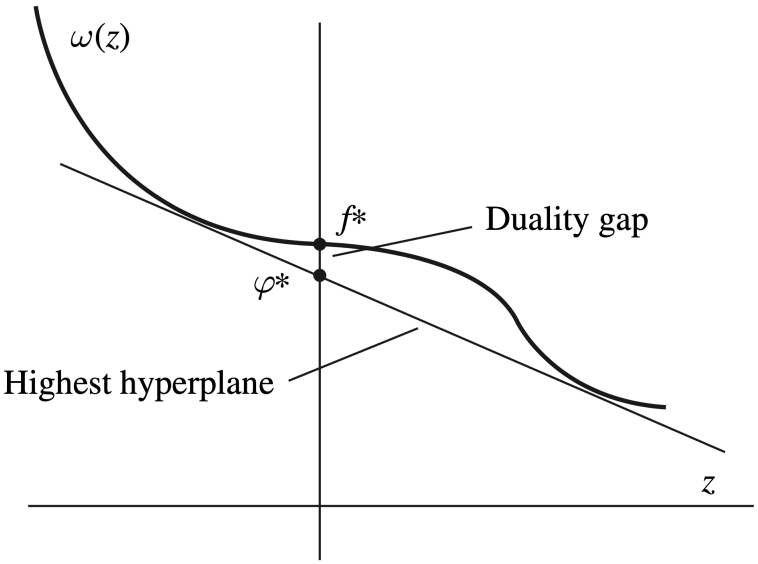


图 7.6: 对偶间距的示意图.

证明 对任意 $\mu \geq 0$ 我们有

$$\begin{aligned}\varphi(\mu) &= \inf\{f(x) + \mu^{\mathsf{T}} g(x) : x \in \Omega\} \\ &\leq \inf\{f(x) + \mu^{\mathsf{T}} g(x) : g(x) \leq 0, x \in \Omega\} \\ &\leq \inf\{f(x) : g(x) \leq 0, x \in \Omega\} = f^*.\end{aligned}$$

由此，$\varphi^* \leq f^*$. □

弱对偶定理也有非常几何的解释。考虑向量 $(\mu, 1) \in \mathbb{R}^p \times \mathbb{R}$，$\mu \geq 0$ 和一个常数 c. 关于 (z, r) 的方程 $(\mu, 1)^{\mathsf{T}}(z, r) = r + \mu^{\mathsf{T}} z = c$ 定义了一个 $\mathbb{R}^p \times \mathbb{R}$ 内的超平面. 不同的 c 得到不同的超平面，他们都是平行的. 对于给定的 $(\mu, 1)$（即平行的超平面），选取一个最低的超平面，使得它刚刚碰到了原始函数上图边界. 假设 x_1 是这个触点，有 $r = f(x_1)$ 和 $z = g(x_1)$. 那么 $c = f(x_1) + \mu^{\mathsf{T}} g(x_1) = \varphi(\mu)$. 注意到此时 $c = \varphi(\mu)$ 就是截距，这就是 $\varphi(\mu)$ 的几何含义.

另一方面，求截距 c（对偶函数值）的最大值 φ^*，就是求位于原始函数之下的超平面的最大截距. 因此至少有 $\varphi^* \leq f^*$，差 $f^* - \varphi^*$ 被称为对偶间距. 这就是弱对偶定理，图示参见图 7.6.

由此可以得到对偶性原理：位于 ω 之下的超平面的最大截距等于刚刚碰到 ω 的超平面的最小截距.

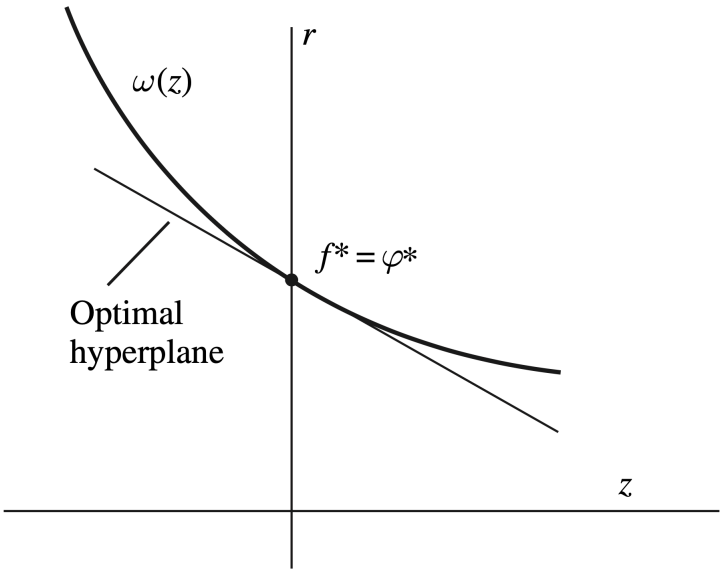


图 7.7: 强对偶定理的示意图.

如果原始函数 ω 是凸的，那么弱对偶定理可以被加强到强对偶定理，此时 φ^* 和 f^* 之间不再存在对偶间距，图 7.6 变成了图 7.7.

下面我们叙述并证明强对偶定理。我们直接考虑一般的优化问题.

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & h(x) = 0, \\ & g(x) \leq 0, \\ & x \in \Omega. \end{aligned} \tag{7.9}$$

其中，h 是 m 维仿射函数，g 是 p 维凸函数，$\Omega \subseteq \mathbb{R}^n$ 是凸集.

原始函数可以写作

$$\omega(y,z) = \inf\{f(x) : \exists x \in \Omega\, h(x) = y, g(x) \leq z\}.$$

对偶函数定义为:

$$\varphi(\lambda,\mu) = \inf\{f(x) + \lambda^{\mathsf{T}} h(x) + \mu^{\mathsf{T}} g(x) : x \in \Omega\}.$$

它的最大值记为

$$\varphi^* = \sup\{\varphi(\lambda,\mu) : \lambda \in \mathbb{R}^m, \mu \in \mathbb{R}^p, \mu \geq 0\}.$$

利用以上定义，我们可以表述强对偶定理如下:

	原料 1	原料 2	原料 3	售价（万元/吨）
清洁剂 A	0.25	0.50	0.25	12
清洁剂 B	0.50	0.50		15
存量（吨）	120	150	50	

表 7.1: 清洁剂原料价格存量表。

定理 7.8 (强对偶定理) *在问题 (7.9) 中，假设 h 是对于 Ω 正规的，在可行域内 g 满足正规性条件. 假设 x^* 是问题 (7.9) 的解，设 $f(x^*) = f^*$. 那么对每个 λ 和 $\mu \geq 0$ 都有*

$$\varphi(\lambda, \mu) \leq f^*.$$

另外，存在 $\lambda, \mu \geq 0$ 使得

$$\varphi(\lambda, \mu) = f^*.$$

因此 $\varphi^* = f^*$. 与此同时，λ, μ 是该问题的 Lagrange 乘子.

证明 由 Lagrange 零阶条件定理（定理 7.6）可知：

$$\begin{aligned} f^* &= \min\{f(x) + \lambda^{\mathsf{T}} h(x) + \mu^{\mathsf{T}} g(x) : x \in \Omega\} \\ &= \varphi(\lambda, \mu) \leq \varphi^* \leq f^*. \end{aligned}$$

因此，$\varphi^* = f^*$，并且取等号的 λ, μ 是 Lagrange 乘子. □

从对偶原理我们可以写出对偶规划的一般形式：

$$\begin{array}{cc|cc} \multicolumn{2}{c|}{\text{原始问题}} & \multicolumn{2}{c}{\text{对偶问题}} \\ \min & \omega(y,z) & \max & \varphi(\lambda,\mu) \\ \text{s.t.} & y = 0, & \text{s.t.} & \lambda \in \mathbb{R}^m, \\ & z \leq 0. & & \mu \geq 0. \end{array}$$

作为例子，下面我们给一个对偶规划的经济学解释.

例 7.3 (线性规划的经济学解释) *表 7.1 描述了公司甲用原料生产清洁剂的价格与存量表。*

甲用 3 种原料混合成 2 种清洁剂. 2 种清洁剂应该如何配制，使总价值最大?

设清洁剂 A 和 B 分别配制 x_1 和 x_2，我们可以把甲的目标写成一个规划问题：

$$\begin{aligned}
\max \quad & z = 12x_1 + 15x_2 \\
s.t. \quad & 0.25x_1 + 0.50x_2 \leq 120, \\
& 0.50x_1 + 0.50x_2 \leq 150, \\
& 0.25x_1 \leq 50, \\
& x_1 \geq 0, \\
& x_2 \geq 0.
\end{aligned}$$

现在有一个公司乙需要这 3 种原料，打算向甲购买，应付出多少钱？

乙向甲购买 3 种原料，出价分别为每吨 y_1, y_2, y_3 万元. 希望总价格尽量小，但不能低于甲用原料生产清洁剂所产生的价值，因此写出规划问题为：

$$\begin{aligned}
\min \quad & w = 120y_1 + 150y_2 + 50y_3 \\
s.t. \quad & 0.25y_1 + 0.50y_2 + 0.25y_3 \geq 12, \\
& 0.50y_1 + 0.50y_2 \geq 15, \\
& y_1 \geq 0, \\
& y_2 \geq 0, \\
& y_3 \geq 0.
\end{aligned}$$

注意到，以上两个规划问题恰好互为对偶问题.

§7.4 应用：支持向量机（SVM）

作为前面极值必要条件的一个具体应用，我们考虑一个经典的机器学习分类器：支持向量机（SVM）.

考虑二分类问题，输入 $x \in \mathbb{R}^n$，函数 f 输出一个 $\{-1, 1\}$ 中的值. 二分类问题的学习问题指的是给定训练集 $\{(x_i, y_i)\}_{i=1}^N$，找到 f 使得 $f(x_i) = y_i$. 假设训练集是线性可分的，例如，存在某个 $w \in \mathbb{R}^n$ 和 $b \in \mathbb{R}$ 使得

$$f(x) = \begin{cases} 1, & w^\mathsf{T} x + b > 0, \\ -1, & w^\mathsf{T} x + b < 0. \end{cases}$$

学习问题的首要目标是找到正确的以及最优的 w 和 b。本质上说，这就是一个找分离超平面的过程。那么，什么才叫最优呢？从几何视角来看，一个自然的想法是最大化分离距离，即训练集中所有点到分离超平面的距离和的最小值，见图 7.8.

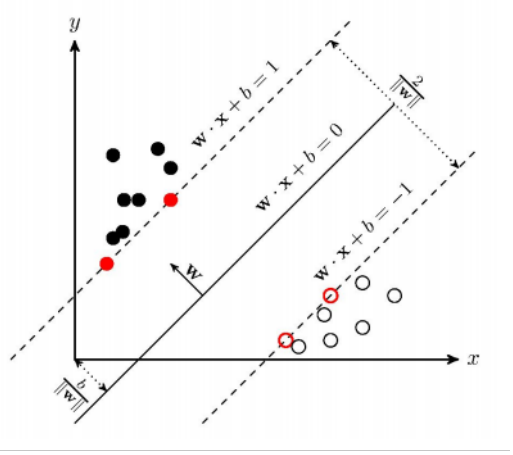


图 7.8: 分离距离示意图.

采样点 x_i 到分离超平面的归一化距离为

$$\gamma_i = y_i \left(\left(\frac{w}{\|w\|_2} \right)^{\mathsf{T}} x + \frac{b}{\|w\|_2} \right).$$

$\gamma = \min_i \gamma_i$ 是最小的归一化距离. 于是我们的任务变成了最大化 γ. 等价地，我们求解如下优化问题

$$\begin{aligned} \max_{w,b} \quad & \gamma \\ \text{s.t.} \quad & \gamma \leq \gamma_i, \quad i = 1, 2, \ldots, N. \end{aligned}$$

$\gamma \leq \gamma_i$ 等价于

$$y_i \left(\left(\frac{w}{\gamma \|w\|_2} \right)^{\mathsf{T}} x + \frac{b}{\gamma \|w\|_2} \right) \geq 1.$$

简洁起见，把 w 替换成 $\frac{w}{\gamma \|w\|_2}$，把 b 替换成 $\frac{b}{\gamma \|w\|_2}$，我们有

$$y_i (w^{\mathsf{T}} x + b) \geq 1.$$

那么最大化 $\gamma = \frac{1}{\|w\|_2}$ 等价于最小化 $\|w\|_2^2$.

我们得到以下凸规划问题:

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2} \|w\|_2^2 \\ \text{s.t.} \quad & y_i (w^{\mathsf{T}} x_i + b) \geq 1, \quad i = 1, 2, \cdots, N. \end{aligned}$$

如何解决这个问题? 利用上面的对偶理论，我们有如下步骤:

- 第一步，用 Lagrange 乘子法，转化成 Lagrange 问题（min-max）.
- 第二步，写出对偶问题（max-min），验证强对偶定理的正规性条件，于是只需要求解对偶规划.
- 第三步，写出 KKT 条件，将对偶规划解为一个二次规划（min），用优化算法求解二次规划.

第八章　不动点理论

考虑优化算法 A，它在函数 f 上的的收敛性如何？算法运行所产生的点列记为 $\{x_n\}$，它满足 $x_{n+1} = A(x_n)$. 如果关注序列 x_n 本身，那么这是一个数学分析的思路，通过寻找不同量之间的联系，来分析收敛性. 如果从算法 A 本身来看，这是一个算子法与泛函分析的思路，研究算法本身的性质，收敛性往往归结为吸收点的存在性. 后者是更加抽象且现代的思路。在本章中，我们将看到，从算子的角度来理解收敛性，最终问题就归结到了不动点理论。

不动点的定义是非常直接的，考虑一个集合 X 以及它到自身的映射 $f : X \to X$，元素 $a \in X$ 称为映射 $f : X \to X$ 的**不动点**，如果 $f(a) = a$. 本章将介绍三种不动点存在性定理。

§8.1　Banach 不动点定理

为了陈述不动点定理，我们需要引入一些数学概念。

定义 8.1 (度量与度量空间) 集合 X 上的**度量**（或**距离**）d 是映射

$$d : X \times X \to \mathbb{R}$$

满足条件

- 非负性：$d(x_1, x_2) \geq 0$，并且 $d(x_1, x_2) = 0 \iff x_1 = x_2$.
- 对称性：$d(x_1, x_2) = d(x_2, x_1)$.
- 三角不等式：$d(x_1, x_3) \leq d(x_1, x_2) + d(x_2, x_3)$.

其中 x_1, x_2, x_3 是 X 的任意元素.

此时，(X, d) 或 X 被称为**度量空间**.

下面给出一些度量的例子。

例 8.1 考虑实数集 $\mathbb{R}$，要成为度量空间，可以装备以下度量：

- 平凡的离散度量：$\forall x_1 \neq x_2\, d(x_1, x_2) \equiv 1, d(x, x) = 0.$
- $d(x_1, x_2) = |x_1 - x_2|.$

考虑向量空间 $\mathbb{R}^n$，要成为度量空间，可以装备以下度量：

- *Minkowski* 度量（L^p 度量）：$d(x_1, x_2) = (\sum_{i=1}^n |x_1^i - x_2^i|^p)^{1/p}$ $(p \geq 1).$
- *Manhattan* 度量（L^1 度量）：$d(x_1, x_2) = \sum_{i=1}^n |x_1^i - x_2^i|.$
- *Euclid* 度量（L^2 度量）：$d(x_1, x_2) = \sqrt{\sum_{i=1}^n |x_1^i - x_2^i|^2}.$
- *Chebyshev* 度量（L^∞ 度量）：$d(x_1, x_2) = \max_i |x_1^i - x_2^i| = \lim_{p\to\infty}(\sum_{i=1}^n |x_1^i - x_2^i|^p)^{1/p}.$

我们的目标是找到一类和实数集非常像的度量空间。实数集一个非常重要的性质是实数列收敛当且仅当它是 Cauchy 列。我们把这一性质抽象出来，就得到了如下定义：

例 8.2 度量空间 (X, d) 的点列 $\{x_n : n \in \mathbb{N}\}$ 称为 ***Cauchy* 列**，如果对于任何 $\epsilon > 0$，都可以找到序号 $N \in \mathbb{N}$，使得对于任何大于 N 的序号 $m, n \in \mathbb{N}$，$d(x_m, x_n) < \epsilon$ 成立.

度量空间 (X, d) 称为**完备的**，如果任意 *Cauchy* 列 $\{x_n : n \in \mathbb{N}\}$ 都收敛：$\exists a \in X, \lim_{n\to\infty} d(a, x_n) = 0.$

度量空间的任何收敛点列显然是 Cauchy 列，完备性本质上只是假设 Cauchy 收敛准则在该空间成立.

下面是一些完备度量空间的例子

例 8.3

- L^p 度量下下 $\mathbb{R}^n$ 是完备的.
- 使用度量 $d(x_1, x_2) = |x_1 - x_2|$，则 $X = \mathbb{R} \setminus \{0\}$ 不是完备度量空间. 考虑 $\{x_n = \frac{1}{n} : n \in \mathbb{N}\}$，它是 *Cauchy* 列，但该点列在 X 中没有极限（极限是 0）.
- $[0, 1]$ 到自身的连续函数空间 $C([0, 1])$ 在 L^∞ 度量下是完备的. 此时

$$d(f, g) = \sup_{x\in[0,1]} |f(x) - g(x)|,$$

完备性由一致收敛得到，函数空间是泛函分析中一个典型的研究对象.

下面我们给出与 Banach 不动点定理相关的概念：

定义 8.2 (压缩映射) 度量空间 (X,d) 到自身的映射 $f: X \to X$ 称为**压缩映射**，如果存在 $0 < q < 1$，使得不等式

$$d(f(x_1), f(x_2)) \leq q \cdot d(x_1, x_2)$$

对于 X 中的任何两个点 x_1, x_2 都成立.

用 $\delta - \epsilon$ 语言容易证明压缩映射一定是连续映射：

引理 8.1 压缩映射 $f: X \to X$ 是连续映射.

压缩映射一定有不动点，这就是 Banach 不动点定理：

定理 8.1 (Banach 不动点定理，压缩映像原理) 完备度量空间 (X,d) 到自身的压缩映射 $f: X \to X$ 具有唯一的不动点 a.

此外，对于任何点 $x_0 \in X$，迭代序列 $x_0, x_1 = f(x_0), \cdots, x_{n+1} = f(x_n), \cdots$ 收敛到 a. 收敛速度由以下估计给出：

$$d(a, x_n) \leq \frac{q^n}{1-q} d(x_1, x_0).$$

证明 首先证明存在性。$d(x_{n+1}, x_n) \leq q d(x_n, x_{n-1}) \leq \cdots \leq q^n d(x_1, x_0)$. 从而

$$\begin{aligned} d(x_{n+k}, x_n) &\leq d(x_n, x_{n+1}) + \cdots + d(x_{n+k-1}, x_{n+k}) \\ &\leq (q^n + \cdots + q^{n+k-1}) d(x_1, x_0) \leq \frac{q^n}{1-q} d(x_1, x_0). \end{aligned}$$

因此 $\{x_n\}$ 是 Cauchy 列，存在极限 $\lim_{n\to\infty} x_n = a \in X$. 结合压缩映射的连续性，有 $a = \lim_{n\to\infty} x_{n+1} = \lim_{n\to\infty} f(x_n) = f(\lim_{n\to\infty} x_n) = f(a)$.

然后证明唯一性。若 f 还有其他不动点 a_1, a_2，则 $0 \leq d(a_1, a_2) = d(f(a_1), f(a_2)) \leq q d(a_1, a_2)$. 而这当且仅当 $d(a_1, a_2) = 0$，即 $a_1 = a_2$ 时才可能成立. 最后证明收敛速度. 对 $d(x_{n+k}, x_n) \leq \frac{q^n}{1-q} d(x_1, x_0)$. 取 $k \to \infty$，得到 $d(a, x_n) \leq \frac{q^n}{1-q} d(x_1, x_0)$. □

例 8.4 (落在地面上的地图) 将一座公园的地图铺开在公园地面上，则地面上恰有唯一一点与地图上对应的点重合. 设公园可以用有界的面闭区域 Ω 表示. 设地图的压缩比是 $\lambda \in (0,1)$. 现在固定一个平面直角坐标系，把地图铺在区域 Ω 内，则从 Ω 内的点 x（公园中的地点）到地图上对应点 x' 的变换由下面的公式给出：

$$x' = f(x) := \lambda R x + b.$$

其中 R 和 b 分别为旋转和平移变换.

考虑 $\|\lambda R\| = \sup_{\|x\|=1} \|\lambda R x\| = \lambda < 1$, 由 *Banach* 不动点定理可知, 压缩映射 $f(x)$ 有唯一不动点 $a = f(a)$.

例 8.5 (梯度下降的收敛性) 我们优化目标是寻找二阶可微凸函数 $f(x), x \in \mathbb{R}^n$ 的最小值. 使用梯度下降方法: 每次往最小梯度方向移动. 假设对任意 $x \in \mathbb{R}^n$,

$$L \leq \lambda_{\min}(\nabla^2 f(x)) \leq \lambda_{\max}(\nabla^2 f(x)) \leq U.$$

其中 $\nabla^2 f(x)$ 是 f 的 *Hessian* 矩阵(二次导数), $U \geq L > 0$ 为给定的常数, $\lambda_{\min}(A), \lambda_{\max}(A)$ 表示矩阵 A 的最小、最大特征值.

我们要证明: 梯度下降能收敛到最小值点, 且具有指数收敛速度.

先看一下证明的思路, 我们要设法证明梯度下降算法是完备度量空间中的一个压缩映射。首先, 二阶可微凸函数的最小值点充分必要地满足 $\nabla f(x) = 0$. 其次, $\nabla f(x) = 0 \iff x \in \mathbb{R}^n$ 是梯度下降算子 $\mathcal{T}^{(\alpha)}$ 的不动点, 其中 $\mathcal{T}^{(\alpha)} : x \mapsto x - \alpha \nabla f(x)$, 这里 $\alpha \in \mathbb{R}_+$ 为步长. 最后, $\mathcal{T}^{(\alpha)}$ 是一个完备度量空间的压缩映射, 其压缩系数为 $q(\alpha) = 1 - L\alpha$. 因此梯度下降可以收敛至唯一的最小值点, 收敛速度可以由压缩系数估计.

为了使 $q(\alpha)$ 确实一个压缩系数, 我们需要 $\alpha < \min L^{-1}$. $\mathcal{T}^{(\alpha)}$ 的不动点恰好满足 $\nabla f(x) = 0$, 因此是最小值点. 我们只需要证明 $\mathcal{T}^{(\alpha)}$ 是压缩映射, 并给出压缩系数

由有限增量原理:

$$\left\| \mathcal{T}^{(\alpha)} x - \mathcal{T}^{(\alpha)} y \right\| \leq \sup_{z \in [x,y]} \left\| I - \alpha \nabla^2 f(z) \right\|_2 \cdot \|x - y\|_2 .$$

最后, 注意到 $\left\| I - \alpha \nabla^2 f(z) \right\|_2$ 等于 $I - \alpha \nabla^2 f(z)$ 特征值的最大模, 根据条件可知特征值的最大模 $\leq 1 - L\alpha$.

§8.2 Brouwer 不动点定理

下面我们考虑更一般的度量空间中的不动点定理, 为此我们需要引入连续映射的概念。

定义 8.3 设 X 和 Y 是度量空间 $(X, d_X), (Y, d_Y)$, 映射 $f : X \to Y$ 在点 $a \in X$ **连续**, 指的是

$$\forall \epsilon > 0 \exists \delta > 0 \forall x \in X (d_X(a, x) < \delta \Rightarrow d_Y(f(a), f(x)) < \epsilon).$$

如果它在每个点 $x \in X$ 连续, 称 f 为**连续映射**. X 到 Y 的连续映射的集合记为 $C(X, Y)$.

当度量空间为欧氏空间时，连续映射的定义与欧氏空间中连续映射的定义相同. 接下来，我们还需要几个集合的概念。

定义 8.4 (开集、闭集和紧集) *考虑度量空间 (X,d), $a \in X$ 的邻域 $B(a,\delta) := \{x \in X | d(a,x) < \delta\}$.*

- *集合 G 是*开集*G 指的是对于任何点 $x \in G$，满足 $B(x,\delta) \subset G$ 的邻域 $B(x,\delta)$ 存在.*
- *集合 F 是*闭集*，如果它的补集 $X \setminus F$ 是 (X,d) 中的开集.*
- *集合 K 是*紧集*，如果从 X 中任何覆盖 K 的开集族中可以选出有限个开集来覆盖 K.*

当度量空间为欧氏空间时，开集和紧集的定义与欧氏空间中的定义相同，紧集等价于有界闭集. 后一条在一般的度量空间不一定成立！

有了上面的准备，我们就可以叙述 Brouwer 不动点定理了：

定理 8.2 (Brouwer 不动点定理) *设 $M \subset \mathbb{R}^n$ 是一个非空紧凸集，而 $F : M \to M$ 是一个连续函数. 则存在 $x \in M$ 使得 $F(x) = x$ 成立.*

Brouwer 不动点定理可以通过该实际的例子来理解：将一张白纸平铺在桌面上，再将它揉成一团（不撕裂），放在原来白纸所在的地方，那么只要它不超出原来白纸平铺时的边界，那么白纸上一定有一点在水平方向上没有移动过. 这个断言依据 Brouwer 不动点定理在 $\mathbb{R}^2$ 的情况，因为把纸揉皱是一个连续的变换过程.

另一个例子：大商场等地方可以看到的平面地图，上面标有“您在此处”的红点. 如果标注足够精确，那么这个点就是把实际地形映射到地图的连续函数的不动点.

下面我们看一个 Brouwer 不动点定理的应用例子。首先引入矩阵**不可约**的概念：对于 n 阶方阵 A 而言，如果存在一个置换矩阵（通过交换单位阵的列获得）P 使得 $P^{\mathsf{T}}AP$ 为一个分块上三角阵，我们就称矩阵 A 是可约的，否则就称该矩阵是不可约的.

定理 8.3 (Perron-Frobenius 定理) *设 $A = (a_{ij})$ 为 $n \times n$ 不可约实矩阵，所有元素均非负，$a_{ij} \geq 0$，则下列结论成立.*

- *存在一个实特征值 r，其他特征值 λ 的模均不超过 r，即 $|\lambda| \leq r$.*
- *存在一个与 r 对应的特征向量，其所有元素恒正.*
- $\min_i \sum_j a_{ij} \leq r \leq \max_i \sum_j a_{ij}$.

证明 首先证明 A 存在一个正的特征值 $r > 0$. 考虑单纯形 $S := \{x \in \mathbb{R}^n | x \geq 0, \sum_i x_i = 1\}$. 则 $\forall x \in S$，有 $Ax \geq 0$.

断言 $Ax > 0$，若不然，A 存在某一列全 0（由 $x \geq 0$ 和 A 非负可得）. 此时可通过置换阵将该 0 列交换到第一列，则得到的矩阵为分块上三角，与不可约性矛盾.

可以在 S 上定义映射

$$T(x) = \frac{1}{\rho(x)} Ax,$$

其中 $\rho(x) > 0$ 使得 $T(x) \in S$.

显然 $T(x)$ 是 $S \to S$ 的连续映射. S 是一个有界凸闭集. 由 Brouwer 不动点定理，存在 $x_0 \in S, x_0 = T(x_0) = \frac{1}{\rho(x_0)} Ax_0$.

令 $r = \rho(x_0)$，则可得 r 为 A 的一个正的特征值.

我们接下来证明，与 r 对应的特征向量所有元素恒正. 由之前的证明，与 r 对应的特征向量 $x_0 \in S$，则 $x_0 \geq 0$. 我们证明 $x_0 > 0$.

考虑 $A = PBP^{-1}$，其中 P 是置换矩阵，则

$$PBP^{-1}x_0 = rx_0 \implies B(P^{-1}x_0) = r(P^{-1}x_0).$$

记 $\tilde{x}_0 = P^{-1}x_0$. 取 B 使得 $\tilde{x}_0 = (\xi, 0)^\mathsf{T}, \xi > 0$. 则

$$\begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \begin{pmatrix} \xi \\ 0 \end{pmatrix} = \begin{pmatrix} r\xi \\ 0 \end{pmatrix}.$$

此时 $B_{21}\xi = 0$，由 $\xi > 0$ 可得 $B_{21} = 0$. 这与不可约矛盾，因此 $x_0 > 0$.

然后我们证明：若 α 是 A 的任意特征值，有 $|\alpha| \leq r$. 设 $0 \leq B \leq A, By = \beta y$. 记 $y^\star = |y| = (|y_i|)_i$. 于是有

$$|\beta| y^\star = |\beta y| = |By| \leq By^\star \leq Ay^\star.$$

由 A^T 不可约，存在特征值 $r_1 > 0$ 和特征向量 $x_1 > 0, A^\mathsf{T}x_1 = r_1 x_1$. 因此有

$$|\beta| x_1^\mathsf{T} y^\star \leq x_1^\mathsf{T} A y^\star = r_1 x_1^\mathsf{T} y^\star.$$

由 $x_1^\mathsf{T} y^\star > 0$，则 $|\beta| \leq r_1$. 令 $B = A$ 可得 $|\alpha| \leq r_1$，特别地 $r \leq r_1$. 同样有 $r_1 \leq r$，故 $r = r_1$.

最后证明：

$$\min_i \sum_j a_{ij} \leq r \leq \max_i \sum_j a_{ij}.$$

以这样的方式获得 $\tilde{A}$：将 A 的每一行都扩增（不减小某个元素），使得每一行都达到 $\max_i \sum_j a_{ij}$. 此时 $\max_i \sum_j a_{ij}$ 成为 A 的一个正特征值，且特征向量 $\tilde{x}_0 = \frac{1}{n} \cdot \mathbf{1} \in S$. 由之前

的结论，假设 $0 \leq A \leq \tilde{A}$，可以得到 $\tilde{A}$ 的正特征值 $\tilde{r} \geq r$. 因此 $r \leq \max_i \sum_j a_{ij}$. 同理缩小 A 可得 $\min_i \sum_j a_{ij} \leq r$. □

Perron-Frobenius 定理在 Markov 链中的有非常重要应用。回忆 Markov 链的平稳分布：满足矩阵方程 $\pi = \pi P$ 和 $\sum_i \pi_i = 1$. 设该 Markov 链状态有限且对应的转移矩阵 P 是（非负实）不可约方阵. 由 Perron-Frobenius 定理，P 存在一个特征值 $1 = \min_i \sum_j a_{ij} \leq r \leq \max_i \sum_j a_{ij} = 1$，对应一个正特征向量 $x_0 \in S = \{x \in \mathbb{R}^n | x \geq 0, \sum_i x_i = 1\}$. 因此不可约有限状态 Markov 链必然存在平稳遍历分布.

§8.3 不动点的一般视角

第四部分

逻辑与博弈

第九章　动态博弈

本章我们讨论每个玩家需要操作多次的博弈，此时，博弈被称为**动态博弈**.

§9.1　输赢博弈

输赢博弈指的是玩家的收益只能取两个值（输或赢）的博弈. 赢博弈中，每个游戏状态只有一个玩家可以进行操作的情况研究最多. 这种情况通常称为**扩展式博弈**. 围棋、象棋、斗地主都是输赢博弈. 输赢博弈的分类见表 9.1.

二人	多人
输赢	输赢平
有限深	无穷深
完全信息	不完全信息
非合作	合作

表 9.1: 输赢博弈的分类.

例 9.1 斗地主是一个多人有限轮不完全信息合作输赢博弈.

我们在本部分主要关注最简单的一种博弈，即完全信息确定性回合制博弈，与之相关的概念如下：

- 局面：博弈的状态包括博弈本身的状态（棋盘状态、出牌情况等）和当前回合是哪个玩家.
- （无记忆）策略：从局面到行动空间的映射 $s_i : C \to \mathcal{A}$.
- 确定性：给定当前格局和所有玩家的行动，可以唯一确定下一回合的格局.

图 9.1: 斗地主.

- 完全信息：所有玩家都知道当前局面，都知道每个玩家的行动，并且这些是共同知识.

这样的博弈可以用博弈树表示出来，例如，井字棋的博弈树见图 9.2.

输赢博弈一个自然的问题是：玩家是否总可以获胜？这就涉及到必胜策略的概念：无论对手如何进行行动，玩家都可以取得胜利的策略. 必胜策略是一种解概念，即给定一个博弈，求解具有一定性质的玩家策略. 如果某个玩家具有必胜策略，那么我们就说这个博弈是被决定的. 什么博弈是被决定的？这一问题的答案由 Zermelo 定理给出.

定理 9.1 (Zermelo 定理, Von Neumann) 如果一个博弈是双人的、有限深的、确定的、完全信息的、输赢的，那么这个博弈是被决定的.

以上限定词缺一不可，缺少了任何一个都可能导致结论不成立.

证明 (证明一：逻辑证明) 设 W_i 表示“玩家 i 获胜”, $i = 1, 2$. 于是 $x \in W_1 \iff x \notin W_2$.

先手玩家有必胜策略当且仅当

$$\exists a_0 \forall b_0 \exists a_1 \forall b_1 \ldots \exists a_n \forall b_n : (a_0 b_0 \ldots a_n b_n) \in W_1.$$

后手玩家有必胜策略当且仅当

$$\forall a_0 \exists b_0 \forall a_1 \exists b_1 \ldots \forall a_n \exists b_n : (a_0 b_0 \ldots a_n b_n) \in W_2.$$

两个命题互为否定，因此二者恰有一个成立！ □

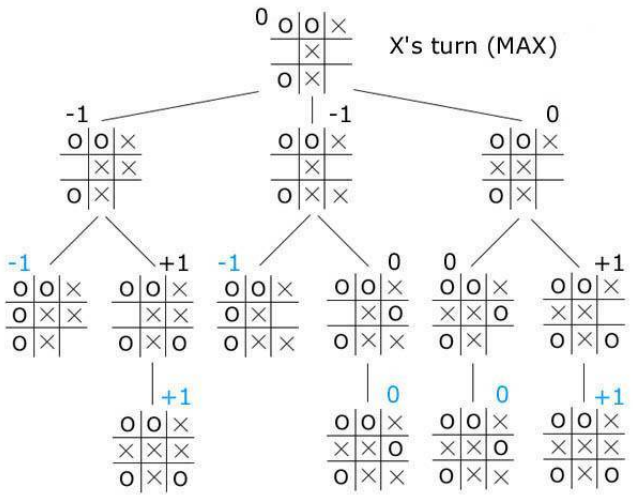


图 9.2: 井字棋的博弈树.

证明 (证明二：后向归纳法) 从博弈树的叶节点往根节点推理.

如果此节点是玩家 i 的回合，那么往后一轮的局面已经完全确定.

- 如果有一种走法使得玩家 i 必胜，那么玩家 i 选择这种走法即可.
- 否则，玩家 i 无论如何也不可能获胜.

当到达到根节点的时候，有一方有必胜策略，另一方必输.

这种证明方式被称为*后向归纳法*: 从最后一期开始往前推理，最终确定一个解概念.□

如果博弈的结局还有平局，我们有如下 Zermelo 定理:

定理 9.2 (有平局的 Zermelo 定理) 如果一个博弈是双人的、有限深的、确定的、完全信息的，博弈的结果有输赢平局三种，那么下面三条有且仅有一条成立:

- 第一个玩家有必胜策略.
- 第二个玩家有必胜策略.
- 双方都有不败策略，因此完全理性的玩家在博弈中必然平局.

证明留做练习。

尽管 Zermelo 定理的第二个证明构造出了必胜策略，但是后向归纳法的搜索空间过于庞大. 例如，充分大但有限的棋盘上，五子棋先手玩家存在不败策略，但是没有经过训练的人类或者简单的算法先手不一定会胜利。究其原因，人的思考以及机器搜索的过程

实际上是前向探索的过程. 如何进行搜索是取得胜利重要的因素. 其中一个非常震撼的例子就是 AlphaGo 的出现，机器战胜了人类围棋高手. 以下我们介绍 AlphaGo 的设计思路.

由 Zermelo 定理可知，围棋也存在必胜策略. 然而标准围棋棋盘大小为 19×19，状态空间量级为 10^{170}，过大的状态空间使得我们无法使用后向归纳法求解出必胜策略. DeepMind 的 AlphaGo、AlphaZero利用深度强化学习的方法取得了围棋博弈的出色表现. 以下我们探讨 AlphaGo 如何通过神经网络建模博弈的过程.

AlphaGo 算法包含*策略网络*, *价值网络*和 *Monte-Carlo 树搜索*（MCTS）.

- 策略网络和价值网络的输入为当前局面状态 $s \in C$.
- 策略网络的输出为下一步落子位置 $a \in \mathcal{A}$（361 维的 one-hot 向量）.
- 价值网络的输出为该局面的价值评估 (期望收益: 输 -1/赢 $+1$).
- MCTS 利用策略网络进行*扩展*，使用价值网络进行*评估*，利用 UCB 公式返回最优的搜索结果作为落子决策.

AlphaGo 使用模仿学习（人类专家对弈数据）、强化学习（自博弈，策略梯度）的方式训练策略网络，使用自我博弈过程中的数据监督训练价值网络.

[lhy: 仔细讨论这部分关于输赢博弈的讨论

- 什么叫完全理性的玩家?
- 表示完整的策略需要多少比特?
- 是否有高效的算法计算必胜策略?
- 如果博弈多方不是完全对抗的（即零和），那么是否还有必胜策略? 是否有其他合理的解概念?

]

我们再看一个有趣的例子：*对话博弈*。在对话博弈中，我们可以把对命题 ϕ 的辩论过程形式化为一个博弈. 博弈中有两个玩家，一个需要证明 ϕ 是真的，被称为*正方* P，一个需要说明 P 的论据是有矛盾的，它被称为*反方* O. 两个玩家可以对命题的某个部分发起*质疑*，或者对某个质疑作出*辩护*. 当正方成功辩护了所有的质疑，而反方已经无法再发出新的质疑，正方获胜；否则反方获胜.

我们以命题逻辑为例，考虑合取和蕴含. 感叹号 ! 表示陈述，问号 ? 表示询问. 当一个玩家 X（P 或者 O）说了一个合取式，另一个玩家 Y 如果想质疑，需要询问合取中的

陈述	质疑	辩护
$X!\varphi \wedge \psi$	$Y?L^{\wedge}$ 或 $Y?R^{\wedge}$	对应地，$X!\varphi$ 或 $X!\psi$

表 9.2: 对话博弈的规则：合取

某个部分（$L^{\wedge}$ 或 $R^{\wedge}$）. X 需要陈述该部分对应的命题作为辩护. 这一规则可以总结为表 9.2.

当一个玩家 X（P 或者 O）说了一个蕴含式，另一个玩家 Y 如果想质疑，需要陈述前提. X 则需要陈述结论作为辩护. 这一规则可以总结为表 9.3.

陈述	质疑	辩护
$X!\varphi \to \psi$	$Y!\varphi$	$X!\psi$

表 9.3: 对话博弈的规则：蕴含

我们考虑一个具体的例子 $(p \wedge q) \to p$. 用一个表格来表示辩论的过程，这个表格有两列，分别表示玩家 O 和 P. 每个玩家分别有三列，A 表示当前操作是第几步（两个玩家统一计数），B 表示当前操作质疑的是哪一步，中间一列表示当前的操作（陈述或者询问）.

O			P		
A		B	B		A

玩家 P 陈述要辩论的命题.

O				P	
				$!p \wedge q \to p$	0

玩家 O 质疑这一陈述.

O				P	
				$!p \wedge q \to p$	0
1	$!p \wedge q$	(0)			

现在又一次轮到了玩家 P，他可以选择为蕴含式辩护，或者质疑这个合取式. 我们假设他这次选择质疑合取式，那么他需要询问左边或者右边，假设他询问了左边：

O			P		
				$!p \wedge q \to p$	0
1	$!p \wedge q$	(0)			
			(1)	$?L^{\wedge}$	2

现在轮到了玩家 O. 他已经没有别的可以进行的操作了，只能对操作 2 进行辩护.

O			P		
				$!p \wedge q \to p$	0
1	$!p \wedge q$	(0)			
3	$!p$		(1)	$?L^{\wedge}$	2

现在轮到了玩家 P. 他已经没有别的可以进行的操作了，只能对操作 1 进行辩护. 因为玩家 O 已经陈述了 p，所以他可以用这个陈述来辩护.

O			P		
				$!p \wedge q \to p$	0
1	$!p \wedge q$	(0)		$!p$	4
3	$!p$		(1)	$?L^{\wedge}$	2

现在轮到了玩家 O. 他已经不能操作了（没有可以质疑的，也没有可以辩护的），所以玩家 P 获胜.

O			P		
				$!p \wedge q \to p$	0
1	$!p \wedge q$	(0)		$!p$	4
3	$!p$		(1)	$?L^{\wedge}$	2

我们还可以定义否定 $\neg p$ 相关的辩论规则，留作练习.

根据 Zermelo 定理，对话博弈一定有一人有必胜策略，我们还有更精细的定理：

定理 9.3 *考虑对命题 ϕ 的对话博弈，ϕ 是重言式当且仅当正方玩家 P 有必胜策略.*

证明只需要考虑对 ϕ 做归纳法. 实际上，我们如此规定对话博弈的规则，就是为了保证这一定理成立. 我们可以将一个命题真值的判定问题转化为玩家 P 博弈必胜策略的存在性问题，这对于一阶逻辑来说非常有用.

§9.2 随机博弈（Markov 博弈）

现在我们考虑无穷博弈中最简单的一种：**随机博弈**，也叫 **Markov 博弈**. 在随机博弈中，玩家的行动是随机的，但是玩家的行动空间是有限的. 为了简化问题，我们考虑两人随机博弈，即两个玩家轮流进行随机行动. 其相关概念如下：

- 有限局面：$C = \{s_1, s_2, \cdots, s_N\}$.

- 有限策略：$\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2$.
  - 每个局面具有自己的行动空间：$\mathcal{A}_p = \{\mathcal{A}_{p,1}, \mathcal{A}_{p,2}, \cdots, \mathcal{A}_{p,N}\}, (p = 1, 2)$.
  - 每个行动空间有限：$|\mathcal{A}_{p,k}| = n_{p,k}, (p = 1, 2; k = 1, 2, \cdots, N)$.
- 在局面 s_k，若玩家 1 选择第 i 个行动 $a_{1,k,i}(1 \leq i \leq n_{1,k})$，玩家 2 选择第 j 个行动 $a_{2,k,j}(1 \leq j \leq n_{2,k})$，则
  - 局面之间有转移概率：
    - 该博弈以概率 s_{ij}^k 停止.
    - 该博弈以概率 p_{ij}^{kl} 转移到状态 s_l.
  - 收益：玩家 1 收获 $Q(a_{1,k,i}, a_{2,k,j}; s_k)$，玩家 2 收获 $-Q(a_{1,k,i}, a_{2,k,j}; s_k)$. 假设 Q 有界.

随机博弈的过程如下。首先，博弈从某一个局面状态 s^0 开始，$s^0 \in C$. 在每个阶段 t，所有玩家同时选择自己的动作 a^t. 环境根据所有玩家的动作 $\boldsymbol{a}^t$ 和状态 s^t，给予每个玩家对应的收益 $q(\boldsymbol{a}^t, s^t)$，并转移到新的状态 $s^{t+1} \in C$.

假设在阶段 T，所有玩家可以观察到所有历史动作 $\{\boldsymbol{a}^t\}_{t \leq T}$. 和一般的动态博弈一样，我们可以定义每个玩家的策略 π——基于历史信息（状态、行动）到当前局面的行动的映射. 玩家在博弈的过程中，其实就是按照某个策略 π 进行行动的. 求解一个博弈也是求解最优策略 π_*. 下面我们定义什么是“最优”.

依赖历史信息的策略 π 一般很复杂. 由于收益只与当前局面、当前玩家的行动有关，我们可以缩小策略空间. 考虑第 p 个玩家 $(p = 1, 2)$，定义*平稳策略*为 N 个概率分布：$\vec{\pi}_p = (\pi_p^1, \pi_p^2, \cdots, \pi_p^N)$，分别对应 N 个状态；每个概率分布 $\pi_p^k = (\pi_{p,1}^k, \pi_{p,2}^k, \cdots, \pi_{p,n_{p,k}}^k)$，分别对应 $|n_{p,k}|$ 个行动. 使用平稳策略时，无论博弈的历史轨迹如何，玩家 p 在状态 s_k 采取行动 $a_{p,k,i}$ 的概率为 $\pi_{p,i}^k$.

假设两个玩家分别用平稳策略 π_1, π_2 进行博弈，则从局面 s^0 开始的随机博弈中，第 p 个玩家（$p = 1, 2$）的*远期收益*：

$$\Pi_p(\pi_1, \pi_2; s^0) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t (-1)^{p-1} Q(\pi_1(s^t), \pi_2(s^t); s^t)\right].$$

以上定义容易扩展为一般随机博弈（多人、非零和）. 随机博弈可以看做 MDP 的多人扩展 $(N, C, \mathcal{A}, \mathcal{P}, \mathcal{Q}, \gamma)$：

- N: 玩家的数量，$N = 1$ 退化为 MDP.

- C: 局面的集合.
- $\mathcal{A}$: 玩家的行动集合. $\mathcal{A} = \mathcal{A}^1 \times \cdots \mathcal{A}^N$. 设 $\mathcal{A}_i(s)$ 表示第 i 个玩家在状态 s 的行动空间.
- $\mathcal{P} : C \times \mathcal{A} \times C \to [0,1]$: 给定玩家的联合动作 $\boldsymbol{a} \in \mathcal{A}$, 局面从状态 $s \in C$ 转移到 $s' \in C$ 的概率 $P(s'|s, \boldsymbol{a})$.
- $\mathcal{Q} : C \times \mathcal{A} \to \mathbb{R}$: 在状态 s，当玩家的联合动作为 $\boldsymbol{a}$ 时，玩家 i 的奖励值 $Q_i(\boldsymbol{a}; s)$（有界）.
- $\gamma \in [0,1]$ 表示折扣系数，用于计算远期收益.

Markov 完美均衡（MPE）是一种解概念. 求解博弈的过程中，我们限制所有玩家使用平稳策略. 此时，面对对手，玩家的最优策略被称为 *Markov 最优反应*：对每个状态 s，给定其他玩家的平稳策略 π_{-i}，玩家 i 的行动 $a_i \in \mathcal{A}_i(s)$ 最大化它的远期收益 $\Pi_i(s; \pi_{-i})$:

$$\Pi_i(s; \pi_{-i}) = \mathbb{E}\left[Q_i(a_i, \pi_{-i}(s); s) + \gamma \sum_{s' \in C} P(s'|a_i, \pi_{-i}(s), s)\Pi_i(s'; \pi_{-i})\right].$$

MPE 被定义为：所有玩家的平稳策略组合，其中每个玩家的行动都是 Markov 最优反应。

我们可以类比 MDP 中求解最优价值的 Bellman 方程（动态规划）的形式. 当假设其他玩家都使用平稳策略，对每个状态 s，存在一个价值函数 $V_i(s; \pi_{-i})$ 取得玩家 i 从 s 出发的最高远期收益，满足:

$$\begin{aligned} V_i(s; \pi_{-i}) = \max_{a_i \in \mathcal{A}_i(s)} \mathbb{E}[&Q_i(a_i, \pi_{-i}(s); s) \\ &+ \gamma \sum_{s' \in C} P(s'|a_i, \pi_{-i}(s), s) V_i(s'; \pi_{-i})]. \end{aligned}$$

定理 9.4 *对于 N 个玩家、有限局面状态、有限动作空间的随机博弈，MPE 存在.*

下面我们介绍 Shapley 关于双人零和随机博弈情形的证明. 对于一般的情况，我们留做习题.

首先介绍一下*矩阵博弈*的概念. 假设 P 是一个 $m \times n$ 的矩阵，玩家 1 有 m 种动作 (动作集合 $\mathcal{A}_1$)，玩家 2 有 n 种动作 (动作集合 $\mathcal{A}_2$)，元素 P_{ij} 表示双方采取动作 (i, j) 时玩家 1 的收益，玩家 2 的收益为 $-P_{ij}$.

回忆在不动点课程中的 minimax 定理**??**:

$$\mathsf{val}(P) = \max_{s_1 \in \Delta(\mathcal{A}_1)} \min_{s_2 \in \Delta(\mathcal{A}_2)} s_1^\top P s_2 = \min_{s_2 \in \Delta(\mathcal{A}_2)} \max_{s_1 \in \Delta(\mathcal{A}_1)} s_1^\top P s_2.$$

s_i 是玩家 i 的混合策略，$\Delta(\mathcal{A}_i)$ 表示玩家 i 所有混合策略的集合。$\mathsf{val}(P)$ 为矩阵 P 定义的*矩阵博弈的值*. 在任意 Nash 均衡中，玩家 1 的期望收益即为 $\mathsf{val}(P)$. 下面我们将看到：双人零和随机博弈也存在值（即可定义均衡收益）.

首先证明一个引理：

引理 9.1 *对任意 $m \times n$ 的矩阵 B, C，成立：*

$$|\mathsf{val}(B) - \mathsf{val}(C)| \le \max_{i,j} |B_{ij} - C_{ij}|.$$

证明 设 (s_1, s_2) 为矩阵博弈 B 的 Nash 均衡，$(\bar{s}_1, \bar{s}_2)$ 为矩阵博弈 C 的 Nash 均衡. 于是由定义有：$s_1^\top B \bar{s}_2 \ge s_1^\top B s_2$，且 $\bar{s}_1^\top C \bar{s}_2 \ge s_1^\top C \bar{s}_2$，因此

$$s_1^\top B s_2 - \bar{s}_1^\top C \bar{s}_2 \le s_1^\top B \bar{s}_2 - s_1^\top C \bar{s}_2 \le \max_{i,j} |B_{ij} - C_{ij}|.$$

□

下面，我们将矩阵博弈的概念迁移到随机博弈. 在双人零和的语境下，我们去掉收益函数 Q 的下标 i. 定义*值迭代*为以下过程：

- 首先，我们选择一个任意的函数 $\alpha : C \to \mathbb{R}$，其中 C 是局面的状态空间，称 α 为*值函数*（value function）.
- 对任意 $s \in C$，定义矩阵 $R_s(\alpha)$ 为

  $$R_s(\alpha)(a_1, a_2) = Q(a_1, a_2; s) + \gamma \sum_{s' \in C} P(s' | a_1, a_2, s) \alpha(s').$$

  其中 $a_1 \in \mathcal{A}_1(s), a_2 \in \mathcal{A}_2(s)$.
- 值函数从 α_0 开始迭代，记 $\alpha_k(s) = \mathsf{val}(R_s(\alpha_{k-1}))$.

如何理解 $\alpha_k(s)$? 假设选取 $\alpha_0(s) \equiv 0$，则 $R_s(\alpha_0) = Q(a_1, a_2; s)$ 是从 s 出发，由 Q 定义的矩阵博弈. $\alpha_1(s) = \mathsf{val}(R_s(\alpha_0)) = \mathsf{val}(Q(\cdot, \cdot; s))$.

再看 $R_s(\alpha_1)(a_1, a_2) = Q(a_1, a_2; s) + \gamma \sum_{s' \in C} P(s' | a_1, a_2, s) \alpha_1(s')$. 我们可以假想有一个被截断的两阶段随机博弈：

- 玩家在第一阶段从状态 s 出发，行动 (a_1, a_2) 待定；
- 在第二阶段，对于每个可能的状态 $s' \in C$，玩家采用矩阵博弈 $R_{s'}(\alpha_0)$ 的 Nash 均衡的行动.

博弈在第二阶段终止，远期收益累积的折扣部分为 $\gamma \sum_{s' \in C} P(s'|a_1, a_2, s)\mathsf{val}(R_{s'}(\alpha_0))$. 从而，矩阵博弈 $R_s(\alpha_1)$ 的值也是这个两阶段随机博弈的值. 更一般地，$\alpha_k(s)$ 是一个被截断的 k 阶段随机博弈的值.

为方便，我们定义迭代算子 $(T\alpha)(s) = \mathsf{val}(R_s(\alpha))$.

$$\begin{aligned}
\left\|T\alpha - T\alpha'\right\|_\infty &= \max_{s \in C} \left|\mathsf{val}(R_s(\alpha)) - \mathsf{val}(R_s(\alpha'))\right| \\
&\leq \gamma \max_{s \in C} \max_{a_1, a_2} \left| \sum_{s' \in C} P(s'|a_1, a_2, s)(\alpha(s') - \alpha'(s'))\right| \\
&\leq \gamma \max_{s' \in C} \left|\alpha(s') - \alpha'(s')\right| \\
&= \gamma \left\|\alpha - \alpha'\right\|_\infty .
\end{aligned}$$

第一个不等式的成立使用了矩阵博弈值的不等式. 对于有折扣的博弈，$\gamma \in (0,1)$，因此 T 是一个压缩映射，由 Banach 不动点定理可知，$\alpha_k \to \alpha^*$ 满足 $T\alpha^* = \alpha^*$.

考虑任意一个从 s 出发的双人零和随机博弈，在前 k 局的博弈中，玩家 1 采用最优策略，后续局面可选择任意动作. 由之前的分析可知，前 k 局构成的截断随机博弈远期收益为 $\alpha_k(s)$. 而对于之后的博弈，玩家 1 损失的累积收益最差不超过 $\gamma^k/(1-\gamma) \cdot \sup|Q|$. 因此，当 $k \to \infty$ 时，玩家 1 的收益至少是 $\alpha^*(s)$. 注意，这个下界是无视玩家 2 的行动得出来的.

另一方面，玩家 2 也可以确保自己的收益至少是 $-\alpha^*(s)$. 由零和，因此均衡时玩家 1 的收益必定是 $\alpha^*(s)$. 因此双人零和随机博弈的均衡收益（值）为 $\alpha^*(s), s \in C$.

这样，我们就证明了 MPE 的存在性。

以上我们只说明了双人零和随机博弈的值存在，还没有指明最优策略如何取得. 类比矩阵博弈，我们有：

定理 9.5 *$R_s(\alpha^*)$ 定义的矩阵博弈的最优策略 (π_1, π_2) 是随机博弈的 MPE.*

证明 固定玩家 2 的一个任意策略 $\hat{\pi}_2$（不一定是平稳策略）. 首先考虑一个 k 阶段截断博弈，我们定义 $\alpha_0 = \alpha^*$，可理解为，原博弈前 k 步动作待定，后面使用策略取得 α^* 的远期收益.

在这个博弈中，玩家 1 可以无视玩家 2 的策略，确保至少取得 α^* 的远期收益（已证明），因此若采用 π_1 也能取得:

$$\mathbb{E}\left[\sum_{t=0}^{k-1} \gamma^t Q(\pi_1(s^t), \hat{\pi}_2(s^t); s^t) + \gamma^k \alpha^*(s^k) \middle| s^0 = s\right] \geq \alpha^*(s).$$

化简可得

$$\mathbb{E}\left[\sum_{t=0}^{k-1} \gamma^t Q(\pi_1(s^t), \hat{\pi}_2(s^t); s^t) \middle| s^0 = s\right] \geq \alpha^*(s) - \gamma^k \left\|\alpha^*\right\|_\infty .$$

因此

$$\Pi(\pi_1, \hat{\pi}_2; s) \geq \alpha^*(s) - \gamma^k \left\|\alpha^*\right\|_\infty - \frac{\gamma^k}{1-\gamma} \sup |Q|.$$

同样地，令 $k \to \infty$ 可得，上式 R.H.S. 趋于 $\alpha^*(s)$. 对于玩家 2 的 π_2 证明对称，因此 (s_1, s_2) 是一个 MPE. □

随机博弈在机器学习中对应着*多智能体强化学习*（MARL），正如 Markov 决策过程对应着（单智能体）强化学习. MARL 是多智能体系统（MAS）研究领域中的一个重要分支，它将强化学习技术、博弈论等应用到多智能体系统，使得多个智能体能在更高维且动态的真实场景中通过交互和决策完成更错综复杂的任务. 解 MDP 的过程是根据环境信息，优化决策，最大化远期收益，这是单智能体强化学习. 解随机博弈的过程，是在 MDP 的基础上引入多玩家，玩家有自己的效用函数，最大化自己的远期收益，这是多智能体强化学习.

第十章　静态博弈

本章我们讨论静态博弈的基本概念和分析方法，并以此为基础，讨论博弈中认知相关的问题。

§10.1　正则形式博弈

动态博弈通常被建模为*扩展形式博弈*. 与之相对的是*正则形式博弈*，即玩家只有一次行动的机会，所有玩家同时操作. 正则博弈通常要求信息是完全的. 这种博弈的过程与时间无关，属于*静态博弈*.

一个正则形式博弈有如下构成要素

- 玩家集合：I，我们总是假设这是一个有限集合.
- 玩家的行动集（纯策略集）：A_i，$i \in I$.
- 玩家的收益：$u_i : \prod_j A_j \to \mathbb{R}$.
- 完全信息：以上内容是所有玩家的共同知识.

所有人的策略拼在一起，即 $s = (s_i)_{i \in I}$，构成博弈的*策略组合*。有以下特殊的正则博弈：

- 当 A_i 有限，我们称之为*矩阵博弈*.
- 当 A_i 和 u_i 都是连续的，我们称之为*连续博弈*.
- 当 $\sum_i u_i = 0$，我们称之为*零和博弈*，当所有策略组合，收益和都是常数时，解概念的分析可以保持一致，我们也可以按零和处理.

如何定义正则博弈的均衡？首先要明确均衡的概念。假设所有人之间是不能交流的，每个人独立做决策. 因此玩家之间不能协调彼此的决策. 因为只能行动一次，所以所谓*均衡*，指的是没有人对自己的决策感到后悔的状态，没有人可以通过改变自己现在的策略来获得更多的收益. 因此我们有如下定义：

定义 10.1 (Nash 均衡)（纯策略）***Nash* 均衡指的是策略组合 s，满足

$$\forall i \in I \, \forall a_i \in A_i : u_i(s_i, s_{-i}) \geq u_i(a_i, s_{-i}).$$

我们也可以用不动点来理解 Nash 均衡。首先定义最优反应：给定对手的策略 s_{-i}，玩家 i 选择的最大化自己收益的策略 s_i. Nash 均衡的等价定义是每个人都达到了自己的最优反应，即最优反应的不动点.

例 10.1 (囚徒困境) 考虑一个经典的非合作博弈，囚徒困境. 一共有两个玩家，行玩家和列玩家. 玩家的第一个选择是保持沉默，第二个选择是认罪并检举对方. 它有如下收益矩阵：

$$\begin{pmatrix} -1,-1 & -10,0 \\ 0,-10 & -5,-5 \end{pmatrix}.$$

矩阵每一项第一个元素是行玩家的收益，第二个是列玩家的收益. 这个博弈有唯一的 *Nash* 均衡：每个人都认罪. 思考：打破 *Nash* 均衡的假设，有没有可能得到更好的结果?

然而，纯策略 Nash 均衡并不一定存在. 考虑如下的输赢（零和）博弈：猜硬币游戏. 行列玩家分别有一枚硬币，他们秘密地抛掷. 如果两个玩家的硬币上面相同，行玩家获胜；否则列玩家获胜. 收益矩阵为：

$$\begin{pmatrix} 1,0 & 0,1 \\ 0,1 & 1,0 \end{pmatrix}.$$

容易验证，这个博弈没有纯策略 Nash 均衡. 更一般地，二人正则输赢博弈中纯策略 Nash 均衡往往不存在. 我们有如下定理：

定理 10.1 设 $G = (I, \{A_i\}_{i\in I}, \{u_i\}_{i\in I})$ 是一个二人正则输赢博弈，其中 $I = \{1,2\}$. 那么，G 存在纯策略 *Nash* 均衡当且仅当其中一个玩家存在必胜策略.

对比动态博弈中的 Zermelo 定理，静态的二人完全信息输赢博弈已经不能够保证必胜策略的存在性。因此，静态输赢博弈的结局往往比动态输赢博弈更加不确定. 我们可以利用这一事实去理解生成对抗网络模型的不稳定性.

10.1.1 生成对抗网络

生成对抗网络（GAN）有两个子模型组成，一个被称为生成模型，一个被称为判别模型. 生成模型的任务是生成看似真实的数据，二判别模型的任务是识别给定的数据是真实的还是伪造的.

假设真实数据的分布为 F_{data}. 生成模型为 $G(x;\theta_g)$，参数为 θ_g，输入向量 x，输出数据向量 z. 当 x 服从分布 F_x，G 的输出会形成一个分布 F_g. 判别模型为 $D(z;\theta_d)$，参数为 θ_d，接受一个数据向量 z，输出一个 $[0,1]$ 中的实数，表示 z 来自分布 F_{data} 的概率. 我们假设 F_{data} 和 F_x 都是连续型分布，有密度函数 p_{data} 和 p_x. 我们再假设 D 和 G 都是连续的.

将 G 和 D 看成两个玩家，于是 GAN 可以被看成一个二人零和博弈，收益函数为:

$$V(G,D) = \mathbb{E}_{z\sim F_{data}}(\log D(z)) + \mathbb{E}_{x\sim F_x}(\log(1-D(G(x)))).$$

D 最大化 V，G 最小化 V.

从博弈论角度出发，一个基本的问题是 Nash 均衡是否存在? 假设 D 和 G 都可以任意选择连续函数. 我们将展示一种通用的方式求解连续博弈的 Nash 均衡. 注意到 $G(x)$ 形成了一个连续分布，密度记为 p_g. 首先证明密度函数存在性定理:

定理 10.2 *设 $X\sim\mathcal{U}(0,1)$. 对于任意密度函数 p，存在一个连续函数 F 使得 $F(X)$ 具有密度 p.*

证明 设 F_p 是 p 对应的分布函数，它是一个单调的连续函数. 取 $F(x) = \inf\{y\in\mathbb{R} : F_p(y)\geq x\}$ 即可. □

因此，G 的行动等价于选择 p_g.

给定 G 的选择 p_g，我们来求 D 的最优反应 D^*.

$$V(G,D) = \int (p_{data}(x)\log D(x) + p_g(x)\log(1-D(x)))\mathrm{d}x.$$

函数 $a\log x + b\log(1-x)$ 最大值在 $x = a/(a+b)$ 的时候取得. 因此，

$$D^*(x) = \frac{p_{data}(x)}{p_{data}(x)+p_g(x)}.$$

现在，给定最优反应 $D^* = p_{data}(x)/(p_{data}(x)+p_g(x))$，我们来求 G 的最优反应. 直观上，G 能做到的最好选择就是 $p_g = p_{data}$. 此时，$D^*(x) = 1/2$，因此对任意 G，$V(G,D^*) = -\log 4$. G 选任何策略都是一样的收益，因此这是一个 Nash 均衡. 我们证明了:

定理 10.3 (GAN 的 Nash 均衡存在性) *在 GAN 的博弈中，G 选择 p_{data}，D 选择 $1/2$ 是一个 Nash 均衡.*

我们刚刚的分析过于理想化，需要考虑一些问题。首先，神经网络的大小是有限的，因此 G 不能选择任何 p_g。因此，我们刚刚找到的 Nash 均衡可能不存在。其次，p_{data} 是一个未知的量，我们只有一些样本。因此，G 和 D 都需要一个算法来找到它们的最优策略。这就是训练 GAN 的过程。

我们接下来给出一种更符合实际的均衡概念。

局部 Nash 均衡 (G^*, D^*) 是指在 G^* 和 D^* 的一个邻域内 (G^*, D^*) 形成了一个 Nash 均衡。*稳定局部 Nash 均衡* (G^*, D^*) 是指 (G^*, D^*) 是一个局部 Nash 均衡，并且在 (G^*, D^*) 的一个邻域内，对任意 (G, D) 都有 $V(G, D^*) \geq V(G, D)$ 和 $V(G^*, D) \leq V(G, D)$。GAN 的训练实际上就是在寻找稳定局部 Nash 均衡的过程。

稳定局部 Nash 均衡表明了，即便对手的策略具有（很小的）不确定性，玩家的策略依然是最优反应。在训练过程中，这样的不确定性很可能出现，源自精度或者误差. 因此，稳定局部 Nash 均衡是一个更有可能被找到的解，不稳定局部 Nash 均衡则很容易偏离. 然而，我们刚刚在理想条件下找到的 Nash 均衡其实也是不稳定的. 实际上，GAN 的训练是一个非常不稳定的过程. 我们有如下结果：

定理 10.4 *设 GAN 博弈的收益函数 V 是解析的，$(0, 0)$ 是稳定局部 Nash 均衡，在 $(0, 0)$ 的一个邻域内，$V(G, D) = C + V^2 f(V) + D^2 g(D) + V^2 D^2 h(G, D)$，其中 f, g, h 都是解析函数，满足 $f(0), g(0) \geq 0$，C 是常数.*

V 要具备这种形式才可能有稳定局部 Nash 均衡. 然而一般的神经网络并不能具备这样的形式，所以很多情况下根本不存在稳定局部 Nash 均衡！

10.1.2 混合策略

我们已经看到，在相当普遍的情况下，纯策略 Nash 均衡并不存在. 所以我们需要允许玩家进行随机行动，这就是**混合策略**。混合策略就是建立在纯策略空间 S 上的一个概率分布. 混合策略空间记为 $\Delta(S)$. 当 S 有 n 个元素（有限），$\Delta(S)$ 可以被表示为标准的 n-单纯形：

$$\Delta(S) = \left\{ x \in \mathbb{R}^n : \sum_{i=1}^{n} x_i = 1, x_j \geq 0, \forall j \right\}.$$

那么，有了混合策略，玩家的决策思考过程是怎么样的？一个非常标准的回答是*期望效用理论*，它由 Von Neumann 和 Morgenstern 提出. 该理论认为，在面对不确定性时，人按照期望效用进行决策. 因此，我们需要计算玩家的期望效用。为此，引入混合*策略组合*：$\sigma = (\sigma_i)_{i \in I}$，其中 $\sigma_i \in \Delta(A_i)$. σ 是一个 $(A_i)_{i \in I}$ 上的概率分布，每一维相互独

立. 当所有玩家选定策略之后，玩家 i 的期望收益是：

$$u_i(\sigma) = \mathbb{E}_{a\sim\sigma} u_i(a).$$

定义 10.2 (Nash 均衡) 对于一个博弈 $G = (I, \{A_i\}_{i\in I}, \{u_i\}_{i\in I})$，混合策略 *Nash* 均衡 σ 满足对于任意玩家 i 和任意 $\sigma_i' \in \Delta(A_i)$，都有

$$u_i(\sigma_i, \sigma_{-i}) \geq u_i(\sigma_i', \sigma_{-i}).$$

Nash 著名的定理是：

定理 10.5 (Nash 均衡存在性定理) 对于任意有限正则形式博弈，都存在一个混合策略 *Nash* 均衡.

我们来看一个例子。

例 10.2 继续考虑猜硬币游戏，收益矩阵为

$$\begin{pmatrix} 1,0 & 0,1 \\ 0,1 & 1,0 \end{pmatrix}.$$

容易证明，唯一的均衡是两个玩家都选择 $(1/2, 1/2)$.

尽管在数学上，混合策略是导出了漂亮的结果，但是混合策略并不是一个非常合理的概念. 如何理解混合策略？我们将在后面通过似然、知识论等方式来解释混合策略.

§10.2 不完全信息博弈（Bayes 博弈）

即便是纯策略 Nash 均衡也可能是不合理的状态. 考虑如下的二人博弈：

$$\begin{pmatrix} 1,1 & 0,0 \\ 0,0 & 0,0 \end{pmatrix}.$$

显然，两个人玩家都选择第二策略达到了 Nash 均衡. 然而，当行玩家对列玩家的选择有任意小的不确定性时，他都更倾向于选择第一个策略. 因此，我们给出的这个 Nash 均衡实际上描述了一种不太可能出现的状态. 这促使我们提出了所谓的颤抖的手完美化：s 是一个纯策略 Nash 均衡，并且当对手玩家的策略有任何微小不确定性的时候，s 中的策略依然是最优反应.

“颤抖的手”给了我们一个例子说明不确定性会影响玩家的决策. 那么如何量化不确定性？经济学的解决方案是 *Bayes 解释*的概率论：每一个玩家对世界有一个先验的信念，信念在数学上被建模为对可能世界的概率分布.

利用这样的建模，我们可以给不完全信息博弈一个正式定义。一个不完全信息博弈有如下组成部分：

- 玩家集合：I.
- 行动空间：$A = (A_i)_{i\in I}$，A_i 表示玩家 A_i 的所有可能行动.
- 类型空间：$\Theta = (\Theta_i)_{i\in I}$，$\Theta_i$ 表示玩家 i 的所有可能类型.
- 收益函数：$u_i : A \times \Theta \to \mathbb{R}$，当所有人的行动和类型都确定的时候，玩家 i 能拿到的收益.

所有玩家的行动 $a = (a_i)_{i\in I}$ 形成了一个行动组合. 所有玩家的类型 $\theta = (\theta_i)_{i\in I}$ 形成了一个类型组合.

$P_i \in \Delta(\Theta_i)$ 是玩家 i 类型的概率分布. P_i 表示了其他玩家对玩家 i 类型的信念. 我们假设 P_i 是相互独立的，因此玩家 i 对其他玩家的信念是 $P_{-i} = \prod_{j\neq i} P_j$. 玩家 i 知道自己的类型.

在使用这一定义的时候需要非常小心，在一般情况下，玩家 i 对这个世界的信念应该包含：

- 其他玩家有谁；
- 自己和对手可能的行动；
- 自己的类型；
- 自己的收益函数；
- ……

然而，在上述标准的经济学模型中，我们做了如下严格的限制：

- 玩家、可能行动、可能类型、收益函数是所有人的共同知识，没有人对这些东西有不一样的信念.
- 玩家对世界的不确定性仅仅在于其他玩家的类型，而且所有人关于每个玩家类型的信念是一致且独立的.

- 自己的类型自己知道并且只有自己知道.

下面我们看一个例子

例 10.3 (合作者) 考虑一个二人博弈,称为"工作-偷懒"博弈。两个人的行动都是"工作"(W)或"偷懒"(S)。行玩家的类型集合是单点集，列玩家的类型是"勤奋"(D) 或"懒惰"(L)。收益矩阵为

$\theta_2 = D$,

	W	S
W	$3,3$	$-1,0$
S	$2,1$	$0,0$

$\theta_2 = L$,

	W	S
W	$1,1$	$-1,2$
S	$2,-1$	$0,0$

在具有不确定性的世界中，玩家的策略如何定义？玩家如何决策？因为玩家知道自己的类型，但在决策的时候不能知道其他人的类型，所以一个完整的（纯）策略应该是 $s_i : \Theta_i \to A_i$，即在给定自己的类型时，应该采取的行动.

关于收益，我们依然沿用期望效用理论. 当玩家 i 具有类型 θ_i，采取行动 a_i，对手的策略是 s_{-i} 时，i 的中期期望收益为:

$$\tilde{u}_i(a_i, \theta_i, s_{-i}) = \mathbb{E}_{\theta_{-i} \sim P_{-i}}[u_i(a_i, s_{-i}(\theta_{-i}), \theta_i, \theta_{-i})].$$

利用期望效用理论，我们很容易定义均衡的概念:

定义 10.3 (Bayesian Nash 均衡，BNE) $s = (s_i)_{i \in I}$ 被称为 Bayesian Nash 均衡，如果

$$\tilde{u}_i(s(\theta_i), \theta_i, s_{-i}) \geq \tilde{u}_i(a_i, \theta_i, s_{-i})$$

对任意 i, θ_i, a_i 都成立.

我们也可以考虑前期期望收益，此时玩家 i 并不知道自己是什么类型，因此他也要对自己的类型求期望:

$$\hat{u}_i(s_i, s_{-i}) = \mathbb{E}_{\theta \sim P}[u_i(s_i(\theta_i), s_{-i}(\theta_{-i}), \theta_i, \theta_{-i})].$$

根据前期期望收益，我们也可以定义 BNE 为:

$$\hat{u}_i(s_i, s_{-i}) \geq u_i(s_i', s_{-i})$$

对任意 i 和任意策略 s_i' 成立.

两个定义是等价的. 首先，前期期望收益是中期期望收益的加权平均. 然后，最大化前期期望收益等价于最大化平均中的每一项中期期望收益，也就是最大化中期收益. 所以这两者是等价的.

当所有的不确定性都消失的时候，我们得到的收益是真实的，被称为后期收益. 前期、中期、后期分别表明了信息的确定程度.

注. 自然，我们也可以定义混合策略的 *BNE*，此时策略 s_i 是一个 Θ_i 到 $\Delta(A_i)$ 的映射.

例 10.4 (猜硬币游戏的 BNE) 考虑猜硬币游戏：

	H	T
H	$1,-1$	$-1,1$
T	$-1,1$	$1,-1$

如果两个人都出 H 的时候收益有微小的扰动，我们就得到了一个 *Bayes* 博弈：

	H	T
H	$1+\epsilon\theta_1,-1+\epsilon\theta_2$	$-1,1$
T	$-1,1$	$1,-1$

其中 $\theta_i \sim \mathcal{U}[-1,1]$.

考虑策略：$s_i:[-1,1]\to\{H,T\}$ 满足

$$s_i(\theta_i)=\begin{cases}H, & \theta_i\in[0,1],\\ T, & \theta_i\in[-1,0).\end{cases}$$

容易证明，(s_1,s_2) 是一个 *BNE*.

注意到，在上面的例子中，策略 (s_1,s_2) 导致的结果实际上是，每个玩家以等概率选择 H 和 T. 当 $\epsilon\to 0$，这个博弈收益矩阵回到了原始博弈. BNE 形成的行动概率分布则趋于原始博弈的混合策略. 通过这样的办法，正则博弈的混合策略均衡被理解为：当不确定性趋于消失时候，BNE 形成的行动概率分布. 这不是偶然的，实际上所有的正则博弈的混合策略均衡都可以用一系列（纯策略的）BNE 纯化.

考虑一个正则博弈 (I,A,u). 给定一个扰动参数 $\epsilon>0$，定义类型为 $\theta=(\theta_i)_{i\in I}$，将收益扰动为：

$$\tilde{u}_i(s,\theta)=u_i(s)+\epsilon\theta_i,\quad \theta_i\in[-1,1].$$

假设 $\theta_i\sim F_i$，相互独立，F_i 是具有连续可微密度的分布. 如此就形成了一个扰动博弈. 当扰动参数 $\epsilon\to 0$ 时候，扰动博弈的 BNE 趋于正则博弈的混合策略均衡，这正是下面的 *Harsanyi* 纯化定理。

定理 10.6 (Harsanyi 纯化定理) 给定玩家集 I 和行动空间 A. 对于一般的收益函数 u 和连续分布族 $\{F_i\}_{i\in I}$，对任意完全信息正则博弈 (I,A,u) 的混合策略 *Nash* 均衡 σ，存在一列扰动博弈纯策略 *BNE* s_ϵ，当扰动参数 $\epsilon\to 0$，$s_\epsilon\to\sigma$.

混合策略均衡可以被看作不确定性趋于消失的时候的纯策略均衡. 这一定理的原始证明需要用到 Brouwer 不动点定理和隐函数定理，并且比较长，这里略去.

人们常说

> “Decision makers do not flip coins in the real world.”

然而，如果玩家对收益的信念有微小的不确定性的时候，他的行为就仿佛在抛硬币. 这是混合策略的似然解释（主观概率论）.

[lhy: 细化这一部分。混合策略的进一步讨论

- 我们之前说过，Bayes 博弈对于玩家信念的刻画是相当受限制的.
  - 当引入不确定性、知识、信念的概念的时候，几乎不可避免需要加入限制条件.
- 另一方面，概率论的 Bayes 学派解释在哲学上也有很多争议.
  - 一旦使用 Bayes 学派的概率论研究不确定性、知识、信念，这一问题也是不可避免的.
- 因而，我们可以考虑完全理性、完全耐心玩家在无穷轮重复的完全信息博弈中的决策行为.
  - 玩家做出行动 a 的极限频率就是行动 a 在混合策略中的概率.
- 这一角度并不涉及不确定性、信念等数学上模糊的概念，单纯讨论混合均衡达到的方式.

]

第五部分

认知逻辑

第十一章　模态逻辑基础

动机

- 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

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（Barbara XXX）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.

三段论

- 三段论可以进行各种形式的扩展.
- 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（mode）：无效、可能、必然、根据情况……

三段论

- 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（Barbara LXL）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（Barbara XLL）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?

三段论

- 类似的例子是，从物（De re）和从言（De dicto）：

我觉得有人作弊.

  - 这句话有两种解读方式.
  - 从物：$\exists x$（我觉得 : x 作弊）.
  - 从言：我觉得（$\exists x$: x 作弊）.

- 模态三段论的讨论并没有流行起来，因为它并没有非常干净漂亮的、符合直观的定义.

非经典逻辑

- 回忆：经典逻辑中的语义等值

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

- 然而，在哲学上，这两者是不一样的含义.
- 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.
- p_n：π 的小数位包含连续的 n 个 1.
  - $p_{100} \to p_{99}$ 是显然的，然而 $\neg p_{100} \vee p_{99}$ 并不直观！

非经典逻辑

- C. I. Lewis 严格蕴含（strict implication）：$p \dashv q$.
  - 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.
- 实质蕴含（materially imply）$p \to q \iff \neg p \vee q$
  - 允许有 p 假但是 q 真（false negatives).
- 换句话说：p 严格蕴含 q 当且仅当必然有 p 实质蕴含 q.

非经典逻辑

- 还有很多重言式也是不合乎常理的.
  - $p \to (q \to p)$.
  - $(p \to q) \vee (q \to r)$.
- Brouwer 直觉逻辑（intuitionistic logic），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.

非经典逻辑

- 非经典逻辑本质上说，都是尝试将*元语言*（meta language）中的概念拿到*对象语言*（objective language）中.
  - 例如，元语言：自然语言，对象语言：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

基本命题模态语言

- 我们只考虑最简单的情况，没有量词，只有一个*模态算子*（modality operator）.
- *基本模态语言*（basic modal logic language）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”.
- 更便捷的记号是使用 Backus-Naur 范式（BNF）：

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

基本命题模态语言

- 类似命题逻辑，我们有如下缩写：
  - $\phi \vee \psi \iff \neg(\neg\phi \wedge \neg\psi)$.
  - $\phi \to \psi \iff \neg\phi \vee \psi$.
  - $\bot \iff \neg\top$.

- 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：多视角下看同一个数学概念.

模态逻辑的例子

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

模态算子的解读：基础语义

- 我们可以把模态算子 $\Box$ 读成“必然”.
- $\Box\phi$：必然有 ϕ.
- $\Diamond\phi$：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.

基础语义：例子

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

模态算子的解读：认知逻辑

- 我们可以把模态算子 $\Box$ 读成“知道”，并写成 K（know）.
- 于是，K 表示某个特定的个体对世界的认知.
- $K\phi$（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.

模态算子的解读：可证性逻辑

- 我们可以把模态算子 $\Box$ 读成“可证明”.
- $\Box\phi$：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.
- Löb 定理：如果 $\mathbf{PA} \vdash \mathsf{Prov}(\ulcorner\phi\urcorner) \to \phi$，那么 $\mathbf{PA} \vdash \phi$.
  - 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.
- 在可证逻辑中，它对应 Löb 公式：$\Box(\Box\phi \to \phi) \to \Box\phi$.

命题模态逻辑：一般情形

- 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（modal similarity type）是一个元组 (O, ρ)，其中 O 是一个模态算子 ∇ 的非空集合，$\rho : O \to \mathbb{N}$ 表示每一个模态算子的元数.

- 多元模态语言的 BNF 为：

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

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

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

例子：时序逻辑

- 基础时序逻辑有两个一元模态算子：G 和 H.
  - $G\phi$：未来总会有 ϕ（always *G*oing to be）.
  - $H\phi$：过去总有 ϕ（always *H*as been）.
- 他们的对偶算子是：F 和 P.
  - $F\phi$：在未来某个时刻会有 ϕ（be true at some *F*uture time）.
  - $P\phi$：在过去某个时刻有 ϕ（was true at some *P*ast time）.
- 还可以加入一个“直到”（*U*ntil）算子 $U(\phi, \psi)$：直到 ϕ 发生都有 ψ.

例子：时序逻辑

- $P\phi \to GP\phi$：如果过去发生过 ϕ，那么 ϕ 在未来总会发生过.
- $F\phi \to FF\phi$：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.
  - 这一公式意味着时间是稠密的.
- McKinsey 公式 $GFp \to FGp$：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.

例子：认知逻辑

- 基本模态算子：K_a：个体 a 知道，B_a：个体 a 相信.
- 例：$K_aK_b\phi \leftrightarrow K_bK_a\phi$：我知道你知道 ϕ 当且仅当你知道我知道 ϕ.
- 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.

  - 注意，$C\phi$ 并不等价于 $K_a\phi$，$\forall a$.

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

* 例子：命题动态逻辑

- 命题动态逻辑（PDL），有无穷多个模态算子.
- 模态算子被记为 $[\pi]$，这里 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.

* 例子：命题动态逻辑

- 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.
- 基本程序：$a, b, \ldots$.
- 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.

* 例子：命题动态逻辑

- $\langle\pi^*\rangle\,\phi \leftrightarrow \phi \vee \langle\pi;\pi^*\rangle\,\phi$.

  - 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）：$[\pi^*](\phi \to [\pi]\phi) \to (\phi \to [\pi^*]\phi)$.
  - 思考：这个公式的含义是什么?

- 如何用模态算子表示 `if` ϕ `then` a `else` b?
  - $(\phi?;a) \cup (\neg\phi?;b)$.

§11.3 Kripke 语义与框架语义

模态逻辑的模型论

- 一个逻辑框架是一个三元组：(语言，模型，语义) $(\mathbf{L}, C, \vDash)$.
- 命题逻辑的例子：
  - 语言：命题公式的集合 $\{\top, \bot, p, p \vee q, \dots\}$.
  - 模型：常值和命题字母的真假：$\top : \top$，$p : \top$，$q : \bot$，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.

Kripke 模型和框架

- 一个 *Kripke 模型*（关系模型，Kripke model）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（frame）是一个没有节点标记的模型.

可能世界语义

- 我们将节点解读为可能世界（possible world）.
- 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.
- 形式化：$\Box\phi$ 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. 一个世界的意义取决于它与其他世界的联系.

状态语义

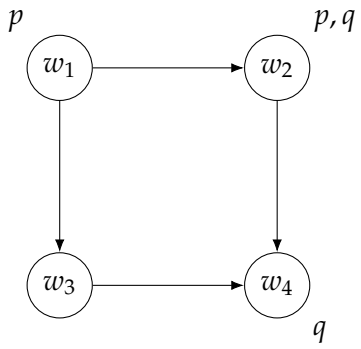
- 我们将节点解读为状态（state）.
- 于是边就被解读为状态的转移.
- $\Box\phi$ 在当前状态为真当且仅当 ϕ 在所有可能转移到的状态上为真.
- PDL 可以用以上语义来理解.
- 在哲学中，状态往往是不完全可观测的，此时模态逻辑可以被理解为不完全信息中可以确定的性质.

对象语义

- 我们将节点解读为对象（object）.
- 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.

Kripke 模型：基本情形

- 考虑 $\mathbf{ML}(\mathbf{P}, \Box)$.
- 一个 *Kripke* 框架（Kripke frame）指的是元组 $\mathcal{F} = (W, R)$，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）.



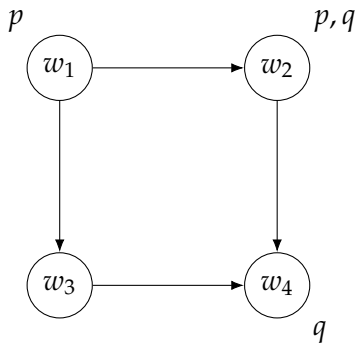
- 一个 *Kripke* 模型（Kripke model）$\mathcal{M}$ 指的是元组 $(\mathcal{F}, V)$，其中 $\mathcal{F}$ 是框架，$V : \mathbf{P} \to 2^W$ 是赋值函数（valuation function）.
- 一个 *Kripke* 点模型（pointed Kripke model）$(\mathcal{M}, w)$ 是 Kripke 模型 $\mathcal{M}$ 加上一个指定的点 $w \in W$.

Kripke 模型：基本情形
Kripke 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，$V : \mathbf{P} \to 2^W$ 是赋值函数（valuation function）.
- 一个 *Kripke* 点模型（pointed Kripke model）$(\mathcal{M}, w)$ 是 Kripke 模型 $\mathcal{M}$ 加上一个指定的点 $w \in W$.

Kripke 语义：基本情形

- 考虑 $\mathbf{ML}(\mathbf{P}, \Box)$.
- 符号 $\mathcal{M}, w \vDash \phi$ 表示 ϕ 在点模型 $\mathcal{M}, w$ 是可满足的（satisfiable）.
- 这一个概念可以递归定义如下
  - $\mathcal{M}, w \vDash \top \iff$ 总是.



  - $\mathcal{M}, w \vDash p \iff p \in V(w)$.
  - $\mathcal{M}, w \vDash (\phi \wedge \psi) \iff \mathcal{M}, w \vDash \phi$ 且 $\mathcal{M}, w \vDash \phi$.
  - $\mathcal{M}, w \vDash \neg\phi \iff \mathcal{M}, w \nvDash \phi$.
  - $\mathcal{M}, w \vDash \Box\phi \iff$ 对所有 v，如果 wRv，那么 $\mathcal{M}, v \vDash \phi$.

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

Kripke 语义：基本情形

- 对哪些 i 来说，$\mathcal{M}, w_i \vDash \Box(p \to \Diamond q)$?

Kripke 语义：一般情形

- 考虑 $\mathbf{ML}(\mathbf{P}, (O, \rho))$.
- 符号 $\mathcal{M}, w \vDash \phi$ 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
  - $\mathcal{M}, w \vDash \top \iff$ 总是.
  - $\mathcal{M}, w \vDash p \iff p \in V(w)$.
  - $\mathcal{M}, w \vDash (\phi \wedge \psi) \iff \mathcal{M}, w \vDash \phi$ 且 $\mathcal{M}, w \vDash \psi$.
  - $\mathcal{M}, w \vDash \neg\phi \iff \mathcal{M}, w \nvDash \phi$.
  - $\mathcal{M}, w \vDash \nabla(\phi_1, \ldots, \phi_{\rho(\nabla)}) \iff$ 对任意 $w_1, w_2, \ldots w_{\rho(\nabla)}$，如果 $R(w, w_1, \ldots, w_{\rho(\nabla)})$，那么存在 w_i 使得 $\mathcal{M}, w_i \vDash \phi_1$.

- 思考：为什么要这么定义 ∇ 的语义?

Kripke 语义：一般情形

- 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（unary）.
- 此时，关系 wRv 可以记为 $w \to_a v$.
- 模态算子一般写作 $\Box_a$.

* 模型验证

- 我们考虑如下两个模型验证（model checking）问题：
  - 局部模型验证：测试 $\mathcal{M}, w \vDash \varphi$ 是否成立；
  - 全局模型验证：计算集合 $\{w \in W_{\mathcal{M}} : \mathcal{M}, w \vDash \varphi\}$.
- 设 $l_R(X) = \{w \in W_{\mathcal{M}} : \forall v : w \to_{\mathcal{M}} v \implies v \in X\}$，我们可以递归定义 $\mathcal{M}$ 中公式的扩张（extension）：

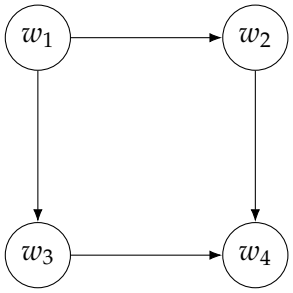
$$\begin{aligned} [\![\top]\!]^{\mathcal{M}} &= W_{\mathcal{M}}, & [\![p]\!]^{\mathcal{M}} &= \{w : p \in V(w)\}, \\ [\![\neg\varphi]\!]^{\mathcal{M}} &= W_{\mathcal{M}} \setminus [\![\varphi]\!]^{\mathcal{M}}, & [\![(\varphi \wedge \psi)]\!]^{\mathcal{M}} &= [\![\varphi]\!]^{\mathcal{M}} \cap [\![\psi]\!]^{\mathcal{M}}, \\ [\![\Box\varphi]\!]^{\mathcal{M}} &= l_R\left([\![\varphi]\!]^{\mathcal{M}}\right). \end{aligned}$$

- 全局模型验证的一个算法：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.
- 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！

模态公式的真值

- 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.
- ϕ 在点模型 $\mathcal{M}, w$ 可满足（satisfiable）指的是 $\mathcal{M}, w \vDash \phi$.
- ϕ 在模型 $\mathcal{M}$ 有效（valid），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 $\mathcal{F}, w$ 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 $\mathcal{F} \vDash \phi$ 指的是 $\mathcal{M} \vDash \phi$ 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 $\vDash_K \phi$ 指的是 $\mathcal{F} \vDash \phi$ 对所有 $\mathcal{F} \in K$ 成立.

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



模态公式的真值

- 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.
- 我们主要讨论高亮的两个部分.

框架语义：例子

- 是否成立 $\mathcal{F} \vDash \Box(p \to \Diamond p)$?

例子：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：$\phi : p \to \Diamond p$.

例子：基本模态逻辑

- 如果一个东西是真的，那么他也是可能的：$\phi : p \to \Diamond p$.
- 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，$\mathcal{M}, v \vDash \phi$ 以及 $\mathcal{F} \vDash \phi$.

例子：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.

例子：时序逻辑

- 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：

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

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

例子：时序逻辑

- 我们再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：$\forall x, y, z(xRy \wedge yRz \to xRz)$.
  - 完全：$\forall x, y(xRy \vee yRx \vee x = y)$.
- 设 $\mathcal{F}$ 是时间框架，是否有：$\mathcal{F} \vDash Pp \to GPp$ 以及 $\mathcal{F} \vDash Fp \to HFp$?

§11.4 模态可定义性

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.
  - 具体的模态公式.
- 这两种方式之间有什么联系?

例子

- $\mathcal{M}, w \vDash \Diamond \top$.
  - 存在一个 v，$w \to v$ 并且 $\mathcal{M}, v \vDash \top$，也就是 w 有一个后继.
- $\mathcal{F} \vDash \Diamond \top$.
  - 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.
- $\mathcal{F} \vDash p \to \Diamond p$.
  - 任意赋值 V 和任意点 w，都有 $\mathcal{M}, w \vDash p \to \Diamond p$.
  - 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.

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.
- 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 $\overline{K}$ 就可以由 $\neg \phi$ 定义.

- 然而，如果 K 由无穷个公式定义，它的补 $\overline{K}$ 不一定可以用无穷个公式定义！
  - 形式上来说，$\overline{K}$ 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.

框架类可定义性：更多例子

- $\Box p \to \Diamond p$.
  - 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.
- $\Box p \to \Box\Box p$.
  - 定义了传递的框架类.
- $\Diamond p \to \Diamond\Diamond p$.
  - 定义了稠密的框架类，即如果 $x \to y$，那么存在 z 满足 $x \to z$ 且 $z \to y$.
- 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！

* 模态可定义与一阶可定义

- 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：$\forall x,y,z(xRy \wedge yRz \to xRz)$.
  - 稠密：$\forall x,y(xRy \to \exists z(xRz \wedge zRy))$.
- 将一阶公式中的变元 $x,y,\ldots$ 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的模型类/框架类.
- 思考：一阶可定义和模态可定义是什么样的关系?

* 模态可定义性：一般结果

- 更一般地，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.

第十二章　认知逻辑与共同知识

§12.1　“泥泞的孩童”谜题

“泥泞的孩童”

- 有 n 个孩子在玩泥巴，他们互相泼泥巴.
- 母亲告诉孩子们，如果他们脸上沾上了泥巴，会受到严厉的惩罚.
- 孩子们不能看到自己的脸，但是可以看到其他所有人的脸.
- 所有孩子都希望保持自己的脸干净，但是弄脏别人的脸.

“泥泞的孩童”

- 此时，孩子的父亲出现了，于是，孩子们停止泼泥巴.
- 孩子们互相不说话.
- 父亲看到了 k（$k \geq 1$）个人脸上有泥巴，于是宣布：

“你们至少有一个人脸上沾了泥巴.”

“泥泞的孩童”

- 之后，父亲会公开地问若干轮如下问题：

“你们知道自己脸上有泥巴了吗?”

- 孩子们回答“知道”或者“不知道”.
- 假设孩子们观察力敏锐、聪慧且诚实，并且每一轮他们都同时回答.

- 接下来会发生什么?

“泥泞的孩童”：谜底

- 假设有 k 个孩子脸上有泥巴.
- 谜底：在前 $k-1$ 轮中，所有孩子都会说“不知道”，在第 k 轮中，所有脸上有泥巴的孩子都会说“知道”.
- “证明”：对 k 归纳.

“泥泞的孩童”：谜底

- 当 $k=1$ 时，脸上沾满泥巴的孩子看到其他人都没有泥巴.
- 既然他知道至少有一个孩子的脸上有泥巴，他就能推出那个人肯定是他自己.

“泥泞的孩童”：谜底

- 现在假设 $k=2$，脸上沾满泥巴的孩子是 a 和 b.
- 一开始，因为他们分别看到了对方的脸上有泥巴，所以他们每个人都回答“不知道”.
- 但是，当 b 回答“不知道”时，a 意识到他自己肯定是脸上有泥巴的那个孩子，否则 b 就会在第一轮中知道泥巴在他的脸上，并回答“知道”. 因此，a 在第二轮回答“知道”.
- b 也会通过同样的推理得出相同的结论.

“泥泞的孩童”：谜底

- 现在假设 $k=3$，脸上沾满泥巴的孩子分别是 a，b 和 c.
- 孩子 a 的论证如下.
  - 假设我没有泥巴落在脸上.
  - 根据 $k=2$ 的情况，b 和 c 在第二轮都会回答“是”.
  - 他们没有这样做，我意识到假设是错误的，我的脸上也有泥巴.
  - 因此在第三轮我会回答“知道”.
- b 和 c 的论证也是类似的.

- $k=3$ 的论证具有一般性，所以归纳假设成立.

“蓝眼睛红眼睛”谜题

- “泥泞的孩童”还有其他流行的陈述方式，比如“蓝眼睛红眼睛”.
- 一个岛上有 100 个人，其中有 5 个红眼睛，95 个蓝眼睛.
- 这个岛有三个奇怪的宗教规则.
  1. 他们不能照镜子，不能看自己眼睛的颜色.
  2. 他们不能告诉别人对方的眼睛是什么颜色.
  3. 一旦有人知道了自己的眼睛是红色，他就必须在当天夜里自杀.
- 岛民是不知道具体有几个红眼睛.

“蓝眼睛红眼睛”谜题

- 某天，有个旅行者到了这个岛上.
- 由于不知道这里的规矩，所以他在和全岛人一起狂欢的时候，一不留神说了一句话：

“你们这里有红眼睛的人.”

- 假设这个岛上的人足够聪明，每个人都可以做出缜密的逻辑推理.
- 请问这个岛上将会发生什么?

为什么会这样?

- 如果 $k>1$，那么所有人都知道 p：“至少有一个人脸上有泥巴”.
- 那么父亲说这句话的意义是什么?
- 如果父亲没有说 p，那么会发生什么?
- 无论父亲问多少轮，所有孩子都只会回答“不知道”!（为什么）
- 因此，父亲公开说了 p，这是谜题的关键.

$k=2$ 的情况分析

- 假设 $k=2$，脸上沾满泥巴的孩子是 a 和 b.

- 在父亲宣布 p 之前，a 和 b 都知道 p.
- 然而，他们并不知道对方知道 p. a 可能会有两种想法：
  - 我的脸上有泥巴，所以 b 知道 p.
  - 我的脸上没有泥巴，b 是唯一一个有泥巴的，所以 b 不知道 p.
- 当父亲宣布 p 之后，a 知道了 b 知道 p.
- 当第一轮 b 回答“不知道”之后，a 可以用“b 知道 p”这一知识推出自己脸上有泥巴.

$k = 3$ 的情况分析

- 假设 $k = 3$，脸上沾满泥巴的孩子是 a，b 和 c.
- 在父亲宣布 p 之前，a，b 和 c 不仅知道 p，而且知道彼此知道 p.
  - 以 a 的视角看，b 能看到 c 脸上有泥巴，所以 a 知道 b 知道 p.
- 但是，a，b，c 都不知道所有人知道所有人知道 p！

父亲宣布 p 的意义

- 用 $E^m p$ 表示所有人知道所有人知道……所有人知道（m 次）p.
- 在一般情况下，父亲没有宣布 p 之前，$E^k p$ 并不成立.
- 父亲宣布了 p 之后，对任意 $m \geq 1$，$E^m p$ 都成立！
- 因此，父亲宣布 p 带来了*共同知识*（common knowledge）.
- 有了共同知识，这一谜题就可以按照我们所讨论的方式进行下去.

共同知识假设

- 我们曾经假设过所有人“观察力敏锐、聪慧且诚实”.
- 然而，这一假设并不足够.
- 我们必须假设所有人都知道所有人“观察力敏锐、聪慧且诚实”，所有人都知道所有人都知道所有人“观察力敏锐、聪慧且诚实”，……
- 换言之，我们需要假设“所有人观察力敏锐、聪慧且诚实”是共同知识.

- 假设还是只有两个孩子 a, b 脸上有泥巴.
- 假如 a 不知道 b 是诚实的，即便 b 回答了“不知道”，a 也无法从 b 的回答中得到任何额外的知识！

共同知识假设

- 除了假设“所有人观察力敏锐、聪慧且诚实”是共同知识，我们还需要假设以下陈述是共同知识：
  - 每个人都能看到所有除自己外的人.
  - 每个人都听到了父亲说的话.
  - 父亲是诚实的.
  - 每个人都在每一轮进行了充分的推理.
  - ……
- 任何假设的破坏都会导致之前的讨论失效.
- 那么，为什么父亲宣布 p 就可以让 p 变成共同知识呢？

共同知识的产生

- 所有人都听到父亲说 p 并不能产生共同知识.
- 假如父亲只是对每一个孩子单独宣布 p.
  - 所有人并不知道所有人都知道 p，因而仅仅可以做到 Ep.
- 那么，所有人都知道所有人听到父亲说 p 会如何呢？
- 进一步假设每个孩子给每一个孩子都安装了窃听器，每个人都能够偷听每个人与父亲的谈话内容.
  - 所有人并不知道所有人都知道所有人都知道 p，因而仅仅有 E^2p.
- 因此，父亲宣布 p 会产生共同知识的核心原因是“公开宣布”，此时对每一个 m 都有 E^mp.

启示

- “泥泞的孩童”谜题足以表明，关于“知道”的讨论远比想象的复杂.

- 关于“知道”（know）和知识（knowledge）的研究在哲学中划归为*知识论*（epistemology）.
- 接下来，我们将使用模态逻辑来形式化关于“知道”和知识的讨论，这被称之为*认知逻辑*（epistemic logic）.

§12.2 认知逻辑的基本模型与性质

命题认知逻辑

- 假设有 n 个人，分别叫 $1, 2, \ldots, n$.
- 基本命题集为 $\mathbf{P}$，用字母 $p, q, r, \ldots$ 表示基本命题.
  - 例如，p 表示“孩子 1 的脸上有泥巴”.
- 回忆：逻辑框架是一个三元组（语言，模型，语义）.
- 命题认知逻辑的的三元组是：
  - 语言 L_n：命题逻辑加上模态算子 $K_i, i = 1, \ldots, n$.
  - 模型 $\mathcal{M}, w$：Kripke 模型
  - 语义 $\vDash$：可能世界语义

命题认知逻辑

- 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.
  - $\mathcal{M}, w \vDash K_i\phi$ 当且仅当对任意 v，如果 $w \to_i v$，那么 $\mathcal{M}, v \vDash \phi$.
- 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：$\mathcal{F} \vDash \phi$.

例子

- 虽然我们没有定义 K_i 的对偶算子，但是 K_i 的对偶相当于 $\Diamond$ 算子.
- $\neg K_i \neg \phi$ 表示的意思是“i 不知道 ϕ 不是真的”，因此 i 会考虑 ϕ 可能是真的. 当然，这也意味着 i 会考虑 $\neg\phi$ 也可能是真的.
- “我不知道上帝存在”vs. “我知道上帝不存在”.
  - $\neg Kp$ vs. $K\neg p$.
  - 显然，前者是更弱的一种表述，因此 $\neg K$ 和 $K\neg$ 的含义完全不同.

例子

- $K_1 K_2 p \wedge \neg K_2 K_1 K_2 p$.
  - 1 知道 2 知道 p，但是 2 并不知道 1 知道 2 知道 p.
- $\neg K_i p \to K_i(\neg K_i p)$.
  - 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，$K_i p \wedge K_i \neg K_i p$.
  - 我知道 p，但是我又知道我不知道 p.

Kripke 模型的限制

- 模态算子 K_i 有特殊的性质，这要求我们对 K_i 对应的关系 R_i 也有额外的要求.
- 我们要求每一个 R_i 都是等价关系 $\sim_i$：
  - 自反：$\forall x\, x \sim_i x$.
  - 传递：$\forall x, y, z (x \sim_i y \wedge y \sim_i z) \to x \sim_i z$.
  - 对称：$\forall x, y (x \sim_i y \leftrightarrow y \sim_i x)$.
- 从可能世界的角度来说，这一要求就是说对 i 来说，她所认为可能的世界之间都是不可区分的.

算子 K_i 的公理

- 从模态可定义性的角度来说，R_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（模式）或推导规则.
- 承认某一条公理（模式）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.

分配公理

- 分配公理（distribution axiom）：$\vDash (K_i(\phi \to \psi) \wedge K_i\phi) \to K_i\psi$.
- 有效性验证.
  - 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.
- 类比：演绎推理的肯定前件（Modus ponens，MP）推导规则.
- 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（logically omniscient）.

知识泛化规则

- 知识泛化规则（knowledge generalization rule）：对所有 $\mathcal{F}$，如果 $\mathcal{F} \vDash \phi$，那么 $\mathcal{F} \vDash K_i\phi$.
- 有效性验证.
  - 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 $\mathcal{F} \vDash K_i\phi$.
- 类比：一阶逻辑推理的泛化规则.
- 前提成立的 ϕ 是关于 $\mathcal{F}$ 本身的性质（特别是关于知识），因此知识泛化规则意味着个体知道关于知识的一切性质.
  - 其实更重要的是，知识的一切性质都是共同知识（练习）.

知识公理

- 知识公理（knowledge axiom）或真理公理（truth axiom）：$\vDash_H K_i\phi \to \phi$.
- 验证留作练习.
- 知识公理意味着，i 知道的命题一定是真的.
- 在知识论中，这一要求实际上反映了“拥有知识”需要付出努力、值得一定的奖励.
- 与此相对应地，信念（belief）则是更加主观、随意的，因而并不具有真理性.
  - 我考试挂了，但不知道我考试挂了.
  - 我考试挂了，但我不相信我考试挂了.

内省公理

- 正内省公理（positive introspective axiom）：$\vDash_H K_i\phi \to K_iK_i\phi$.
- 负内省公理（positive introspective axiom）：$\vDash_H \neg K_i\phi \to K_i\neg K_i\phi$.
- 验证留作练习.
- 这两条公理意味着个体会通过内省来知道自己的处境，特别是“我知道什么”和“我不知道什么”.
- 思考：（负）内省公理是否合理?

其他公理

- 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，包含了无穷条公理.
- 此外，从哲学的角度讨论，还有一些别的公理.
- 一致性公理（consistency axiom）：$\vDash_H \neg K_i\bot$.
  - 因此，个体不能够知道假的陈述，以此区别于信念.

公理与模型结构

- 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?

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
$\neg K_i\varphi \to K_i\neg K_i\varphi$	欧氏性
$\neg K_i\bot$	序列性
$\varphi \to K_i\neg K_i\neg\varphi$	对称性

- 我们总结如下表：

公理与模型结构

- 欧氏性：$\forall x, y, z(xR_iy \wedge xR_iz \to yR_iz)$.
  - 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.
- 以上验证都留作习题.
- 思考：从可能世界角度，R_i 的这些性质有什么直观的含义？
- 思考：为什么有些公理没有对应的结构性质？（提示：注意观察一个公理/规则是否有 H 出现）

公理与模型结构

- 可以看出来，以上关系其实并不是孤立的，我们有：

  引理 12.1
  - 如果 R_i 是自反和欧氏的，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 是欧氏的.
  - 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 是自反和欧氏的.

- 证明留作练习.

引入共同知识算子

- 下面我们将认知逻辑语言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，共同知识算子 C 的语义定义为：

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

模型含义

- 我们可以从图结构来理解共同知识算子.
- $\mathcal{M}, w \vDash E^k\phi$ 的含义是，从 w 出发走 k 步可到达的可能世界 v 上都有 $\mathcal{M}, v \vDash \phi$.
- $\mathcal{M}, w \vDash C\phi$ 的含义则是，从 w 出发可到达的可能世界 v 上都有 $\mathcal{M}, v \vDash \phi$.

共同知识的公理

- 类似算子 K_i，C 也有它对应的公理和推导规则.
- 不动点公理（fixed-point axiom）：$\vDash C\phi \leftrightarrow E(\phi \wedge C\phi)$.
  - 共同知识是一个递归方程的（最小）解，这是一种不动点的视角.
- 归纳规则（induction rule）：如果 $\mathcal{F} \vDash \phi \to E(\phi \wedge \psi)$，那么 $\mathcal{F} \vDash \phi \to C\psi$.
  - 每个人都可以从真实中得到的知识，一定是共同知识.
- 将 S5 公理系统中加入关于 E 和 C 的公理，我们就扩展了认知逻辑.
- 因为 E 和 C 是用 K_i 定义的，因此他们本身并不会带来 Kripke 模型新的结构性质.

“泥泞的孩童”再回顾

- 现在，我们可以形式化、严格讨论“泥泞的孩童”这一谜题了.
- 这一问题对应的逻辑语言是认知逻辑语言.
- 可能世界是 $\{0,1\}^n$ 的元素 $x = (x_1, \dots, x_n)$，$x_i = 1$ 表示孩子 i 脸上有泥巴，$x_i = 0$ 表示 i 脸上没有泥巴.

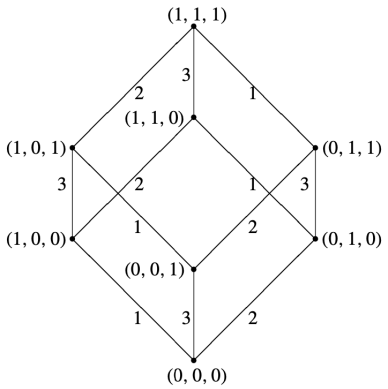
- 假设每个孩童 i 对应的 R_i 都是一个等价关系.
- 当我们如此假设时，每个孩子唯一不是共同知识的事情就是脸上泥巴的状态，其他的所有事情都被隐含在了共同知识之中.

“泥泞的孩童”再回顾

- 原子命题 p_i：孩子 i 脸上有泥巴，p：至少有一个孩子脸上有泥巴.
- 假设现在父亲还没有说出 p.
- 对于孩子 i，他的认知中只有两个可能世界：我的脸上有泥巴，或者我的脸上没有泥巴，其他对他来说都是确定的.
  - xR_iy 当且仅当 $x_j = y_j$ 对任意 $j \neq i$ 成立.
- 因此，框架 $\mathcal{F}$ 会对应一个 n 维超立方体.

“泥泞的孩童”再回顾

- $n = 3$ 的例子：



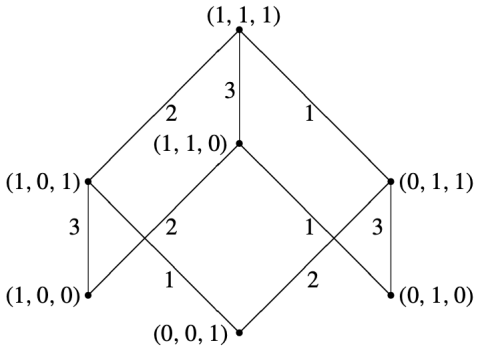
“泥泞的孩童”再回顾

- 从框架 $\mathcal{F}$ 到模型 $\mathcal{M}$，我们还需要确定赋值 V.
- $w \in V(p_i)$ 当且仅当 $w_i = 1$.
- $w \in V(p)$ 当且仅当所有分量 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，于是我们就可以讨论模态公式的可满足性.
- 例如：$\mathcal{M}, (1,0,1) \vDash Ep$，但是 $\mathcal{M}, (1,0,1) \vDash \neg E^2 p$.

“泥泞的孩童”再回顾

- 假设现在父亲宣布了 p，那么 $\mathcal{F}$ 将会发生变化：



“泥泞的孩童”再回顾

- 在 i 眼中，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，但是她却说“不知道”，说明真实世界不是 s.
- 同理，所有那些只有一个 1 的可能世界都会被消掉.
- 因此，归纳可得，第 k 轮的时候，所有那些只有 k 个 1 的可能世界会被消掉.

“泥泞的孩童”再回顾

- 如果父亲没有宣布 p，那么 $\mathcal{M}$ 是一个超立方体.
- 无论在任何轮，每一个孩子都会觉得两个可能世界，因此不会有任何可能世界被消掉！

- 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.
- 这一套方法可以将类似的智力谜题都用算法化的方式得到解答.

基于事件的认知理论

- 如果我们限制 Kripke 模型中 R_i 为等价关系，那么认知逻辑还可以有另一种理解方式.
- 回忆：概率论（或者似然）理解世界的方式基于“事件”.
- 我们只能感知事件的发生与否，而不能具体知道是哪个样本点.
- 用事件的方式理解认知，得到的结构被称为 *Aumann 结构*.

Aumann 结构

- 考虑全集 Ω，理解为样本空间.
- 事件 $e \subseteq \Omega$ 是样本点的集合.
- 一次观测会落实在一个样本点 ω 上.
- 事件 e 发生当且仅当 $\omega \in e$.
- 在 Kripke 模型中，我们将 i 的知识刻画为了等价关系 R_i.
- 在 Aumann 结构中，对每一个个体 i，它的知识被刻画为 Ω 的划分 $\mathcal{P}_i = \{\Omega_j\}$，$\Omega_j$ 是 Ω 的子集.
- S_j 被称为 i 的*信息集*（information sets），可以被理解为 i 能够感知到的最基本的事件.

Aumann 结构

- $\mathcal{P}_i(\omega)$ 被定义为 ω 所属于的那个信息集.
- 现在我们重新定义算子 $K_i : 2^\Omega \to 2^\Omega$ 为

$$K_i(e) := \{\omega \in \Omega : \mathcal{P}_i(\omega) \subseteq e\}.$$

- $K_i(e)$ 定义了“个体 i 知道事件 e”这一事件.

- 思想：将所有关于知识的讨论转化为关于事件的讨论.

Aumann 结构

- 类似地，可以定义算子 $E : 2^\Omega \to 2^\Omega$ 为：

$$E(e) = \bigcap_{i=1}^{n} K_i(e).$$

- 定义共同知识算子 $C : 2^\Omega \to 2^\Omega$ 为：

$$C(e) = \bigcap_{k=1}^{\infty} E^k(e).$$

Kripke 模型与 Aumann 结构的关系

- 我们再一次看到逻辑和集合的对应，我们总结如下表：

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 R_i	i 的划分
逻辑连接词	集合操作

Kripke 模型与 Aumann 结构的关系

- Kripke 语义偏好逻辑，Aumann 结构偏好（Bayes）概率论，因此用数学来研究知识论就有了两种风格，一种是计算机、逻辑、哲学的风格，另一种是经济学、信息论的风格.
- 但是，这两种对应关系完全取决于我们对知识的那些基本假设，所以如果这些假设被打破，那么这样的对应关系就不再成立！
- 下面我们分别用这两种风格来讨论共同知识的意义.

§12.3 对不一致达成一致

问题背景

- 本部分将用模态逻辑的方式来探讨达成一致与共同知识的关系.
- 最早由 Aumann 给出.
- 我们将要证明，对于有相同决策方式的两个个体来说，他们不可能对采取不同行动这件事具有共同知识.
- 典型故事：同样的 AI 之间会发生交易吗?
- 交易发生意味着买家和卖家有不一样的决策（一个买一个卖）.
- 因此，如果两个人按照相同的规则来行事，那么不会有交易发生！

players cannot “agree to disagree”.

模型

- 假想一个含时的系统，有两个玩家 1 和 2.
- 在任意时刻，每个玩家处于一个状态 s_i 之中.
- 每个玩家分别有一个自己的局部状态空间 S_i.
- 整个系统的全局状态是 $(s_1, s_2) \in S_1 \times S_2 = \mathcal{G}$.
- 时刻是离散的，用非负整数 m 表示，初始时刻是 0.
- 系统的一次运行（run）指的是函数 $r : m \mapsto (s_1, s_2)$.
  - 运行描述了系统每一时刻的全局状态.
- 系统（system）$\mathcal{R}$ 指的是 $\mathcal{G}$ 上所有可能运行的集合.
- 给定 $r \in \mathcal{R}$，(r, m) 被称为系统 $\mathcal{R}$ 的一个点.

模型

- 玩家处于某个状态的时候可以采取某种行动.

- 为了反映“玩家按照相同的规则行事”这件事，我们规定两个玩家的行动集都是 A，并且这一集合不依赖于全局或局部的状态.
- 给定所有人的行动和一个全局状态，我们可以定义系统的*转移函数*（transition function）为 $\tau : A^2 \times \mathcal{G} \to \mathcal{G}$.
  - 因此，转移函数描述了所有人的行动如何导致系统从一个状态到另一个状态.

模型

- 如何描述“按照规则行事”?
- 我们用*协议*（protocol）来描述这种概念.
- 玩家 i 的协议 P_i 是一个从局部状态 S_i 到行动集 A 的映射.
  - 处于什么状态就做什么事.
- 两个玩家的联合协议记为 $P = (P_1, P_2)$.

模型

- 一个联合协议要执行起来，还需要初始状态.
- 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协议.
- 我们把元组 $\gamma = (\mathcal{G}_0, \tau)$ 称为系统的*上下文*（context）.

模型

- 给定上下文 $\gamma = (\mathcal{G}_0, \tau)$ 和一个联合协议 P，我们可以讨论 P 产生的所有可能运行.
- 一个运行 r 与 P *相容*（compatible）指的是
  - $r(0) \in \mathcal{G}_0$.
  - 对任意时刻 m，如果 $r(m) = (s_1, s_2)$，那么 $r(m+1) = \tau(P(s_1), P(s_2))(s_1, s_2)$.
- 换言之，r 是从跟某个可能的初始状态开始执行协议产生的运行.
- 一个系统 $\mathcal{R}$ 表示了上下文 γ 和联合协议 P，指的是所有 $r \in \mathcal{R}$ 都与 P 相容. 这样的系统我们用记号 $\mathcal{R}^{rep}(P, \gamma)$ 来表示.

模型

- 接下来我们引入 Kripke 模型.
- Kripke 模型的点是系统的点.
- 设原子命题集 **P**，它的元素是 $perf_i(a)$，表示玩家 i 采取行动 a.
- 接下来我们定义赋值函数 V.
- 从 **P** 的定义来看，赋值应该只依赖状态，而不依赖时间，所以我们赋值函数实际上需要分两步来定义：
  - 定义 V 为从 **P** 到全局状态集合的映射，
  - 然后再扩展为到系统点集合的映射：$(r,m) \in V(p) \iff r(m) \in V(p)$.
- 第一步定义如下：状态 $s \in V(perf_i(a))$ 当且仅当在状态 s 玩家 i 采取过行动 a.

模型

- 然后我们引入知识算子 K_i 的语义.
- 同样，我们假设 K_i 对应的是等价关系 $\sim_i$.
- 玩家 i 只能区分自己的局部状态 s_i，他执行协议时，只有状态，没有时间的概念.
- 因此我们定义 $(r,m) \sim_i (r',m') \iff r(m)_i = r'(m')_i$.
- 从 Aumann 结构来说，每一个局部状态 s_i 对应了一个信息集

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

- 这样，我们就得到了 Kripke 点模型 $\mathcal{M},(r,m)$.
- K_i 的语义按照基本认知逻辑定义即可.

模型

- 接下来，我们引入关于时间的模态算子.
- 特别地，我们只引入算子 X，表示“下一时刻”.

- 它的语义定义为

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

- 有了算子 X，我们可以用公式表达“将要采取行动”：

$$act_i(a) = \neg perf_i(a) \wedge Xperf_i(a).$$

模型

- 接下来，我们定义关于 Kripke 模型的*决策函数*（decision function），用它来在点模型的角度讨论协议的执行.
- 设 Kripke 模型的点集为 S.
- 玩家 i 的决策函数 D 是从 S 的某些子集到行动集 A 的映射.
  - 我们没有写决策函数的下标，表明两个玩家采取了相同的决策策略.
- 决策函数描述的是：知道什么样的信息，就采取什么样的行动.
  - 回忆 Aumann 结构，S 的子集是事件.

模型

- 我们要求协议 P_i 和决策函数 D 是相容的，也就是决策函数在某个信息集上采取的行动恰好是这个协议在该状态要执行的行动：

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

- 反过来说，联合协议 P 在上下文 γ 中*实现*（implement）了决策函数 D，如果对所有 i，P_i 与 D 在系统 $\mathcal{R}^{rep}(P, \gamma)$ 中是相容的.

模型

- 协议和决策函数是两个非常容易混淆的概念，尽管他们有密切联系.
- 直观来说，协议就是处于什么局部状态采取什么行动，这并不涉及知识的内容.
- 而决策函数指的是，知道什么信息就采取什么行动，这完全是知识的内容.
- 在我们的背景下，

$$\text{知道的信息} = \text{处于的局部状态}.$$

- 因此二者其实是从不同角度描述同一个概念.

模型

- 我们对决策函数 D 有一个额外的技术要求，我们要求 D 是并-一致的（union-consistent）.
- 具体来说，给定 S 一列互不相交的子集 $T_1, \ldots, T_k$，每一个都有 $D(T_i) = a$，那么我们要求 $D(\cup_i T_i) = a$.
  - 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.
- 性质：任何联合协议都可以从某个并-一致的决策函数产生.

模型

- 我们现在回顾一下这个模型.
- 两个玩家处于同一个系统中.
- 每个玩家可能知道不同的东西（局部状态空间不同，信息集不同）.
- 但是他们的行动集相同、决策函数相同.
- 决策函数要求是并-一致的，由某个联合协议实现.
- 给定可能的初始状态和系统的转移函数（上下文），系统可以产生一系列可能的运行.

达成一致定理

定理 12.1 (Aumann，达成一致定理，Agreement theorem) 给定联合协议 P，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.
- 他们不可能对不一致达成一致（agree to disagree）.

达成一致定理：证明

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

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

- 我们证明 $a = b$.
- 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 $a = b$.

达成一致定理：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.

达成一致定理：证明

- 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 $a = b$.

达成一致定理：拓展

- 我们的定理是对于确定性的协议证明的.
- 然而，一个协议可能是非确定的，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.
- 这个时候，达成一致定理依然成立，但是我们需要恰当地定义 Kripke 模型、决策函数以适应非确定性的协议.
- 当协议具有非确定性时，我们可以用这一模型来理解带有先验知识（分布）、风险或者不确定性下的达成一致定理.
  - 只要协议能够对应一个并-一致的决策函数，结论都有效.

§12.4 Rubinstein 电子邮件博弈

Rubinstein 电子邮件博弈

- 接下来我们使用 Bayes 概率论来说明在二人静态博弈中，共同知识对到底实现哪一个 Nash 均衡非常关键.
- 此时，知道一件事与否被赋予了不确定性的含义：我确定或不确定某件事发生.

Decision Letter on the“E-Mail Game”, AER, 1989

30 March, 1988

Dear Ariel,

Please find enclosed two thoughtful reports (actually, one is thoughtful; the other one is just silly) on your stimulating paper “the e-mail game”, which you submitted for publication in the AER. As you will see, it’s quite impossible to understand what they’re trying to say, or what their final recommendation is.
Now, I didn’t feel like reading your paper myself. You seem like a nice guy, the title of the paper is catchy, and people say you write pretty good papers, so I decided to accept your paper for publication. You also won a free ticket to a Broadway musical of your choice.

Sincerely yours,
The Editor

Rubinstein 电子邮件博弈

	A	B
A	$(0,0)$	$(-10,1)$
B	$(1,-10)$	$(8,8)$

	A	B
A	$(8,8)$	$(-10,1)$
B	$(1,-10)$	$(0,0)$

- 两个玩家和两个可能的矩阵.
- 在左边的矩阵中：
  - (B,B) 是唯一的 Nash 均衡.
- 在右边的矩阵中：
  - 这是一个协作博弈.
  - 多个 Nash 均衡：(A,A) 和 (B,B).
  - 观察：(A,A) 给出比 (B,B) 更高的收益，但行动 A 比 B 更有风险.

Rubinstein 电子邮件博弈

	A	B
A	$(0,0)$	$(-10,1)$
B	$(1,-10)$	$(8,8)$

	A	B
A	$(8,8)$	$(-10,1)$
B	$(1,-10)$	$(0,0)$

- 左边矩阵被选择的概率是 $p > 1/2$.
- 玩家 1 知道真实的矩阵，而玩家 2 不知道.
- 如果选择了右边矩阵，玩家 1 会给玩家 2 发送一条消息.

Rubinstein 电子邮件博弈

	A	B
A	$(0,0)$	$(-10,1)$
B	$(1,-10)$	$(8,8)$

	A	B
A	$(8,8)$	$(-10,1)$
B	$(1,-10)$	$(0,0)$

- 如果玩家 2 收到了消息，她会回复.

- 如果玩家 1 收到了回复，她会发送第二条消息来确认她收到了玩家 2 的回复.
- 以此类推.

Rubinstein 电子邮件博弈

	A	B
A	$(0,0)$	$(-10,1)$
B	$(1,-10)$	$(8,8)$

	A	B
A	$(8,8)$	$(-10,1)$
B	$(1,-10)$	$(0,0)$

- 每条消息都以 ϵ 的概率独立等可能丢失.
- 注意：发送电子邮件不是一个行动，而是一个规则.

Rubinstein 电子邮件博弈

- $\Theta_i = \{\theta_i^0, \theta_i^1, \theta_i^2, \dots\}$.
- θ_i^m：m 是玩家 i 发的邮件数量.
- θ_i^m 的解释：
  - θ_1^0：真实收益矩阵是左边的.
  - θ_1^1：真实收益矩阵是右边的，1 发送了一封电子邮件，但 2 没有收到.
- θ 包含了所有可能的情况：
  - (θ_1^0, θ_2^0)：真实收益矩阵是左边的.
  - (θ_1^1, θ_2^0)：真实收益矩阵是右边的，1 发送了一封电子邮件，但 2 没有收到.
  - (θ_1^1, θ_2^1)：真实收益矩阵是右边的，2 收到了第一封电子邮件，但 1 没有收到 2 的回复.

Rubinstein 电子邮件博弈

- 以概率 p 选择左边的矩阵.
- 而且没有人发送消息.
- 因此，(θ_1^0, θ_2^0) 的概率是 p.

Rubinstein 电子邮件博弈

左	θ_2^0	θ_2^1	θ_2^2	$\cdots$
θ_1^0	p	0	0	$\cdots$
θ_1^1	0	0	0	$\cdots$
θ_1^2	0	0	0	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

右	θ_2^0	θ_2^1	θ_2^2	$\cdots$
θ_1^0	0	0	0	$\cdots$
θ_1^1	$\epsilon(1-p)$	$\epsilon(1-\epsilon)(1-p)$	0	$\cdots$
θ_1^2	0	$\epsilon(1-\epsilon)^2(1-p)$	$\epsilon(1-\epsilon)^3(1-p)$	$\cdots$
θ_1^3	0	0	$\epsilon(1-\epsilon)^4(1-p)$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

- 以概率 $1-p$ 选择右边的矩阵.
- 玩家 1 发送一条消息，它会以概率 ϵ 丢失.
- 因此，(θ_1^1, θ_2^0) 的概率是 $\epsilon(1-p)$.

Rubinstein 电子邮件博弈

- 对类型 θ_i^m，收益矩阵是到第 m 层的共同知识，即 E^m.
- 所以对于很大的 m，收益矩阵是“几乎公共知识”.
- 信息结构是玩家的共同知识，玩家们进行博弈.
- 问题：这个博弈的 BNE 是什么?

Rubinstein 电子邮件博弈

- 我们需要弄清楚对每个类型 θ_i^m，玩家会做什么.
- 假设玩家 1 的类型为 θ_1^0.
- 玩家 1 知道 (θ_1^0, θ_2^0) 是真实的类型.
- $\Longrightarrow$ 左边的矩阵被选择.
- 据此推理：玩家 1 选择占优策略 B.

Rubinstein 电子邮件博弈

- 假设玩家 2 的类型为 θ_2^0.
- Bayes 定理意味着：
  - $\Pr(\theta_1^0|\theta_2^0) = \frac{p}{p+\epsilon(1-p)} := \mu_2^0 \implies$ 左边的矩阵被选择.
  - $\Pr(\theta_1^1|\theta_2^0) = 1 - \mu_2^0 \implies$ 右边的矩阵被选择.
- 选择 B 的期望收益至少是 $8\mu_2^0$.
  - 推理：类型 θ_1^0 时肯定选择 B.
  - $\implies$ 最坏的情况是 θ_1^1 选择 B.
- 选择 A 的期望收益最多是 $-10\mu_2^0 + 8(1-\mu_2^0)$.
  - 推理：类型 θ_0^1 肯定选择 B.
  - $\implies$ 最好的情况是 θ_1^1 选择 A.
- $\implies$ B 更好，因为对于所有 ϵ，$\mu_2^0 \geq p > \frac{1}{2}$.

Rubinstein 电子邮件博弈

- 假设玩家 1 的类型为 $\theta_1^1 \implies$ 右边的矩阵被选择.
- Bayes 定理意味着：
  - $\Pr(\theta_2^0|\theta_1^1) = \frac{\epsilon(1-p)}{\epsilon(1-p)+\epsilon(1-\epsilon)(1-p)} = \frac{1}{2-\epsilon} := \mu_1^1$.
  - $\Pr(\theta_2^1|\theta_1^1) = 1 - \mu_1^1$.
- 选择 B 的期望收益至少为 0.
  - 推理：类型 θ_2^0 肯定选择 B.
  - $\implies$ 最坏的情况是 θ_2^1 选择 B.
- 选择 A 的期望收益最多为 $-10\mu_1^1 + 8(1-\mu_1^1)$.
  - 推理：类型 θ_2^0 肯定选择 B.
  - $\implies$ 最好的情况是 θ_2^1 选择 A.
- $\implies$ B 更好，因为对于所有 ϵ，$\mu_1^1 > \frac{1}{2}$.

Rubinstein 电子邮件博弈

- 逐步迭代上述过程.
- $\Longrightarrow$ 在唯一的 BNE 中，所有类型都选择 B.
- 回忆：如果右边的矩阵是共同知识，(A, A) 是一个严格 Nash 均衡.
- 结论：即便收益矩阵是“几乎共同知识”，Nash 均衡也不一定是一个可实现的均衡.

关于均衡的进一步思考

- 用 $Nash(x)$ 表示“x 是 Nash 均衡”，那么 $\exists x C(Nash(x))$ 和 $C(\exists x C(Nash(x)))$ 的含义是否一样?
- 如果不引入不确定性，在完全信息下，实现特定的 Nash 均衡是否还需要共同知识?
- 如果玩家不是逻辑全知的，或者说她的推理、计算能力是有限的，那么 Nash 均衡还是否会达到？是否可接近?

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