機率與統計：從入門到精通

第 2 章　條件機率與貝氏定理

Conditional Probability and Bayes' Theorem

難度：入門

幾乎所有真正有用的機率，都是**條件機率**——在「已經知道某些事」的前提下，問另一件事發生的可能性。檢測結果呈陽性，患病的機率是多少？郵件裡出現「free」這個詞，它是垃圾郵件的機率是多少？給定前面的文字，下一個詞會是什麼？第 1 章的機率像一張靜態快照；本章要讓它**動**起來：**條件化**（conditioning）是機率吸收證據的方式，而**貝氏定理**是把方向反轉的引擎——它把容易取得的「已知原因下，證據出現的機率」，翻轉成我們真正想要的「看到證據後，原因成立的機率」。若說機器學習的本質是「從資料推論未知」，那麼本章就是這句話的數學——全書「機器學習 = 推論」主線上最重要的一章。

學習目標

讀完本章後，你將能夠：

- 使用定義 $\mathbb{P}(A \mid B) = \mathbb{P}(A \cap B)/\mathbb{P}(B)$ 計算條件機率，並以「縮小樣本空間 + 重新正規化」的觀點解釋它。
- 運用**乘法規則**與**鏈式法則**，把複雜的聯合機率拆成簡單條件機率的乘積。
- 判斷事件的**獨立性**與**條件獨立性**，清楚分辨「獨立」與「互斥」。
- 用**全機率定理**「分情況再重組」，並說明它與邊際化、貝氏定理分母（證據項）的關係。
- 敘述並使用**貝氏定理**更新信念，完整解釋**基率謬誤**，並執行序貫貝氏更新。
- 說明「後驗 $\propto$ 概似 $\times$ 先驗」如何連結最大概似估計（MLE）、最大後驗估計（MAP）與自迴歸語言模型，並用 Python 驗證本章的計算。

2.1 條件機率：資訊如何改變機率

本章開頭要問一個實際的問題：**機率會變嗎？**會——只要你知道的事情變了。骰子還沒擲，出現 6 的機率是 1/6；若有人偷看後告訴你「點數是偶數」，同一次投擲出現 6 的機率立刻變成 1/3。骰子沒有變，變的是**你所知道的事**：機率不是事件的固有屬性，而是事件與資訊共同決定的量。條件機率（conditional probability）就是這件事的精確語言。

定義 2.1　條件機率

設 A, B 為事件且 $\mathbb{P}(B) > 0$。在「已知 B 發生」之下 A 的**條件機率**（conditional probability）定義為

$$\mathbb{P}(A \mid B) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)},$$

讀作「給定 B 之下 A 的機率」。當 $\mathbb{P}(B) = 0$ 時此式沒有定義——你無法對不可能的事條件化（連續情形的細緻處理見第 3 章）。

這個公式值得慢慢咀嚼，因為它有兩層直觀，各自對應公式的一半：

- **分子——縮小樣本空間**：一旦知道 B 發生，B 之外的所有結果全部出局，B 成為**新的宇宙**。在新宇宙裡，A 還能發生的部分只剩 $A \cap B$。
- **分母——重新正規化**：新宇宙的總機率只有 $\mathbb{P}(B)$，不再是 1。除以 $\mathbb{P}(B)$ 是把尺度重新拉回「總和為一」，讓 $\mathbb{P}(\cdot \mid B)$ 在 B 之內仍然是一個合法的機率分配。

在等可能的有限樣本空間中，兩層直觀合成一句話：$\mathbb{P}(A \mid B) = |A \cap B| / |B|$——「$B$ 裡面有多大比例也屬於 A」。

例 2.1　擲骰子：資訊改變機率

(a) 擲一顆公正骰子。已知點數為偶數，求點數大於 3 的機率。
(b) 擲兩顆公正骰子。已知點數和為 8，求第一顆是 5 的機率。

解　(a) 條件 $B = \{2, 4, 6\}$ 是新宇宙，其中大於 3 的結果為 $A \cap B = \{4, 6\}$，故

$$\mathbb{P}(A \mid B) = \frac{|A \cap B|}{|B|} = \frac{2}{3},$$

高於無條件的 $\mathbb{P}(A) = 3/6 = 1/2$：「偶數」這條資訊把機率往上推了。

(b) 和為 8 的結果有 $B = \{(2,6), (3,5), (4,4), (5,3), (6,2)\}$，共 5 個等可能結果；其中第一顆是 5 的只有 $(5,3)$，故

$$\mathbb{P}(\text{第一顆是5} \mid \text{和為8}) = \frac{1}{5},$$

高於無條件的 $1/6$。同樣的條件也能把機率打到零：$\mathbb{P}(\text{第一顆是1} \mid \text{和為8}) = 0$。資訊可以推升、壓低機率，也可以直接排除可能性。

直觀理解

把條件化想成**聚光燈**：原本整個 Ω 都亮著，得知 B 發生後，光圈縮到 B 上，機率在 B 之內**按原比例**重新分配——這就是「除以 $\mathbb{P}(B)$」的幾何意義。先埋一顆種子：「得知 B 發生，於是更新所有機率」正是模型**看到資料時做的事**——過濾器讀到「free」、語言模型讀到前文、診斷模型讀到報告，每一次「看到」都是一次條件化。

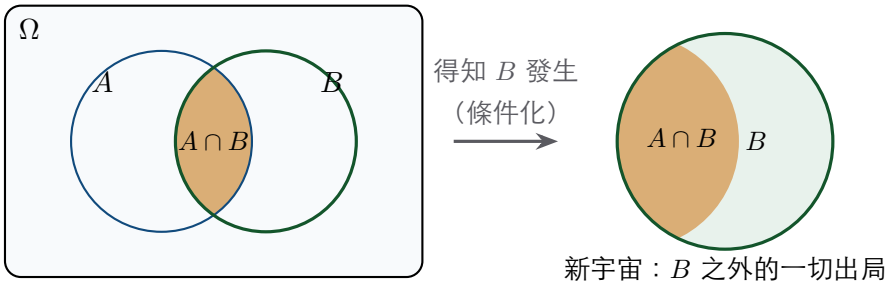


圖 2.1: 條件化就是換宇宙：得知 B 發生後，B 之外的可能性全部作廢，B 成為新的樣本空間；$\mathbb{P}(A \mid B)$ 就是 $A \cap B$ 在新宇宙中所佔的比例。

命題 2.2　條件機率是合法的機率

固定 B 且 $\mathbb{P}(B) > 0$，把 $\mathbb{P}(\cdot \mid B)$ 視為「事件 A 的函數」，則它滿足 Kolmogorov 三公理：非負性、$\mathbb{P}(\Omega \mid B) = 1$，以及互斥事件的可加性。

證明　非負性顯然。規範性：$\mathbb{P}(\Omega \mid B) = \mathbb{P}(\Omega \cap B)/\mathbb{P}(B) = \mathbb{P}(B)/\mathbb{P}(B) = 1$。可加性：若 $A_1, A_2, \ldots$ 互斥，則 $A_1 \cap B,\ A_2 \cap B, \ldots$ 也互斥，故

$$\mathbb{P}\Big(\bigcup_i A_i \,\Big|\, B\Big) = \frac{\mathbb{P}\big(\bigcup_i (A_i \cap B)\big)}{\mathbb{P}(B)} = \frac{\sum_i \mathbb{P}(A_i \cap B)}{\mathbb{P}(B)} = \sum_i \mathbb{P}(A_i \mid B). \blacksquare$$

這個命題的價值是實用性的：條件化之後**不需要學任何新規則**，第 1 章的補集規則、單調性與排容原理在「$\mid B$ 的世界」原封不動可用，例如 $\mathbb{P}(A^c \mid B) = 1 - \mathbb{P}(A \mid B)$。

常見陷阱

條件機率有兩個最常見的錯誤。**其一：交換條件。**$\mathbb{P}(A \mid B) \neq \mathbb{P}(B \mid A)$：「患病者驗出陽性」的機率可以是 99%，「陽性者真的患病」卻可能只有 50%（見 2.5 節）。法庭版稱為**檢察官謬誤**（prosecutor's fallacy）：「若被告無辜，出現此證據的機率僅百萬分之一」**不等於**「出現此證據，被告無辜的機率僅百萬分之一」——差多少，由基率（先驗）決定。**其二：跨條件互補。**$\mathbb{P}(A \mid B) + \mathbb{P}(A^c \mid B) = 1$ 恆成立，但 $\mathbb{P}(A \mid B) + \mathbb{P}(A \mid B^c)$ 一般**不**等於 1——它們活在兩個不同的宇宙，不能混加。

2.2　乘法規則與鏈式法則

把條件機率的定義移項，就得到一個看似平凡、實則威力巨大的等式。視角就此翻轉：定義說「用聯合機率算條件機率」，乘法規則說「用條件機率**建構**聯合機率」——而後者才是機率建模的日常。

定理 2.3　乘法規則

若 $\mathbb{P}(B) > 0$，則

$$\mathbb{P}(A \cap B) = \mathbb{P}(A \mid B)\,\mathbb{P}(B).$$

對稱地，若 $\mathbb{P}(A)>0$，亦有 $\mathbb{P}(A\cap B)=\mathbb{P}(B\mid A)\,\mathbb{P}(A)$。「兩件事同時發生」的機率 =「第一件發生」的機率 ×「在第一件發生下，第二件也發生」的機率。

例 2.2　連抽三張不放回，張張紅心

從一副 52 張撲克牌中連抽三張、不放回，求三張全是紅心的機率。

解　設 H_i =「第 i 張是紅心」。逐步條件化：第一張是紅心的機率為 $13/52$；已知第一張是紅心，牌堆剩 51 張、紅心剩 12 張，故 $\mathbb{P}(H_2\mid H_1)=12/51$；再抽走一張紅心後 $\mathbb{P}(H_3\mid H_1\cap H_2)=11/50$。三段相乘：

$$\mathbb{P}(H_1\cap H_2\cap H_3)=\frac{13}{52}\times\frac{12}{51}\times\frac{11}{50}=\frac{11}{850}\approx 0.0129.$$

用第 1 章的組合計數驗算：$\binom{13}{3}/\binom{52}{3}=286/22100=11/850$，一致。條件化法的好處，是把「不放回」的動態過程攤開成一串**每一步都簡單**的條件機率。

把乘法規則反覆使用，就得到本章第一個「通往現代機器學習」的公式。

定理 2.4　鏈式法則

設 $A_1,\dots,A_n$ 為事件且 $\mathbb{P}(A_1\cap\cdots\cap A_{n-1})>0$，則

$$\mathbb{P}(A_1\cap A_2\cap\cdots\cap A_n)=\mathbb{P}(A_1)\,\mathbb{P}(A_2\mid A_1)\,\mathbb{P}(A_3\mid A_1\cap A_2)\cdots\mathbb{P}(A_n\mid A_1\cap\cdots\cap A_{n-1}).$$

任何聯合機率都可以拆解成「一步一步條件化」的連乘積。

證明　把右邊每個條件機率寫回定義，得到望遠鏡式的連乘：

$$\mathbb{P}(A_1)\cdot\frac{\mathbb{P}(A_1\cap A_2)}{\mathbb{P}(A_1)}\cdot\frac{\mathbb{P}(A_1\cap A_2\cap A_3)}{\mathbb{P}(A_1\cap A_2)}\cdots\frac{\mathbb{P}(A_1\cap\cdots\cap A_n)}{\mathbb{P}(A_1\cap\cdots\cap A_{n-1})},$$

相鄰的分子與分母兩兩對消，只剩 $\mathbb{P}(A_1\cap\cdots\cap A_n)$。條件 $\mathbb{P}(A_1\cap\cdots\cap A_{n-1})>0$ 保證途中每個分母皆為正（見練習 2.9）。 ■

鏈式法則的深意是**複雜的整體可以分解成簡單的局部**：直接對一整句話指定機率毫無頭緒，「給定前文，下一個詞的機率」卻是可以學習的對象——把條件機率串起來，聯合分布就自動組裝完成。這是從單純貝氏到大型語言模型共用的骨幹。

例 2.3　玩具語言模型：一句話的機率

一個語言模型對每段「前文」輸出下一個詞的條件分布。考慮句子「今天天氣很好」，設模型依序給出四個條件機率：$\mathbb{P}(\text{今天})=0.04$（句首）、$\mathbb{P}(\text{天氣}\mid\text{今天})=0.18$、$\mathbb{P}(\text{很}\mid\text{今天天氣})=0.35$、$\mathbb{P}(\text{好}\mid\text{今天天氣很})=0.30$。求整句的機率，並與把最後一詞換成「香蕉」（設 $\mathbb{P}(\text{香蕉}\mid\text{今天天氣很})=0.0001$）比較。

解　由鏈式法則，整句機率是四個條件機率的連乘：

$$\mathbb{P}(\text{今天天氣很好})=0.04\times0.18\times0.35\times0.30=7.56\times10^{-4}.$$

換成「香蕉」後，前三個因子不變，最後一個因子從 0.30 掉到 0.0001：

$$\mathbb{P}(\text{今天天氣很香蕉}) = 0.04 \times 0.18 \times 0.35 \times 0.0001 = 2.52 \times 10^{-7},$$

小了 3000 倍——模型用機率說出「這句話不自然」。實務註腳：每個因子都是模型一次前向傳播的 softmax 輸出；連乘積很快小到浮點數存不住，故實務上取對數，把乘法變加法：$\log \mathbb{P} = \sum_t \log \mathbb{P}(x_t \mid x_{<t})$。

這如何驅動機器學習／ AI

GPT 這一類**自迴歸**（autoregressive）語言模型分解句子機率的方式，一字不差就是鏈式法則：

$$\mathbb{P}(x_1, x_2, \ldots, x_n) = \prod_{t=1}^{n} \mathbb{P}(x_t \mid x_1, \ldots, x_{t-1}).$$

注意這是**恆等式**而非近似——模型架構（Transformer）的全部工作，只是**學習**那些條件分布。訓練目標是最大化語料的 $\sum_t \log \mathbb{P}(x_t \mid x_{<t})$，等價於最小化**交叉熵**（cross-entropy）損失；生成則是逐詞元（token）從條件分布抽樣。單純貝氏（2.3 節）用條件獨立把鏈式法則砍到最簡，大型語言模型則保留完整條件結構、用上億參數逼近它——同一條公式的兩個極端。

2.3 獨立性與條件獨立

條件化之後機率通常會變，但有一種重要的例外：資訊進來了，機率**紋風不動**。這就是獨立性——「B 的發生與否，對 A 毫無情報價值」。

定義 2.5　獨立性

事件 A 與 B **獨立**（independent），記作 $A \perp B$，若

$$\mathbb{P}(A \cap B) = \mathbb{P}(A)\,\mathbb{P}(B).$$

不獨立時稱兩事件**相依**（dependent）。

命題 2.6　獨立性的等價刻畫

若 $\mathbb{P}(B) > 0$，則

$$A \perp B \iff \mathbb{P}(A \mid B) = \mathbb{P}(A).$$

即：知道 B 發生，完全不改變 A 的機率。

證明　由定義 $\mathbb{P}(A \mid B) = \mathbb{P}(A \cap B)/\mathbb{P}(B)$，故 $\mathbb{P}(A \mid B) = \mathbb{P}(A) \iff \mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B)$。■

乘法形式對稱、且在 $\mathbb{P}(B) = 0$ 時仍有意義，故作為正式定義。獨立性可以有物理直覺支持（兩顆骰子互不接觸），但**終究是要用數字驗證的等式**——直覺對「看起來有關 / 無關」的判斷經

常出錯，下面的例子就是警告。

例 2.4　獨立性必須驗證，不能用感覺

擲兩顆公正骰子。設 $A=$「第一顆是偶數」、$B=$「點數和為 7」、$C=$「點數和為 8」。問：A 與 B 獨立嗎？A 與 C 呢？

解　$\mathbb{P}(A)=1/2$，$\mathbb{P}(B)=6/36=1/6$，$\mathbb{P}(C)=5/36$。

$A\cap B=$「第一顆偶數且和為 7」$=\{(2,5),(4,3),(6,1)\}$，機率 $3/36=1/12$。檢驗：

$$\mathbb{P}(A)\,\mathbb{P}(B)=\frac{1}{2}\times\frac{1}{6}=\frac{1}{12}=\mathbb{P}(A\cap B),$$

所以 $A\perp B$——「和為 7」對第一顆骰子的奇偶**毫無情報**。原因很漂亮：不論第一顆擲出多少，第二顆都恰有一種點數能湊成 7，因此「和為 7」的機率永遠是 $1/6$，與第一顆完全脫鉤。

$A\cap C=\{(2,6),(4,4),(6,2)\}$，機率 $3/36=1/12$，但

$$\mathbb{P}(A)\,\mathbb{P}(C)=\frac{1}{2}\times\frac{5}{36}=\frac{5}{72}\neq\frac{1}{12}=\frac{6}{72},$$

所以 A 與 C **相依**：和為 8 排除了「第一顆是 1」的可能，偶數因此稍占便宜。兩個「長得幾乎一樣」的事件，一個獨立、一個相依——獨立性藏在算術裡，不在直覺裡。

常見陷阱

「獨立」不是「互斥」——它們幾乎是相反的概念。互斥說「不能同時發生」：一旦知道 B 發生，A 就確定不發生，$\mathbb{P}(A\mid B)=0$。若 $\mathbb{P}(A)>0$，這是資訊對機率最劇烈的改寫——互斥且機率為正的兩個事件，其實是**最強烈的相依**。文氏圖上：互斥是「兩塊不相交」，獨立是「按精確比例重疊」。另一個常犯的錯：獨立是**強而易碎**的假設——身高與體重、昨天與今天的股價、句子裡相鄰的詞，幾乎都相依；把相依當獨立會系統性算錯聯合機率，這是第 1 章「忽略重疊」錯誤的進階版。

獨立性還有一個更微妙的親戚。很多時候，兩件事「整體上」相依，但那份相依完全是**第三個變數**造成的；一旦固定第三個變數，相依就消失了。

定義 2.7　條件獨立

事件 A 與 B 在給定 C（$\mathbb{P}(C)>0$）之下**條件獨立**（conditionally independent），若

$$\mathbb{P}(A\cap B\mid C)=\mathbb{P}(A\mid C)\,\mathbb{P}(B\mid C).$$

等價地（當相關機率為正）：$\mathbb{P}(A\mid B\cap C)=\mathbb{P}(A\mid C)$——在已知 C 的世界裡，B 對 A 不再提供額外資訊。

例 2.5　冰淇淋與溺水：相關不等於因果

某濱海小鎮的資料顯示：冰淇淋銷量高的日子，溺水事故也特別多。把一年粗分為夏、冬各半（$\mathbb{P}(S)=\mathbb{P}(S^c)=0.5$，$S=$ 夏天），設 $I=$「當日冰淇淋銷量高」、$D=$「當日發生溺水事故」，且在**給定季節**之下 I 與 D 條件獨立，數值為

$$\mathbb{P}(I\mid S)=0.8,\quad \mathbb{P}(D\mid S)=0.4,\qquad \mathbb{P}(I\mid S^c)=0.2,\quad \mathbb{P}(D\mid S^c)=0.1.$$

問：I 與 D（不給定季節時）獨立嗎？

解　先算邊際機率（用全機率定理，正式敘述見 2.4 節）：

$$\mathbb{P}(I)=0.5\times 0.8+0.5\times 0.2=0.5,\qquad \mathbb{P}(D)=0.5\times 0.4+0.5\times 0.1=0.25.$$

再算聯合機率，在每個季節內用條件獨立拆開：

$$\mathbb{P}(I\cap D)=0.5\times(0.8\times 0.4)+0.5\times(0.2\times 0.1)=0.16+0.01=0.17.$$

但 $\mathbb{P}(I)\mathbb{P}(D)=0.5\times 0.25=0.125\neq 0.17$：整體來看 I 與 D **正相依**——知道冰淇淋大賣，溺水機率從 0.25 升到 $\mathbb{P}(D\mid I)=0.17/0.5=0.34$。可是冰淇淋當然不會把人拖下水：兩者的關聯完全來自共同的原因「夏天」（天熱使人買冰、也使人下水）。給定季節後，關聯煙消雲散。這種第三變數稱為**混淆變數**（confounder），而這個例子就是「相關不蘊涵因果」這句話的機率骨架。

方向也可以反過來：兩件事整體上獨立，卻在條件化之後變得相依——對「共同結果」條件化會把原本無關的原因綁在一起（練習 2.8 會用兩枚硬幣造出這個現象）。條件獨立與獨立**互不蘊涵**，判斷時必須明說「在什麼條件下」。

這如何驅動機器學習／ AI

單純貝氏（naive Bayes，又譯樸素貝氏）分類器是條件獨立假設的招牌應用。判斷垃圾郵件需要聯合概似 $\mathbb{P}(x_1,\dots,x_n\mid \text{類別})$（$x_i$ 為各字詞是否出現），但它難以估計。單純貝氏大刀一揮，假設**特徵在給定類別下條件獨立**：

$$\mathbb{P}(x_1,\dots,x_n\mid \text{類別})\approx\prod_{i=1}^{n}\mathbb{P}(x_i\mid \text{類別}),$$

每個因子數一數詞頻即可估計。這個假設幾乎必然是錯的（「free」與「winner」顯然結伴出現），所以叫 naive；但分類只需**排序正確**、不需機率精準，所以它常出奇地好用——快、省資料、難過擬合；代價是冗餘特徵被重複計分，輸出過度自信。何時信任一個明知是錯的假設，是機器學習永恆的工程判斷。

2.4　全機率定理

有些機率直接算很難，「分情況」卻很簡單。例如：你的通勤時間受天氣影響——晴天 15 分鐘內到的機率是 0.9，雨天只有 0.4；氣象預報明天降雨機率 0.3。那「明天 15 分鐘內到」的機率是多少？直覺的算法是加權平均：$0.7 \times 0.9 + 0.3 \times 0.4 = 0.75$。這個直覺就是全機率定理。

定理 2.8　全機率定理

設 $B_1, B_2, \ldots, B_n$ 為樣本空間的一個**分割**（partition）：兩兩互斥、聯集為 Ω、且每個 $\mathbb{P}(B_i) > 0$。則對任何事件 A，

$$\mathbb{P}(A) = \sum_{i=1}^{n} \mathbb{P}(A \cap B_i) = \sum_{i=1}^{n} \mathbb{P}(A \mid B_i)\,\mathbb{P}(B_i).$$

最常用的特例是二分割 $\{B, B^c\}$：$\mathbb{P}(A) = \mathbb{P}(A \mid B)\mathbb{P}(B) + \mathbb{P}(A \mid B^c)\mathbb{P}(B^c)$。

證明　分割把 A 切成互斥的碎片：$A = \bigcup_i (A \cap B_i)$，且各碎片互斥（因為 B_i 互斥）。由可加性 $\mathbb{P}(A) = \sum_i \mathbb{P}(A \cap B_i)$，再對每一項套用乘法規則 $\mathbb{P}(A \cap B_i) = \mathbb{P}(A \mid B_i)\mathbb{P}(B_i)$ 即得。 ■

一句話總結：**對每種情況條件化，再用情況的機率加權重組**。「先把難題切成好算的情境，再平均」這個模式無所不在——它就是「分情況討論」的機率版。

例 2.6　兩個甕

甕 I 裝有 3 顆紅球、2 顆白球；甕 II 裝有 1 顆紅球、4 顆白球。擲一枚公正硬幣決定從哪個甕中隨機抽一顆球。求抽到紅球的機率。

解　以「選到哪個甕」分割：$\mathbb{P}(\text{I}) = \mathbb{P}(\text{II}) = 1/2$，而每個甕內的條件機率一目瞭然：$\mathbb{P}(R \mid \text{I}) = 3/5$、$\mathbb{P}(R \mid \text{II}) = 1/5$。由全機率定理

$$\mathbb{P}(R) = \frac{1}{2} \times \frac{3}{5} + \frac{1}{2} \times \frac{1}{5} = \frac{2}{5} = 0.4.$$

注意 0.4 恰好落在 0.2 與 0.6 之間——全機率定理算出的是條件機率的**加權平均**，權重是各情境的機率。現在把問題反過來：**已知抽到紅球，它來自甕 I 的機率是多少？**「由結果反推原因」——這正是下一節的主角。

直觀理解

全機率定理是**邊際化**（marginalization）的離散版：$\mathbb{P}(A) = \sum_i \mathbb{P}(A \cap B_i)$ 把聯合機率**沿著情境變數加總**——行話叫把 B「加掉（sum out）」或「積掉（integrate out）」，第 3 章的連續版把和換成積分。它同時是貝氏定理的**分母製造機**：證據項 $\mathbb{P}(E)$ 正是各假設下概似的加權平均。

2.5　貝氏定理：反轉方向的引擎

我們常掌握「原因 → 結果」方向的機率：患病時檢測呈陽性的機率、垃圾郵件含有「free」的機率。但決策需要反方向：**看到**陽性，患病的機率？**看到**「free」，是垃圾郵件的機率？橋樑是乘法

規則的一體兩面：$\mathbb{P}(A\cap B)=\mathbb{P}(A\mid B)\mathbb{P}(B)=\mathbb{P}(B\mid A)\mathbb{P}(A)$。令兩端相等、移項，就得到以 18 世紀英國牧師 Thomas Bayes 命名的定理——形式簡單到近乎樸素，卻是整個統計推論的樞紐。

定理 2.9　貝氏定理

設 $\mathbb{P}(E)>0$。對任何事件 H，

$$\mathbb{P}(H\mid E)=\frac{\mathbb{P}(E\mid H)\,\mathbb{P}(H)}{\mathbb{P}(E)}.$$

若 $H_1,\dots,H_n$ 為互斥且窮盡的假設（分割），分母可用全機率定理展開：

$$\mathbb{P}(H_i\mid E)=\frac{\mathbb{P}(E\mid H_i)\,\mathbb{P}(H_i)}{\sum_{j=1}^{n}\mathbb{P}(E\mid H_j)\,\mathbb{P}(H_j)}.$$

證明　由乘法規則 $\mathbb{P}(H\cap E)=\mathbb{P}(E\mid H)\mathbb{P}(H)=\mathbb{P}(H\mid E)\mathbb{P}(E)$，兩端同除以 $\mathbb{P}(E)$ 即得第一式；分母以全機率定理展開即得第二式。 ■

慣例上以 H 記假設（hypothesis）、E 記證據（evidence），四個組成部分各有名字——這套詞彙將伴隨你走完整本書：

$$\underbrace{\mathbb{P}(H\mid E)}_{\text{後驗}}=\frac{\overbrace{\mathbb{P}(E\mid H)}^{\text{概似}}\times\overbrace{\mathbb{P}(H)}^{\text{先驗}}}{\underbrace{\mathbb{P}(E)}_{\text{證據}}}.$$

- **先驗**（prior）$\mathbb{P}(H)$：看到證據**之前**對假設的信念。
- **概似**（likelihood）$\mathbb{P}(E\mid H)$：**假如**假設為真，看到這份證據的機率——「正向」而容易取得的量。
- **證據**（evidence）$\mathbb{P}(E)$：證據的總機率（由全機率定理加權平均而得）；不含 H，作用只是正規化。
- **後驗**（posterior）$\mathbb{P}(H\mid E)$：看到證據**之後**更新過的信念——真正想要的答案。

核心觀念

貝氏定理是**理性信念隨證據更新**的方式，一個永不停歇的循環：先驗 → 乘上概似 → 除以證據（正規化）→ 後驗。由於分母與 H 無關，這個循環常被壓縮成一行口訣：

$$\text{後驗}\ \propto\ \text{概似}\times\text{先驗}.$$

「你原本信什麼」乘上「證據多支持它」，再調成總和為一。這一行撐起了貝氏推論、垃圾郵件過濾、醫療診斷與整個機率式機器學習觀。循環還可以疊代：**今天的後驗，就是明天的先驗**——證據一批批進來，信念一步步更新（例 2.8）。

先兌現上一節的伏筆：$\mathbb{P}(\mathrm{I}\mid R)=\dfrac{\mathbb{P}(R\mid \mathrm{I})\mathbb{P}(\mathrm{I})}{\mathbb{P}(R)}=\dfrac{(3/5)(1/2)}{2/5}=\dfrac{3}{4}$——紅球是甕 I 的招牌商品，看到紅球，信念便從五五開移向甕 I。接著是本章最重要的例子。先定義兩個檢測術語：**靈敏度**（sensitivity）$=\mathbb{P}(+\mid \text{患病})$，患病者被驗出的機率；**特異度**（specificity）$=\mathbb{P}(-\mid \text{健康})$，

健康者被正確放行的機率。

例 2.7　基率謬誤：陽性了，然後呢？

某疾病盛行率為 1%。檢測的靈敏度與特異度皆為 99%。你的檢測結果呈陽性，你患病的機率是多少？

解　直覺大喊「99%」——檢測都這麼準了。貝氏定理說：慢著，先驗呢？設 D =「患病」，$+$ =「檢測陽性」。已知 $\mathbb{P}(D)=0.01$、$\mathbb{P}(+\mid D)=0.99$、$\mathbb{P}(+\mid D^c)=1-0.99=0.01$（偽陽性率 $=1-$ 特異度）。先用全機率定理算證據：

$$\mathbb{P}(+)=\mathbb{P}(+\mid D)\mathbb{P}(D)+\mathbb{P}(+\mid D^c)\mathbb{P}(D^c)=0.99\times 0.01+0.01\times 0.99=0.0198.$$

再套貝氏定理：

$$\mathbb{P}(D\mid +)=\frac{0.99\times 0.01}{0.0198}=\frac{0.0099}{0.0198}=\frac{1}{2}.$$

只有 **50**%！癥結在於疾病太罕見：真陽性來自 1% 患病者中的 99%，偽陽性來自 99% 健康者中的 1%——兩股人馬恰好一樣多，陽性報告因此只有一半可信。用**自然頻率**（natural frequencies）重講一次，直覺立刻歸位：10,000 人受檢，患病 100 人中約 99 人陽性；健康 9,900 人中約 99 人偽陽性——陽性共 198 人，真患病恰好一半。心理學家 Gigerenzer 發現，把機率翻譯成「人數」後，多數人（包括醫師）都能答對；出錯的不是人腦的算術，是題目的表述方式。

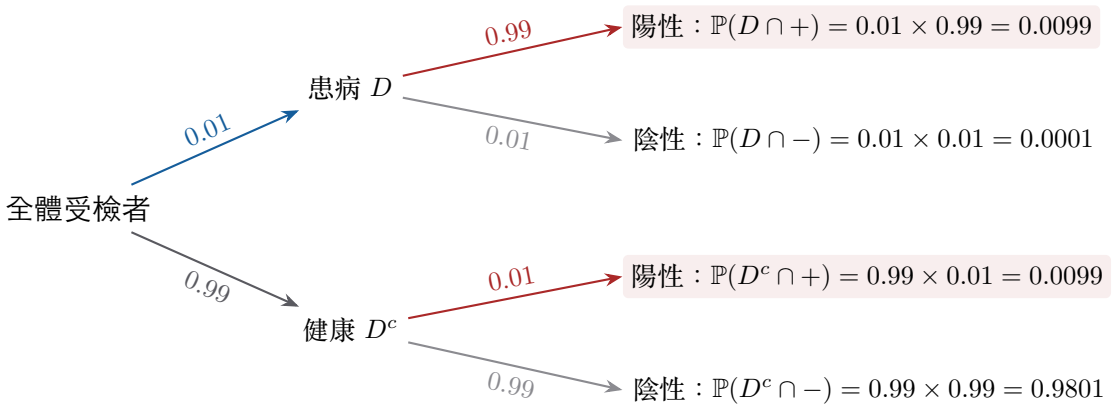


圖 2.2: 檢測問題的樹狀圖：第一層依患病與否分割（先驗），第二層是檢測結果（概似），葉端為聯合機率。兩條通往「陽性」的路徑（紅底）機率相同、各 0.0099——真陽性與偽陽性一樣多，後驗因此恰為 50%；兩片紅底相加即證據 $\mathbb{P}(+)=0.0198$。

常見陷阱

基率謬誤（base rate fallacy）：只看證據多有力（概似），忘記假設原本多罕見（先驗）——把 $\mathbb{P}(+\mid D)=0.99$ 錯當 $\mathbb{P}(D\mid +)$ 的昂貴化身。**醫學**：罕見疾病的陽性報告，直覺答案可以離正解一倍以上。**法律**：檢察官謬誤把「無辜者恰好吻合證據」的小機率，偷換成「吻合

證據者無辜」的小機率——資料庫夠大時，隨機吻合者可以多到翻案。**機器學習**：偵測罕見事件（詐欺、入侵、瑕疵）時，警報的精確率（precision）就是後驗 $\mathbb{P}(\text{真事件} \mid \text{警報})$——正類越罕見，精確率越被偽陽性稀釋；「準確率 99%」的偵測器可能毫無用處，因為全猜「沒事」的模型準確率也有 99%。

直觀理解

先驗很重要：罕見的假設，需要很強的證據才會變得可信——「非凡的主張需要非凡的證據」的定量版。檢測例子裡，證據力道是概似比 $\mathbb{P}(+ \mid D)/\mathbb{P}(+ \mid D^c) = 99$ 倍，但先驗是 $1:99$ 的劣勢，兩相抵銷、後驗打平於 50%；想到 99%，需要近萬倍的概似比，或兩次獨立的證據（例 2.8）。這解釋了機器學習的兩個日常災難：**類別不平衡**讓分類器學到「永遠猜多數類」這種高準確率、零價值的策略；**異常偵測**若忽視基率，警報清單會被偽陽性淹沒。處方——調閾值、重加權、代價敏感學習——本質都是**把先驗放回決策裡**。

例 2.8　序貫貝氏更新：證據會累積

延續例 2.7：第一次檢測陽性後，你患病的機率是 0.5。假設在給定健康狀態下各次檢測**條件獨立**（同一台儀器不會「記得」上一次的結果），你再做一次獨立檢測，又是陽性。現在患病的機率是多少？

解　貝氏更新的美妙之處：**把上一輪的後驗當成這一輪的先驗**，公式原封不動再跑一次。這次 $\mathbb{P}(D) = 0.5$：

$$\mathbb{P}(D \mid ++) = \frac{0.99 \times 0.5}{0.99 \times 0.5 + 0.01 \times 0.5} = \frac{0.495}{0.500} = 0.99.$$

兩次陽性把信念從 1% 推到 50%、再推到 99%。一次算完也行：對複合證據「連續兩次陽性」，概似為 $\mathbb{P}(++ \mid D) = 0.99^2$、$\mathbb{P}(++ \mid D^c) = 0.01^2$（條件獨立在此登場），代入貝氏定理得 $0.009801/0.009900 = 0.99$，與逐步更新完全一致——貝氏更新對「證據一次進來或分批進來」無動於衷。

另解（勝算形式）：定義事件的**勝算**（odds）為 $\mathbb{P}(D)/\mathbb{P}(D^c)$。把貝氏定理對 D 與 D^c 各寫一次、相除，證據項對消，得到

$$\underbrace{\frac{\mathbb{P}(D \mid E)}{\mathbb{P}(D^c \mid E)}}_{\text{後驗勝算}} = \underbrace{\frac{\mathbb{P}(E \mid D)}{\mathbb{P}(E \mid D^c)}}_{\text{概似比}} \times \underbrace{\frac{\mathbb{P}(D)}{\mathbb{P}(D^c)}}_{\text{先驗勝算}}.$$

每次陽性就是把勝算乘上概似比 99：先驗勝算 $1:99$，一次陽性後 $1:1$（即 50%），兩次後 $99:1$（即 99%），三次後 $9801:1$（約 99.99%）。乘法接力，不需要每次重算分母——垃圾郵件過濾器就是這樣把幾十個字詞的證據一路乘上去的。

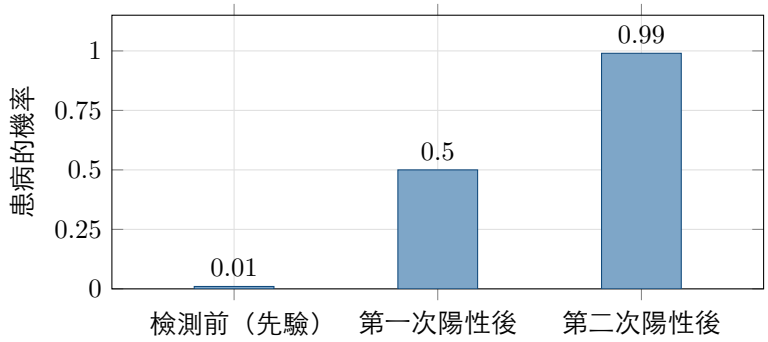


圖 2.3: 貝氏序貫更新：連續兩次陽性把患病機率從 0.01 推到 0.50、再到 0.99。每一步都是「後驗變先驗、乘概似、正規化」——證據不會一錘定音，但會不斷累積。

例 2.9　蒙提霍爾問題：換門還是不換？

遊戲節目中有三扇門，一扇後面是汽車，另兩扇是山羊，位置等可能。你選了 1 號門。主持人**知道**汽車在哪；他必定打開一扇「你沒選、且後面是山羊」的門（若兩扇都符合，等機率隨機挑一扇）。主持人打開 3 號門，露出山羊，並問你要不要改選 2 號門。換，還是不換？

解　**法一（快速論證）**：不換門獲勝 $\iff$ 你最初就選中汽車，機率 $1/3$；換門獲勝 $\iff$ 你最初選錯（此時主持人被迫把另一扇山羊門排除，剩下那扇必是汽車），機率 $2/3$。換門讓勝率加倍。

法二（貝氏完整分析）：設 $C_i =$「汽車在 i 號門後」，先驗各 $1/3$。證據 $E =$「主持人打開 3 號門」。逐一計算概似：若車在 1 號門（你的門），2、3 號皆山羊，主持人隨機挑，開 3 號的機率是 $1/2$；若車在 2 號門，他只能開 3 號；若車在 3 號門，他絕不會開：

$$\mathbb{P}(E \mid C_1) = \frac{1}{2}, \qquad \mathbb{P}(E \mid C_2) = 1, \qquad \mathbb{P}(E \mid C_3) = 0.$$

證據項：$\mathbb{P}(E) = \frac{1}{3}\cdot\frac{1}{2} + \frac{1}{3}\cdot 1 + \frac{1}{3}\cdot 0 = \frac{1}{2}$。後驗：

$$\mathbb{P}(C_1 \mid E) = \frac{(1/2)(1/3)}{1/2} = \frac{1}{3}, \qquad \mathbb{P}(C_2 \mid E) = \frac{(1)(1/3)}{1/2} = \frac{2}{3}, \qquad \mathbb{P}(C_3 \mid E) = 0.$$

你的門停在 $1/3$，被主持人「跳過」的 2 號門吸走了 3 號門釋出的全部機率。該換。

變體（健忘的主持人）：若主持人**不知道**汽車在哪，隨手開了 3 號門、**碰巧**是山羊，證據變成 $E' =$「開 3 號門且是山羊」，概似成為 $\mathbb{P}(E' \mid C_1) = 1/2$、$\mathbb{P}(E' \mid C_2) = 1/2$、$\mathbb{P}(E' \mid C_3) = 0$，後驗是 $1/2$、$1/2$——換不換都一樣！你看到的畫面一模一樣（3 號門開了、是山羊），後驗卻不同，因為**資訊的產生機制不同**：知情主持人的開門被規則逼出、攜帶著汽車位置的資訊，健忘主持人的開門只是運氣。教訓：條件化的對象是「你如何知道」的完整事件，不只是「你知道的事實」。

這如何驅動機器學習／ AI

把貝氏定理裡的 H 換成**模型參數** θ、E 換成**資料** D，就得到機率式機器學習的總綱：

$$\mathbb{P}(\theta \mid D) \;\propto\; \mathbb{P}(D \mid \theta)\,\mathbb{P}(\theta).$$

三條路從這裡分岔。**(1) 只最大化概似** $\mathbb{P}(D \mid \theta)$：**最大概似估計**（maximum likelihood estimation, MLE，第 7 章）——分類的交叉熵損失、迴歸的平方誤差損失，都是「負對數概似」換裝登場。**(2) 最大化概似 × 先驗**：**最大後驗估計**（maximum a posteriori, MAP，第 9 章）——先驗（例如「權重不要太大」）取對數後就是正則化項，權重衰減（weight decay）正是高斯先驗的化身。**(3) 保留整個後驗分布**：完整的貝氏推論，不只給點估計，還給不確定性。配上 2.2 節的鏈式法則（GPT 逐詞元分解句子機率），本章兩條公式撐起現代 AI 的大半骨架——**機器學習 = 推論**，從本章開始有了精確的意思。

2.6 Python 實作

理論說「陽性後只有五成」，程式來驗證。以下先把貝氏更新寫成函數，重現基率謬誤與序貫更新；再掃過不同盛行率，看先驗如何左右後驗。

```python
# --- 1. 貝氏更新：一次陽性、兩次陽性（序貫更新） ---
def bayes_update(prior, sens, spec):
    """輸入先驗機率，回傳「檢測陽性」之後的後驗機率"""
    pos_if_sick    = sens                # P(+ | 患病) = 靈敏度
    pos_if_healthy = 1 - spec            # P(+ | 健康) = 偽陽性率
    evidence = pos_if_sick * prior + pos_if_healthy * (1 - prior)
    return pos_if_sick * prior / evidence

p1 = bayes_update(0.01, 0.99, 0.99)      # 第一次陽性後
p2 = bayes_update(p1, 0.99, 0.99)        # 第二次陽性後（後驗變先驗）
print(round(p1, 4), round(p2, 4))        # -> 0.5 0.99

# --- 2. 先驗（盛行率）如何左右後驗 ---
for prev in [0.001, 0.01, 0.1, 0.5]:
    post = bayes_update(prev, 0.99, 0.99)
    print(f"盛行率 {prev:>5}: 陽性後患病機率 {post:.4f}")
# -> 0.001: 0.0902 / 0.01: 0.5000 / 0.1: 0.9167 / 0.5: 0.9900
```

同一個檢測，後驗從 9% 一路變到 99%——變的從來不是檢測，是先驗。接著用蒙地卡羅模擬直接「數人頭」驗證理論值，順便重現蒙提霍爾問題與鏈式法則的計算（固定亂數種子以利重現）。

```python
import numpy as np
rng = np.random.default_rng(42)
```

```python

# --- 3. 蒙地卡羅驗證基率謬誤：一百萬人接受檢測 ---
N = 1_000_000
sick = rng.random(N) < 0.01              # 1% 的人真的患病
r = rng.random(N)
pos = np.where(sick, r < 0.99, r < 0.01) # 患病 99% 陽性；健康 1% 偽陽性
print(round(sick[pos].mean(), 4))        # 陽性者中患病比例 -> 約 0.50

# --- 4. 蒙提霍爾問題：不換門 vs. 換門 ---
M = 100_000
car  = rng.integers(0, 3, size=M)        # 汽車藏在哪扇門後
pick = rng.integers(0, 3, size=M)        # 玩家最初的選擇
print(round(np.mean(car == pick), 3))    # 不換門勝率 -> 約 0.333
print(round(np.mean(car != pick), 3))    # 換門勝率   -> 約 0.667

# --- 5. 鏈式法則驗證：連抽三張不放回、張張紅心 ---
decks = np.tile(np.arange(52), (100_000, 1))
draws = rng.permuted(decks, axis=1)[:, :3]    # 每副牌洗勻後抽前三張
p_sim = np.all(draws < 13, axis=1).mean()     # 編號 0..12 代表紅心
print(round(p_sim, 4), round(11/850, 4))      # -> 約 0.0129 vs 0.0129
```

三點觀察：模擬的 $\mathbb{P}(\text{患病} \mid \text{陽性}) \approx 0.50$ 與貝氏定理一致，誤差來自有限樣本的隨機波動；蒙提霍爾的模擬用了「換門獲勝若且唯若最初選錯」的事實，故不必模擬主持人（練習 2.10 會老老實實模擬，並驗證條件機率版本的答案）；鏈式法則乘積 $11/850$ 與洗牌模擬吻合。理論給精確值，模擬給信心——兩邊都會，才算真的懂。

本章總結

- **條件機率** $\mathbb{P}(A \mid B) = \mathbb{P}(A \cap B)/\mathbb{P}(B)$：把 B 變成新宇宙（分子），再重新正規化（分母）——這是機率吸收證據的方式。切記 $\mathbb{P}(A \mid B) \neq \mathbb{P}(B \mid A)$，混用即是檢察官謬誤。
- **乘法規則** $\mathbb{P}(A \cap B) = \mathbb{P}(A \mid B)\mathbb{P}(B)$ 把視角反轉成「用條件建構聯合」；**鏈式法則**把任何聯合機率拆成條件機率的連乘——自迴歸語言模型逐詞元生成句子用的正是它，訓練目標即交叉熵。
- **獨立** $\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B)$ 表示資訊不改變機率；**互斥**反而是最強的相依。**條件獨立**與獨立互不蘊涵：條件化可以消滅相依（冰淇淋與溺水），也可以製造相依。單純貝氏靠條件獨立假設換取簡單與效率。
- **全機率定理** $\mathbb{P}(A) = \sum_i \mathbb{P}(A \mid B_i)\mathbb{P}(B_i)$：分情況、加權、重組——邊際化的離散版，也是貝氏定理分母（證據項）的計算引擎。
- **貝氏定理**：後驗 $\propto$ 概似 $\times$ 先驗。**基率謬誤**警告先驗不可忽略——罕見假設需要很強

的證據（盛行率 1%、準確率 99% 的檢測，陽性後也只有五成）。證據會累積：後驗變先驗，勝算乘上概似比，兩次陽性即達 99%。

- 把 H 換成參數 θ、E 換成資料 D：$\mathbb{P}(\theta \mid D) \propto \mathbb{P}(D \mid \theta)\mathbb{P}(\theta)$。只取概似是 **MLE**（交叉熵與平方誤差的源頭），加上先驗是 **MAP**（正則化的源頭）——「機器學習 = 推論」由此起步。

2.7 練習題

建議先獨立完成再對照詳解。題目依難度標示為**基礎**、**進階**、**挑戰**三級；標示「程式」者需要動手寫程式。

練習 2.1 (基礎)　擲兩顆公正骰子。(a) 已知點數和為 6，求「至少一顆是 2」的機率。(b) 已知「至少一顆是 2」，求點數和為 6 的機率。(c) 兩個答案為何不同？這說明了條件機率的什麼性質？

練習 2.2 (基礎)　從一副 52 張撲克牌連抽兩張、不放回。(a) 求兩張都是 A 的機率。(b) 求「第一張是 A 且第二張是 K」的機率。(c) 用全機率定理求「第二張是 A」的機率，並解釋為什麼答案恰好等於 $\mathbb{P}(\text{第一張是 A}) = 1/13$。

練習 2.3 (基礎)　(a) 擲兩顆公正骰子，設 $A =$「第一顆是奇數」、$B =$「點數和為奇數」。判斷 A 與 B 是否獨立（請用定義驗算）。(b) 證明：若 A 與 B 獨立，則 A 與 B^c 也獨立。(c) 進一步證明 A^c 與 B^c 也獨立。

練習 2.4 (基礎)　某工廠有三台機器：甲機台佔總產量 50%、瑕疵率 1%；乙機台佔 30%、瑕疵率 2%；丙機台佔 20%、瑕疵率 5%。隨機抽驗一件產品。(a) 求它是瑕疵品的機率。(b) 已知它是瑕疵品，分別求它出自甲、乙、丙機台的機率。(c) 若要優先檢修一台機器，該選哪台？為什麼？

練習 2.5 (基礎)　沿用例 2.7 的檢測（靈敏度 99%、特異度 99%），改變受檢族群：(a) 高風險族群，盛行率 10%；(b) 一般健檢族群，盛行率 0.1%。分別求檢測陽性後患病的機率。(c) 連同例 2.7 的 1% 情境，比較三個答案，說明「同一個檢測，為何意義天差地遠」，並指出這對機器學習模型部署（訓練分布與部署分布不同）的警示。

練習 2.6 (進階)　已知 40% 的郵件是垃圾郵件。「free」出現在 25% 的垃圾郵件與 2% 的正常郵件中。(a) 求任一封郵件出現「free」的機率。(b) 已知某郵件出現「free」，求它是垃圾郵件的機率。(c) 另一詞「winner」出現在 15% 的垃圾郵件與 1% 的正常郵件中。假設兩詞在給定郵件類別下條件獨立（單純貝氏假設），求「free」與「winner」同時出現時，該郵件

是垃圾郵件的機率。

練習 2.7 (進階)　**計程車問題**：某城市 85% 的計程車是綠色、15% 是藍色。一輛計程車夜間肇事逃逸。目擊者指認肇事車是**藍色**；測試顯示該目擊者在同樣條件下辨色的正確率為 80%（藍車說藍、綠車說綠的機率皆為 0.8）。求肇事車確為藍色的機率，並評論「目擊者八成可靠，所以八成是藍車」這句話錯在哪裡。

練習 2.8 (進階)　擲兩枚公正硬幣，設 $A=$「第一枚正面」、$B=$「第二枚正面」、$C=$「兩枚結果相同」。(a) 驗證 A、B、C 兩兩獨立。(b) 驗證 A 與 B 在給定 C 之下**不**條件獨立。(c) 驗證 $\mathbb{P}(A\cap B\cap C)\neq\mathbb{P}(A)\mathbb{P}(B)\mathbb{P}(C)$，即三事件並非相互獨立。(d) 本題與例 2.5 的冰淇淋例子方向相反——請用一句話總結「條件化對獨立性的影響」。

練習 2.9 (挑戰)　(a) 用數學歸納法證明鏈式法則：

$$\mathbb{P}(A_1\cap\cdots\cap A_n)=\mathbb{P}(A_1)\prod_{k=2}^{n}\mathbb{P}(A_k\mid A_1\cap\cdots\cap A_{k-1}).$$

(b) 說明為何只需假設 $\mathbb{P}(A_1\cap\cdots\cap A_{n-1})>0$，式中所有條件機率就都有定義。(c) 寫出 $n=4$ 的展開式，並用「連抽四張不放回、張張紅心」驗證：分別以鏈式法則與組合計數計算，確認兩者相等。

練習 2.10 (挑戰 · 程式)　(a) 用 NumPy 模擬蒙提霍爾問題 3×10^5 次：**明確模擬主持人**的開門規則（不開玩家選的門、不開有汽車的門、兩扇皆可開時等機率挑一扇），驗證「不換門」與「換門」的獲勝率分別約為 1/3 與 2/3；再僅挑出「主持人開了 2 號門」的場次，驗證 $\mathbb{P}(\text{汽車在另一扇門}\mid\text{主持人開 2 號門})\approx2/3$。(b) 模擬 10^6 人接受例 2.7 的檢測（盛行率 1%、靈敏度 99%、特異度 99%），驗證陽性者中患病比例約為 0.5。

2.8　練習題詳解

練習 2.1 詳解　(a) 和為 6 的結果：$B=\{(1,5),(2,4),(3,3),(4,2),(5,1)\}$，共 5 個等可能結果。其中至少一顆是 2 的有 $(2,4)$ 與 $(4,2)$，故

$$\mathbb{P}(\text{至少一顆是 2}\mid\text{和為 6})=\frac{2}{5}=0.4.$$

(b)「至少一顆是 2」的結果數：第一顆是 2 有 6 個，第二顆是 2 有 6 個，扣掉重複計算的 $(2,2)$，共 $6+6-1=11$ 個。其中和為 6 的仍是 $(2,4),(4,2)$ 兩個，故

$$\mathbb{P}(\text{和為 6}\mid\text{至少一顆是 2})=\frac{2}{11}\approx0.182.$$

(c) 分子相同（交集不變），分母不同：條件事件的「新宇宙」大小不一樣（5 個結果對 11 個結果）。這正說明 $\mathbb{P}(A\mid B)\neq\mathbb{P}(B\mid A)$——條件機率對「誰當條件」極其敏感，兩個方向的

量回答的是不同的問題。

練習 2.2 詳解　(a) 乘法規則：

$$\mathbb{P}(\text{兩張皆 A}) = \frac{4}{52} \times \frac{3}{51} = \frac{12}{2652} = \frac{1}{221} \approx 0.0045.$$

第一張抽走一張 A 後，剩 51 張中只有 3 張 A。

(b)

$$\mathbb{P}(\text{A 後 K}) = \frac{4}{52} \times \frac{4}{51} = \frac{16}{2652} = \frac{4}{663} \approx 0.0060,$$

略大於 (a)：第一張 A 沒有消耗任何 K，第二步的條件機率是 $4/51$ 而非 $3/51$。

(c) 以第一張是否為 A 分割，用全機率定理：

$$\mathbb{P}(A_2) = \mathbb{P}(A_2 \mid A_1)\mathbb{P}(A_1) + \mathbb{P}(A_2 \mid A_1^c)\mathbb{P}(A_1^c) = \frac{3}{51}\cdot\frac{4}{52} + \frac{4}{51}\cdot\frac{48}{52} = \frac{12+192}{2652} = \frac{204}{2652} = \frac{1}{13}.$$

恰等於第一張是 A 的機率。原因是**對稱性**：洗勻的牌堆中，每張牌落在「第二個位置」的機會完全均等，於是在**不知道**第一張是什麼的情況下，第二張的分布與第一張相同。「不放回會改變機率」只在你**看到**第一張之後才成立——相依是透過資訊起作用的。

練習 2.3 詳解　(a) $\mathbb{P}(A) = 1/2$；和為奇數需一奇一偶，$\mathbb{P}(B) = 18/36 = 1/2$。交集「第一顆奇數且和為奇數」等價於「第一顆奇數且第二顆偶數」，共 $3 \times 3 = 9$ 個結果：

$$\mathbb{P}(A \cap B) = \frac{9}{36} = \frac{1}{4} = \frac{1}{2} \times \frac{1}{2} = \mathbb{P}(A)\mathbb{P}(B),$$

故 $A \perp B$。（值得玩味：B 同時牽涉兩顆骰子，卻與 A 獨立——不論第一顆奇偶，第二顆讓和變奇數的機率都是 $1/2$。）

(b) 把 A 拆成互斥兩塊：$A = (A \cap B) \cup (A \cap B^c)$，故

$$\mathbb{P}(A \cap B^c) = \mathbb{P}(A) - \mathbb{P}(A \cap B) = \mathbb{P}(A) - \mathbb{P}(A)\mathbb{P}(B) = \mathbb{P}(A)\big(1 - \mathbb{P}(B)\big) = \mathbb{P}(A)\mathbb{P}(B^c),$$

即 $A \perp B^c$。

(c) 對 (b) 的結論「$A \perp B^c$」再套用一次 (b)（把 B^c 視為新的「A」、A 視為新的「B」，利用獨立的對稱性）：$B^c \perp A \Rightarrow B^c \perp A^c$，即 $A^c \perp B^c$。 ■

練習 2.4 詳解　設 $F =$「瑕疵品」，以出產機台 $\{$ 甲, 乙, 丙 $\}$ 分割。

(a) 全機率定理：

$$\mathbb{P}(F) = 0.5 \times 0.01 + 0.3 \times 0.02 + 0.2 \times 0.05 = 0.005 + 0.006 + 0.010 = 0.021,$$

即 2.1%。

(b) 貝氏定理，分母即 (a)：

$$\mathbb{P}(\text{甲} \mid F) = \frac{0.005}{0.021} = \frac{5}{21} \approx 0.238, \qquad \mathbb{P}(\text{乙} \mid F) = \frac{0.006}{0.021} = \frac{2}{7} \approx 0.286,$$

$$\mathbb{P}(\text{丙} \mid F) = \frac{0.010}{0.021} = \frac{10}{21} \approx 0.476.$$

（三者相加為 1，可作驗算。）

(c) 應優先檢修**丙**：它只佔兩成產量，卻要為將近一半（47.6%）的瑕疵品負責——後驗機率把「產量占比（先驗）」與「瑕疵率（概似）」兩股力量正確地合成，單看其中一項都會誤判。

練習 2.5 詳解　偽陽性率 $= 1 -$ 特異度 $= 0.01$。

(a) 盛行率 10%：

$$\mathbb{P}(D \mid +) = \frac{0.99 \times 0.10}{0.99 \times 0.10 + 0.01 \times 0.90} = \frac{0.099}{0.108} = \frac{11}{12} \approx 0.917.$$

(b) 盛行率 0.1%：

$$\mathbb{P}(D \mid +) = \frac{0.99 \times 0.001}{0.99 \times 0.001 + 0.01 \times 0.999} = \frac{0.00099}{0.01098} = \frac{11}{122} \approx 0.090.$$

(c) 同一個檢測，後驗依先驗不同而為 9%、50%（例 2.7）、92%：報告的意義不在檢測本身，而在**它被用在誰身上**——這正是「不對無症狀低風險族群濫行篩檢」的數學理由。對機器學習的警示：模型（概似）不變，**部署環境的基率變了，預測的可信度就全變**。在正類 10% 的資料上校準的偵測模型，搬到正類僅 0.1% 的真實流量上，精確率會崩跌——部署前必須重新校準或調整閾值，把新環境的先驗放回來。

練習 2.6 詳解　設 $S =$「垃圾郵件」、$F =$「出現 free」、$W =$「出現 winner」。已知 $\mathbb{P}(S) = 0.4$、$\mathbb{P}(F \mid S) = 0.25$、$\mathbb{P}(F \mid S^c) = 0.02$。

(a) 全機率定理：

$$\mathbb{P}(F) = 0.25 \times 0.4 + 0.02 \times 0.6 = 0.10 + 0.012 = 0.112.$$

(b) 貝氏定理：

$$\mathbb{P}(S \mid F) = \frac{0.25 \times 0.4}{0.112} = \frac{0.10}{0.112} = \frac{25}{28} \approx 0.893.$$

(c) 單純貝氏假設下，$\mathbb{P}(F \cap W \mid S) = 0.25 \times 0.15$、$\mathbb{P}(F \cap W \mid S^c) = 0.02 \times 0.01$。由「後驗 $\propto$ 概似 $\times$ 先驗」：

$$\mathbb{P}(S \mid F \cap W) = \frac{0.4 \times 0.25 \times 0.15}{0.4 \times 0.25 \times 0.15 + 0.6 \times 0.02 \times 0.01} = \frac{0.015}{0.015 + 0.00012} = \frac{0.015}{0.01512} \approx 0.992.$$

用勝算形式驗算更漂亮：先驗勝算 $0.4/0.6 = 2/3$，兩個概似比分別為 $0.25/0.02 = 12.5$ 與 $0.15/0.01 = 15$，後驗勝算 $= \frac{2}{3} \times 12.5 \times 15 = 125$，故後驗 $= 125/126 \approx 0.992$，一致。每個證據詞都是一次「勝算乘法」——這正是垃圾郵件過濾器的實際運作方式。

練習 2.7 詳解　設 $B=$「肇事車是藍色」、$W_B=$「證人說藍色」。已知 $\mathbb{P}(B)=0.15$、$\mathbb{P}(W_B\mid B)=0.8$、$\mathbb{P}(W_B\mid B^c)=0.2$。貝氏定理：

$$\mathbb{P}(B\mid W_B)=\frac{0.8\times 0.15}{0.8\times 0.15+0.2\times 0.85}=\frac{0.12}{0.12+0.17}=\frac{12}{29}\approx 0.414.$$

儘管證人指認藍色，肇事車更可能是**綠色**（機率約 58.6%）。「八成可靠，所以八成是藍車」錯在把概似 $\mathbb{P}(W_B\mid B)$ 直接當成後驗 $\mathbb{P}(B\mid W_B)$，完全忽略了 85 : 15 的懸殊基率：綠車實在太多，即使只有兩成的誤認率，被誤認成藍色的綠車（$0.85\times 0.2=0.17$）仍多於被正確指認的藍車（$0.15\times 0.8=0.12$）。這是 Kahneman 與 Tversky 的經典實驗題——多數受試者的直覺答案正是 80%，活生生的基率謬誤。

練習 2.8 詳解　樣本空間 $\{\mathrm{HH},\mathrm{HT},\mathrm{TH},\mathrm{TT}\}$ 各 $1/4$；$\mathbb{P}(A)=\mathbb{P}(B)=\mathbb{P}(C)=1/2$（$C=\{\mathrm{HH},\mathrm{TT}\}$）。

(a) $\mathbb{P}(A\cap B)=\mathbb{P}(\mathrm{HH})=1/4=\mathbb{P}(A)\mathbb{P}(B)$，故 $A\perp B$。$\mathbb{P}(A\cap C)=\mathbb{P}(\mathrm{HH})=1/4=\mathbb{P}(A)\mathbb{P}(C)$，故 $A\perp C$；同理 $B\perp C$。三事件**兩兩獨立**。

(b) $\mathbb{P}(A\mid C)=\mathbb{P}(B\mid C)=\frac{1/4}{1/2}=\frac{1}{2}$，但

$$\mathbb{P}(A\cap B\mid C)=\frac{\mathbb{P}(\mathrm{HH})}{\mathbb{P}(C)}=\frac{1/4}{1/2}=\frac{1}{2}\neq\frac{1}{4}=\mathbb{P}(A\mid C)\mathbb{P}(B\mid C),$$

故給定 C 之後 A 與 B **相依**——知道「兩枚相同」後，第一枚是正面就**迫使**第二枚也是正面。

(c) $\mathbb{P}(A\cap B\cap C)=\mathbb{P}(\mathrm{HH})=\frac{1}{4}\neq\frac{1}{8}=\mathbb{P}(A)\mathbb{P}(B)\mathbb{P}(C)$。兩兩獨立不蘊涵相互獨立（mutual independence）——這是一個經典反例。

(d) **條件化可以消滅相依，也可以製造相依。**例 2.5 中，對共同**原因**（季節）條件化消除了冰淇淋與溺水的關聯；本題中，對共同**結果**（C 是 A,B 的函數）條件化，反而把原本獨立的兩枚硬幣綁在一起（貝氏網路稱此為對撞結構的 explaining away 效應）。獨立性永遠是「相對於當前資訊」的陳述。

練習 2.9 詳解　(a) 對 n 歸納。基底 $n=2$ 即乘法規則 $\mathbb{P}(A_1\cap A_2)=\mathbb{P}(A_1)\mathbb{P}(A_2\mid A_1)$。設命題對 $n-1$ 成立。令 $B=A_1\cap\cdots\cap A_{n-1}$（依假設 $\mathbb{P}(B)>0$），由乘法規則

$$\mathbb{P}(A_1\cap\cdots\cap A_n)=\mathbb{P}(B\cap A_n)=\mathbb{P}(B)\,\mathbb{P}(A_n\mid B).$$

再以歸納假設展開 $\mathbb{P}(B)=\mathbb{P}(A_1)\prod_{k=2}^{n-1}\mathbb{P}(A_k\mid A_1\cap\cdots\cap A_{k-1})$，代入即得 n 的情形。　■

(b) 對每個 $k\le n-1$，交更多集合只會更小：$A_1\cap\cdots\cap A_{n-1}\subseteq A_1\cap\cdots\cap A_k$。由機率的單調性

$$\mathbb{P}(A_1\cap\cdots\cap A_k)\ge\mathbb{P}(A_1\cap\cdots\cap A_{n-1})>0,$$

故每個條件機率的分母皆為正、皆有定義——一個假設保全整條公式。

(c) $n=4$：

$$\mathbb{P}(A_1\cap A_2\cap A_3\cap A_4)=\mathbb{P}(A_1)\,\mathbb{P}(A_2\mid A_1)\,\mathbb{P}(A_3\mid A_1\cap A_2)\,\mathbb{P}(A_4\mid A_1\cap A_2\cap A_3).$$

四張皆紅心：鏈式法則給

$$\frac{13}{52}\times\frac{12}{51}\times\frac{11}{50}\times\frac{10}{49}=\frac{17160}{6497400}=\frac{11}{4165}\approx 0.00264;$$

組合計數給

$$\binom{13}{4}\Big/\binom{52}{4}=\frac{715}{270725}=\frac{11}{4165},$$

兩者相等，驗證完成。

練習 2.10 詳解　參考程式與輸出（0、1、2 號門；不失一般性令玩家永遠先選 0 號門）：

```python
import numpy as np
rng = np.random.default_rng(0)
# (a) 蒙提霍爾：明確模擬主持人的規則
N = 300_000
car  = rng.integers(0, 3, size=N)          # 汽車位置
pick = np.zeros(N, dtype=int)              # 玩家永遠先選 0 號門
coin = rng.integers(1, 3, size=N)          # 車在 0 號門時隨機開 1 或 2
host = np.where(car == 0, coin, 3 - car)   # 車在 1 開 2；車在 2 開 1
switch = 3 - host                          # 換到另一扇沒開的門
print(round(np.mean(car == pick), 3))      # 不換 -> 約 0.333
print(round(np.mean(car == switch), 3))    # 換門 -> 約 0.667
m = host == 2                              # 只看「主持人開 2 號門」的場次
print(round(np.mean(car[m] == 1), 3))      # P(車在 1 號 | 開 2 號) -> 約 0.667

# (b) 檢測問題：盛行率 1%、靈敏度 99%、特異度 99%
K = 1_000_000
sick = rng.random(K) < 0.01
r = rng.random(K)
pos = np.where(sick, r < 0.99, r < 0.01)
print(round(sick[pos].mean(), 3))          # P(患病 | 陽性) -> 約 0.5
```

說明：(a) 輸出依序約 0.333、0.667、0.667；第三個數限制在「主持人開 2 號門」的場次，直接驗證例 2.9 法二的條件後驗 2/3。(b) 輸出約 0.5：真陽性與偽陽性各約 9900 人，與貝氏定理一致。

下一章：隨機變數與機率分布——把隨機結果編碼成數字，讓期望值與變異數登場。