機率與統計：從入門到精通

第 13 章　機率與統計在機器學習、深度學習與 AI 中的角色

Probability and Statistics in ML, DL, and AI

難度：總整合

一切在此匯流。前十二章的零件——擲硬幣、極限定理、後驗、KL 散度——將在本章組裝成一部完整的機器：**機器學習就是規模化的應用機率與統計**。它假設資料背後有一個機率模型、用（帶懲罰的）最大概似估計參數、量化預測的不確定性、並宣稱能從樣本泛化到母體。線性代數與微積分給了我們「怎麼算」的工具；機率與統計給了我們「這些計算為什麼是對的」理由。本章要展示：任何模型的訓練——從估計一枚硬幣的偏差，到訓練一個擴散 transformer——都是同一個統計想法的重複。

學習目標

讀完本章後，你將能夠：

- 把任何監督式學習的訓練流程改寫成**統計推論**：模型指定概似，訓練是最大概似（MLE）點估計或貝氏後驗，預測靠最大化或邊際化。
- 從雜訊模型出發**推導**對應的損失函數（均方誤差、交叉熵、平均絕對誤差、卜瓦松損失），並解釋「最小化損失 = 最大化概似 = 最小化 KL 散度」這條等式鏈。
- 把常見的正則化技巧（權重衰減、lasso、dropout、label smoothing、early stopping）解讀為**參數的先驗信念**，並完整推導「權重衰減 = 高斯先驗的 MAP」。
- 用偏差–變異數權衡、極限定理、交叉驗證與假設檢定，評估模型的**泛化**與 benchmark 比較是否誠實。
- 區分**偶然**（aleatoric）與**認知**（epistemic）不確定性，計算期望校準誤差（ECE），並說明溫度縮放、深度集成等校準方法的原理。
- 以本書的機率概念解構四大**生成模型**家族：自迴歸、VAE、擴散模型、GAN。

13.1　學習就是統計推論

把任何深度學習教學影片靜音重播，你會看到同一組動作反覆出現：定義模型、選損失、加正則化、跑梯度下降、用驗證集調參、報告測試分數。本章的任務是逐一揭露：**每個動作都是統計學**

的化名。第一步，先把「模型」與「訓練」翻譯成機率的語言。

定義 13.1　機率式模型、訓練與預測

一個**機率式機器學習模型**（probabilistic model）是一族由參數 θ 索引的條件分布：$p(y \mid x, \theta)$（判別式：給輸入預測輸出）或 $p(x \mid \theta)$（生成式：直接建模資料）。給定 $\overset{\text{i.i.d.}}{\sim}$ 資料集 $D = \{(x_i, y_i)\}_{i=1}^{n}$，

- **訓練**（training）= 從 D 估計 θ：頻率式取點估計 $\hat{\theta}_{\text{MLE}} = \arg\max_\theta p(D \mid \theta)$；貝氏式求整個後驗 $p(\theta \mid D) \propto p(D \mid \theta)\, p(\theta)$（第 7、9 章）。
- **預測**（prediction）= 使用估計：點估計把 $\hat{\theta}$ 代入 $p(y \mid x^*, \hat{\theta})$ 後取 $\arg\max_y$ 或期望值；貝氏式對參數**邊際化**，得後驗預測分布 $p(y \mid x^*, D) = \int p(y \mid x^*, \theta)\, p(\theta \mid D)\, \mathrm{d}\theta$。

注意分類器的 softmax 輸出天生滿足第 1 章的公理：每項非負、總和為一——它是「類別」這個樣本空間上的合法機率分布。深度網路無論多深，最後一層做的事情始終是「輸出一個分布」。

核心觀念

兩種訓練哲學、同一個概似。本書的兩種估計學派直接對應訓練的兩種觀點：

$$\hat{\theta}_{\text{MLE}} = \arg\max_\theta p(D \mid \theta) \qquad \text{vs.} \qquad p(\theta \mid D) \propto p(D \mid \theta)\, p(\theta).$$

左邊是**頻率式點估計**——挑出「最會解釋資料」的那組參數，深度學習的絕大多數屬於此類；右邊是**貝氏後驗**——保留所有可能參數的加權信念，是不確定性感知（uncertainty-aware）機器學習的基礎。兩者都從同一個**概似**（likelihood）出發：第 7 章與第 9 章是同一枚硬幣的兩面。

全書最小的「機器學習模型」足以看清這件事：擲一枚偏差未知的硬幣 n 次得 k 次正面，概似 $L(\theta) = \theta^k (1-\theta)^{n-k}$ 在 $\hat{\theta} = k/n$ 處最大（第 7 章）——「數比例」這個直覺動作，其實是在爬概似的山坡。換成 GPT：訓練目標是 $\arg\max_\theta \sum_t \log p(x_t \mid x_{<t}, \theta)$——同一個 $\arg\max$、同一個對數概似，只是白努利換成上兆個串接的條件 softmax、解方程式換成隨機梯度下降。**從硬幣估計器到擴散 transformer，訓練的統計本質從未改變。**

順帶把訓練迴圈翻譯完畢：`forward pass` 計算這批資料的概似，`loss.backward()` 算對數概似的梯度，`optimizer.step()` 往概似高處走一步，驗證集評估則檢查「泛化」這個統計主張（第 13.4 節）——SGD 只是登山工具，山的形狀由統計決定。本章路線圖如圖 13.1：從資料經概似、損失、先驗到訓練完成的模型，再分流到泛化、不確定性與生成三個統計任務，每個節點都標了所依靠的章節——它同時是全書的概念地圖。

13.2　損失函數從哪裡來

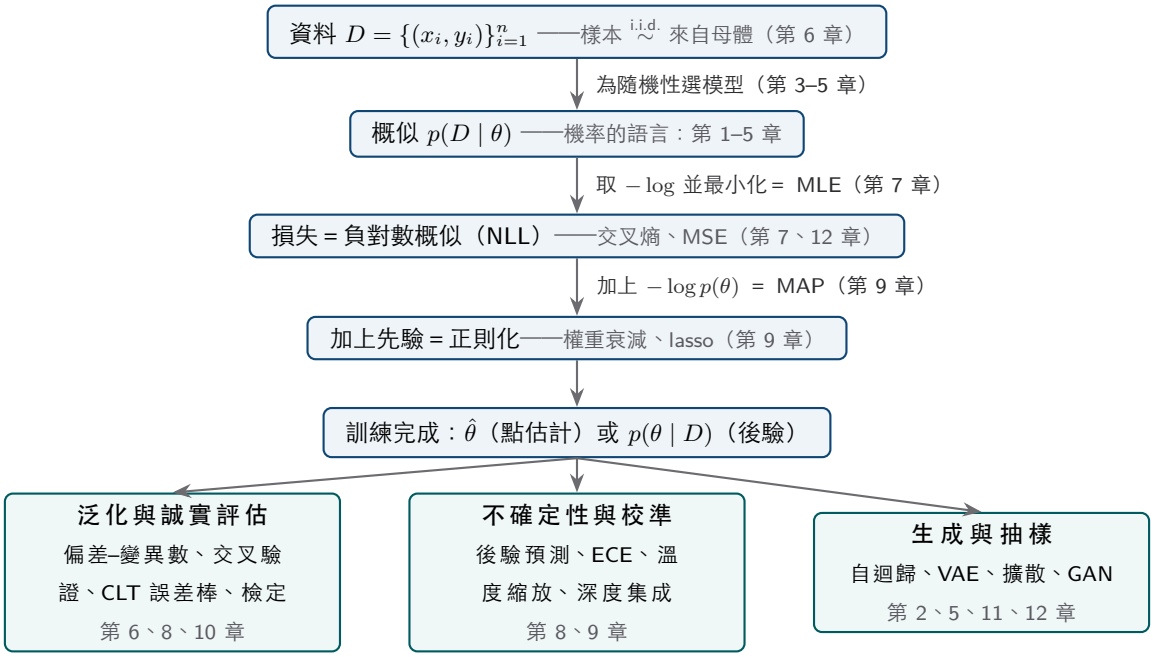


圖 13.1: 全書概念地圖，同時是本章路線圖：任何模型的訓練都沿著這條統計流水線前進，訓練完成後分流到三個統計任務。

打開任何深度學習框架的文件，你會看到一份損失函數型錄：`MSELoss`、`CrossEntropyLoss`、`L1Loss`、`PoissonNLLLoss`……它們看似互不相關的工程選項，其實全部出自同一條生產線：**先為資料的隨機性選一個分布，再取負對數概似**。最後一個名字直接洩了底——NLL。

定義 13.2　負對數概似損失

給定模型 $p(y\mid x,\theta)$ 與 $\overset{\text{i.i.d.}}{\sim}$ 資料 D，**負對數概似**（negative log-likelihood, NLL）損失為

$$\mathcal{L}(\theta)=-\log p(D\mid\theta)=-\sum_{i=1}^{n}\log p(y_i\mid x_i,\theta).$$

因為 $-\log$ 嚴格遞減，最小化 $\mathcal{L}$ 等價於最大化概似：**訓練 = MLE**。

例 13.1　高斯雜訊模型 $\Longrightarrow$ 均方誤差：一步步推導

設迴歸模型

$$y_i=f_\theta(x_i)+\varepsilon_i,\qquad \varepsilon_i\overset{\text{i.i.d.}}{\sim}\mathcal{N}(0,\sigma^2),$$

其中 f_θ 可以是線性函數，也可以是上億參數的神經網路。推導訓練 f_θ 該用的損失。

解　**第 1 步：把模型寫成條件分布。**雜訊假設等價於

$$p(y_i\mid x_i,\theta)=\frac{1}{\sqrt{2\pi\sigma^2}}\exp\!\Big(-\frac{(y_i-f_\theta(x_i))^2}{2\sigma^2}\Big).$$

第 2 步：寫出概似。樣本獨立，概似是 n 個上式的乘積：$p(D\mid\theta)=\prod_{i=1}^n p(y_i\mid x_i,\theta)$。

第 3 步：取負對數。乘積變加總：

$$-\log p(D \mid \theta) = \frac{n}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - f_\theta(x_i))^2.$$

第 4 步：丟掉與 θ 無關的部分。第一項是常數，第二項的係數 $\frac{1}{2\sigma^2} > 0$ 不影響 $\arg\min$，所以

$$\arg\min_\theta \big[-\log p(D \mid \theta)\big] = \arg\min_\theta \frac{1}{n}\sum_{i=1}^{n}(y_i - f_\theta(x_i))^2 = \arg\min_\theta \mathrm{MSE}(\theta).$$

結論：用均方誤差（mean squared error, MSE）訓練迴歸網路，就是在「殘差為同方差高斯」的假設下做最大概似。若把雜訊換成厚尾的 Laplace 分布，同樣四步會給出**平均絕對誤差**（mean absolute error, MAE）——這正是 MAE 對離群值較穩健的機率根源（練習 13.2）。

例 13.2　類別分布 $\Longrightarrow$ 交叉熵

K 類分類器對輸入 x 輸出 softmax 機率 $p_k(x;\theta)$，$k = 1, \dots, K$。資料為 $\{(x_i, y_i)\}_{i=1}^n$，$y_i \in \{1, \dots, K\}$。推導訓練損失。

解　每筆觀測「y_i 落在類別 y_i」的機率就是模型指給該類的機率，故概似與 NLL 為

$$p(D \mid \theta) = \prod_{i=1}^{n} p_{y_i}(x_i;\theta), \qquad \mathcal{L}(\theta) = -\sum_{i=1}^{n}\log p_{y_i}(x_i;\theta).$$

把第 i 筆的標籤寫成 one-hot 分布 $q^{(i)}$（$q_k^{(i)} = \mathbf{1}\{k = y_i\}$），單筆損失可改寫為

$$-\log p_{y_i}(x_i;\theta) = -\sum_{k=1}^{K} q_k^{(i)}\log p_k(x_i;\theta) = H\big(q^{(i)},\, p(x_i;\theta)\big),$$

恰是第 12 章的**交叉熵**（cross-entropy）。再用恆等式 $H(q,p) = H(q) + \mathrm{KL}(q \,\|\, p)$ 與 one-hot 的 $H(q^{(i)}) = 0$，得 $H(q^{(i)}, p) = \mathrm{KL}(q^{(i)} \,\|\, p) \geq 0$，等號成立若且唯若模型把全部機率押在正確類別上。**結論**：最小化交叉熵 = 最大化類別概似 = 在 KL 意義下把模型分布推向資料標籤。

各種標準損失與其背後雜訊模型的對照如表 13.1——這張表把第 4 章（分布動物園）、第 7 章（MLE）與第 12 章（資訊理論）綁在同一條線上。

表 13.1: 每個標準損失都是某個雜訊模型的負對數概似。

雜訊／輸出模型（第 4 章）	NLL 給出的損失	典型應用
高斯 $\mathcal{N}(f_\theta(x), \sigma^2)$	均方誤差（MSE）	迴歸、自編碼器的重建
Laplace（位置 $f_\theta(x)$、尺度 b）	平均絕對誤差（MAE）	穩健迴歸
白努利 $\mathrm{Bern}(p_\theta(x))$	二元交叉熵	二元分類、邏輯斯迴歸
類別分布（softmax 輸出）	（多類）交叉熵	影像分類、語言模型
卜瓦松 $\mathrm{Poi}(\lambda_\theta(x))$	卜瓦松損失	計數迴歸（點閱數、事故數）

核心觀念

選擇如何建模隨機性，就是選擇你的損失函數。交叉熵是類別模型的 NLL，等價於「模型到資料」的 KL 散度（第 12 章）；平方誤差是高斯雜訊的 NLL。於是有一條貫穿所有監督式學習的等式鏈：

$$\text{最小化損失} \quad = \quad \text{最大化概似} \quad = \quad \text{最小化與真實資料分布的 KL 散度。}$$

第 7 章（MLE）、第 10 章（迴歸與 GLM）與第 12 章（資訊理論）在此合而為一：一切標準損失都是「分布動物園裡挑一隻，取 $-\log$」的結果。

常見陷阱

損失不是可以隨意轉的旋鈕。每換一個損失，就默默換了一個雜訊假設。兩個常見翻車現場：(1) 把類別編號 $1,\dots,K$ 當實數、用 MSE 訓練分類器——隱含類別間有歐氏距離，「把 1 誤判成 3」被罰成誤判成 2 的四倍；(2) 對重尾資料（金融報酬、極端流量）用 MSE——離群值的平方項主導梯度，模型被少數樣本綁架。正確反應不是調小學習率，而是回頭改雜訊模型（Laplace、Student-t、Huber）。

13.3 正則化是先驗

第 10 章看過純 MLE 的病症：參數多、資料少時，它連雜訊都背下來——過擬合，偏差–變異數天平的變異數端。工程上的處方是在損失後面加一個懲罰項；而貝氏的 X 光（第 9 章）照出這個懲罰項的真身：**它是先驗**。

定義 13.3　帶懲罰的訓練與 MAP

實務上的訓練目標幾乎都是

$$\min_{\theta} \underbrace{-\log p(D \mid \theta)}_{\text{NLL（資料項）}} + \lambda\, R(\theta),$$

其中 $R(\theta)$ 是**正則化項**（regularizer）、$\lambda > 0$ 是其強度。若取 $R(\theta) = -\log p(\theta)$（相差常數與正倍數），這個目標恰為**最大後驗估計**（maximum a posteriori, MAP）：

$$\hat{\theta}_{\text{MAP}} = \arg\max_{\theta}\, p(\theta \mid D) = \arg\max_{\theta}\, p(D \mid \theta)\, p(\theta) = \arg\min_{\theta} \big[-\log p(D \mid \theta) - \log p(\theta)\big].$$

例 13.3　權重衰減 = 高斯先驗的 MAP：完整推導

設高斯雜訊迴歸模型（同例 13.1，σ^2 已知），並對 d 維參數放獨立高斯先驗 $\theta_j \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \tau^2)$，即 $\theta \sim \mathcal{N}(0, \tau^2 I)$。推導 MAP 目標。

解　**第 1 步：先驗的負對數。**

$$p(\theta)=\prod_{j=1}^{d}\frac{1}{\sqrt{2\pi\tau^2}}\exp\!\Big(-\frac{\theta_j^2}{2\tau^2}\Big)\quad\Longrightarrow\quad -\log p(\theta)=\frac{d}{2}\log(2\pi\tau^2)+\frac{\|\theta\|^2}{2\tau^2}.$$

第 2 步：組出 MAP 目標。丟掉與 θ 無關的常數：

$$J(\theta)=\frac{1}{2\sigma^2}\sum_{i=1}^{n}\big(y_i-f_\theta(x_i)\big)^2+\frac{1}{2\tau^2}\|\theta\|^2.$$

第 3 步：整理成熟悉的樣子。乘上正常數 $2\sigma^2$ 不改變最小值位置：

$$\tilde{J}(\theta)=\sum_{i=1}^{n}\big(y_i-f_\theta(x_i)\big)^2+\lambda\|\theta\|^2,\qquad\boxed{\lambda=\frac{\sigma^2}{\tau^2}}.$$

這正是**脊迴歸**（ridge regression，第 10 章），在深度學習裡叫**權重衰減**（weight decay）。λ 的公式會說話：先驗越緊（τ 小），罰得越重；雜訊越大（σ^2 大、資料越不可信），越該依賴先驗。同樣的計算把 Laplace 先驗變成 **lasso**（L1 正則化）：$-\log p(\theta)=\|\theta\|_1/b+$ 常數，其在 0 處的尖點誘導**稀疏解**（練習 13.3）。

把常見的正則化技巧逐一翻譯成先驗語言：

- **L2 ／權重衰減** = 高斯先驗 $\theta\sim\mathcal{N}(0,\tau^2 I)$（脊迴歸）；
- **L1 ／ lasso** = Laplace 先驗（尖點誘導稀疏，兼做特徵選擇）；
- **dropout ≈ 貝氏模型平均**（Bayesian model averaging）的廉價近似：每步隨機遮蔽神經元 = 訓練一個「子網路」，測試時的期望輸出近似對「哪些神經元存在」的模型平均；
- **label smoothing** = 標籤上的先驗：不完全相信 one-hot 標註，把 ε 的機率分給其他類別，防 logits 發散、緩解過度自信；
- **early stopping** ≈ 隱式收縮：走不遠的梯度下降把參數留在初始值附近，效果類似以初始點為中心的高斯先驗。

直觀理解

先驗就是「還沒看資料前你相信什麼」：權重衰減說「權重不會太大」，lasso 說「多數權重根本是 0」，label smoothing 說「標註偶爾會錯」；λ 是你堅持的程度。天平會自己傾斜：概似是 n 項之和、隨資料成長，先驗固定不動，資料夠多時終將壓過先驗（第 9 章的後驗集中）——所以大資料時代 λ 通常很小，而小樣本領域（醫療影像、罕見語言）正則化直接決定成敗。

這如何驅動機器學習／ AI

實務中的注意事項。(1) 調 weight decay 就是在調「先驗信心」：λ 加大，偏差上升、變異數下降——第 10 章那座天平。(2) 用驗證集掃 λ = 讓資料反推合理的先驗強度，是經驗貝氏（empirical Bayes）的精神。(3) 現代訓練配方「權重衰減 + dropout + label smoothing + early stopping」等於同時押四個先驗信念；AdamW 與 SGD 的實作差異（解耦與否）只改變最佳化路徑，不改變「懲罰 = 先驗」的統計語義。

13.4 泛化：從樣本到母體

模型在**樣本**上訓練，卻在**母體**上被評判——這是一次純然統計的跳躍。**真風險**（true risk）是母體上的期望損失（第 3 章），**經驗風險**（empirical risk）是它的樣本版本：

$$R(\theta) = \mathbb{E}_{(x,y)\sim P}\big[\ell\big(y, f_\theta(x)\big)\big], \qquad \widehat{R}_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell\big(y_i, f_\theta(x_i)\big).$$

對**固定的** θ，大數法則保證 $\widehat{R}_n(\theta) \to R(\theta)$，集中不等式（第 6 章）還給出定量的高機率保證。但訓練偏偏挑了讓 $\widehat{R}_n$ 最小的 $\hat{\theta}$——被挑中的點幾乎必然「運氣好」，訓練誤差因此是真風險的**樂觀（往下偏）估計**。統計學的標準武器全數上場：

- **偏差–變異數權衡**（第 10 章）：期望測試誤差 = 偏差 ^{2} + 變異數 + 不可約雜訊 σ^2。容量太小輸在偏差（欠擬合），太大輸在變異數（過擬合）；正則化在這座天平上移動砝碼。記下那個 σ^2——下一節它將以「偶然不確定性」的身分再登場。
- **交叉驗證**（第 7、10 章）：輪流留出資料，用**沒參與訓練**的部分估計母體誤差；k 折平均進一步壓低估計的變異數。
- **信賴區間與假設檢定**（第 8 章）：benchmark 分數是隨機變數。在 $n = 1000$ 筆的測試集上，單一準確率（約 87%）的 95% 信賴區間半寬 $1.96\sqrt{\hat{p}(1-\hat{p})/n} \approx 2.1$ 個百分點——0.5% 的「進步」多半只是抽樣雜訊。比較模型必須附誤差棒，並以 z 檢定、McNemar 檢定或 bootstrap（第 8 章）檢驗差距。

訓練內部的隨機性同樣歸極限定理管：mini-batch 梯度是全資料梯度的**樣本平均**（第 3、6 章），中央極限定理說它的雜訊近似 $\mathcal{N}(0, \Sigma/B)$——batch size 放大 4 倍，梯度雜訊減半。SGD 之所以「吵但能用」，是 LLN 與 CLT 在背後撐腰。

核心觀念

「這個模型泛化」不是工程口號，而是一個統計主張：「**訓練（或驗證）誤差是真風險的良好估計**」。大數法則保證估計收斂、中央極限定理給出誤差棒、偏差–變異數分解指認失敗模式、假設檢定裁決改進是否超出隨機波動——恰好是統計學被發明出來審理的那一類主張。

常見陷阱

測試集不是免費的。每一次「看了測試分數再回頭改模型」，都在把測試集悄悄變成訓練訊號；重複幾十次後樂觀偏差不斷累積，你以為的 SOTA 只是對這一千筆資料的過擬合——**排行榜過擬合**（leaderboard overfitting）。Kaggle 的 public / private 排行榜分裂、「測試集只碰一次」的學術紀律，守護的都是同一件事：**估計泛化誤差的資料，不可以參與任何決策**。

13.5 不確定性與校準

準確率回答「模型多常猜對」；校準回答「模型自稱的把握，值不值得信」。可信賴的模型必須**知道自己不知道什麼**——尤其當輸出要被當成機率、餵進下游的醫療分診、自駕決策或金融授信時。先把「不知道」拆成兩種。

定義 13.4　偶然與認知不確定性

- **偶然不確定性**（aleatoric uncertainty）：資料生成過程**本身**的隨機性——量測雜訊、標籤歧義、骰子的物理，即偏差–變異數分解裡的 σ^2。**不可約**：參數估得再準、資料再多，它都在。
- **認知不確定性**（epistemic uncertainty）：因資料有限而對**參數**（或模型結構）的無知，由後驗 $p(\theta \mid D)$ 的散布刻畫（第 9 章）。**可約**：資料越多它越小。

預測的總不確定性是兩者之和；第 10 章「預測區間比信賴區間寬」正是「偶然 + 認知」對上「只有認知」的差別。

定義 13.5　校準與期望校準誤差

設分類器對每筆輸入輸出預測類別與**信心**（confidence，即最大 softmax 機率）。模型是**校準的**（calibrated），若

$$\mathbb{P}(\text{預測正確} \mid \text{信心} = c) = c \qquad \text{對所有} c,$$

即「在所有以 80% 信心做出的預測中，約 80% 應該正確」。把 n 筆預測依信心分成 M 桶 $B_1, \dots, B_M$，**期望校準誤差**（expected calibration error, ECE）為

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{n_m}{n} \left| \mathrm{acc}(B_m) - \mathrm{conf}(B_m) \right|,$$

其中 n_m 是桶內筆數、acc 是桶內準確率、conf 是桶內平均信心。

例 13.4　三個信心桶的 ECE 計算

某分類器在 100 筆驗證資料上的預測依信心分成三桶：桶 $[0.5, 0.7)$ 有 20 筆、平均信心 0.60、命中 11（準確率 0.55）；桶 $[0.7, 0.9)$ 有 30 筆、平均信心 0.80、命中 21（0.70）；桶 $[0.9, 1.0]$ 有 50 筆、平均信心 0.95、命中 43（0.86）。求 ECE，並判斷模型是過度自信還是信心不足。

解　各桶差距為 $|0.55 - 0.60| = 0.05$、$|0.70 - 0.80| = 0.10$、$|0.86 - 0.95| = 0.09$。按桶的權重（筆數占比）加權：

$$\mathrm{ECE} = \frac{20}{100}(0.05) + \frac{30}{100}(0.10) + \frac{50}{100}(0.09) = 0.010 + 0.030 + 0.045 = 0.085.$$

三桶的準確率**全部**低於平均信心，模型系統性**過度自信**（overconfident），需要 $T > 1$ 的溫度縮放（見下）。

深度網路是出了名的過度自信：現代架構越深、越準，softmax 卻常把 70% 把握講成 99%（交叉熵鼓勵把 logits 推向極端、容量大到能記住標籤）。修補的工具箱由淺入深：

- **溫度縮放**（temperature scaling）：把 softmax 改成 $\mathrm{softmax}(z/T)$，在**驗證集**上以 NLL 擬合單一參數 T；$T > 1$ 壓平分布、調低信心。除以正常數不改變 logits 排序，**預測類別與準確率完全不變**（練習 13.5）。
- **MC dropout**：測試時保留 dropout，抽 M 次遮罩、平均預測——把隨機子網路當後驗樣本。
- **深度集成**（deep ensembles）：以不同初始化訓練 M 個網路（$3 \sim 5$ 個就很有效）平均預測；成員間的分歧直接度量認知不確定性。
- **貝氏神經網路**（Bayesian neural network）：直接近似 $p(\theta \mid D)$（變分推論、MCMC，第 9、11 章）。

這些方法殊途同歸：都在近似第 9 章的**後驗預測分布** $p(y \mid x, D) = \int p(y \mid x, \theta)\, p(\theta \mid D)\, \mathrm{d}\theta \approx \frac{1}{M}\sum_m p(y \mid x, \theta^{(m)})$——用「對參數的平均」取代「單一參數的斷言」，把生分數變成誠實機率。

核心觀念

校準是機率輸出的品質保證。「所有說八成把握的場合，長期約八成正確」是一個**頻率學派**的驗收標準（第 1 章），套在**貝氏味**的信念輸出上——兩大詮釋在此握手。我們用 ECE 量差距、用溫度縮放與集成修補，而修補全都在近似貝氏後驗預測（第 9 章）。

常見陷阱

把準確率當校準，是上線前最貴的一課。兩個測試準確率同為 90% 的模型可以天差地遠：一個錯的時候會說「我沒把握」，另一個永遠喊 99%、錯得斬釘截鐵——若下游按信心分診病患、決定自駕何時交還控制權，後者會系統性放大災難。同樣危險的是忽略**認知**不確定性：模型對訓練分布外（out-of-distribution）的輸入往往依然自信滿滿。

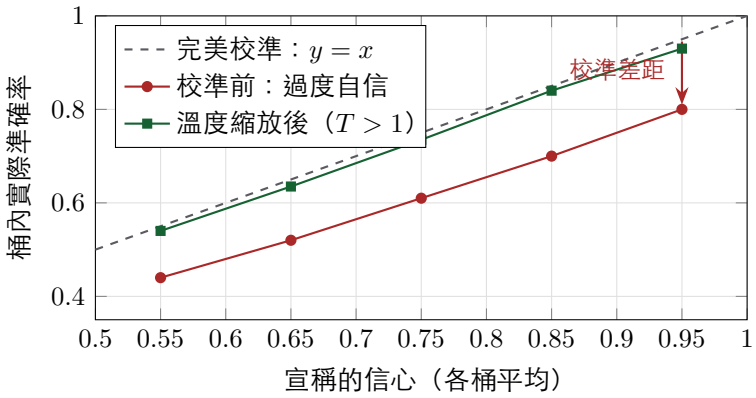


圖 13.2: 可靠度圖（reliability diagram）。對角線代表完美校準；紅色曲線整段低於對角線，是典型「過度自信」的深度網路。溫度縮放（$T > 1$）把信心整體調低、曲線貼回對角線，而預測類別與準確率完全不變。

這如何驅動機器學習／ AI

實務中的注意事項。只要輸出被當成機率參與決策，就必須校準。處方按成本排序：驗證集上量 ECE、畫可靠度圖；試溫度縮放（單一參數、幾乎零成本）；要求更高就上深度集成（同時改善準確率、校準與 OOD 偵測，代價是 M 倍訓練與推論）；MC dropout 介於其間。報告 benchmark 時，把 ECE 與誤差棒（第 13.4 節）和準確率並列，是負責任的最低標準。

13.6 生成模型：學會整個分布

到目前為止的主角是**判別式模型**（discriminative model）：學 $p(y \mid x)$，回答「給我輸入，我告訴你標籤」。**生成式模型**（generative model）野心更大：學整個資料分布 $p(x)$，因此能**抽樣**——寫出句子、畫出圖片、設計出蛋白質。四大家族看似天差地遠，其實每一個都是本書機率概念的直接建構。

（一）自迴歸模型（autoregressive model）：鏈式法則的化身。第 2 章的鏈式法則對**任何**分布恆成立：

$$p(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} p(x_t \mid x_{<t}),$$

GPT 把每個條件因子參數化成 softmax 分類器（「預測下一個 token」），逐因子用交叉熵擬合——大型語言模型的訓練損失就是 13.2 節的交叉熵，換上兆級的規模。分解本身不含任何近似；近似發生在參數化與有限資料。

例 13.5　三個 token 的 GPT：鏈式分解與逐 token 交叉熵

設詞彙表已定，模型逐步為句子「我愛統計」給出條件機率：

$$p(\text{我}) = 0.2, \qquad p(\text{愛} \mid \text{我}) = 0.1, \qquad p(\text{統計} \mid \text{我愛}) = 0.4.$$

求整句機率、每 token 平均 NLL 與困惑度（perplexity）。

解　鏈式法則給出整句機率

$$p(\text{我愛統計}) = 0.2 \times 0.1 \times 0.4 = 0.008.$$

總 NLL 與每 token 平均：

$$-\log 0.008 = 4.828 \text{ nats}, \qquad \bar{\ell} = \frac{4.828}{3} = 1.609 \text{ nats/token}.$$

困惑度定義為 $\text{PPL} = e^{\bar{\ell}} = e^{1.609} = 5.0$。讀法：$0.008 = 0.2^3$，幾何平均機率恰為 0.2，困惑度是它的倒數——模型平均像在 **5 個等可能的候選 token 中瞎猜**。GPT 的訓練損失就是 $\bar{\ell}$ 在上兆 token 上的平均，每次 loss 下降都是全體條件分布向資料分布靠近一步（第 12 章）。

（二）變分自編碼器（variational autoencoder, VAE）：ELBO 的舞台。VAE 假設資料由潛在變數生成：$z \sim \mathcal{N}(0, I)$（第 5 章的多元常態當先驗），$x \sim p_\theta(x \mid z)$（解碼器）。概似 $p_\theta(x) = \int p_\theta(x \mid z)\, p(z)\, \mathrm{d}z$ 積不出來，於是引入編碼器 $q_\phi(z \mid x)$ 當後驗的替身，改而最大化**證據下界**（evidence lower bound, ELBO）：

$$\text{ELBO}(x) = \underbrace{\mathbb{E}_{q_\phi(z|x)}\big[\log p_\theta(x \mid z)\big]}_{\text{重建項}} - \underbrace{\mathrm{KL}\big(q_\phi(z \mid x) \,\big\|\, p(z)\big)}_{\text{對先驗的 KL}} \;\le\; \log p_\theta(x).$$

訓練用**重參數化技巧**（reparameterization trick）：把抽樣寫成 $z = \mu_\phi(x) + \sigma_\phi(x) \odot \varepsilon$、$\varepsilon \sim \mathcal{N}(0, I)$，隨機性被搬到與參數無關的 ε 上，梯度才能穿過抽樣節點（第 5 章高斯的線性變換性質）。

例 13.6　ELBO 兩項各在說什麼

證明 $\log p_\theta(x) = \text{ELBO}(x) + \mathrm{KL}\big(q_\phi(z \mid x) \,\|\, p_\theta(z \mid x)\big)$，並給兩項白話的解讀。

解　由條件機率定義 $p_\theta(z \mid x) = p_\theta(x, z)/p_\theta(x)$，對任意 q_ϕ：

$$\begin{aligned}\mathrm{KL}\big(q_\phi(z \mid x) \,\big\|\, p_\theta(z \mid x)\big) &= \mathbb{E}_{q_\phi}\big[\log q_\phi(z \mid x) - \log p_\theta(x, z)\big] + \log p_\theta(x) \\ &= -\,\text{ELBO}(x) + \log p_\theta(x),\end{aligned}$$

其中用了 $\text{ELBO}(x) = \mathbb{E}_{q_\phi}[\log p_\theta(x, z) - \log q_\phi(z \mid x)]$（把 $p_\theta(x, z) = p_\theta(x \mid z)p(z)$ 展開即得正文的「重建 − KL」形式）。因為 KL 非負（第 12 章），$\text{ELBO} \le \log p_\theta(x)$，且等號成立若且唯若 $q_\phi(z \mid x)$ 恰為真後驗。

白話解讀：**重建項**說「編碼再解碼要還原得好」——壓縮不失真；**KL 項**說「編碼分布不許離先驗太遠」——壓縮碼要說先驗的語言，生成時才能直接從 $p(z)$ 抽 z 交給解碼器。拿掉 KL 項，VAE 退化成普通自編碼器（潛在空間亂放、無法抽樣生成）；KL 項太強則**後驗塌縮**（posterior collapse）：$q_\phi(z \mid x) \approx p(z)$ 對所有 x，編碼不帶資訊，重建退化成「畫平均臉」（練習 13.7）。

（三）擴散模型（diffusion model）：會倒帶的馬可夫鏈。前向過程是一條把資料逐步淹沒在高

斯雜訊裡的馬可夫鏈（第 11 章）：

$$q(x_t \mid x_{t-1}) = \mathcal{N}\big(x_t;\ \sqrt{1-\beta_t}\,x_{t-1},\ \beta_t I\big), \qquad t = 1, \dots, T,$$

走到 x_T 時幾乎是純雜訊 $\mathcal{N}(0, I)$。模型學的是**反向鏈** $p_\theta(x_{t-1} \mid x_t)$——一步步去噪。訓練目標是分層的變分下界，拆開是一串 KL 項 $\sum_t \mathrm{KL}\big(q(x_{t-1} \mid x_t, x_0) \,\|\, p_\theta(x_{t-1} \mid x_t)\big)$；兩邊都是高斯，每項 KL 有閉式解（第 5、12 章），化簡後竟變成「用 MSE 預測所加的雜訊」——**又回到高斯 NLL**。同一條馬可夫語法在強化學習裡升級為**馬可夫決策過程**（Markov decision process, MDP），而策略梯度的核心 $\nabla_\theta \log \pi_\theta$ 依然是對數概似的梯度。

（四）生成對抗網路（generative adversarial network, GAN）：用賽局逼近散度。生成器 G 把雜訊 z 映成假樣本，判別器 D 判斷樣本真偽，兩者對弈：

$$\min_G \max_D \ \mathbb{E}_{x \sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z \sim p(z)}[\log(1 - D(G(z)))].$$

固定 G 時最優判別器是 $D^*(x) = p_{\text{data}}(x) / \big(p_{\text{data}}(x) + p_G(x)\big)$——這是一個**貝氏後驗**（第 2 章）：「這張圖來自真資料的機率」。把 D^* 代回去，生成器的目標化為 $2\,\mathrm{JSD}(p_{\text{data}} \,\|\, p_G) - \log 4$，其中 JSD 是對稱化的 KL（第 12 章）；均衡點恰是 $p_G = p_{\text{data}}$。GAN 不寫概似（隱式生成模型），但它的靶心仍然是「最小化兩個分布之間的散度」。

直觀理解

四大家族共用一個套路：**把難以直接建模的分布，分解成一串簡單的機率零件**——自迴歸拆成「下一步預測」的條件分布，擴散拆成小步高斯轉移，VAE 繞道高斯潛在空間，GAN 用貝氏判別器當量尺。零件全部來自前十二章；讀懂零件，新模型不過是新的排列組合。

這如何驅動機器學習／ AI

現代 LLM 一次用齊全套。訓練 = 自迴歸分解（第 2 章）+ 交叉熵（第 7、12 章）+ 權重衰減（第 9 章）+ 驗證困惑度監控泛化（第 6、8 章）。部署時你在 API 裡調的 `temperature` 就是 13.5 節的 T：$T < 1$ 壓低熵（第 12 章）、輸出保守集中，$T > 1$ 拉高熵、多樣但易失控，$T \to 0$ 退化成貪婪 $\arg\max$。校準的溫度與解碼的溫度是同一個數學物件——一個管「信心的誠實」，一個管「生成的膽量」。

13.7 全書總覽：一張表與一段話

現在回頭看圖 13.1：從資料到概似、從概似到損失、從先驗到正則化、再分流到泛化、校準與生成——本章走過的每個箭頭，你其實在前十二章都已各別走過。表 13.2 把整本書的對應關係攤開成一張總表。

核心觀念

一段話說完機器學習。機器學習是「從資料推斷機率模型，並在不確定性下用它做預測」的實務。模型指定概似；最大概似把概似變成損失（交叉熵、平方誤差）；先驗變成正則化

表 13.2: 全書總覽：每個機率與統計概念在 AI 中的角色。

機率／統計概念	章節	在 AI 中的角色
機率公理；事件	1	合法的模型輸出；softmax 作為機率分布
條件機率與鏈式法則	2	自迴歸分解 $p(x)=\prod_t p(x_t \mid x_{<t})$
貝氏定理	2	學習的模板；垃圾郵件過濾與醫療診斷；先驗的引入
隨機變數；期望值	3	風險 = $\mathbb{E}$[損失]；特徵的標準化與正規化
分布動物園	4	選模型 = 選損失（表 13.1）
共變異數；多元常態	5	PCA、白化、高斯過程；VAE 與擴散模型的先驗
大數法則與中央極限定理	6	mini-batch SGD 的雜訊；蒙地卡羅方法；誤差棒
最大概似估計	7	交叉熵與 MSE 損失的統計出身
偏差–變異數；交叉驗證	7, 10	過擬合與欠擬合的診斷；容量與正則化的選擇
信賴區間與假設檢定	8	誠實的 benchmark 比較；A/B 測試；bootstrap 指標
貝氏推論；MAP	9	權重衰減（L2）、lasso（L1）；校準的不確定性
迴歸與廣義線性模型	10	線性 / 邏輯斯 / softmax = 神經網路的輸出層
馬可夫鏈與 MCMC	11	強化學習與 MDP；後驗抽樣；擴散模型的加噪鏈
熵、交叉熵、KL、互資訊	12	分類損失；ELBO；變分推論；對比學習

（權重衰減、lasso）；偏差–變異數權衡與極限定理決定它是否泛化；貝氏定理與後驗預測供應校準的不確定性；馬可夫鏈與蒙地卡羅負責生成樣本；資訊理論度量模型與世界的差距。機率說「該期待什麼」，統計說「該相信什麼」——合起來，它們不是機器學習的先修課，**它們就是機器學習**；線性代數與微積分則是讓這些計算跑得動的引擎。

接下來去哪？　比較多個模型或實驗組時，第 14 章的變異數分析把第 8 章的檢定延伸到 k 組。除此之外，本書給了你地基，往上有幾條自然的路：想知道「泛化為什麼成立」，讀**統計學習理論**（VC 維、泛化界）；想知道「上兆參數怎麼訓得動」，讀**大規模最佳化**；想動手蓋房子，進入**深度學習課程**本身。無論走哪條路，你都會不斷認出本書的老朋友——它們就住在每一條路的地基裡。

13.8 Python 實作

下面的程式把本章三個關鍵論點各自濃縮成幾行可執行的驗證：自迴歸 = 鏈式法則 + 交叉熵、ECE 的計算、溫度縮放的效果。

```python
import numpy as np

# --- 1. 自迴歸 = 鏈式法則 + 逐 token 交叉熵（例 13.5） ---
p_steps = np.array([0.2, 0.1, 0.4])          # p(x_t | x_{<t})
joint = p_steps.prod()                       # 整句機率 0.008
nll_tok = -np.log(p_steps).mean()            # 每 token 平均 NLL
print(joint, nll_tok.round(4), np.exp(nll_tok).round(2))  # 困惑度 5.0
```

```

# --- 2. 期望校準誤差（ECE） ---
def ece(conf, correct, n_bins=10):
    edges = np.linspace(0, 1, n_bins + 1)
    total = 0.0
    for lo, hi in zip(edges[:-1], edges[1:]):
        m = (conf > lo) & (conf <= hi)
        if m.any():
            total += m.mean() * abs(correct[m].mean() - conf[m].mean())
    return total

# 信心系統性高於準確率 10% 時，ece() 約回報 0.10；
# 完整實驗見練習 13.10 詳解

# --- 3. 溫度縮放：信心下降、argmax 不變（練習 13.5） ---
def softmax(Z, T=1.0):                    # 對最後一軸作用，
    Z = np.asarray(Z, dtype=float) / T    # 一維向量、二維批次皆可
    Z = Z - Z.max(axis=-1, keepdims=True)
    E = np.exp(Z)
    return E / E.sum(axis=-1, keepdims=True)

z = np.array([2.0, 0.0, -1.0])
for T in (1.0, 2.0, 4.0):
    p = softmax(z, T)
    print(T, p.round(3), int(p.argmax()))
# T=1: [0.844 0.114 0.042]; T=2: [0.629 0.231 0.14]; argmax 恆為 0
```

第 1 段用三行程式重現例 13.5 的整句機率、每 token NLL 與困惑度；第 2、3 段是 13.5 節的兩件工具——量化校準差距，以及不動準確率地修理它（輸出中信心隨 T 上升而下降、$\arg\max$ 恆為 0）。

本章總結

- **學習是統計推論**：模型指定概似 $p(D \mid \theta)$；訓練是 MLE 點估計（深度學習主流）或貝氏後驗（不確定性感知），預測靠最大化或對參數邊際化。
- **損失是負對數概似**：高斯給 MSE、類別給交叉熵、Laplace 給 MAE、卜瓦松給計數損失。最小化損失 = 最大化概似 = 最小化到資料分布的 KL 散度——選雜訊模型就是選損失。
- **正則化是先驗**：權重衰減 = 高斯先驗（$\lambda = \sigma^2/\tau^2$）、lasso = Laplace 先驗（稀疏）、dropout ≈ 貝氏模型平均、label smoothing = 標籤先驗、early stopping ≈ 隱式收縮。
- **泛化、校準與誠實評估是統計主張**：偏差–變異數與極限定理決定樣本到母體的跳躍；

交叉驗證估計母體誤差；檢定與信賴區間讓模型比較誠實；ECE 量化信心誠實度、溫度縮放與集成負責修補。

- **生成模型是本書概念的直接建構**：自迴歸 = 鏈式法則 + 交叉熵；VAE = ELBO（重建 − KL）+ 重參數化；擴散 = 高斯馬可夫鏈的反轉；GAN = 以貝氏判別器為量尺的散度最小化賽局。
- **機率與統計不是機器學習的背景知識——它們就是機器學習的地基**；線性代數與微積分是讓計算跑起來的工具。

13.9 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。

練習 13.1 (基礎)　把左欄的雜訊 / 輸出模型與右欄的損失函數配對，並各用一句話說明理由（提示：寫出該模型單筆觀測的 NLL 形式）。
左欄：(甲) 高斯 $\mathcal{N}(f_\theta(x), \sigma^2)$；(乙) Laplace；(丙) 白努利；(丁) 類別分布（softmax）；(戊) 卜瓦松。
右欄：(A) 多類交叉熵；(B) 均方誤差；(C) 卜瓦松損失；(D) 二元交叉熵；(E) 平均絕對誤差。

練習 13.2 (基礎)　（從雜訊模型推導損失）設迴歸模型 $y_i = f_\theta(x_i) + \varepsilon_i$，$\varepsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$，$\sigma^2$ 已知。(a) 寫出概似與 NLL，證明 $\arg\min_\theta \text{NLL} = \arg\min_\theta \text{MSE}$，並說明為何「加法常數」與「正的乘法係數」都不影響最佳解。(b) 改設 $\varepsilon_i \overset{\text{i.i.d.}}{\sim} \text{Laplace}(0, b)$（密度 $\frac{1}{2b} e^{-|\varepsilon|/b}$），推導對應的損失，並藉由比較單一殘差 r 的貢獻與梯度，說明 MAE 為何對離群值較穩健。(c) 換成 K 類分類器（softmax 輸出 $p_k(x;\theta)$、標籤 y_i）：寫出概似與 NLL，指出後者是逐樣本交叉熵之和。(d) 設 q 為 one-hot 標籤分布，證明 $H(q,p) = \text{KL}(q \,\|\, p)$，並解釋「最小化交叉熵就是在 KL 意義下把模型分布推向資料」。

練習 13.3 (進階)　（正則化與先驗）(a) 對高斯雜訊迴歸（雜訊變異數 σ^2）放先驗 $\theta \sim \mathcal{N}(0, \tau^2 I)$，完整推導 MAP 目標，並以 σ^2, τ^2 表示等價的權重衰減強度 λ。(b) 改放獨立 Laplace 先驗（尺度 b），證明 MAP 等價於 L1 正則化，並說明尖點與稀疏性的關係。(c) 若把 τ 減半，λ 變為原來的幾倍？方向是否符合直覺？(d) 把 dropout、label smoothing、early stopping 各自對應到其貝氏 / 收縮解讀。

練習 13.4 (基礎)　某分類器在 100 筆驗證資料上的預測分成三個信心桶：桶 $[0.6, 0.8)$ 有 40 筆、平均信心 0.72、命中 26；桶 $[0.8, 0.9)$ 有 35 筆、平均信心 0.85、命中 28；桶 $[0.9, 1.0]$ 有 25 筆、平均信心 0.96、命中 21。(a) 求每桶的準確率與校準差距。(b) 計算 ECE。(c) 模

型過度自信還是信心不足？建議 $T>1$ 還是 $T<1$ 的溫度縮放？

練習 13.5 (進階)　（溫度縮放）設 logits $z=(2,0,-1)$。(a) 證明對任意 $T>0$，$\mathrm{softmax}(z/T)$ 的 $\arg\max$ 與 $\mathrm{softmax}(z)$ 相同。(b) 分別計算 $T=1$ 與 $T=2$ 的 softmax 機率與最大信心（取三位小數）。(c) 求 $T\to\infty$ 與 $T\to 0^{+}$ 的極限分布。(d) 為什麼 T 要在驗證集而非訓練集上擬合？溫度縮放會改變測試準確率嗎？

練習 13.6 (進階)　判斷下列情境的不確定性主要是**偶然**（aleatoric）還是**認知**（epistemic），並說明哪些可以靠收集更多資料縮小：(a) 擲一顆公正骰子，下一次的點數；(b) 只用 50 張標註 X 光片訓練的診斷模型，其參數的正確值；(c) 相機感光元件在夜間拍攝時的熱雜訊；(d) 貓狗分類器收到一張卡車照片時的預測；(e) 一張連人類也無法分辨是「1」還是「7」的潦草手寫數字；(f) 兩個架構相同、隨機初始化不同的網路，對同一輸入給出不同預測。另外：情境 (d) 需要什麼樣的資料才有幫助？

練習 13.7 (進階)　（ELBO）定義 $\mathrm{ELBO}(x)=\mathbb{E}_{q(z\mid x)}[\log p(x\mid z)]-\mathrm{KL}(q(z\mid x)\,\|\,p(z))$。(a) 證明 $\log p(x)=\mathrm{ELBO}(x)+\mathrm{KL}(q(z\mid x)\,\|\,p(z\mid x))$，從而 $\mathrm{ELBO}(x)\le\log p(x)$，並指出等號成立的條件。(b) 用白話說明 ELBO 兩項各自的角色，以及 KL 項為何是「能從先驗抽樣生成」的關鍵。(c) β-VAE 把 KL 項乘上係數 β：討論 $\beta\to 0$ 與 β 過大時各會發生什麼事。

練習 13.8 (進階)　（自迴歸分解）(a) 由條件機率的定義出發，用數學歸納法證明鏈式分解 $p(x_1,\dots,x_T)=\prod_{t=1}^{T}p(x_t\mid x_{<t})$ 對任何分布恆成立（不需要任何獨立性假設）。(b) 某語言模型對三個 token 的句子給出 $p(x_1)=0.625$、$p(x_2\mid x_1)=0.4$、$p(x_3\mid x_1,x_2)=0.5$，求整句機率、每 token 平均 NLL 與困惑度。(c) 對照 n-gram 模型與 Transformer：兩者各如何近似條件因子 $p(x_t\mid x_{<t})$？n-gram 的哪個假設對應第 11 章的馬可夫性質？

練習 13.9 (挑戰)　是非題（每小題請附一兩句理由）：

(a) 最小化交叉熵等價於最大化訓練資料在類別模型下的概似。

(b) 權重衰減只是工程技巧，沒有機率上的詮釋。

(c) 測試準確率 90% 的模型必然校準良好。

(d) 偶然不確定性可以靠收集更多訓練資料消除。

(e) 溫度縮放會改變模型的測試準確率。

(f) VAE 的 ELBO 是 $\log p(x)$ 的下界。

(g) GPT 類模型的訓練損失本質上是逐 token 的交叉熵。

(h) 訓練誤差是母體風險的無偏估計。

練習 13.10 (挑戰·程式)　用 NumPy 寫一個實驗：(a) 實作 ECE 函數與帶溫度的 softmax。(b) 造一個「過度自信」的模擬分類器：真實標籤從 $\mathrm{softmax}(z)$ 抽出，但模型用放大的 logits $2.5z$ 預測。(c) 比較 $T=1$ 與 $T=2.5$ 的 ECE 與準確率，驗證校準大幅改善而準確率不變，並解釋為何 $T=2.5$ 恰是理論完美溫度。

13.10 練習題詳解

練習 13.1 詳解　配對：**甲–B、乙–E、丙–D、丁–A、戊–C**。理由（寫出單筆觀測的 NLL，略去與 θ 無關的常數）：甲（高斯）$-\log p=(y-f_\theta(x))^2/(2\sigma^2)$，平方誤差，平均即 **MSE**；乙（Laplace）$-\log p=|y-f_\theta(x)|/b$，絕對誤差，平均即 **MAE**；丙（白努利）$-\log p=-[\,y\log p_\theta(x)+(1-y)\log(1-p_\theta(x))\,]$，**二元交叉熵**；丁（類別）$-\log p=-\log p_y(x;\theta)$，**多類交叉熵**；戊（卜瓦松）由 $\mathbb{P}(Y=y)=e^{-\lambda}\lambda^y/y!$ 得 $-\log p=\lambda_\theta(x)-y\log\lambda_\theta(x)$，**卜瓦松損失**。

練習 13.2 詳解　(a) 概似與 NLL：

$$p(D\mid\theta)=\prod_{i=1}^{n}\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(y_i-f_\theta(x_i))^2}{2\sigma^2}},\qquad \mathrm{NLL}(\theta)=\frac{n}{2}\log(2\pi\sigma^2)+\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i-f_\theta(x_i))^2.$$

第一項是常數、第二項係數為正，故 $\arg\min$ 落在平方和（即 MSE）最小處。一般地，若 $g=a\,h+c$ 且 $a>0$，則 g 與 h 的大小比較完全一致——保序變換不動最佳解，推導中可放心丟常數、丟正係數。

(b) Laplace 雜訊給 $\mathrm{NLL}(\theta)=n\log(2b)+\frac{1}{b}\sum_i|y_i-f_\theta(x_i)|$，最小化即 **MAE**。殘差 r 在 MSE 中貢獻 r^2、在 MAE 中貢獻 $|r|$：離群殘差放大 10 倍，MSE 的罰放大 100 倍、MAE 只放大 10 倍；梯度上 MSE 正比於 r（離群值主導更新）、MAE 正比於 $\mathrm{sign}(r)$（票票等值）。極端例子：擬合常數 $f\equiv c$ 時 MSE 給樣本平均、MAE 給樣本中位數（第 3 章）。

(c) 概似為 $p(D\mid\theta)=\prod_i p_{y_i}(x_i;\theta)$，取負對數

$$\mathrm{NLL}(\theta)=-\sum_{i=1}^{n}\log p_{y_i}(x_i;\theta)=\sum_{i=1}^{n}H\big(q^{(i)},\,p(x_i;\theta)\big),$$

其中 $q^{(i)}$ 是第 i 筆的 one-hot 分布，因為 $H(q^{(i)},p)=-\log p_{y_i}$——正是深度學習框架中「交叉熵對 batch 求和」的形式。

(d) 由恆等式 $H(q,p)=H(q)+\mathrm{KL}(q\,\|\,p)$（第 12 章）與 one-hot 的熵 $H(q)=0$（約定 $0\log 0=0$），得 $H(q,p)=\mathrm{KL}(q\,\|\,p)\ge 0$。每一步梯度下降都在縮小標籤與模型分布的 KL——訓練分類器就是分布匹配。

練習 13.3 詳解　(a) MAP 目標為 $-\log p(D \mid \theta) - \log p(\theta)$。高斯先驗給 $-\log p(\theta) = \frac{\|\theta\|^2}{2\tau^2}$ + 常數，高斯概似給 $\frac{1}{2\sigma^2}\sum_i (y_i - f_\theta(x_i))^2$ + 常數，合併：

$$J(\theta) = \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - f_\theta(x_i))^2 + \frac{1}{2\tau^2}\|\theta\|^2 \quad \xrightarrow{\times 2\sigma^2} \quad \sum_{i=1}^{n}(y_i - f_\theta(x_i))^2 + \lambda\|\theta\|^2, \qquad \lambda = \frac{\sigma^2}{\tau^2}.$$

(b) Laplace 先驗 $p(\theta_j) = \frac{1}{2b}e^{-|\theta_j|/b}$ 給 $-\log p(\theta) = d\log(2b) + \|\theta\|_1/b$，故 MAP $= \arg\min_\theta \big[\mathrm{NLL}(\theta) + \|\theta\|_1/b\big]$，即 L1 正則化（lasso）。$|\theta_j|$ 在 0 處有尖點：只要資料項在 $\theta_j = 0$ 附近的梯度小於 $1/b$，最佳解就被「卡」在 0——這是稀疏性的幾何來源；光滑的 L2 只把權重推小、永遠推不到恰好為 0。
(c) $\lambda = \sigma^2/\tau^2$，$\tau \to \tau/2$ 使 $\lambda \to 4\lambda$：先驗緊一倍，罰重四倍。方向符合直覺——信念越強越該壓制大權重；平方關係來自高斯的負對數是二次式。
(d) dropout ≈ **貝氏模型平均**（對隨機「子網路」的不確定性平均）；label smoothing = **標籤的先驗**（以 $1-\varepsilon$ 相信標註、ε 均分其他類）；early stopping ≈ **隱式收縮**（參數留在初始值附近，近似以初始點為中心的高斯先驗）。

練習 13.4 詳解　(a) 各桶準確率：$26/40 = 0.65$、$28/35 = 0.80$、$21/25 = 0.84$。校準差距 $|\mathrm{acc} - \mathrm{conf}|$：$|0.65 - 0.72| = 0.07$、$|0.80 - 0.85| = 0.05$、$|0.84 - 0.96| = 0.12$。
(b) 權重為筆數占比 $0.40, 0.35, 0.25$：$\mathrm{ECE} = 0.40(0.07) + 0.35(0.05) + 0.25(0.12) = 0.028 + 0.0175 + 0.030 = 0.0755 \approx 7.6\%$。
(c) 三桶的準確率全部低於平均信心：**過度自信**，應使用 $T > 1$ 把 softmax 壓平、信心調低。（驗算：整體平均信心 0.8255 減整體準確率 0.75 恰為 0.0755——當所有桶往同一邊偏時，兩種算法必然一致。）

練習 13.5 詳解　(a) 設 $z_i > z_j$。因 $T > 0$，$z_i/T > z_j/T$；指數嚴格遞增給 $e^{z_i/T} > e^{z_j/T}$；除以同一個正配分函數不改變大小關係——softmax 排序與 logits 排序一致，$\arg\max$ 對任意 $T > 0$ 不變。
(b) $T = 1$：$e^2 = 7.389$、$e^0 = 1$、$e^{-1} = 0.368$，總和 8.757，得 $p = (0.844, 0.114, 0.042)$，最大信心 0.844。$T = 2$：$z/T = (1, 0, -0.5)$ 給 $(2.718, 1, 0.607)$，總和 4.325，得 $p = (0.629, 0.231, 0.140)$，最大信心 0.629——信心下降、預測類別不變。
(c) $T \to \infty$：$z/T \to (0,0,0)$，softmax 趨於均勻分布 $(1/3, 1/3, 1/3)$（最大熵）。$T \to 0^+$：最大 logit 優勢被無限放大，趨於 one-hot $(1,0,0)$（零熵、貪婪 $\arg\max$）。
(d) 訓練集上模型已把標籤「背熟」、信心被人為推高；驗證集才是「未見過資料」的代理，在其上以 NLL 擬合 T 才能校正部署時的信心。由 (a)，溫度縮放不改變預測類別，**測試準確率完全不變**；變的是 NLL、ECE 等機率品質指標。

練習 13.6 詳解　(a) **偶然**：就算確定骰子公正（參數無不確定性），下一擲仍不可預測。(b) **認知**：50 筆資料下後驗 $p(\theta \mid D)$ 很寬，更多標註會使它收窄。(c) **偶然**：物理雜訊屬於資料

生成過程，模型再好、資料再多都在。(d) **認知**（為主）：模型從未見過該區域的輸入；照片若模糊另疊偶然成分。(e) **偶然**：標籤本身歧義，貝氏最優分類器也只能五五開——不可約的類別重疊。(f) **認知**：資料不足以唯一決定參數，初始化差異存活到訓練後；深度集成正是用成員分歧**估計**認知不確定性。
可靠縮小者：(b)、(d)、(f)——都是「對參數的無知」。(d) 的補充：需要**涵蓋該輸入區域**的資料（把卡車等類別納入訓練，或建立 OOD 偵測），單純加貓狗照片幫助有限。

練習 13.7 詳解　(a) 由 $p(z \mid x) = p(x, z)/p(x)$：

$$\begin{aligned}\mathrm{KL}\big(q(z \mid x) \,\big\|\, p(z \mid x)\big) &= \mathbb{E}_q\Big[\log \frac{q(z \mid x)}{p(z \mid x)}\Big] = \mathbb{E}_q\big[\log q(z \mid x) - \log p(x, z)\big] + \log p(x) \\ &= -\underbrace{\mathbb{E}_q\big[\log p(x, z) - \log q(z \mid x)\big]}_{=\,\mathrm{ELBO}(x)} + \log p(x),\end{aligned}$$

($\log p(x)$ 不依賴 z，可提出期望外；把 $p(x, z) = p(x \mid z)\, p(z)$ 展開即得題目的「重建 − KL」形式。) 移項得 $\log p(x) = \mathrm{ELBO}(x) + \mathrm{KL} \geq \mathrm{ELBO}(x)$，等號成立若且唯若 $q(z \mid x) = p(z \mid x)$——編碼器完美復刻真後驗。
(b) **重建項** $\mathbb{E}_q[\log p(x \mid z)]$：從編碼 z 解碼回 x 的對數概似要高——壓縮後要還原得回來。**KL 項**：編碼分布必須貼近先驗 $p(z)$——壓縮碼要說先驗的語言。生成時直接從 $p(z)$ 抽樣丟給解碼器，KL 項確保「編碼器實際使用的區域」與「生成時抽樣的區域」是同一塊，抽出的 z 解碼器才看得懂。
(c) $\beta \to 0$：退化為普通自編碼器——重建極佳，但潛在空間與先驗脫鉤，無法從 $p(z)$ 抽樣生成。β 過大：**後驗塌縮**——最省事的解是 $q(z \mid x) \approx p(z)$ 對所有 x，KL 歸零但 z 不帶資訊，重建退化成「輸出資料的平均樣貌」。β-VAE 刻意取 $\beta > 1$，犧牲部分重建換取更解耦（disentangled）的表示——與 13.3 節的 λ 同宗的權衡。

練習 13.8 詳解　(a) 歸納法。$T = 1$：$p(x_1) = p(x_1)$，成立。假設對 $T - 1$ 成立。由條件機率定義（第 2 章），對任何滿足 $p(x_1, \ldots, x_{T-1}) > 0$ 的點：

$$p(x_1, \ldots, x_T) = p(x_1, \ldots, x_{T-1})\; p(x_T \mid x_1, \ldots, x_{T-1}),$$

把歸納假設代入前項即得 $\prod_{t=1}^{T} p(x_t \mid x_{<t})$。全程只用條件機率定義——**不需要**任何獨立性假設，分解是恆等式；模型之間的差別只在如何參數化這些條件因子。
(b) 整句機率 $= 0.625 \times 0.4 \times 0.5 = 0.125 = (1/2)^3$。每 token 平均 NLL 為 $\bar{\ell} = -\frac{1}{3}\log 0.125 = \log 2 \approx 0.693$ nats/token，困惑度 $\mathrm{PPL} = e^{\bar{\ell}} = 2$：模型平均每步像在兩個等可能候選中擇一。
(c) n-gram 假設 $p(x_t \mid x_{<t}) \approx p(x_t \mid x_{t-n+1}, \ldots, x_{t-1})$——只看最近 $n - 1$ 個 token，正是第 11 章的（高階）**馬可夫性質**。Transformer 不做截斷，以自注意力對**整段前文**條件化，用共享參數的網路輸出 softmax。n-gram 的致命傷是資料稀疏（長脈絡幾乎從未重複出現），深度參數化以泛化能力補上了這個洞。

練習 13.9 詳解　(a) **對**。類別模型的 NLL 恰為交叉熵之和（練習 13.2），$-\log$ 嚴格遞減，最小化其一即最大化其二。
(b) **錯**。權重衰減是「高斯先驗下的 MAP」：$\lambda\|\theta\|^2$ 就是 $-\log p(\theta)$ 在 $\theta \sim \mathcal{N}(0, \tau^2 I)$ 時的樣子，$\lambda = \sigma^2/\tau^2$（練習 13.3）。
(c) **錯**。準確率只看 $\arg\max$ 是否命中，校準看信心是否誠實；90% 準確的模型可以永遠宣稱 99.9% 信心。
(d) **錯**。偶然不確定性是資料生成過程固有的隨機性（量測雜訊、標籤歧義）；更多資料縮小的是**認知**不確定性。
(e) **錯**。$T > 0$ 下 z/T 保持排序，$\arg\max$ 不變、準確率原封不動；變的是信心分布（NLL、ECE）。
(f) **對**。$\log p(x) = \text{ELBO} + \text{KL}(q \,\|\, p(z \mid x))$ 而 KL 非負；最大化 ELBO 同時推高下界、拉近變分後驗與真後驗。
(g) **對**。鏈式分解把 $\log p(x)$ 寫成 $\sum_t \log p(x_t \mid x_{<t})$，每個因子是 softmax 分類器，其 NLL 就是該位置的交叉熵。
(h) **錯**。$\hat{\theta}$ 是專門挑來最小化訓練誤差的，幾乎必然「沾了樣本雜訊的光」，訓練誤差因此**往下偏**（樂觀）；無偏估計需要未參與決策的 held-out 資料或交叉驗證。

練習 13.10 詳解　(a) 的 `ece` 與帶溫度的 `softmax` 直接沿用 13.8 節的實作（後者本就對最後一軸作用，一次處理整個批次）；接續該環境，(b)(c) 的實驗如下：

```
rng = np.random.default_rng(1)

# (b) 真實標籤來自 softmax(z)，模型卻用放大的 logits 2.5z
n, K = 20000, 3
z_true = rng.normal(0.0, 1.2, size=(n, K))
y = np.array([rng.choice(K, p=p) for p in softmax(z_true)])
logits = 2.5 * z_true                    # 過度自信的模型

# (c) 比較 T = 1 與 T = 2.5
for T in (1.0, 2.5):
    P = softmax(logits, T)
    conf = P.max(axis=1)
    correct = (P.argmax(axis=1) == y).astype(float)
    print(T, round(ece(conf, correct), 3), round(correct.mean(), 3))
# 1.0   0.117   0.62x   <- ECE 大：宣稱的信心遠高於實際
# 2.5   0.006   0.62x   <- 同樣的準確率，校準幾乎完美
```

解讀：兩列的**準確率完全相同**（$\arg\max$ 不變，練習 13.5a）；$T = 2.5$ 恰為**完美溫度**——標籤真的從 $\text{softmax}(z)$ 抽出，$\text{softmax}(2.5z/2.5) = \text{softmax}(z)$ 正是真實條件分布，殘餘的 0.006 是抽樣雜訊（第 6 章）。