機率與統計：從入門到精通

第 10 章　統計模型與迴歸

Statistical Models and Regression

難度：進階

迴歸是機率與統計「變成可預測的機器學習」之處。前幾章我們學會了描述隨機性、估計參數（最大概似，第 7 章）與納入先驗（第 9 章）；本章把這些工具組裝成**統計模型**——描述輸出如何依賴輸入、再加上隨機性——並用**迴歸**把這種依賴關係擬合到資料上。線性迴歸與邏輯斯迴歸是最簡單的兩個這類模型，而且並非巧合地，它們正是整個監督式學習工具箱（一路到深度神經網路）的概念模板：機率模型決定損失函數、最大概似給出擬合原理、梯度下降負責優化。本章最後將形式化**偏差–變異數權衡**——它將支配你未來的每一個擬合決策。

學習目標

讀完本章後，你將能夠：

- 寫出線性迴歸的機率模型 $y = \boldsymbol{w}^\top \boldsymbol{x} + b + \varepsilon$，證明**最小平方法**等價於高斯雜訊下的最大概似估計，並推導**正規方程**。
- 對小型資料集手算簡單線性迴歸的 $\hat{w}$、$\hat{b}$、**殘差**與判定係數 R^2，並正確解讀其意義。
- 用 sigmoid 把線性分數轉成機率，推導**邏輯斯迴歸**的交叉熵損失與「預測減實際」形式的梯度，並說明 softmax 的多類推廣。
- 以「分布 + 連結函數」的觀點理解**廣義線性模型**（GLM），說明分布如何經由最大概似決定損失，並用卜瓦松迴歸處理計數資料。
- 敘述並證明**偏差–變異數分解**，診斷欠擬合與過度擬合，並運用正則化（脊迴歸）與交叉驗證調整模型複雜度。
- 用 NumPy 實作最小平方、脊迴歸與邏輯斯迴歸的梯度下降，親手驗證本章理論。

10.1　統計模型與線性迴歸

前幾章的主角是「分布」：給定一個機率模型，資料會長什麼樣？從本章起我們把方向反過來：給定資料，哪個模型能夠**預測**？監督式學習（supervised learning）的舞台上有一組成對資料 $(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_n, y_n)$，其中 $\boldsymbol{x}_i \in \mathbb{R}^d$ 是**特徵**（feature），y_i 是想預測的**目標**（target）。統計模型就是對「y 如何依賴 $\boldsymbol{x}$」的機率式描述。

定義 10.1　統計模型

統計模型（statistical model）是一族由參數 θ 索引的條件分布 $p(y \mid \boldsymbol{x};\theta)$，描述在輸入 $\boldsymbol{x}$ 給定之下，輸出 y 的分布——也就是「系統性關係 + 隨機性」。**擬合**（fit）模型是指用資料選出參數估計 $\hat{\theta}$；**迴歸**（regression）泛指把輸出對輸入的依賴關係擬合到資料的整個過程。

定義 10.2　線性迴歸模型

線性迴歸（linear regression）把連續目標建模為特徵的線性函數加上高斯雜訊：

$$y = \boldsymbol{w}^{\top}\boldsymbol{x} + b + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0,\sigma^2),$$

各筆資料的雜訊 $\varepsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,\sigma^2)$。$\boldsymbol{w} \in \mathbb{R}^d$ 稱為**權重**（weights），b 稱為**截距**（intercept；機器學習中常稱 bias 項）。等價的寫法是

$$y \mid \boldsymbol{x} \sim \mathcal{N}\big(\boldsymbol{w}^{\top}\boldsymbol{x} + b,\ \sigma^2\big).$$

特徵只有一維時（$y = wx + b + \varepsilon$）稱為**簡單線性迴歸**（simple linear regression）。

直觀理解

線性迴歸主張「**訊號加雜訊**」：y 的期望值是 $\boldsymbol{x}$ 的線性函數（訊號），而雜訊項 ε 收容所有未被特徵捕捉的因素——被省略的變數、測量誤差、真正的隨機性。另外，「線性」指的是**對參數線性**：你可以把特徵換成 $x, x^2, \log x$ 等非線性轉換，模型仍然是線性迴歸（多項式迴歸就是這樣來的）。這正是線性迴歸適用範圍出乎意料廣的原因。

怎樣的直線才算「擬合得好」？最自然的準則是讓預測與觀測的**平方誤差總和**最小。這就是 Legendre 與 Gauss 在十九世紀初提出的**最小平方法**（method of least squares）：

$$\min_{w,\,b}\ S(w,b) = \sum_{i=1}^{n}\big(y_i - wx_i - b\big)^2.$$

命題 10.3　簡單線性迴歸的閉式解

給定資料 $(x_1,y_1),\dots,(x_n,y_n)$，記 $\bar{x} = \frac{1}{n}\sum_i x_i$、$\bar{y} = \frac{1}{n}\sum_i y_i$。最小平方解為

$$\hat{w} = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sum_i (x_i - \bar{x})^2} = \frac{S_{xy}}{S_{xx}}, \qquad \hat{b} = \bar{y} - \hat{w}\,\bar{x}.$$

分子分母同除以 n 可見 $\hat{w}$ 是「樣本共變異數除以樣本變異數」，也可寫成 $\hat{w} = r_{xy}\,(s_y/s_x)$，其中 r_{xy} 為樣本相關係數。

證明　對 b 偏微分並設為零：

$$\frac{\partial S}{\partial b} = -2\sum_i \big(y_i - wx_i - b\big) = 0 \quad\Longrightarrow\quad b = \bar{y} - w\bar{x}.$$

把它代回、再對 w 偏微分：

$$\frac{\partial S}{\partial w} = -2\sum_i x_i\big(y_i - \bar y - w(x_i - \bar x)\big) = 0 \quad\Longrightarrow\quad \sum_i x_i(y_i - \bar y) = w\sum_i x_i(x_i - \bar x).$$

由於 $\sum_i (y_i - \bar y) = 0$，把左式的 x_i 換成 $x_i - \bar x$ 不改變總和；右式同理。故 $S_{xy} = w\,S_{xx}$，即 $\hat w = S_{xy}/S_{xx}$。S 是 (w, b) 的凸二次函數，此駐點即全域最小值。 ■

定義 10.4　殘差、SSE 與判定係數

擬合完成後，$\hat y_i = \hat w x_i + \hat b$ 稱為**擬合值**，

$$e_i = y_i - \hat y_i$$

稱為**殘差**（residual）——模型「沒解釋掉」的部分。殘差平方和 $\mathrm{SSE} = \sum_i e_i^2$、總平方和 $\mathrm{SST} = \sum_i (y_i - \bar y)^2$，而**判定係數**（coefficient of determination）

$$R^2 = 1 - \frac{\mathrm{SSE}}{\mathrm{SST}}$$

衡量「y 的變異中有多大比例被模型解釋」。含截距的最小平方擬合保證 $0 \le R^2 \le 1$，且殘差總和 $\sum_i e_i = 0$（來自 $\partial S/\partial b = 0$）。對簡單線性迴歸，$R^2 = r_{xy}^2$。

例 10.1　手算簡單線性迴歸

某公司記錄五個月的廣告支出 x（萬元）與銷售額 y（萬元）：$(1,2),(2,3),(3,5),(4,6),(5,9)$。求最小平方直線、殘差與 R^2。

解　先算平均：$\bar x = 3$、$\bar y = 25/5 = 5$。列表計算離差：

x_i	y_i	$x_i - \bar x$	$y_i - \bar y$	$(x_i - \bar x)(y_i - \bar y)$	$(x_i - \bar x)^2$
1	2	−2	−3	6	4
2	3	−1	−2	2	1
3	5	0	0	0	0
4	6	1	1	1	1
5	9	2	4	8	4
			合計	$S_{xy} = 17$	$S_{xx} = 10$

因此

$$\hat w = \frac{17}{10} = 1.7, \qquad \hat b = 5 - 1.7\times 3 = -0.1, \qquad \hat y = 1.7x - 0.1.$$

擬合值依序為 1.6, 3.3, 5.0, 6.7, 8.4，殘差為 0.4, −0.3, 0, −0.7, 0.6（總和為 0，驗算通過）。於是

$$\mathrm{SSE} = 0.16 + 0.09 + 0 + 0.49 + 0.36 = 1.10, \qquad \mathrm{SST} = 9 + 4 + 0 + 1 + 16 = 30,$$

$$R^2 = 1 - \frac{1.10}{30} \approx 0.963.$$

交叉驗算：$r_{xy} = 17/\sqrt{10\times 30} = 0.9815$，$r_{xy}^2 = 0.963$，一致。解讀：廣告每多投一萬元，銷售額平均增加 1.7 萬元；銷售變異的 96.3% 由廣告支出解釋。

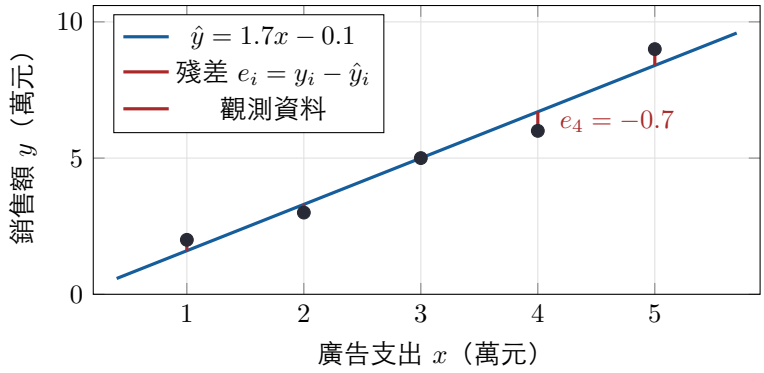


圖 10.1: 例 10.1 的資料、最小平方擬合線與殘差（垂直線段；$x = 3$ 處殘差恰為 0）。最小平方法選出使殘差平方和最小的那條直線。

核心觀念

用最大概似擬合線性迴歸模型，與最小平方法完全等價。因為雜訊是高斯的，整組資料的對數概似等於

$$-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\big(y_i - \boldsymbol{w}^\top \boldsymbol{x}_i - b\big)^2 + \text{常數},$$

所以「最大化概似」就是「最小化平方誤差和」。Legendre 與 Gauss 發明的最小平方法，原來就是高斯雜訊假設下的 MLE（第 7 章）。這揭示一個貫穿全書的原理：**機率模型與損失函數是同一件事的兩面**——選定雜訊分布，損失函數自動誕生。

例 10.2　最小平方 = 高斯 MLE：完整推導

證明在線性迴歸模型下，$(\boldsymbol{w}, b)$ 的最大概似估計恰為最小平方解，並求 σ^2 的 MLE。

解　各筆觀測獨立，且 $y_i \mid \boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{w}^\top \boldsymbol{x}_i + b, \sigma^2)$，概似函數為

$$L(\boldsymbol{w}, b, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\!\Big(-\frac{(y_i - \boldsymbol{w}^\top \boldsymbol{x}_i - b)^2}{2\sigma^2}\Big).$$

取對數：

$$\ell(\boldsymbol{w}, b, \sigma^2) = -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\underbrace{\sum_{i=1}^{n}(y_i - \boldsymbol{w}^\top \boldsymbol{x}_i - b)^2}_{\mathrm{SSE}(\boldsymbol{w}, b)}.$$

對任何固定的 $\sigma^2 > 0$，第一項與 $(\boldsymbol{w}, b)$ 無關，第二項的係數 $-1/(2\sigma^2)$ 為負，故最大化 ℓ 等價於最小化 SSE：MLE 就是最小平方解 $(\hat{\boldsymbol{w}}, \hat{b})$，而且這個結論與 σ^2 的值無關。接著把 SSE 的最小值記為 $\widehat{\mathrm{SSE}}$，對 σ^2 微分：

$$\frac{\partial \ell}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{\widehat{\mathrm{SSE}}}{2\sigma^4} = 0 \quad\Longrightarrow\quad \hat{\sigma}^2 = \frac{\widehat{\mathrm{SSE}}}{n}.$$

雜訊變異數的 MLE 就是殘差的均方。

把截距吸收進權重可以讓多特徵的情形更俐落：令**設計矩陣**（design matrix）$X \in \mathbb{R}^{n\times(d+1)}$ 的第 i 列為 $(1, \boldsymbol{x}_i^\top)$，並把 b 併入 $\boldsymbol{w}$ 的第一個分量，則所有預測可一次寫成 $X\boldsymbol{w}$，最小平方問題成為 $\min_{\boldsymbol{w}} \|\boldsymbol{y} - X\boldsymbol{w}\|^2$。

定理 10.5　正規方程

若 $X^\top X$ 可逆，則 $\min_{\boldsymbol{w}} \|\boldsymbol{y} - X\boldsymbol{w}\|^2$ 有唯一解

$$\hat{\boldsymbol{w}} = (X^\top X)^{-1} X^\top \boldsymbol{y},$$

稱為**正規方程**（normal equations）的解——統計與線性代數在此精準相遇。

證明　展開目標函數：

$$L(\boldsymbol{w}) = \|\boldsymbol{y} - X\boldsymbol{w}\|^2 = \boldsymbol{y}^\top\boldsymbol{y} - 2\boldsymbol{w}^\top X^\top \boldsymbol{y} + \boldsymbol{w}^\top X^\top X \boldsymbol{w}.$$

利用梯度恆等式 $\nabla_{\boldsymbol{w}}(\boldsymbol{w}^\top \boldsymbol{a}) = \boldsymbol{a}$ 與 $\nabla_{\boldsymbol{w}}(\boldsymbol{w}^\top A\boldsymbol{w}) = 2A\boldsymbol{w}$（$A$ 對稱；逐分量證明見練習 10.9），

$$\nabla_{\boldsymbol{w}} L = -2X^\top \boldsymbol{y} + 2X^\top X\boldsymbol{w} = \boldsymbol{0} \quad\Longrightarrow\quad X^\top X\,\hat{\boldsymbol{w}} = X^\top \boldsymbol{y}.$$

Hessian 為 $2X^\top X$，對任意 $\boldsymbol{v}$ 有 $\boldsymbol{v}^\top X^\top X\boldsymbol{v} = \|X\boldsymbol{v}\|^2 \ge 0$，故 L 是凸函數、駐點即全域最小；$X^\top X$ 可逆時解唯一。 ∎

$X^\top X$ 可逆的條件是 X 的各行（各特徵）線性獨立；若特徵**共線**（例如同一特徵以不同單位放了兩次），解就不唯一——10.4 節的脊迴歸正好能修復這一點。以例 10.1 驗算：$X^\top X = \left(\begin{smallmatrix} 5 & 15 \\ 15 & 55 \end{smallmatrix}\right)$、$X^\top \boldsymbol{y} = \left(\begin{smallmatrix} 25 \\ 92 \end{smallmatrix}\right)$，解此 2×2 方程組得 $\hat b = -0.1$、$\hat w = 1.7$，與命題 10.3 一致——正規方程是同一個閉式解的矩陣包裝，優點是任意多個特徵都適用。

直觀理解

正規方程有一幅漂亮的幾何圖像：$X\hat{\boldsymbol{w}}$ 是 $\boldsymbol{y}$ 在 X 的行空間（所有可能預測構成的子空間）上的**正交投影**。改寫 $X^\top(\boldsymbol{y} - X\hat{\boldsymbol{w}}) = \boldsymbol{0}$ 可見：**殘差向量垂直於每一個特徵向量**——最小平方 = 找子空間中離 $\boldsymbol{y}$ 最近的點 = 投影。這也解釋了殘差總和為零：全 1 的截距行也在被垂直的名單裡。

常見陷阱

（一）外插風險。擬合結果只在資料涵蓋的範圍內可信。例 10.1 的 $x \in [1, 5]$：拿 $\hat y = 1.7x - 0.1$ 去預測 $x = 20$ 純屬臆測——廣告效果很可能早已飽和，直線關係在遠處未必成立。**（二）R^2 高不代表因果，也不代表模型正確。**冰淇淋銷量能高度「解釋」溺水人數（兩者都被夏天驅動），但禁售冰淇淋救不了人；而明顯彎曲的資料硬配直線，也可能得到看似體面的 R^2。永遠畫散點圖與殘差圖。

10.2　邏輯斯迴歸：把分數變成機率

當目標是**類別**——垃圾郵件或正常、違約或正常——我們建模的是「屬於類別 1 的機率」而非連續值。直接把標籤 0/1 當連續值做線性迴歸行不通：輸出會跑出 $[0,1]$ 之外，也對離群點極度敏感。正確的做法是讓線性分數通過一個「壓縮函數」變成機率。

定義 10.6　邏輯斯迴歸

邏輯斯迴歸（logistic regression）把線性分數通過 **sigmoid 函數**變成機率：

$$\mathbb{P}(y=1 \mid \boldsymbol{x}) = \sigma\big(\boldsymbol{w}^{\top}\boldsymbol{x}+b\big), \qquad \sigma(z)=\frac{1}{1+e^{-z}},$$

等價地 $y \mid \boldsymbol{x} \sim \mathrm{Bernoulli}\big(\sigma(\boldsymbol{w}^{\top}\boldsymbol{x}+b)\big)$。線性分數 $z=\boldsymbol{w}^{\top}\boldsymbol{x}+b$ 常稱為 **logit**。

命題 10.7　sigmoid 的基本性質

對所有 $z\in\mathbb{R}$：(a) $0<\sigma(z)<1$ 且 σ 嚴格遞增；(b) $\sigma(-z)=1-\sigma(z)$；(c) $\sigma'(z)=\sigma(z)\big(1-\sigma(z)\big)$；(d) $p=\sigma(z) \iff z=\log\dfrac{p}{1-p}$，即 z 是**對數勝算**（log-odds）。

證明　(a) 顯然。(b) $\sigma(-z)=\frac{1}{1+e^{z}}=\frac{e^{-z}}{e^{-z}+1}=1-\sigma(z)$。(c) $\sigma'(z)=\frac{e^{-z}}{(1+e^{-z})^2}=\frac{1}{1+e^{-z}}\cdot\frac{e^{-z}}{1+e^{-z}}=\sigma(z)\big(1-\sigma(z)\big)$。(d) 由 $p(1+e^{-z})=1$ 得 $e^{-z}=\frac{1-p}{p}$，取對數即得。 ■

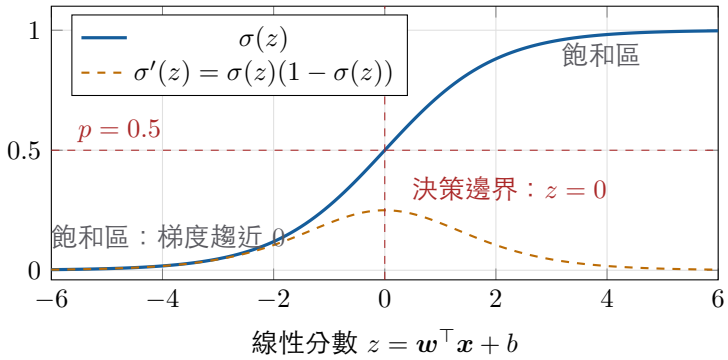


圖 10.2: sigmoid 把整條實數軸壓縮到 $(0,1)$；$z=0$ 對應機率 0.5，是預設的決策邊界。虛線是其導數，在 $z=0$ 處最大（0.25），兩端趨近於零。

以 $p\ge 0.5$ 判為類別 1 時，**決策邊界**（decision boundary）恰是超平面 $\boldsymbol{w}^{\top}\boldsymbol{x}+b=0$——邏輯斯迴歸是**線性分類器**。性質 (d) 給出最好的參數解讀：x_j 每增加一單位，對數勝算增加 w_j，等於「勝算乘上 e^{w_j}」。

怎麼擬合？仍然用最大概似。標籤 $y_i\in\{0,1\}$、$p_i=\sigma(\boldsymbol{w}^{\top}\boldsymbol{x}_i+b)$，白努利概似為

$$L(\boldsymbol{w},b)=\prod_{i=1}^{n} p_i^{\,y_i}(1-p_i)^{1-y_i}, \qquad -\log L=-\sum_{i=1}^{n}\Big[y_i\log p_i+(1-y_i)\log(1-p_i)\Big].$$

右邊這個負對數概似（NLL）正是機器學習裡無所不在的**交叉熵**（cross-entropy）損失。

核心觀念

用最大概似擬合邏輯斯迴歸 = 最小化交叉熵（白努利 NLL）。再一次，分布決定了損失。與線性迴歸不同的是：sigmoid 的非線性讓「令梯度為零」不再有閉式解，必須用**梯度下降**（gradient descent）迭代求解 $\boldsymbol{w} \leftarrow \boldsymbol{w} - \eta \nabla_{\boldsymbol{w}} \mathrm{NLL}$。幸運的是交叉熵對 $(\boldsymbol{w}, b)$ 是凸函數，梯度下降能收斂到全域最小值，而且梯度本身有極乾淨的形式——下例。

例 10.3　邏輯斯迴歸的梯度：「預測減實際」

推導單筆資料的損失 $\ell = -\big[y \log p + (1-y)\log(1-p)\big]$（$p = \sigma(z)$、$z = \boldsymbol{w}^{\top}\boldsymbol{x} + b$）對 $\boldsymbol{w}$ 與 b 的梯度。

解　連鎖法則分三步。第一步，對 p 微分：

$$\frac{\partial \ell}{\partial p} = -\frac{y}{p} + \frac{1-y}{1-p}.$$

第二步，乘上 $\dfrac{\partial p}{\partial z} = p(1-p)$（命題 10.7(c)）：

$$\frac{\partial \ell}{\partial z} = \left(-\frac{y}{p} + \frac{1-y}{1-p}\right) p(1-p) = -y(1-p) + (1-y)\,p = p - y.$$

第三步，z 對 $\boldsymbol{w}$ 的梯度是 $\boldsymbol{x}$、對 b 是 1，故

$$\nabla_{\boldsymbol{w}} \ell = (p-y)\,\boldsymbol{x}, \qquad \frac{\partial \ell}{\partial b} = p - y.$$

全體資料的梯度就是 $\sum_i (p_i - y_i)\boldsymbol{x}_i$：**預測減實際、再用特徵加權**。sigmoid 與對數的導數在中途完全相消，這不是巧合——10.3 節會揭示它是 GLM 的結構性後果。順帶一提，線性迴歸的梯度 $\sum_i (\hat{y}_i - y_i)\boldsymbol{x}_i$ 也是同一形式。

例 10.4　垃圾郵件過濾器：機率計算與決策邊界

某垃圾郵件模型使用兩個特徵：x_1 = 可疑關鍵字數、x_2 = 寄件者信譽分數，已擬合出 $w_1 = 1.2$、$w_2 = -0.8$、$b = -0.5$。

(a) 一封郵件有 $x_1 = 3$、$x_2 = 2$，是垃圾郵件的機率？(b) 若 $x_1 = 1$、$x_2 = 3$ 呢？(c) 寫出決策邊界。(d) 要「至少九成把握」判為垃圾郵件，線性分數需多大？

解　(a) $z = 1.2 \times 3 - 0.8 \times 2 - 0.5 = 3.6 - 1.6 - 0.5 = 1.5$，

$$p = \sigma(1.5) = \frac{1}{1 + e^{-1.5}} = \frac{1}{1 + 0.2231} \approx 0.818,$$

判為垃圾郵件（勝算 $e^{1.5} \approx 4.48$，約 $4.5 : 1$）。

(b) $z = 1.2 - 2.4 - 0.5 = -1.7$，$p = \sigma(-1.7) = 1/(1+e^{1.7}) \approx 0.154$，判為正常。

(c) $p = 0.5 \iff z = 0$，即直線 $1.2x_1 - 0.8x_2 - 0.5 = 0$，整理為 $x_2 = 1.5x_1 - 0.625$：關鍵字越多，需要越高的信譽分數才能「洗白」。

(d) $\sigma(z) \geq 0.9 \iff z \geq \log \frac{0.9}{0.1} = \log 9 \approx 2.197$。例如 $x_1 = 3$ 時需 $3.6 - 0.8x_2 - 0.5 \geq 2.197$，

即 $x_2 \le 1.13$。

類別超過兩個時，把 sigmoid 換成 **softmax**：對 K 類各配一組參數 $(\boldsymbol{w}_k, b_k)$，令 $z_k = \boldsymbol{w}_k^\top \boldsymbol{x} + b_k$，

$$\mathbb{P}(y = k \mid \boldsymbol{x}) = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}},$$

每項為正、總和為一，是合法的機率分布。用類別分布的 MLE 擬合，損失是**多類交叉熵** $-\log p_y$，對分數的梯度依然是「預測減實際」：$\partial / \partial z_k = p_k - \mathbf{1}\{k = y\}$。$K = 2$ 時 softmax 化約回 sigmoid（只有分數差 $z_1 - z_0$ 有意義）。

直觀理解

邏輯斯迴歸就是「**單層、sigmoid 輸出的神經網路**」：輸入特徵、一層線性權重、一個非線性輸出。疊上更多層與非線性，就是深度學習；但損失（交叉熵）、擬合原理（最大概似）、優化器（梯度下降）全都不變。這是邏輯斯迴歸被稱為「現代分類器概念祖先」的原因——用機率的眼光理解它，你就理解了幾乎每一個深度模型的輸出層。

這如何驅動機器學習／ AI

打開任何一個深度分類網路：最後一層正是 softmax 迴歸，訓練損失正是交叉熵，也就是類別分布的最大概似。網路前面的所有層只是在**學特徵**——把原始輸入變換成讓「最後那個邏輯斯／ softmax 迴歸」好做的表示。因此「深度網路的預測是機率」不是修辭而是字面事實：它服從第 1 章的公理，可以（也應該）用機率的工具檢驗——例如第 13 章的校準（calibration）。

常見陷阱

完全可分的資料會讓 MLE 發散。若有一個超平面能把兩類完全分開，那麼把 $\boldsymbol{w}$ 沿該方向無限放大，概似會單調上升趨近 1——最佳解不存在，$\|\boldsymbol{w}\| \to \infty$，輸出機率全被推向 0 或 1。症狀是係數暴走、機率過度自信；解方是正則化（10.4 節）或 early stopping。另外提醒：分類準確率高不代表機率輸出可信，把輸出當機率用之前，先檢查校準。

10.3 廣義線性模型：一個原理統一所有損失

回顧一下已經出現的模式：目標是連續值，選高斯分布，MLE 給出平方誤差；目標是二元類別，選白努利分布，MLE 給出交叉熵。這不是兩個孤立的技巧，而是同一台機器的兩次運轉。把它一般化，就得到**廣義線性模型**。

定義 10.8　廣義線性模型

廣義線性模型（generalized linear model, GLM）由三個組件構成：

1. **線性預測子**：$\eta = \boldsymbol{w}^\top \boldsymbol{x} + b$；
2. **目標分布**：為 $y \mid \boldsymbol{x}$ 選一個分布（高斯、白努利、卜瓦松……）；
3. **連結函數**（link function）g：把該分布的均值 $\mu = \mathbb{E}[y \mid \boldsymbol{x}]$ 接到線性預測子上：$g(\mu) = \eta$，即 $\mu = g^{-1}(\eta)$。

擬合一律用最大概似：**分布（經由 MLE）決定損失函數**。線性迴歸（恆等連結）與邏輯斯迴歸（logit 連結）都是特例。

表 10.1: 常用 GLM 一覽：選定分布與連結，損失自動誕生。

目標分布	連結函數 $g(\mu)$	均值模型 $\mu(\boldsymbol{x})$	MLE 給出的損失	典型任務
高斯	恆等：μ	$\boldsymbol{w}^\top \boldsymbol{x} + b$	平方誤差	房價等連續目標
白努利	logit：$\log \frac{\mu}{1-\mu}$	$\sigma(\boldsymbol{w}^\top \boldsymbol{x} + b)$	二元交叉熵	垃圾郵件過濾
類別（K 類）	多類 logit	$\mathrm{softmax}(\boldsymbol{z})$	多類交叉熵	影像分類
卜瓦松	對數：$\log \mu$	$e^{\boldsymbol{w}^\top \boldsymbol{x} + b}$	卜瓦松 NLL	來電數等計數資料

例 10.5　卜瓦松迴歸：計數資料的 GLM

某客服中心想用「當日促銷檔次 x」預測「每小時來電數 y」。來電數是非負整數，適合卜瓦松分布：$y \mid x \sim \mathrm{Poisson}(\mu(x))$，取對數連結 $\log \mu = b + wx$（保證 $\mu > 0$）。已擬合出 $\hat{b} = 0.8$、$\hat{w} = 0.3$。

(a) 無促銷與四檔促銷時的預期來電數？(b) 如何解讀 $\hat{w} = 0.3$？(c) 四檔促銷時整小時無來電的機率？(d) 寫出 NLL 的梯度。

解　(a) $x = 0$：$\mu = e^{0.8} \approx 2.23$ 通；$x = 4$：$\mu = e^{0.8 + 1.2} = e^{2} \approx 7.39$ 通。

(b) 對數連結給出**乘法效應**：x 每加一檔，μ 乘上 $e^{0.3} \approx 1.35$，即來電數平均增加約 35%——而非「增加固定通數」。

(c) $\mathbb{P}(y = 0 \mid x = 4) = e^{-\mu} = e^{-7.389} \approx 0.0006$：幾乎必有來電。

(d) 卜瓦松 pmf 為 $e^{-\mu}\mu^y / y!$，單筆 NLL 為 $\mu - y \log \mu + \log(y!)$。以 $\mu = e^z$、$z = b + wx$ 代入並對 z 微分：

$$\frac{\partial}{\partial z}\left(e^z - yz\right) = e^z - y = \mu - y \quad \Longrightarrow \quad \nabla_w \mathrm{NLL} = \sum_i (\mu_i - y_i)\, x_i.$$

又是「預測減實際」！

核心觀念

GLM 把機器學習「損失的動物園」統一在一個機率原理之下：**選分布 + 選連結，MLE 決定損失**。平方誤差、交叉熵、卜瓦松 NLL 的梯度全都長成「(預測 − 實際) × 特徵」，不是巧合，而是指數族分布配上典型連結（canonical link）的結構性後果。遇到新任務時，先問「什麼分布適合我的 y ?」——正值連續可用 Gamma、過度離散的計數可用負二項——損失函數就會自己出現。

10.4　偏差–變異數權衡

我們已經會擬合模型了，該面對更深的問題：擬合出來的模型對**沒見過的資料**表現如何？衡量它的量是**測試誤差**（泛化誤差），而非訓練誤差。本節的分解定理把測試誤差拆成三種來源，成為所有模型容量決策的透鏡。（提醒：此處「偏差」與截距的別名 bias 項無關，只是撞名。）

定理 10.9　偏差–變異數分解

固定測試點 x，設資料由 $y = f(x) + \varepsilon$ 生成，$\mathbb{E}[\varepsilon] = 0$、$\mathrm{Var}(\varepsilon) = \sigma^2$。$\hat{f}$ 是在隨機抽出的訓練集 D 上訓練得到的預測函數（ε 為測試點的新雜訊，與 D 獨立）。則在平方誤差損失下，

$$\underbrace{\mathbb{E}\Big[\big(y - \hat{f}(x)\big)^2\Big]}_{\text{期望測試誤差}} = \underbrace{\big(f(x) - \mathbb{E}[\hat{f}(x)]\big)^2}_{\text{偏差}^2\text{（太僵硬）}} + \underbrace{\mathrm{Var}\big(\hat{f}(x)\big)}_{\text{變異數（太敏感）}} + \underbrace{\sigma^2}_{\text{不可約雜訊}},$$

期望對訓練集 D 與測試雜訊 ε 同時取。

證明　記 $\bar{f}(x) = \mathbb{E}_D[\hat{f}(x)]$（「平均模型」的預測）。第一步拆出雜訊：

$$\mathbb{E}\big[(y - \hat{f})^2\big] = \mathbb{E}\big[(\varepsilon + f - \hat{f})^2\big] = \mathbb{E}[\varepsilon^2] + 2\,\mathbb{E}\big[\varepsilon\,(f - \hat{f})\big] + \mathbb{E}\big[(f - \hat{f})^2\big].$$

因 ε 與 D（故與 $\hat{f}$）獨立且均值為零，交叉項為 0；$\mathbb{E}[\varepsilon^2] = \sigma^2$。第二步對剩下的一項加減 $\bar{f}$：

$$\mathbb{E}\big[(f - \hat{f})^2\big] = \mathbb{E}\big[(f - \bar{f} + \bar{f} - \hat{f})^2\big] = (f - \bar{f})^2 + 2(f - \bar{f})\,\mathbb{E}[\bar{f} - \hat{f}] + \mathbb{E}\big[(\bar{f} - \hat{f})^2\big].$$

$f - \bar{f}$ 是常數而 $\mathbb{E}[\bar{f} - \hat{f}] = 0$，交叉項再度消失；末項正是 $\mathrm{Var}(\hat{f})$。合併即得。 ■

三個詞的白話翻譯：**偏差**——就算把「所有可能訓練集的平均模型」拿來用，仍然學不到真相（模型族太僵硬，例如用直線硬套曲線）；**變異數**——換一批訓練資料，答案就劇烈變動（模型太有彈性，把訓練雜訊也背了下來）；σ^2——資料本身的隨機性，任何模型都躲不掉的地板。

直觀理解

把「訓練一次模型、在 x 處做預測」想成朝靶心（真值 $f(x)$）射一箭；「換一批訓練資料重訓」就是再射一箭。**偏差**是箭群中心偏離靶心的距離（系統性瞄偏），**變異數**是箭群本身散得多開（手抖）。簡單模型：箭群緊密但整體偏一邊；複雜模型：繞著靶心亂噴。分解式裡的期望與變異數，說的正是這個「重複整套訓練流程」的思想實驗。

模型太簡單，偏差主導，稱為**欠擬合**（underfitting）：連訓練資料都學不好。模型太有彈性，變異數主導，稱為**過度擬合**（overfitting）：訓練誤差近乎完美、測試誤差卻糟糕——它背下了雜訊。隨著模型複雜度增加，偏差單調下降、變異數單調上升，期望測試誤差因此呈 U 形，最小值落在中間某處。

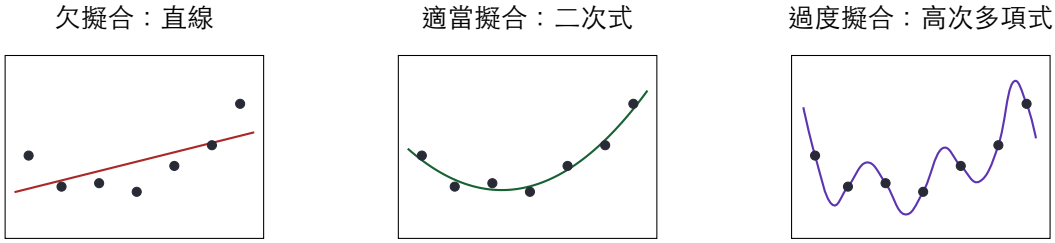


圖 10.3: 同一組資料、三種複雜度。直線抓不到彎曲趨勢（高偏差）；二次式恰到好處；高次多項式穿過每一點卻在點與點之間劇烈震盪（高變異數）——換一批資料，它會變成完全不同的曲線。

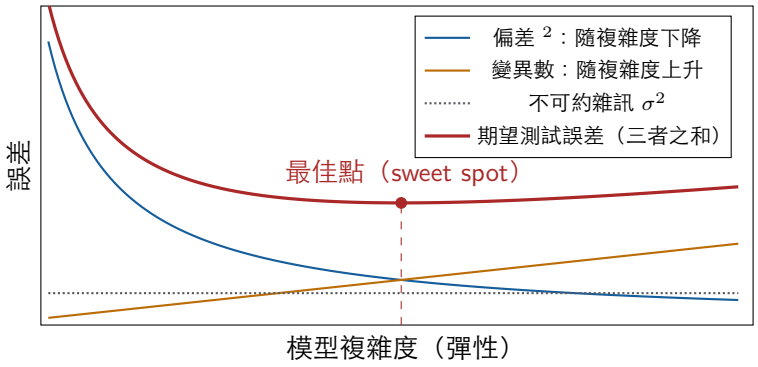


圖 10.4: 偏差–變異數權衡示意（曲線取 偏差$^2 = 4/x$、變異數 $= x/8$、$\sigma^2 = 0.5$）：左側偏差主導（欠擬合）、右側變異數主導（過度擬合），總誤差呈 U 形，最小值在中間。所有容量決策都是在這條曲線上移動。

核心觀念

偏差–變異數權衡是模型擬合的中心張力。太簡單的模型（高偏差）欠擬合——抓不到模式；太有彈性的模型（高變異數）過度擬合——背下訓練雜訊、跨樣本劇烈變動。泛化誤差在中間某處最小。實務上的兩大工具：**正則化**故意加一點偏差，換掉大量變異數；**交叉驗證**誠實地估計測試誤差，用來找出平衡點。訓練誤差永遠偏袒複雜模型，絕不能拿來做容量決策。

正則化。正則化（regularization）在損失上加一個懲罰項，抑制係數的大小。最經典的是**脊迴歸**（ridge regression）：

$$\min_{\boldsymbol{w}} \ \|\boldsymbol{y} - X\boldsymbol{w}\|^2 + \lambda\|\boldsymbol{w}\|^2, \qquad \lambda \geq 0,$$

閉式解為 $\hat{\boldsymbol{w}}_\lambda = (X^\top X + \lambda I)^{-1} X^\top \boldsymbol{y}$（推導見練習 10.10；慣例上不懲罰截距）。$\lambda$ 越大，係數被拉向 0 越多：偏差上升、變異數下降。而且對任何 $\lambda > 0$，$X^\top X + \lambda I$ 必定可逆——共線特徵的

問題順便被治好。從第 9 章的貝氏觀點看，脊迴歸恰是「權重取高斯先驗 $\boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \tau^2 I)$ 時的 MAP 估計」，$\lambda = \sigma^2/\tau^2$：**正則化 = 先驗**。近親**套索迴歸**（lasso）改罰 $\lambda \sum_j |w_j|$，會把部分係數壓到恰好為零，兼做特徵選擇；**early stopping**（提早停止梯度下降）則是另一種隱式正則化。

例 10.6　脊迴歸的縮小效果

沿用例 10.1 中心化後的資料（$S_{xx} = 10$、$S_{xy} = 17$，此時無截距），脊迴歸的一維閉式解為

$$\hat{w}_\lambda = \frac{S_{xy}}{S_{xx} + \lambda} = \frac{17}{10 + \lambda} = \frac{10}{10 + \lambda} \times \hat{w}_{\text{OLS}}.$$

計算不同 λ 的係數：

λ	0	2	5	10	40	$\to \infty$
$\hat{w}_\lambda$	1.700	1.417	1.133	0.850	0.340	$\to 0$
縮小因子	1.00	0.83	0.67	0.50	0.20	$\to 0$

解　$\lambda = 0$ 還原最小平方；λ 增大時係數平滑地縮向 0（但不會恰好等於零——那是 lasso 的專長）；$\lambda = S_{xx}$ 時恰好縮半。縮小的代價是偏差、報酬是變異數：雜訊大、資料少、特徵共線時，適度的 λ 常讓測試誤差顯著下降。

交叉驗證。該怎麼選 λ（以及一切超參數）？標準解是 k **摺交叉驗證**（k-fold cross-validation）：把訓練資料均分為 k 摺，輪流讓每一摺當驗證集、其餘 $k-1$ 摺訓練，把 k 個驗證誤差取平均作為測試誤差的估計；對每個候選超參數重複此流程，選平均誤差最小者，再用全部訓練資料重新訓練。真正的測試集全程鎖在保險箱，只在最後動用一次。

例 10.7　五摺交叉驗證選 λ

以 100 筆資料、$k = 5$（每摺 20 筆）為脊迴歸選 λ，候選 $\{0.01, 0.1, 1, 10, 100\}$。每個 λ 訓練 5 次（共 25 次擬合），平均驗證 MSE 如下：

λ	0.01	0.1	1	10	100
平均驗證 MSE	2.41	2.18	**1.93**	2.06	3.12

解　選 $\lambda^* = 1$，再用全部 100 筆以 $\lambda = 1$ 重訓。注意誤差對 λ 呈 U 形——λ 太小過度擬合（變異數殘留）、太大欠擬合（偏差入侵），正是偏差–變異數曲線的再現。

常見陷阱

訓練誤差的假象。加大模型、多塞特徵，訓練誤差保證不升——所以用訓練誤差（或未調整的 R^2）選模型，永遠會選到最複雜的那個，正中過度擬合的陷阱。同樣危險的是**反覆用測試集調參**：每看一次測試分數再回頭改超參數，測試集就洩漏一分資訊，久而久之你「過度擬合了測試集」，成績再漂亮也只是自欺。紀律：訓練集學參數、驗證集（或交叉驗證）選超參數、測試集只在最後開封一次。連特徵標準化、特徵選擇也必須只用訓練摺的統計量，

否則資訊照樣滲漏。

這如何驅動機器學習／ AI

本章是**監督式學習的骨架**。線性迴歸 = 高斯雜訊下的 MLE = 最小平方；邏輯斯 / softmax 迴歸 = 白努利 / 類別分布下的 MLE = 交叉熵——正是深度分類器的輸出層；GLM 用一個原理解釋整個損失動物園。而偏差-變異數權衡是每個容量決策的透鏡：網路加寬加深（降偏差）、正則化與 weight decay（降變異數）、early stopping、dropout、資料增強、集成方法（ensemble 靠平均消變異數）、蒐集更多資料（降變異數、不動偏差）——全都在同一條 U 形曲線上移動。深度學習還加了著名的轉折：**double descent**——測試誤差在「參數恰好能背下資料」的內插門檻處衝高後，隨參數繼續增加反而再度下降（練習 10.11）。這不推翻分解式（它是恆等式），只是提醒：偏差與變異數隨複雜度變化的形狀，取決於模型族與訓練演算法。

10.5 Python 實作

動手驗證本章的兩件大事：正規方程與 `lstsq` 給出相同的最小平方解、並精準恢復生成資料的真實權重；脊迴歸（= 高斯先驗的 MAP）把係數縮向零。邏輯斯迴歸的梯度下降實作見練習 10.12。

```
import numpy as np

rng = np.random.default_rng(42)

# --- 1. 產生線性迴歸資料：y = X w_true + 高斯雜訊 ---
n = 200
X = np.column_stack([np.ones(n), rng.normal(size=(n, 2))])  # 首欄 1 吸收截距
w_true = np.array([1.0, 2.0, -0.5])
y = X @ w_true + rng.normal(scale=0.5, size=n)

# --- 2. 最小平方：正規方程 vs. lstsq（兩者應一致）---
w_ne = np.linalg.solve(X.T @ X, X.T @ y)      # (X^T X)^{-1} X^T y
w_ls, *_ = np.linalg.lstsq(X, y, rcond=None)  # 數值上更穩健的做法
print(np.round(w_ne, 3), np.round(w_ls, 3))   # 皆約 [1.0, 2.0, -0.5]

# --- 3. 脊迴歸 = 高斯先驗的 MAP：lambda 越大，係數越縮向 0 ---
for lam in [0.0, 10.0, 100.0, 1000.0]:
    w_r = np.linalg.solve(X.T @ X + lam * np.eye(3), X.T @ y)
    print(lam, np.round(w_r, 3))              # 係數逐步縮小
```

第 2 段印出的兩組係數幾乎等於 `w_true`：閉式的正規方程與數值穩健的 `lstsq` 完全一致，

且有了足夠資料，最小平方確實恢復真實權重。第 3 段可看到 λ 從 0 到 1000 時係數一路縮向 0——正則化以偏差換變異數的具體模樣；把它與例 10.6 的一維縮小因子公式對照，每一行輸出都有理論背書。

本章總結

- **統計模型**描述輸出如何依賴輸入加上隨機性；**迴歸**把這種依賴擬合到資料。
- **線性迴歸** = 高斯雜訊下的 MLE = **最小平方法**，閉式解由**正規方程** $\hat{\boldsymbol{w}} = (X^\top X)^{-1} X^\top \boldsymbol{y}$ 給出；殘差垂直於特徵（投影觀點），R^2 衡量被解釋的變異比例。
- **邏輯斯迴歸**把線性分數通過 sigmoid 變成機率 = 白努利 MLE = 最小化**交叉熵**；無閉式解但為凸，梯度是「預測減實際」；softmax 是多類推廣，正是神經網路分類器的輸出層。
- **GLM** 統一全局：選分布 + **連結函數**，MLE 決定損失——卜瓦松迴歸（log 連結）處理計數資料，係數有乘法解讀。
- **偏差–變異數分解**：期望測試誤差 = 偏差^2 + 變異數 + σ^2；欠擬合 = 偏差主導、過度擬合 = 變異數主導，測試誤差呈 U 形。
- **正則化**（脊迴歸 = 高斯先驗的 MAP、lasso、early stopping）用一點偏差換掉大量變異數；**交叉驗證**誠實估計測試誤差以選超參數；訓練誤差永遠偏袒複雜模型，不可用於容量決策。
- 這些觀念驅動機器學習中每一個容量決策；double descent 提醒我們曲線形狀依模型族而變，但分解式的詞彙永遠是基礎。

10.6 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 10.1 (基礎)　一位學生記錄每週讀書時數 x 與小考分數 y（滿分 10）：$(0,1),(1,3),(2,4),(3,4),(4,8)$。(a) 求最小平方直線 $\hat{y} = \hat{w}x + \hat{b}$。(b) 解讀 $\hat{w}$ 與 $\hat{b}$。(c) 預測 $x = 2.5$ 小時的分數，並說明為何預測 $x = 40$ 不可信。

練習 10.2 (基礎)　延續練習 10.1：(a) 計算五個擬合值與殘差，驗證殘差總和為 0。(b) 計算 SSE、SST 與 R^2。(c) 用一句話解釋這個 R^2 的意義；R^2 高是否代表「讀書導致高分」？

練習 10.3 (基礎)　信用卡盜刷偵測模型：$\mathbb{P}(\text{盜刷} \mid \boldsymbol{x}) = \sigma(2x_1 - x_2 - 0.5)$，其中 x_1 = 異常交易指標、x_2 = 帳戶使用年資（皆已標準化）。(a) 求 $\boldsymbol{x} = (1,1)$ 的盜刷機率並分類（門檻 0.5）。(b) 求 $\boldsymbol{x} = (0.5,2)$ 的盜刷機率並分類。(c) 寫出決策邊界方程式，並說明它是平面上的什麼圖形。(d) 盜刷機率要達 0.9 以上，線性分數 z 需滿足什麼條件？

練習 10.4 (基礎)　GLM 配對題。分布：(A1) 高斯、(A2) 卜瓦松、(A3) 白努利、(A4) 類別（K 類）。連結：(B1) logit、(B2) 恆等、(B3) 多類 logit（softmax）、(B4) 對數。損失：(C1) 卜瓦松 NLL、(C2) 多類交叉熵、(C3) 平方誤差、(C4) 二元交叉熵。任務：(D1) 影像十類辨識、(D2) 房價預測、(D3) 每日客訴件數、(D4) 垃圾郵件過濾。(a) 為每個分布配上正確的連結、損失與任務。(b) 給出卜瓦松迴歸採用對數連結而非恆等連結的兩個理由。

練習 10.5 (進階)　設 $y_i = \boldsymbol{w}^\top \boldsymbol{x}_i + b + \varepsilon_i$，$\varepsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$。(a) 寫出對數概似函數，證明 $(\boldsymbol{w}, b)$ 的 MLE 與最小平方解相同。(b) 求 σ^2 的 MLE。(c) 說明為什麼「換一個雜訊分布，損失就不再是平方誤差」——若 ε_i 改為 Laplace 分布（密度 $\propto e^{-|t|/s}$），MLE 對應哪種損失？

練習 10.6 (進階)　(a) 從白努利概似 $L = \prod_i p_i^{y_i}(1-p_i)^{1-y_i}$ 出發，推導交叉熵損失。(b) 說明單筆損失在 $y=1$ 時為 $-\log p$，並解釋「信心滿滿卻答錯」時損失如何爆炸。(c) 若改用平方損失 $(p-y)^2$（$p = \sigma(z)$），計算 $\partial \ell / \partial z$，並說明在「$p \approx 0$ 但 $y = 1$」時會發生什麼事；這給了交叉熵什麼優勢？

練習 10.7 (進階)　卜瓦松迴歸模型 $\log \mu = 0.8 + 0.3x$（x = 促銷檔次）。(a) 求 $x = 0$ 時 $\mathbb{P}(Y = 3)$。(b)「每加一檔促銷，平均來電數增加 35%」這句話從何而來？(c) 從單筆 NLL $\mu - y \log \mu + \log(y!)$ 出發，證明對 w 的梯度為 $(\mu - y)x$。(d) 若改用恆等連結 $\mu = b + wx$，會出現什麼數學上的麻煩？

練習 10.8 (進階)　偏差–變異數情境判斷。對以下每個情境，指出主要問題（高偏差 / 高變異數 / 已達標 / 正則化失衡）並開出處方：(a) 訓練錯誤率 28%、驗證 29%，目標 5%。(b) 訓練錯誤率 1%、驗證 16%。(c) 訓練錯誤率 0.5%、驗證 0.7%，目標 1%。(d) 把 λ 大幅調高後，訓練錯誤率從 3% 升到 15%、驗證從 9% 升到 16%。並回答：哪些情境中「蒐集更多資料」幫得上忙？

練習 10.9 (挑戰)　正規方程的完整推導。(a) 逐分量證明梯度恆等式 $\nabla_{\boldsymbol{w}}(\boldsymbol{w}^\top \boldsymbol{a}) = \boldsymbol{a}$ 與 $\nabla_{\boldsymbol{w}}(\boldsymbol{w}^\top A \boldsymbol{w}) = (A + A^\top)\boldsymbol{w}$（$A$ 對稱時為 $2A\boldsymbol{w}$）。(b) 展開 $\|\boldsymbol{y} - X\boldsymbol{w}\|^2$ 並用 (a) 推出正規方程 $X^\top X \boldsymbol{w} = X^\top \boldsymbol{y}$。(c) 證明 Hessian $2X^\top X$ 半正定，故駐點是全域最小。(d) $X^\top X$ 何時不可逆？舉一個具體的特徵設計錯誤為例，並說明此時最小平方解的集合長什麼樣。

練習 10.10 (挑戰)　脊迴歸閉式解。(a) 對 $J(\boldsymbol{w}) = \|\boldsymbol{y} - X\boldsymbol{w}\|^2 + \lambda \|\boldsymbol{w}\|^2$ 求梯度，推出 $\hat{\boldsymbol{w}}_\lambda = (X^\top X + \lambda I)^{-1} X^\top \boldsymbol{y}$。(b) 證明對任何 $\lambda > 0$，$X^\top X + \lambda I$ 必可逆（提示：考慮 $\boldsymbol{v}^\top (X^\top X + \lambda I)\boldsymbol{v}$）。(c) 在單一特徵、資料已中心化的情形，證明 $\hat{w}_\lambda = \frac{S_{xx}}{S_{xx} + \lambda} \hat{w}_{\text{OLS}}$。(d) 從 $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \tau^2 I)$ 的先驗出發，證明 MAP 估計正是脊迴歸，並給出 λ 與 σ^2, τ^2 的關係。

練習 10.11 (挑戰)　double descent 概念題。(a) 描述古典的測試誤差曲線形狀及其成因。(b) 描述 double descent 現象：當參數量跨過「恰好能完美擬合訓練資料」的內插門檻（interpolation threshold）後，測試誤差如何變化？(c) 這與偏差–變異數分解矛盾嗎？為什麼？(提示：分解式是恆等式；改變的是什麼？) (d) 給出一條實務準則：在 double descent 存在的世界裡，該如何決定模型大小？

練習 10.12 (挑戰·程式)　用 NumPy 完成：(a) 產生 $y = X\boldsymbol{w}_{\text{true}} + \varepsilon$（$\boldsymbol{w}_{\text{true}} = (1, 2, -0.5)$，含截距欄），用 `np.linalg.lstsq` 估計係數並驗證接近真值。(b) 對 $\lambda \in \{0, 10, 100, 1000\}$ 計算脊迴歸解，觀察係數範數如何隨 λ 縮小。(c) 產生一維邏輯斯迴歸資料（真實參數 $w = 3$、$b = -1$），用「預測減實際」的梯度下降擬合，驗證收斂到真值附近。

10.7 練習題詳解

練習 10.1 詳解　(a) $\bar{x} = 2$、$\bar{y} = 20/5 = 4$。離差為 $x_i - \bar{x} = (-2, -1, 0, 1, 2)$、$y_i - \bar{y} = (-3, -1, 0, 0, 4)$，故

$$S_{xy} = 6 + 1 + 0 + 0 + 8 = 15, \qquad S_{xx} = 4 + 1 + 0 + 1 + 4 = 10,$$

得 $\hat{w} = 15/10 = 1.5$，$\hat{b} = 4 - 1.5 \times 2 = 1$，$\hat{y} = 1.5x + 1$。
(b) 每多讀一小時，分數平均提高 1.5 分；完全不讀書的預期分數為 1 分（截距）。
(c) $\hat{y}(2.5) = 1.5 \times 2.5 + 1 = 4.75$ 分——落在資料範圍 $[0, 4]$ 內，屬內插，可信。$x = 40$ 給出 $\hat{y} = 61$ 分，遠超滿分 10 分：資料範圍外的外插毫無根據，線性關係（且分數有上限）早已失效。

練習 10.2 詳解　(a) 擬合值 $\hat{y} = 1.5x + 1$ 依序為 $1, 2.5, 4, 5.5, 7$；殘差 $e_i = y_i - \hat{y}_i$ 為

$$0, \quad 0.5, \quad 0, \quad -1.5, \quad 1,$$

總和 $0 + 0.5 + 0 - 1.5 + 1 = 0$，驗證通過（含截距的最小平方必然如此）。
(b)

$$\text{SSE} = 0 + 0.25 + 0 + 2.25 + 1 = 3.5, \qquad \text{SST} = 9 + 1 + 0 + 0 + 16 = 26,$$

$$R^2 = 1 - \frac{3.5}{26} = \frac{22.5}{26} \approx 0.865.$$

(c) 分數變異的約 86.5% 可由讀書時數的線性關係解釋。但 R^2 只度量**關聯**不度量**因果**：可能是「本來就擅長此科的人讀得更起勁」，或有第三變數（睡眠、興趣）同時驅動兩者。要談因果需要實驗設計或因果推論工具，迴歸本身給不了。

練習 10.3 詳解　(a) $z = 2(1) - 1 - 0.5 = 0.5$，

$$p = \sigma(0.5) = \frac{1}{1 + e^{-0.5}} = \frac{1}{1 + 0.6065} \approx 0.622 > 0.5,$$

判為盜刷。（對數勝算 0.5，勝算 $e^{0.5} \approx 1.65$。）
(b) $z = 2(0.5) - 2 - 0.5 = -1.5$，$p = \sigma(-1.5) = 1 - \sigma(1.5) \approx 1 - 0.818 = 0.182 < 0.5$，判為正常。
(c) 決策邊界為 $z = 0$：$2x_1 - x_2 - 0.5 = 0$，即直線 $x_2 = 2x_1 - 0.5$。直線上方（x_2 較大，老帳戶）判為正常、下方判為盜刷——邏輯斯迴歸是線性分類器。
(d) $\sigma(z) \geq 0.9 \iff z \geq \log \dfrac{0.9}{0.1} = \log 9 \approx 2.197$。

練習 10.4 詳解　(a) 配對結果：

分布	連結	損失	任務
(A1) 高斯	(B2) 恆等	(C3) 平方誤差	(D2) 房價預測
(A3) 白努利	(B1) logit	(C4) 二元交叉熵	(D4) 垃圾郵件過濾
(A2) 卜瓦松	(B4) 對數	(C1) 卜瓦松 NLL	(D3) 每日客訴件數
(A4) 類別（K 類）	(B3) softmax	(C2) 多類交叉熵	(D1) 影像十類辨識

(b) 理由一：卜瓦松均值必須為正，$\mu = e^{\eta}$ 對任何 $\eta \in \mathbb{R}$ 自動保證 $\mu > 0$；恆等連結 $\mu = \eta$ 可能給出負的均值，模型自相矛盾。理由二：對數連結給出**乘法效應**的自然解讀（每單位 x 讓均值乘上 e^{w}），符合計數現象常見的倍增結構；此外它是卜瓦松的典型連結，使 NLL 對參數為凸、梯度呈「預測減實際」的乾淨形式。

練習 10.5 詳解　(a) 各筆獨立且 $y_i \mid \boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{w}^{\top}\boldsymbol{x}_i + b, \sigma^2)$：

$$\ell(\boldsymbol{w}, b, \sigma^2) = -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}\big(y_i - \boldsymbol{w}^{\top}\boldsymbol{x}_i - b\big)^2.$$

固定 σ^2 時，第一項是常數、第二項係數為負，最大化 ℓ 等價於最小化 $\mathrm{SSE}(\boldsymbol{w}, b)$——MLE 即最小平方解。
(b) 記最小化後的殘差平方和為 $\widehat{\mathrm{SSE}}$，令 $\partial\ell/\partial\sigma^2 = -\dfrac{n}{2\sigma^2} + \dfrac{\widehat{\mathrm{SSE}}}{2\sigma^4} = 0$，解得 $\hat{\sigma}^2 = \widehat{\mathrm{SSE}}/n$（可驗證是最大值點）。
(c) 損失是「負對數雜訊密度」的化身。Laplace 密度 $\propto e^{-|t|/s}$ 的負對數是 $|t|/s$ 加常數，故 MLE 等價於最小化**絕對誤差和** $\sum_i |y_i - \boldsymbol{w}^{\top}\boldsymbol{x}_i - b|$（最小絕對偏差迴歸，對離群點更穩健）。高斯給 L^2、Laplace 給 L^1：換分布就換損失，正是「機率模型與損失函數是同一件事的兩面」。

練習 10.6 詳解　(a) 取負對數：

$$-\log L = -\sum_{i=1}^{n}\Big[y_i \log p_i + (1-y_i)\log(1-p_i)\Big],$$

即二元交叉熵。最大化概似 = 最小化交叉熵。

(b) $y=1$ 時只剩 $-\log p$。若模型自信地答錯（$p \to 0$ 但 $y=1$），$-\log p \to \infty$：交叉熵對「過度自信的錯誤」施以無上限的懲罰，逼模型輸出誠實的機率。

(c) 平方損失 $\ell = (p-y)^2$ 經 $p=\sigma(z)$ 的連鎖法則：

$$\frac{\partial \ell}{\partial z} = 2(p-y)\,\sigma'(z) = 2(p-y)\,p(1-p).$$

在 $p \approx 0$、$y=1$（大錯特錯）時，因子 $p(1-p) \approx 0$ 使梯度趨近 0——模型「站在錯誤裡動彈不得」（梯度消失），且此損失對 z 非凸。交叉熵的梯度是 $p-y$，此時約為 -1：錯得越離譜、修正力道越大，而且目標函數是凸的。這就是分類問題採用交叉熵而非平方損失的核心理由。

練習 10.7 詳解　(a) $x=0$ 時 $\mu = e^{0.8} \approx 2.2255$。

$$\mathbb{P}(Y=3) = e^{-\mu}\frac{\mu^3}{3!} = e^{-2.2255} \times \frac{11.02}{6} \approx 0.1080 \times 1.837 \approx 0.198.$$

(b) x 每增加 1，$\log\mu$ 增加 0.3，故 μ 乘上 $e^{0.3} \approx 1.3499$——增加約 35%。對數連結把「線性的加法」翻譯成「均值的乘法」。

(c) 以 $\mu = e^z$、$z = b + wx$ 代入單筆 NLL（略去與參數無關的 $\log(y!)$）：

$$\text{NLL}(z) = e^z - yz \quad\Longrightarrow\quad \frac{\partial \text{NLL}}{\partial z} = e^z - y = \mu - y,$$

再乘上 $\partial z/\partial w = x$ 得 $(\mu - y)x$。全體資料相加即 $\sum_i (\mu_i - y_i)x_i$。

(d) 恆等連結 $\mu = b + wx$ 無法保證 $\mu > 0$：只要 x 落在某個範圍，模型就宣稱「平均來電數為負」，概似（含 $\log\mu$）直接未定義；優化時必須加約束，凸性與數值穩定性也一併惡化。對數連結一勞永逸地解決這些問題。

練習 10.8 詳解　(a) 訓練與驗證誤差幾乎相同、皆遠高於目標：**高偏差（欠擬合）**。處方：加大模型容量、增加特徵或非線性轉換、減弱正則化、訓練更久。

(b) 訓練近乎完美、驗證落後 15 個百分點：**高變異數（過度擬合）**。處方：更多資料、資料增強、加強正則化（調大 λ、dropout）、縮小模型、early stopping。

(c) 兩個誤差都低於目標 1%：**已達標**。收工部署，之後監控資料飄移即可；再堆容量只是徒增過度擬合風險。

(d) 訓練誤差暴升且與驗證誤差貼近：**正則化過強，被推入欠擬合**。處方：把 λ 調回中間值——用交叉驗證在 λ 網格上掃描，找 U 形谷底。

「蒐集更多資料」只對 (b) 有效：更多資料壓低變異數，但對偏差無能為力——(a)、(d) 的瓶頸是模型太僵硬（或被懲罰得太僵硬），資料再多也學不出它表達不了的模式。

練習 10.9 詳解　(a) 第一式：$\dfrac{\partial}{\partial w_j}\sum_k a_k w_k = a_j$，收集所有分量即 $\nabla_{\boldsymbol{w}}(\boldsymbol{w}^\top \boldsymbol{a}) = \boldsymbol{a}$。第二式：

$$\frac{\partial}{\partial w_j}\sum_{k,l} A_{kl} w_k w_l = \sum_l A_{jl} w_l + \sum_k A_{kj} w_k = (A\boldsymbol{w})_j + (A^\top \boldsymbol{w})_j,$$

即 $\nabla = (A + A^\top)\boldsymbol{w}$；$A$ 對稱時為 $2A\boldsymbol{w}$。

(b) 展開

$$L(\boldsymbol{w}) = (\boldsymbol{y} - X\boldsymbol{w})^\top(\boldsymbol{y} - X\boldsymbol{w}) = \boldsymbol{y}^\top\boldsymbol{y} - 2\boldsymbol{w}^\top X^\top \boldsymbol{y} + \boldsymbol{w}^\top (X^\top X)\boldsymbol{w}.$$

以 (a) 求梯度（$X^\top X$ 對稱）：$\nabla L = -2X^\top \boldsymbol{y} + 2X^\top X\boldsymbol{w}$；設為零即 $X^\top X\boldsymbol{w} = X^\top \boldsymbol{y}$。

(c) 對任意 $\boldsymbol{v}$：$\boldsymbol{v}^\top(2X^\top X)\boldsymbol{v} = 2\|X\boldsymbol{v}\|^2 \ge 0$，Hessian 半正定，$L$ 為凸函數，故駐點是全域最小值。

(d) $X^\top X$ 不可逆 $\iff$ X 的行（特徵）線性相依。具體例：同一溫度以攝氏、華氏各放一欄（$x_F = 1.8\,x_C + 32$，與截距欄合成線性相依）。此時解不唯一：沿 $X^\top X$ 零空間平移，擬合值與訓練誤差完全不變，解集是仿射子空間。`lstsq` 以最小範數解回應；脊迴歸加上 λI 後恢復唯一性。

練習 10.10 詳解　(a) $J(\boldsymbol{w}) = \|\boldsymbol{y} - X\boldsymbol{w}\|^2 + \lambda \boldsymbol{w}^\top \boldsymbol{w}$。梯度為

$$\nabla J = -2X^\top \boldsymbol{y} + 2X^\top X\boldsymbol{w} + 2\lambda \boldsymbol{w} = \boldsymbol{0} \quad\Longrightarrow\quad (X^\top X + \lambda I)\,\boldsymbol{w} = X^\top \boldsymbol{y},$$

故 $\hat{\boldsymbol{w}}_\lambda = (X^\top X + \lambda I)^{-1} X^\top \boldsymbol{y}$。Hessian $2(X^\top X + \lambda I)$ 正定（見 (b)），故為唯一全域最小。

(b) 對任意 $\boldsymbol{v} \neq \boldsymbol{0}$：

$$\boldsymbol{v}^\top (X^\top X + \lambda I)\boldsymbol{v} = \|X\boldsymbol{v}\|^2 + \lambda\|\boldsymbol{v}\|^2 \ge \lambda\|\boldsymbol{v}\|^2 > 0,$$

矩陣正定故可逆——不論 X 是否共線。這就是脊迴歸「治好」共線性的原因。

(c) 中心化單特徵時 $X^\top X = S_{xx}$、$X^\top \boldsymbol{y} = S_{xy}$（皆為純量），故

$$\hat{w}_\lambda = \frac{S_{xy}}{S_{xx} + \lambda} = \frac{S_{xx}}{S_{xx} + \lambda} \cdot \frac{S_{xy}}{S_{xx}} = \frac{S_{xx}}{S_{xx} + \lambda}\,\hat{w}_{\text{OLS}},$$

縮小因子 $S_{xx}/(S_{xx} + \lambda) \in (0, 1]$。

(d) 後驗 $\propto$ 概似 $\times$ 先驗：

$$p(\boldsymbol{w} \mid \text{資料}) \propto \exp\!\Big(-\frac{\|\boldsymbol{y} - X\boldsymbol{w}\|^2}{2\sigma^2}\Big)\exp\!\Big(-\frac{\|\boldsymbol{w}\|^2}{2\tau^2}\Big).$$

取負對數並乘以 $2\sigma^2$（不影響最小化位置）：

$$\hat{\boldsymbol{w}}_{\text{MAP}} = \arg\min_{\boldsymbol{w}}\ \|\boldsymbol{y} - X\boldsymbol{w}\|^2 + \frac{\sigma^2}{\tau^2}\|\boldsymbol{w}\|^2,$$

正是脊迴歸，且 $\lambda = \sigma^2/\tau^2$：雜訊越大或先驗越緊（$\tau^2$ 越小），懲罰越重。「正則化 = 先驗」由此坐實。

練習 10.11 詳解　(a) 古典圖像：複雜度低時偏差高（欠擬合）、高時變異數高（過度擬合），期望測試誤差呈 U 形，最佳點在中間（圖 10.4）。
(b) 參數量跨過**內插門檻**（恰能把訓練資料背到零誤差）時，測試誤差先衝到峰值，隨後在超大參數化區再度下降——曲線出現第二次下降，故名 double descent。
(c) 不矛盾。分解式 $\mathbb{E}[(y-\hat{f})^2] = \text{偏差}^2 + \text{變異數} + \sigma^2$ 是代數恆等式，在任何複雜度都成立；會變的是**偏差與變異數隨複雜度變化的形狀**。古典 U 形假設「變異數單調上升」，這在超大參數化區失效：內插解有無窮多個，訓練演算法（如 SGD、最小範數解）隱性偏好平滑的那一個，這種**隱式正則化**使變異數在門檻後反而回落。
(d) 實務準則：不要用「參數多寡」推斷泛化好壞，永遠用驗證集或交叉驗證直接測量泛化誤差；在資源允許下，「加大模型 + 適當正則化 + early stopping」常是安全的預設。

練習 10.12 詳解　參考程式與輸出：

```python
import numpy as np
rng = np.random.default_rng(1)
# (a) lstsq 恢復 w_true
n = 300
X = np.column_stack([np.ones(n), rng.normal(size=(n, 2))])
w_true = np.array([1.0, 2.0, -0.5])
y = X @ w_true + rng.normal(scale=0.5, size=n)
w_hat, *_ = np.linalg.lstsq(X, y, rcond=None)
print(np.round(w_hat, 3))        # 約 [1.0, 2.0, -0.5]
for lam in [0, 10, 100, 1000]:      # (b) 脊迴歸：lambda 越大係數越小
    w_r = np.linalg.solve(X.T @ X + lam * np.eye(3), X.T @ y)
    print(lam, np.round(w_r, 3), round(float(np.linalg.norm(w_r)), 3))
def sigmoid(z):                     # (c) 一維邏輯斯迴歸的梯度下降
    return 1 / (1 + np.exp(-z))
xb = rng.normal(size=400)
yb = rng.binomial(1, sigmoid(3 * xb - 1))
w, b = 0.0, 0.0
for _ in range(3000):
    p = sigmoid(w * xb + b)
    w -= 0.5 * np.mean((p - yb) * xb)   # 預測減實際，加權平均
    b -= 0.5 * np.mean(p - yb)
print(round(w, 2), round(b, 2))         # 接近 3 與 -1
```

(a) 印出約 $[1.0,\ 2.0,\ -0.5]$：最小平方精準恢復真實權重。(b) 隨 λ 增大，三個係數同步縮向 0、範數單調下降。(c) 梯度下降收斂到約 $(3.0, -1.0)$，親證例 10.3 的「預測減實際」梯度。

模型—損失—優化：你已握有監督式學習的完整骨架，之後的每一個學習演算法都是它的變奏。