

機器學習參考手冊

Afshine AMIDI 和 Shervine AMIDI

December 15, 2019

1 監督學習

翻譯: kevingo. 審閱: 詹志傑.

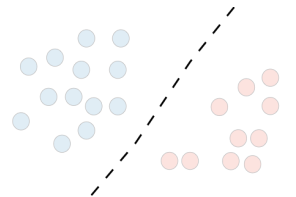
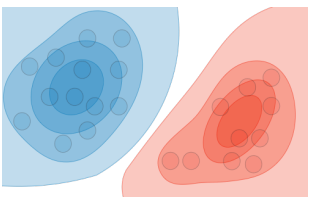
1.1 監督式學習介紹

給定一組資料點 $\{x^{(1)}, \dots, x^{(m)}\}$ ，以及對應的一組輸出 $\{y^{(1)}, \dots, y^{(m)}\}$ ，我們希望建立一個分類器，用來學習如何從 x 來預測 y

□ **預測的種類** – 根據預測的種類不同，我們將預測模型分為底下幾種：

	迴歸	分類器
結果	連續	類別
範例	線性迴歸	邏輯迴歸, 支援向量機(SVM), 單純貝式分類器

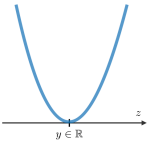
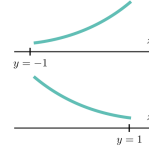
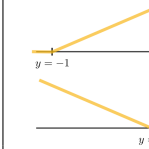
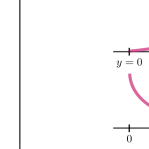
□ **模型種類** – 不同種類的模型歸納如下表：

	判別模型	生成模型
目標	直接估計 $P(y x)$	先估計 $P(x y)$ ，然後推論出 $P(y x)$
學到什麼	決策分界線	資料的機率分佈
示意圖		
範例	迴歸, 支援向量機(SVM)	高斯判別分析(GDA), 單純貝氏(Naive Bayes)

1.2 符號及一般概念

□ **假設** – 我們使用 h_θ 來代表所選擇的模型，對於給定的輸入資料 $x^{(i)}$ ，模型預測的輸出是 $h_\theta(x^{(i)})$

□ **損失函數** – 損失函數是一個函數 $L: (z, y) \in \mathbb{R} \times Y \mapsto L(z, y) \in \mathbb{R}$ ，目的在於計算預測值 z 和實際值 y 之間的差距。底下是一些常見的損失函數：

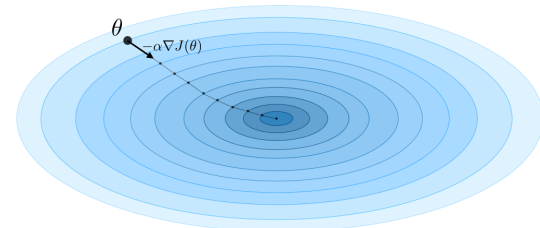
最小平方方法	Logistic 損失函數	Hinge 損失函數	交叉熵
$\frac{1}{2}(y - z)^2$	$\log(1 + \exp(-yz))$	$\max(0, 1 - yz)$	$-\left[y \log(z) + (1 - y) \log(1 - z)\right]$
			
線性迴歸	邏輯迴歸	支援向量機(SVM)	神經網路

□ **代價函數** – 代價函數 J 通常用來評估一個模型的表現，它可以透過損失函數 L 來定義：

$$J(\theta) = \sum_{i=1}^m L(h_\theta(x^{(i)}), y^{(i)})$$

□ **梯度下降** – 使用 $\alpha \in \mathbb{R}$ 表示學習速率，我們透過學習速率和代價函數來使用梯度下降的方法找出網路參數更新的方法可以表示為：

$$\theta \leftarrow \theta - \alpha \nabla J(\theta)$$



注意：隨機梯度下降法(SGD) 使用每一個訓練資料來更新參數。而批次梯度下降法則是透過一個批次的訓練資料來更新參數。

□ **概似估計** – 在給定參數 θ 的條件下，一個模型 $L(\theta)$ 的概似估計的目的是透過最大概似估計法來找到最佳的參數。實務上，我們會使用對數概似估計函數(log-likelihood) $\ell(\theta) = \log(L(\theta))$ ，會比較容易最佳化。如下：

$$\theta^{\text{opt}} = \arg \max_{\theta} L(\theta)$$

□ **牛頓演算法** – 牛頓演算法是一個數值方法，目的在於找到一個 θ ，讓 $\ell'(\theta) = 0$ 。其更新的規則為：

$$\theta \leftarrow \theta - \frac{\ell'(\theta)}{\ell''(\theta)}$$

注意：多維度正規化的方法，或又被稱之為牛頓-拉弗森(Newton-Raphson) 演算法，是透過以下的規則更新：

$$\theta \leftarrow \theta - \left(\nabla_{\theta}^2 \ell(\theta)\right)^{-1} \nabla_{\theta} \ell(\theta)$$

1.2.1 線性迴歸

我們假設 $y|x; \theta \sim \mathcal{N}(\mu, \sigma^2)$

□ **正規方程法** – 我們使用 X 代表矩陣，讓代價函數最小的 θ 值有一個封閉解，如下：

$$\theta = (X^T X)^{-1} X^T y$$

□ **最小均方演算法 (LMS)** – 我們使用 α 表示學習速率，針對 m 個訓練資料，透過最小均方演算法的更新規則，或是叫做 Widrow-Hoff 學習法如下：

$$\forall j, \quad \theta_j \leftarrow \theta_j + \alpha \sum_{i=1}^m [y^{(i)} - h_{\theta}(x^{(i)})] x_j^{(i)}$$

注意：這個更新的規則是梯度上升的一種特例

□ **LWR** – 局部加權迴歸，又稱為 LWR，是線性迴歸的變形，通過 $w^{(i)}(x)$ 對其成本函數中的每個訓練樣本進行加權，其中參數 $\tau \in \mathbb{R}$ 定義為：

$$w^{(i)}(x) = \exp\left(-\frac{(x^{(i)} - x)^2}{2\tau^2}\right)$$

1.2.2 分類與邏輯迴歸

□ **Sigmoid 函數** – Sigmoid 函數 g ，也可以稱為邏輯函數定義如下：

$$\forall z \in \mathbb{R}, \quad g(z) = \frac{1}{1 + e^{-z}} \in [0, 1]$$

□ **邏輯迴歸** – 我們假設 $y|x; \theta \sim \text{Bernoulli}(\phi)$ ，請參考以下：

$$\phi = p(y = 1|x; \theta) = \frac{1}{1 + \exp(-\theta^T x)} = g(\theta^T x)$$

注意：對於這種情況的邏輯迴歸，並沒有一個封閉解

□ **Softmax 迴歸** – Softmax 迴歸又稱做多分類邏輯迴歸，目的是用在超過兩個以上的分類時的迴歸使用。按照慣例，我們設定 $\theta_K = 0$ ，讓每一個類別的 Bernoulli 參數 ϕ_i 等同於：

$$\phi_i = \frac{\exp(\theta_i^T x)}{\sum_{j=1}^K \exp(\theta_j^T x)}$$

1.2.3 廣義線性模型

□ **指數族分佈** – 一個分佈如果可以透過自然參數(或稱之為正準參數或連結函數) η 、充分統計量 $T(y)$ 和對數區分函數(log-partition function) $a(\eta)$ 來表示時，我們就稱這個分佈是屬於指數族分佈。該分佈可以表示如下：

$$p(y; \eta) = b(y) \exp(\eta^T T(y) - a(\eta))$$

注意：我們經常讓 $T(y) = y$ ，同時， $\exp(-a(\eta))$ 可以看成是一個正規化的參數，目的在於讓機率總和為一。

底下是最常見的指數分佈：

分佈	η	$T(y)$	$a(\eta)$	$b(y)$
白努利(Bernoulli)	$\log\left(\frac{\phi}{1-\phi}\right)$	y	$\log(1 + \exp(\eta))$	1
高斯(Gaussian)	μ	y	$\frac{\eta^2}{2}$	$\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{y^2}{2}\right)$
卜瓦松(Poisson)	$\log(\lambda)$	y	e^η	$\frac{1}{y!}$
幾何(Geometric)	$\log(1 - \phi)$	y	$\log\left(\frac{e^\eta}{1 - e^\eta}\right)$	1

□ **廣義線性模型的假設** – 廣義線性模型(GLM) 的目的在於，給定 $x \in \mathbb{R}^{n+1}$ ，要預測隨機變數 y ，同時它依賴底下三個假設：

$$(1) \quad y|x; \theta \sim \text{ExpFamily}(\eta) \quad (2) \quad h_{\theta}(x) = E[y|x; \theta] \quad (3) \quad \eta = \theta^T x$$

注意：最小平方法和邏輯迴歸是廣義線性模型的一種特例

1.3 支援向量機

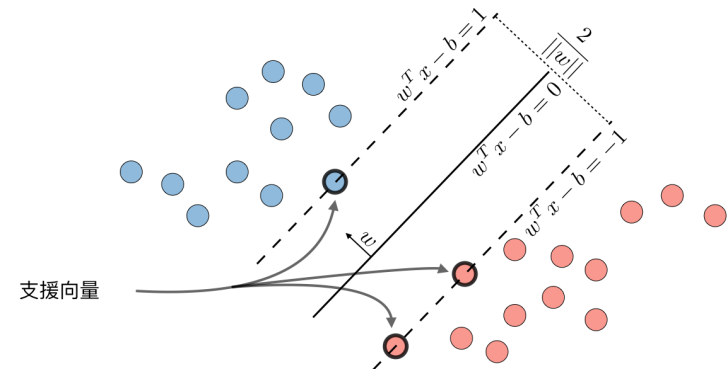
支援向量機的目的在於找到一條決策邊界和資料樣本之間最大化最小距離的線

□ **最佳的邊界分類器** – 最佳的邊界分類器可以表示為：

$$h(x) = \text{sign}(w^T x - b)$$

其中， $(w, b) \in \mathbb{R}^n \times \mathbb{R}$ 是底下最佳化問題的答案：

$$\min \frac{1}{2} \|w\|^2 \quad \text{使得} \quad y^{(i)}(w^T x^{(i)} - b) \geq 1$$



注意：該條直線定義為 $w^T x - b = 0$

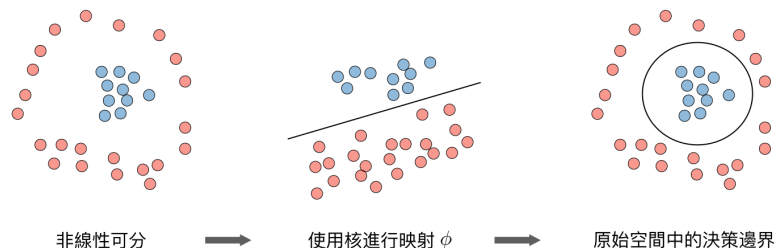
□ **Hinge 損失函數** – Hinge 損失函數用在支援向量機上，定義如下：

$$L(z, y) = [1 - yz]_+ = \max(0, 1 - yz)$$

□ **核(函數)** – 給定特徵轉換 ϕ ，我們定義核(函數) K 為：

$$K(x, z) = \phi(x)^T \phi(z)$$

實務上， $K(x, z) = \exp\left(-\frac{\|x-z\|^2}{2\sigma^2}\right)$ 定義的核(函數) K ，一般稱作高斯核(函數)。這種核(函數)經常被使用



注意：我們使用“核(函數)技巧”來計算代價函數時，不需要真正的知道映射函數 ϕ ，這個函數非常複雜。相反的，我們只需要知道 $K(x, z)$ 的值即可。

□ **Lagrangian** – 我們將Lagrangian $\mathcal{L}(w, b)$ 定義如下：

$$\mathcal{L}(w, b) = f(w) + \sum_{i=1}^l \beta_i h_i(w)$$

注意：係數 β_i 稱為Lagrange 乘數

1.4 生成學習

生成模型嘗試透過預估 $P(x|y)$ 來學習資料如何生成，而我們可以透過貝氏定理來預估 $P(y|x)$

1.4.1 高斯判別分析

□ **設定** – 高斯判別分析針對 y 、 $x|y=0$ 和 $x|y=1$ 進行以下假設：

$$y \sim \text{Bernoulli}(\phi)$$

$$x|y=0 \sim \mathcal{N}(\mu_0, \Sigma) \quad \text{和} \quad x|y=1 \sim \mathcal{N}(\mu_1, \Sigma)$$

□ **估計** – 底下的表格總結了我們在最大似估計時的估計值：

$\hat{\phi}$	$\hat{\mu}_j \quad (j=0,1)$	$\hat{\Sigma}$
$\frac{1}{m} \sum_{i=1}^m 1_{\{y^{(i)}=1\}}$	$\frac{\sum_{i=1}^m 1_{\{y^{(i)}=j\}} x^{(i)}}{\sum_{i=1}^m 1_{\{y^{(i)}=j\}}}$	$\frac{1}{m} \sum_{i=1}^m (x^{(i)} - \mu_{y^{(i)}})(x^{(i)} - \mu_{y^{(i)}})^T$

1.4.2 單純貝氏

□ **假設** – 單純貝氏模型會假設每個資料點的特徵都是獨立的。

$$P(x|y) = P(x_1, x_2, \dots | y) = P(x_1|y)P(x_2|y)\dots = \prod_{i=1}^n P(x_i|y)$$

□ **解決方法** – 最大化對數概似估計來給出以下解答， $k \in \{0,1\}, l \in [1, L]$

$$P(y=k) = \frac{1}{m} \times \#\{j|y^{(j)}=k\}$$

$$P(x_i=l|y=k) = \frac{\#\{j|y^{(j)}=k \text{ 和 } x_i^{(j)}=l\}}{\#\{j|y^{(j)}=k\}}$$

注意：單純貝氏廣泛應用在文字分類和垃圾信件偵測上

1.5 基於樹狀結構的學習和整體學習

這些方法可以應用在迴歸或分類問題上

□ **CART** – 分類與迴歸樹(CART)，通常稱之為決策數，可以被表示為二元樹。它的優點是具有可解釋性。

□ **隨機森林** – 這是一個基於樹狀結構的方法，它使用大量經由隨機挑選的特徵所建構的決策樹。與單純的決策樹不同，它通常具有高度不可解釋性，但它的效能通常很好，所以是一個相當流行的演算法。

注意：隨機森林是一種整體學習方法

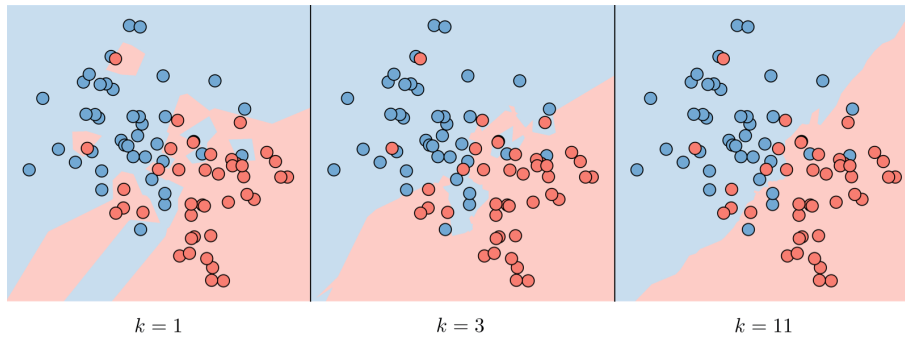
□ **增強學習(Boosting)** – 增強學習方法的概念是結合數個弱學習模型來變成強學習模型。主要的分類如下：

自適應增強	梯度增強
- 在下一輪的提升步驟中，錯誤的部分會被賦予較高的權重 - "Adaboost"	- 弱學習器會負責訓練剩下的錯誤

1.6 其他非參數方法

□ **k-最近鄰** – k-最近鄰演算法，又稱之為k-NN，是一個非參數的方法，其中資料點的決定是透過訓練集中最近的k 個鄰居而決定。它可以用在分類和迴歸問題上。

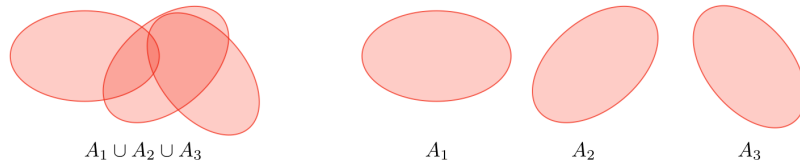
注意：參數k 的值越大，偏差越大。k 的值越小，變異越大。



1.7 學習理論

□ **聯集上界** – 令 A_1, \dots, A_k 為 k 個事件，我們有：

$$P(A_1 \cup \dots \cup A_k) \leq P(A_1) + \dots + P(A_k)$$



□ **霍夫丁不等式** – 令 Z_1, \dots, Z_m 為 m 個從參數 ϕ 的白努利分佈中抽出的獨立同分佈(iid) 的變數。令 $\hat{\phi}$ 為其樣本平均、固定 $\gamma > 0$ ，我們可以得到：

$$P(|\phi - \hat{\phi}| > \gamma) \leq 2 \exp(-2\gamma^2 m)$$

注意：這個不等式也被稱之為 *Chernoff* 界線

□ **訓練誤差** – 對於一個分類器 h ，我們定義訓練誤差為 $\hat{\epsilon}(h)$ ，也可以稱為經驗風險或經驗誤差。定義如下：

$$\hat{\epsilon}(h) = \frac{1}{m} \sum_{i=1}^m 1_{\{h(x^{(i)}) \neq y^{(i)}\}}$$

□ **可能近似正確(PAC)** – PAC 是一個框架，有許多學習理論都證明其有效性。它包含以下假設：

- 訓練和測試資料集具有相同的分佈
- 訓練資料集之間彼此獨立

□ **打散(Shattering)** – 給定一個集合 $S = \{x^{(1)}, \dots, x^{(d)}\}$ 以及一組分類器的集合 \mathcal{H} ，如果對於任何一組標籤 $\{y^{(1)}, \dots, y^{(d)}\}$ ， \mathcal{H} 都能打散 S ，定義如下：

$$\exists h \in \mathcal{H}, \quad \forall i \in [1, d], \quad h(x^{(i)}) = y^{(i)}$$

□ **上限定理** – 令 \mathcal{H} 是一個有限假設類別，使 $|\mathcal{H}| = k$ 且令 δ 和樣本大小 m 固定，結著，在機率至少為 $1 - \delta$ 的情況下，我們得到：

$$\epsilon(\hat{h}) \leq \left(\min_{h \in \mathcal{H}} \epsilon(h) \right) + 2\sqrt{\frac{1}{2m} \log \left(\frac{2k}{\delta} \right)}$$

□ **VC 維度** – 一個有限假設類別的 Vapnik-Chervonenkis (VC) 維度 $VC(\mathcal{H})$ 指的是 \mathcal{H} 最多能夠打散的數量

注意： $\mathcal{H} = \{2 \text{ 維的線性分類器}\}$ 的 VC 維度為 3



□ **理論(Vapnik)** – 令 \mathcal{H} 已給定， $VC(\mathcal{H}) = d$ 且 m 是訓練資料級的數量，在機率至少為 $1 - \delta$ 的情況下，我們得到：

$$\epsilon(\hat{h}) \leq \left(\min_{h \in \mathcal{H}} \epsilon(h) \right) + O \left(\sqrt{\frac{d}{m} \log \left(\frac{m}{d} \right)} + \frac{1}{m} \log \left(\frac{1}{\delta} \right) \right)$$

2 無監督學習

翻譯: kevingo. 審閱: imironhead, 徐承漢.

2.1 非監督式學習介紹

□ **動機** – 非監督式學習的目的是要找出未標籤資料 $\{x^{(1)}, \dots, x^{(m)}\}$ 之間的隱藏模式

□ **Jensen's 不等式** – 令 f 為一個凸函數、 X 為一個隨機變數，我們可以得到底下這個不等式：

$$E[f(X)] \geq f(E[X])$$

2.1.1 最大期望值

□ **潛在變數 (Latent variables)** – 潛在變數指的是隱藏/沒有觀察到的變數，這會讓問題的估計變得困難，我們通常使用 z 來代表它。底下是潛在變數的常見設定

設定	潛在變數 z	$x z$	評論
k 元高斯模型	$\text{Multinomial}(\phi)$	$\mathcal{N}(\mu_j, \Sigma_j)$	$\mu_j \in \mathbb{R}^n, \phi \in \mathbb{R}^k$
因素分析	$\mathcal{N}(0, I)$	$\mathcal{N}(\mu + \Lambda z, \psi)$	$\mu_j \in \mathbb{R}^n$

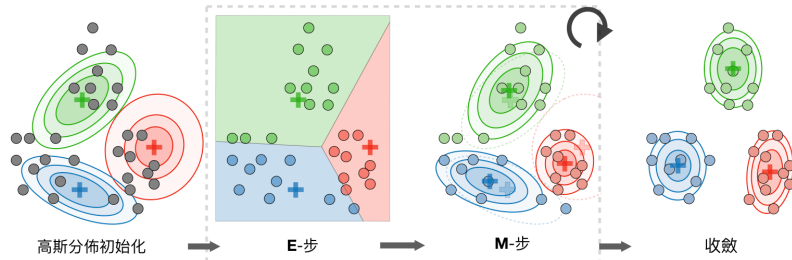
□ **演算法** – 最大期望演算法 (EM Algorithm) 透過重複建構一個概似函數的下界 (E-step) 和最佳化下界 (M-step) 來進行最大概似估計給出參數 θ 的高效率估計方法：

- **E-step:** 評估後驗機率 $Q_i(z^{(i)})$ ，其中每個資料點 $x^{(i)}$ 來自於一個特定的群集 $z^{(i)}$ ，如下：

$$Q_i(z^{(i)}) = P(z^{(i)} | x^{(i)}; \theta)$$

- **M-step:** 使用後驗機率 $Q_i(z^{(i)})$ 作為資料點 $x^{(i)}$ 在群集中特定的權重，用來分別重新估計每個群集，如下：

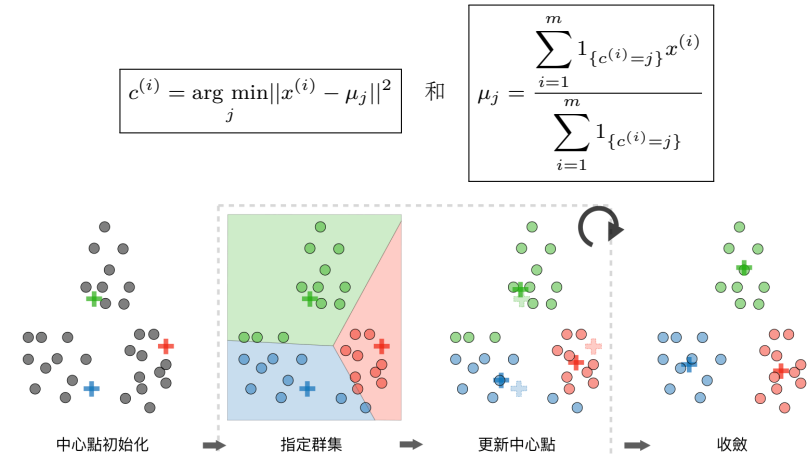
$$\theta_i = \operatorname{argmax}_{\theta} \sum_i \int_{z^{(i)}} Q_i(z^{(i)}) \log \left(\frac{P(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})} \right) dz^{(i)}$$



2.1.2 k -平均算法分群法

我們使用 $c^{(i)}$ 表示資料 i 屬於某群，而 μ_j 則是群 j 的中心

□ **演算法** – 在隨機初始化群集中心點 $\mu_1, \mu_2, \dots, \mu_k \in \mathbb{R}^n$ 後， k -平均算法演算法重複以下步驟直到收斂：



□ **畸變函數** – 為了確認演算法是否收斂，我們定義以下的畸變函數：

$$J(c, \mu) = \sum_{i=1}^m \|x^{(i)} - \mu_{c^{(i)}}\|^2$$

2.1.3 階層式分群法

□ **演算法** – 階層式分群法是透過一種階層架構的方式，將資料建立為一種連續層狀結構的形式。

□ **類型** – 底下是幾種不同類型的階層式分群法，差別在於要最佳化的目標函式的不同，請參考底下：

Ward 鏈結距離	平均鏈結距離	完整鏈結距離
最小化群內距離	最小化各群彼此的平均距離	最小化各群彼此的最大距離

2.1.4 分群衡量指標

在非監督式學習中，通常很難去評估一個模型的好壞，因為我們沒有擁有像在監督式學習任務中正確答案的標籤

□ **輪廓係數 (Silhouette coefficient)** – 我們指定 a 為一個樣本點和相同群集中其他資料點的平均距離、 b 為一個樣本點和下一個最接近群集其他資料點的平均距離，輪廓係數 s 對於此一樣本點的定義為：

$$s = \frac{b - a}{\max(a, b)}$$

□ **Calinski-Harabaz 指標** – 定義 k 是群集的數量， B_k 和 W_k 分別是群內和群集之間的離差矩陣 (dispersion matrices)：

$$B_k = \sum_{j=1}^k n_{c(i)} (\mu_{c(i)} - \mu)(\mu_{c(i)} - \mu)^T, \quad W_k = \sum_{i=1}^m (x^{(i)} - \mu_{c(i)})(x^{(i)} - \mu_{c(i)})^T$$

Calinski-Harabaz 指標 $s(k)$ 指出分群模型的好壞，此指標的值越高，代表分群模型的表現越好。定義如下：

$$s(k) = \frac{\text{Tr}(B_k)}{\text{Tr}(W_k)} \times \frac{N - k}{k - 1}$$

2.1.5 主成份分析

這是一個維度縮減的技巧，在於找到投影資料的最大方差

□ **特徵值、特徵向量** – 給定一個矩陣 $A \in \mathbb{R}^{n \times n}$ ，我們說 λ 是 A 的特徵值，當存在一個特徵向量 $z \in \mathbb{R}^n \setminus \{0\}$ ，使得：

$$Az = \lambda z$$

□ **譜定理** – 令 $A \in \mathbb{R}^{n \times n}$ ，如果 A 是對稱的，則 A 可以透過正交矩陣 $U \in \mathbb{R}^{n \times n}$ 對角化。當 $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$ ，我們得到：

$$\exists \Lambda \text{ 對角線, } A = U\Lambda U^T$$

注意：與特徵值所關聯的特徵向量就是 A 矩陣的主特徵向量

□ **演算法** – 主成份分析 (PCA) 是一種維度縮減的技巧，它會透過尋找資料最大變異的方式，將資料投影在 k 維空間上：

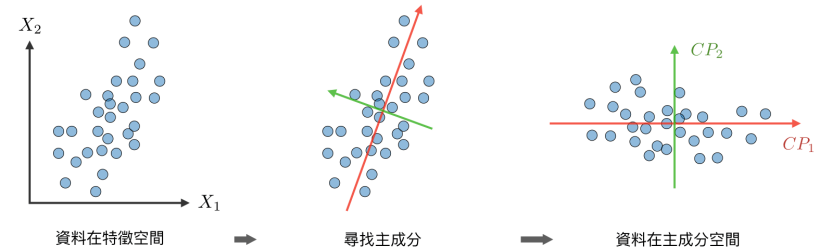
- 第一步: 正規化資料，讓資料平均為 0，變異數為 1

$$x_j^{(i)} \leftarrow \frac{x_j^{(i)} - \mu_j}{\sigma_j} \quad \text{哪裡} \quad \mu_j = \frac{1}{m} \sum_{i=1}^m x_j^{(i)} \quad \text{和} \quad \sigma_j^2 = \frac{1}{m} \sum_{i=1}^m (x_j^{(i)} - \mu_j)^2$$

- 第二步: 計算 $\Sigma = \frac{1}{m} \sum_{i=1}^m x^{(i)} x^{(i)T} \in \mathbb{R}^{n \times n}$ ，即對稱實際特徵值

- 第三步: 計算 $u_1, \dots, u_k \in \mathbb{R}^n$ ， k 個正交主特徵向量的總和 Σ ，即是 k 個最大特徵值的正交特徵向量

- 第四步: 將資料投影到 $\text{span}_{\mathbb{R}}(u_1, \dots, u_k)$



2.1.6 獨立成分分析

這是用來尋找潛在生成來源的技巧

□ **假設** – 我們假設資料 x 是從 n 維的來源向量 $s = (s_1, \dots, s_n)$ 產生， s_i 為獨立變數，透過一個混合與非奇異矩陣 A 產生如下：

$$x = As$$

目的在於找到一個 unmixing 矩陣 $W = A^{-1}$

□ **Bell 和 Sejnowski 獨立成份分析演算法** – 此演算法透過以下步驟來找到 unmixing 矩陣：

- 紀錄 $x = As = W^{-1}s$ 的機率如下：

$$p(x) = \prod_{i=1}^n p_s(w_i^T x) \cdot |W|$$

- 在給定訓練資料 $\{x^{(i)}, i \in [1, m]\}$ 的情況下，其對數概似估計函數與定義 g 為 sigmoid 函數如下：

$$l(W) = \sum_{i=1}^m \left(\sum_{j=1}^n \log \left(g'(w_j^T x^{(i)}) \right) + \log |W| \right)$$

因此，梯度隨機下降學習規則對每個訓練樣本 $x^{(i)}$ 來說，我們透過以下方法來更新 W ：

$$W \leftarrow W + \alpha \left(\begin{pmatrix} 1 - 2g(w_1^T x^{(i)}) \\ 1 - 2g(w_2^T x^{(i)}) \\ \vdots \\ 1 - 2g(w_n^T x^{(i)}) \end{pmatrix} x^{(i)T} + (W^T)^{-1} \right)$$

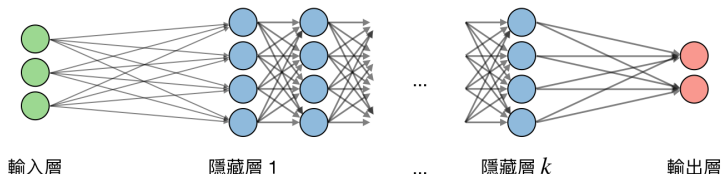
3 深度學習

翻譯: kevingo. 審閱: TobyOoO.

3.1 神經網路

神經網路是一種透過layer來建構的模型。經常被使用的神經網路模型包括了卷積神經網路(CNN)和遞迴式神經網路(RNN)。

□ **架構** – 神經網路架構所需要用到的詞彙描述如下：



我們使用 i 來代表網路的第 i 層、 j 來代表某一層中第 j 個隱藏神經元的話，我們可以得到下面得公式：

$$z_j^{[i]} = w_j^{[i]T} x + b_j^{[i]}$$

其中，我們分別使用 w 來代表權重、 b 代表偏差項、 z 代表輸出的結果。

□ **激活函數** – 激活函數是為了在每一層尾端的神經元帶入非線性轉換而設計的。底下是一些常見激活函數：

Sigmoid	Tanh	ReLU	Leaky ReLU
$g(z) = \frac{1}{1 + e^{-z}}$	$g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	$g(z) = \max(0, z)$	$g(z) = \max(\epsilon z, z)$ 與 $\epsilon \ll 1$

□ **交叉熵損失函式** – 在神經網路中，交叉熵損失 $L(z, y)$ 通常如下定義：

$$L(z, y) = - \left[y \log(z) + (1 - y) \log(1 - z) \right]$$

□ **學習速率** – 學習速率通常用 α 或 η 來表示，目的是用來控制權重更新的速度。學習速度可以是一個固定值，或是隨著訓練的過程改變。現在最熱門的最佳化方法叫作Adam，是一種隨著訓練過程改變學習速率的最佳化方法。

□ **反向傳播演算法** – 反向傳播演算法是一種在神經網路中用來更新權重的方法，更新的基準是根據神經網路的實際輸出值和期望輸出值之間的關係。權重的導數是根據連鎖律(chain rule)來計算，通常會表示成下面的形式：

$$\frac{\partial L(z, y)}{\partial w} = \frac{\partial L(z, y)}{\partial a} \times \frac{\partial a}{\partial z} \times \frac{\partial z}{\partial w}$$

因此，權重會透過以下的方式來更新：

$$w \leftarrow w - \eta \frac{\partial L(z, y)}{\partial w}$$

□ **更新權重** – 在神經網路中，權重的更新會透過以下步驟進行：

- **步驟一**: 取出一個批次(batch)的資料
- **步驟二**: 執行前向傳播演算法(forward propagation)來得到對應的損失值
- **步驟三**: 將損失值透過反向傳播演算法來得到梯度
- **步驟四**: 使用梯度來更新網路的權重

□ **丟棄法** – 丟棄法是一種透過丟棄一些神經元，來避免過擬和的技巧。在實務上，神經元會透過機率值的設定來決定要丟棄或保留

3.2 卷積神經網路

□ **卷積層的需求** – 我們使用 W 來表示輸入資料的維度大小、 F 代表卷積層的filter尺寸、 P 代表對資料墊零(zero padding)使資料長度齊一後的長度、 S 代表卷積後取出的特徵stride數量，則輸出的維度大小可以透過以下的公式表示：

$$N = \frac{W - F + 2P}{S} + 1$$

□ **批次正規化(Batch normalization)** – 它是一個藉由 γ, β 兩個超參數來正規化每個批次 $\{x_i\}$ 的過程。每一次正規化的過程，我們使用 μ_B, σ_B^2 分別代表平均數和變異數。請參考以下公式：

$$x_i \leftarrow \gamma \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} + \beta$$

批次正規化的動作通常在全連接層/卷積層之後、在非線性層之前進行。目的在於接納更高的學習速率，並且減少該批次學習初期對取樣資料特徵的依賴性。

3.3 遞歸神經網路(RNN)

□ **開的種類** – 在傳統的遞歸神經網路中，你會遇到幾種開：

輸入開	遺忘開	輸出開	開
要不要將資料寫入到記憶區塊中？	要不要將存在在記憶區塊中的資料清除？	要不要將資料從記憶區塊中取出？	要寫多少資料到記憶區塊？

□ **長短期記憶模型** – 長短期記憶模型是一種遞歸神經網路，藉由導入遺忘開的設計來避免梯度消失的問題

3.4 強化學習及控制

強化學習的目標就是為了讓代理(agent) 能夠學習在環境中進化

□ **馬可夫決策過程** – 一個馬可夫決策過程(MDP) 包含了五個元素($\mathcal{S}, \mathcal{A}, \{P_{sa}\}, \gamma, R$) :

- \mathcal{S} 是一組狀態的集合
- \mathcal{A} 是一組行為的集合
- $\{P_{sa}\}$ 指的是, 當 $s \in \mathcal{S}, a \in \mathcal{A}$ 時, 狀態轉移的機率
- $\gamma \in [0, 1[$ 是衰減係數
- $R: \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ 或 $R: \mathcal{S} \rightarrow \mathbb{R}$ 指的是獎勵函數, 也就是演算法想要去最大化的目標函數

□ **策略** – 一個策略 π 指的是一個函數 $\pi: \mathcal{S} \rightarrow \mathcal{A}$, 這個函數會將狀態映射到行為

注意: 我們會說, 我們給定一個策略 π , 當我們給定一個狀態 s 我們會採取一個行動 $a = \pi(s)$

□ **價值函數** – 給定一個策略 π 和狀態 s , 我們定義價值函數 V^π 為:

$$V^\pi(s) = E \left[R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \dots | s_0 = s, \pi \right]$$

□ **貝爾曼方程** – 最佳的貝爾曼方程是將價值函數 V^{π^*} 和策略 π^* 表示為:

$$V^{\pi^*}(s) = R(s) + \max_{a \in \mathcal{A}} \gamma \sum_{s' \in \mathcal{S}} P_{sa}(s') V^{\pi^*}(s')$$

注意: 對於給定一個狀態 s , 最佳的策略 π^* 是:

$$\pi^*(s) = \operatorname{argmax}_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} P_{sa}(s') V^*(s')$$

□ **價值迭代演算法** – 價值迭代演算法包含兩個步驟:

- 1) 針對價值初始化:

$$V_0(s) = 0$$

- 根據之前的值, 迭代此價值的值:

$$V_{i+1}(s) = R(s) + \max_{a \in \mathcal{A}} \left[\sum_{s' \in \mathcal{S}} \gamma P_{sa}(s') V_i(s') \right]$$

□ **最大概似估計** – 針對狀態轉移機率的最大概似估計為:

$$P_{sa}(s') = \frac{\# \text{從狀態 } s \text{ 到 } s' \text{ 所採取行為的次數}}{\# \text{從狀態 } s \text{ 所採取行為的次數}}$$

□ **Q學習演算法** – Q學習演算法是針對 Q 的一個model-free 的估計, 如下:

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[R(s, a, s') + \gamma \max_{a'} Q(s', a') - Q(s, a) \right]$$

4 機器學習秘訣和技巧參

翻譯: kevingo. 審閱: kentropy.

4.1 分類器的評估指標

在二元分類的問題上，底下是主要用來衡量模型表現的指標

□ **混淆矩陣** – 混淆矩陣是用來衡量模型整體表現的指標

		預測類別	
		+	-
真實類別	+	TP True Positives	FN False Negatives Type II error
	-	FP False Positives Type I error	TN True Negatives

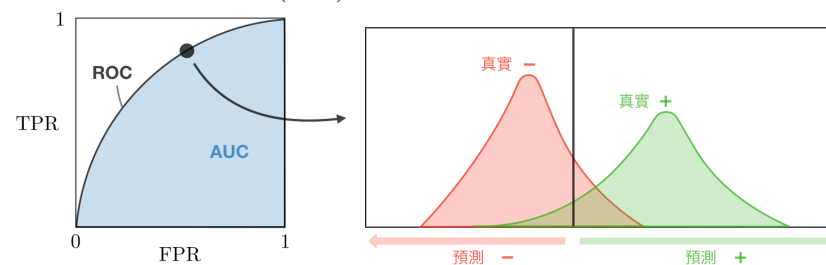
□ **主要的衡量指標** – 底下的指標經常用在評估分類模型的表現

指標	公式	解釋
準確度	$\frac{TP + TN}{TP + TN + FP + FN}$	模型的整體表現
Precision	$\frac{TP}{TP + FP}$	預測的類別有多精準的比例
Recall Sensitivity	$\frac{TP}{TP + FN}$	實際正的樣本的覆蓋率有多少
Specificity	$\frac{TN}{TN + FP}$	實際負的樣本的覆蓋率
F1分數	$\frac{2TP}{2TP + FP + FN}$	對於非平衡類別相當有用的混合指標

□ **ROC** – 接收者操作特徵曲線(ROC Curve)，又被稱為ROC，是透過改變閾值來表示TPR 和FPR 之間關係的圖形。這些指標總結如下：

衡量指標	公式	等同於
True Positive Rate TPR	$\frac{TP}{TP + FN}$	Recall, sensitivity
False Positive Rate FPR	$\frac{FP}{TN + FP}$	1-specificity

□ **AUC** – 在接收者操作特徵曲線(ROC) 底下的面積，也稱為AUC 或AUROC：



4.2 回歸器的評估指標

□ **基本的指標** – 給定一個迴歸模型 f ，底下是經常用來評估此模型的指標：

總平方和	被解釋平方和	殘差平方和
$SS_{\text{tot}} = \sum_{i=1}^m (y_i - \bar{y})^2$	$SS_{\text{reg}} = \sum_{i=1}^m (f(x_i) - \bar{y})^2$	$SS_{\text{res}} = \sum_{i=1}^m (y_i - f(x_i))^2$

□ **決定係數** – 決定係數又被稱為 R^2 or r^2 ，它提供了模型是否具備復現觀測結果的能力。定義如下：

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

□ **主要的衡量指標** – 藉由考量變數 n 的數量，我們經常用使用底下的指標來衡量迴歸模型的表現：

Mallow's Cp	AIC	BIC	Adjusted R^2
$\frac{SS_{\text{res}} + 2(n+1)\hat{\sigma}^2}{m}$	$2[(n+2) - \log(L)]$	$\log(m)(n+2) - 2\log(L)$	$1 - \frac{(1-R^2)(m-1)}{m-n-1}$

當中， L 代表的是概似估計， $\hat{\sigma}^2$ 則是變異數的估計

4.3 模型選擇

□ **詞彙** – 當進行模型選擇時，我們會針對資料進行以下區分：

訓練資料集	驗證資料集	測試資料集
- 用來訓練模型 - 通常是80	- 用來評估模型 - 又被稱為hold-out 資料集或開發資料集	- 模型用來預測用的資料集

當模型被選擇後，就會使用整個資料集來做訓練，並且在沒看過的資料集上做測試。你可以參考以下的圖表：



□ **交叉驗證** – 交叉驗證，又稱之為CV，它是一種不特別依賴初始訓練集來挑選模型的方法。幾種不同的方法如下：

<i>k</i> -fold	Leave- <i>p</i> -out
<ul style="list-style-type: none"> - 把資料分成<i>k</i>份，利用<i>k</i> - 1份資料來訓練，剩下的一份用來評估模型效能 - 一般來說<i>k</i> = 5 或10 	<ul style="list-style-type: none"> - 在<i>n</i> - <i>p</i>份資料上進行訓練，剩下的<i>p</i>份資料用來評估模型效能 - 當<i>p</i> = 1時，又稱為leave-one-out

最常用到的方法叫做*k*-fold 交叉驗證。它將訓練資料切成*k*份，在*k* - 1份資料上進行訓練，而剩下的一份用來評估模型的效能，這樣的流程會重複*k*次。最後計算出來的模型損失是*k*次結果的平均，又稱為交叉驗證損失值。

Fold	Dataset	Validation error	Cross-validation error
1		ϵ_1	$\frac{\epsilon_1 + \dots + \epsilon_k}{k}$
2		ϵ_2	
\vdots	\vdots	\vdots	
<i>k</i>		ϵ_k	
	Train Validation		

□ **正規化** – 正歸化的目的是為了避免模型對於訓練資料過擬合，進而導致高方差。底下的表格整理了常見的正規化技巧：

LASSO	Ridge	Elastic Net
<ul style="list-style-type: none"> - 將係數縮減為0 - 有利變數的選擇 	將係數變得更小	在變數的選擇和小係數之間作權衡
$\dots + \lambda \ \theta\ _1$ $\lambda \in \mathbb{R}$	$\dots + \lambda \ \theta\ _2^2$ $\lambda \in \mathbb{R}$	$\dots + \lambda \left[(1 - \alpha) \ \theta\ _1 + \alpha \ \theta\ _2^2 \right]$ $\lambda \in \mathbb{R}, \alpha \in [0, 1]$

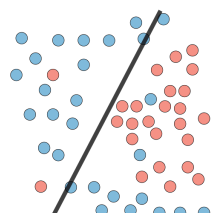
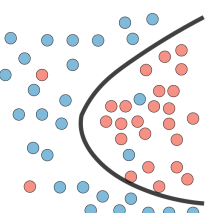
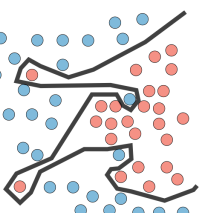
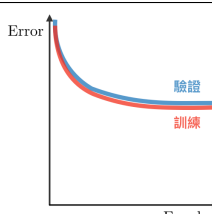
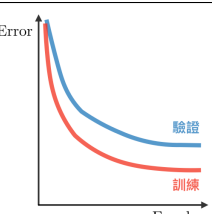
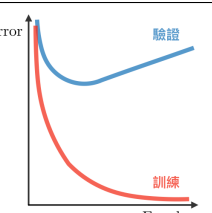
4.4 診斷

□ **偏差** – 模型的偏差指的是模型預測值與實際值之間的差異

□ **變異** – 變異指的是模型在預測資料時的變異程度

□ **偏差/變異的權衡** – 越簡單的模型，偏差就越大。而越複雜的模型，變異就越大

	Underfitting	Just right	Overfitting
現象	<ul style="list-style-type: none"> - 訓練錯誤較高 - 訓練錯誤和測試錯誤接近 - 高偏差 	<ul style="list-style-type: none"> - 訓練誤差會稍微比測試誤差低 	<ul style="list-style-type: none"> - 訓練誤差很低 - 訓練誤差比測試誤差低很多 - 高變異
迴歸圖示			

分類圖示			
深度學習圖示			
可能的解法	<ul style="list-style-type: none"> - 使用較複雜的模型 - 增加更多特徵 - 訓練更久 		<ul style="list-style-type: none"> - 採用正規化化的方法 - 取得更多資料

□ **誤差分析(Error analysis)** – 誤差分析指的是分析目前使用的模型和最佳模型之間差距的根本原因

□ **銷蝕分析(Ablative analysis)** – 銷蝕分析指的是分析目前模型和基準模型之間差異的根本原因

5 回顧

5.1 機率和統計

翻譯: *kevingo*. 審閱: 徐承漢.

5.2 幾率與組合數學介紹

□ **樣本空間** – 一個實驗的所有可能結果的集合稱之為這個實驗的樣本空間，記做 S

□ **事件** – 樣本空間的任何子集合 E 被稱之為一個事件。也就是說，一個事件是實驗的可能結果的集合。如果該實驗的結果包含 E ，我們稱我們稱 E 發生

□ **機率公理** – 對於每個事件 E ，我們用 $P(E)$ 表示事件 E 發生的機率

$$(1) \quad 0 \leq P(E) \leq 1 \quad (2) \quad P(S) = 1 \quad (3) \quad P\left(\bigcup_{i=1}^n E_i\right) = \sum_{i=1}^n P(E_i)$$

□ **排列** – 排列指的是從 n 個相異的物件中，取出 r 個物件按照固定順序重新安排，這樣安排的數量用 $P(n, r)$ 來表示，定義為：

$$P(n, r) = \frac{n!}{(n-r)!}$$

□ **組合** – 組合指的是從 n 個物件中，取出 r 個物件，但不考慮他的順序。這樣組合要考慮的數量用 $C(n, r)$ 來表示，定義為：

$$C(n, r) = \frac{P(n, r)}{r!} = \frac{n!}{r!(n-r)!}$$

注意：對於 $0 \leq r \leq n$ ，我們會有 $P(n, r) \geq C(n, r)$

5.3 條件機率

□ **貝氏定理** – 對於事件 A 和 B 滿足 $P(B) > 0$ 時，我們定義如下：

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

注意： $P(A \cap B) = P(A)P(B|A) = P(A|B)P(B)$

□ **分割** – 令 $\{A_i, i \in [1, n]\}$ 對所有的 i ， $A_i \neq \emptyset$ ，我們說 $\{A_i\}$ 是一個分割，當底下成立時：

$$\forall i \neq j, A_i \cap A_j = \emptyset \quad \text{和} \quad \bigcup_{i=1}^n A_i = S$$

注意：對於任何在樣本空間的事件 B 來說， $P(B) = \sum_{i=1}^n P(B|A_i)P(A_i)$

□ **貝氏定理的擴展** – 令 $\{A_i, i \in [1, n]\}$ 為樣本空間的一個分割，我們定義：

$$P(A_k|B) = \frac{P(B|A_k)P(A_k)}{\sum_{i=1}^n P(B|A_i)P(A_i)}$$

□ **獨立** – 當以下條件滿足時，兩個事件 A 和 B 為獨立事件：

$$P(A \cap B) = P(A)P(B)$$

5.4 隨機變數

□ **隨機變數** – 一個隨機變數 X ，它是一個將樣本空間中的每個元素映射到實數域的函數

□ **累積分佈函數(CDF)** – 累積分佈函數 F 是單調遞增的函數，其 $\lim_{x \rightarrow -\infty} F(x) = 0$ 且 $\lim_{x \rightarrow +\infty} F(x) = 1$ ，定義如下：

$$F(x) = P(X \leq x)$$

注意： $P(a < X \leq b) = F(b) - F(a)$

□ **機率密度函數** – 機率密度函數 f 是隨機變數 X 在兩個相鄰的實數值附近取值的機率

□ **機率密度函數和累積分佈函數的關係** – 底下是一些關於離散(D) 和連續(C) 的情況下的重要屬性

情況	累積分佈函數 F	機率密度函數 f	機率密度函數的屬性
(D)	$F(x) = \sum_{x_i \leq x} P(X = x_i)$	$f(x_j) = P(X = x_j)$	$0 \leq f(x_j) \leq 1$ 和 $\sum_j f(x_j) = 1$
(C)	$F(x) = \int_{-\infty}^x f(y)dy$	$f(x) = \frac{dF}{dx}$	$f(x) \geq 0$ 和 $\int_{-\infty}^{+\infty} f(x)dx = 1$

□ **變異數** – 隨機變數的變異數通常表示為 $\text{Var}(X)$ 或 σ^2 ，用來衡量一個分佈離散程度的指標。其表示如下：

$$\text{Var}(X) = E[(X - E[X])^2] = E[X^2] - E[X]^2$$

□ **標準差** – 一個隨機變數的標準差通常表示為 σ ，用來衡量一個分佈離散程度的指標，其單位和實際的隨機變數相容，表示如下：

$$\sigma = \sqrt{\text{Var}(X)}$$

□ **分佈的期望值和動差** – 底下是期望值 $E[X]$ 、一般期望值 $E[g(X)]$ 、第 k 個動差和特徵函數 $\psi(\omega)$ 在離散和連續的情況下的表示式：

情況	$E[X]$	$E[g(X)]$	$E[X^k]$	$\psi(\omega)$
(D)	$\sum_{i=1}^n x_i f(x_i)$	$\sum_{i=1}^n g(x_i) f(x_i)$	$\sum_{i=1}^n x_i^k f(x_i)$	$\sum_{i=1}^n f(x_i) e^{i\omega x_i}$
(C)	$\int_{-\infty}^{+\infty} x f(x) dx$	$\int_{-\infty}^{+\infty} g(x) f(x) dx$	$\int_{-\infty}^{+\infty} x^k f(x) dx$	$\int_{-\infty}^{+\infty} f(x) e^{i\omega x} dx$

□ **隨機變數的轉換** – 令變數 X 和 Y 由某個函式連結在一起。我們定義 f_X 和 f_Y 是 X 和 Y 的分佈函式，可以得到：

$$f_Y(y) = f_X(x) \left| \frac{dx}{dy} \right|$$

□ **萊布尼茲積分法則** – 令 g 為 x 和 c 的函數， a 和 b 是依賴於 c 的的邊界，我們得到：

$$\frac{\partial}{\partial c} \left(\int_a^b g(x) dx \right) = \frac{\partial b}{\partial c} \cdot g(b) - \frac{\partial a}{\partial c} \cdot g(a) + \int_a^b \frac{\partial g}{\partial c}(x) dx$$

□ **柴比雪夫不等式** – 令 X 是一隨機變數，期望值為 μ 。對於 $k, \sigma > 0$ ，我們有以下不等式：

$$P(|X - \mu| \geq k\sigma) \leq \frac{1}{k^2}$$

5.5 聯合分佈隨機變數

□ **條件密度** – X 對於 Y 的條件密度，通常用 $f_{X|Y}$ 表示如下：

$$f_{X|Y}(x) = \frac{f_{XY}(x, y)}{f_Y(y)}$$

□ **獨立** – 當滿足以下條件時，我們稱隨機變數 X 和 Y 互相獨立：

$$f_{XY}(x, y) = f_X(x) f_Y(y)$$

□ **邊緣密度和累積分佈** – 從聯合密度機率函數 f_{XY} 中我們可以得到：

種類	邊緣密度函數	累積函數
(D)	$f_X(x_i) = \sum_j f_{XY}(x_i, y_j)$	$F_{XY}(x, y) = \sum_{x_i \leq x} \sum_{y_j \leq y} f_{XY}(x_i, y_j)$
(C)	$f_X(x) = \int_{-\infty}^{+\infty} f_{XY}(x, y) dy$	$F_{XY}(x, y) = \int_{-\infty}^x \int_{-\infty}^y f_{XY}(x', y') dx' dy'$

□ **獨立** – 當滿足以下條件時，我們稱隨機變數 X 和 Y 互相獨立：

$$\psi_{X+Y}(\omega) = \psi_X(\omega) \times \psi_Y(\omega)$$

□ **共變異數** – 我們定義隨機變數 X 和 Y 的共變異數為 σ_{XY}^2 或 $\text{Cov}(X, Y)$ 如下：

$$\text{Cov}(X, Y) \triangleq \sigma_{XY}^2 = E[(X - \mu_X)(Y - \mu_Y)] = E[XY] - \mu_X \mu_Y$$

□ **相關性** – 我們定義 σ_X 、 σ_Y 為 X 和 Y 的標準差，而 X 和 Y 的相關係數 ρ_{XY} 定義如下：

$$\rho_{XY} = \frac{\sigma_{XY}^2}{\sigma_X \sigma_Y}$$

注意一：對於任何隨機變數 X 和 Y 來說， $\rho_{XY} \in [-1, 1]$ 成立

注意二：當 X 和 Y 獨立時， $\rho_{XY} = 0$

□ **主要的分佈** – 底下是我們需要熟悉的幾個主要的不等式：

種類	分佈	PDF	$\psi(\omega)$	$E[X]$	$\text{Var}(X)$
(D)	$X \sim \mathcal{B}(n, p)$ Binomial	$P(X = x) = \binom{n}{x} p^x q^{n-x}$ $x \in \llbracket 0, n \rrbracket$	$(pe^{i\omega} + q)^n$	np	npq
	$X \sim \text{Po}(\mu)$ Poisson	$P(X = x) = \frac{\mu^x}{x!} e^{-\mu}$ $x \in \mathbb{N}$	$e^{\mu(e^{i\omega} - 1)}$	μ	μ
(C)	$X \sim \mathcal{U}(a, b)$ Uniform	$f(x) = \frac{1}{b-a}$ $x \in [a, b]$	$\frac{e^{i\omega b} - e^{i\omega a}}{(b-a)i\omega}$	$\frac{a+b}{2}$	$\frac{(b-a)^2}{12}$
	$X \sim \mathcal{N}(\mu, \sigma)$ Gaussian	$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$ $x \in \mathbb{R}$	$e^{i\omega\mu - \frac{1}{2}\omega^2\sigma^2}$	μ	σ^2
	$X \sim \text{Exp}(\lambda)$ Exponential	$f(x) = \lambda e^{-\lambda x}$ $x \in \mathbb{R}_+$	$\frac{1}{1 - \frac{i\omega}{\lambda}}$	$\frac{1}{\lambda}$	$\frac{1}{\lambda^2}$

5.6 參數估計

□ **隨機抽樣** – 隨機抽樣指的是 n 個隨機變數 X_1, \dots, X_n 和 X 獨立且同分佈的集合

□ **估計量** – 估計量是一個資料的函數，用來推斷在統計模型中未知參數的值

□ **偏差** – 一個估計量的偏差 $\hat{\theta}$ 定義為 $\hat{\theta}$ 分佈期望值和真實值之間的差距：

$$\text{Bias}(\hat{\theta}) = E[\hat{\theta}] - \theta$$

注意：當 $E[\hat{\theta}] = \theta$ 時，我們稱為不偏估計量

□ **樣本平均** – 一個隨機樣本的樣本平均是用來預估一個分佈的真實平均 μ ，通常我們用 \bar{X} 來表示，定義如下：

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

注意：當 $E[\bar{X}] = \mu$ 時，則為不偏樣本平均

□ **樣本變異數** – 一個隨機樣本的樣本變異數是用來估計一個分佈的真實變異數 σ^2 ，通常使用 s^2 或 $\hat{\sigma}^2$ 來表示，定義如下：

$$s^2 = \hat{\sigma}^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$$

注意：當 $E[s^2] = \sigma^2$ 時，稱之為不偏樣本變異數

□ **中央極限定理** – 當我們有一個隨機樣本 X_1, \dots, X_n 滿足一個給定的分佈，其平均數為 μ ，變異數為 σ^2 ，我們有：

$$\bar{X} \underset{n \rightarrow +\infty}{\sim} \mathcal{N}\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$$

5.7 線性代數與微積分

翻譯: kevingo. 審閱: Miyaya.

5.8 通用符號

□ **向量** – 我們定義 $x \in \mathbb{R}^n$ 是一個向量，包含 n 維元素， $x_i \in \mathbb{R}$ 是第 i 維元素：

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n$$

□ **矩陣** – 我們定義 $A \in \mathbb{R}^{m \times n}$ 是一個 m 列 n 行的矩陣， $A_{i,j} \in \mathbb{R}$ 代表位在第 i 列第 j 行的元素：

$$A = \begin{pmatrix} A_{1,1} & \cdots & A_{1,n} \\ \vdots & & \vdots \\ A_{m,1} & \cdots & A_{m,n} \end{pmatrix} \in \mathbb{R}^{m \times n}$$

注意：上述定義的向量 x 可以視為 $n \times 1$ 的矩陣，或是更常被稱為行向量

□ **單位矩陣** – 單位矩陣 $I \in \mathbb{R}^{n \times n}$ 是一個方陣，其主對角線皆為 1，其餘皆為 0

$$I = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{pmatrix}$$

注意：對於所有矩陣 $A \in \mathbb{R}^{n \times n}$ ，我們有 $A \times I = I \times A = A$

□ **對角矩陣** – 對角矩陣 $D \in \mathbb{R}^{n \times n}$ 是一個方陣，其主對角線為非 0，其餘皆為 0

$$D = \begin{pmatrix} d_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & d_n \end{pmatrix}$$

注意：我們令 D 為 $\text{diag}(d_1, \dots, d_n)$

5.9 矩陣運算

□ **向量-向量** – 有兩種類型的向量-向量相乘：

- 內積：對於 $x, y \in \mathbb{R}^n$ ，我們可以得到：

$$x^T y = \sum_{i=1}^n x_i y_i \in \mathbb{R}$$

- 外積：對於 $x \in \mathbb{R}^m, y \in \mathbb{R}^n$ ，我們可以得到：

$$xy^T = \begin{pmatrix} x_1 y_1 & \cdots & x_1 y_n \\ \vdots & & \vdots \\ x_m y_1 & \cdots & x_m y_n \end{pmatrix} \in \mathbb{R}^{m \times n}$$

□ **矩陣-向量** – 矩陣 $A \in \mathbb{R}^{m \times n}$ 和向量 $x \in \mathbb{R}^n$ 的乘積是一個大小為 \mathbb{R}^m 的向量，使得：

$$Ax = \begin{pmatrix} a_{r,1}^T x \\ \vdots \\ a_{r,m}^T x \end{pmatrix} = \sum_{i=1}^n a_{c,i} x_i \in \mathbb{R}^m$$

其中 $a_{r,i}^T$ 是 A 的列向量、 $a_{c,j}$ 是 A 的行向量、 x_i 是 x 的元素

□ **矩陣-矩陣** – 矩陣 $A \in \mathbb{R}^{m \times n}$ 和 $B \in \mathbb{R}^{n \times p}$ 的乘積為一個大小 $\mathbb{R}^{m \times p}$ 的矩陣，使得：

$$AB = \begin{pmatrix} a_{r,1}^T b_{c,1} & \cdots & a_{r,1}^T b_{c,p} \\ \vdots & & \vdots \\ a_{r,m}^T b_{c,1} & \cdots & a_{r,m}^T b_{c,p} \end{pmatrix} = \sum_{i=1}^n a_{c,i} b_{r,i}^T \in \mathbb{R}^{m \times p}$$

其中， $a_{r,i}^T$ 、 $b_{r,i}^T$ 和 $a_{c,j}$ 、 $b_{c,j}$ 分別是 A 和 B 的列向量與行向量

□ **轉置** – 一個矩陣的轉置矩陣 $A \in \mathbb{R}^{m \times n}$ ，記作 A^T ，指的是其中元素的翻轉：

$$\forall i, j, \quad A_{i,j}^T = A_{j,i}$$

注意：對於矩陣 A 、 B ，我們有 $(AB)^T = B^T A^T$

□ **可逆** – 一個可逆矩陣 A 記作 A^{-1} ，存在唯一的矩陣，使得：

$$AA^{-1} = A^{-1}A = I$$

注意：並非所有的方陣都是可逆的。同樣的，對於矩陣 A 、 B 來說，我們有 $(AB)^{-1} = B^{-1}A^{-1}$

□ **跡** – 一個方陣 A 的跡，記作 $\text{tr}(A)$ ，指的是主對角線元素之合：

$$\text{tr}(A) = \sum_{i=1}^n A_{i,i}$$

注意：對於矩陣 A 、 B 來說，我們有 $\text{tr}(A^T) = \text{tr}(A)$ 及 $\text{tr}(AB) = \text{tr}(BA)$

□ **行列式** – 一個方陣 $A \in \mathbb{R}^{n \times n}$ 的行列式，記作 $|A|$ 或 $\det(A)$ ，可以透過 $A_{\setminus i, \setminus j}$ 來遞迴表示，它是一個沒有第 i 列和第 j 行的矩陣 A ：

$$\det(A) = |A| = \sum_{j=1}^n (-1)^{i+j} A_{i,j} |A_{\setminus i, \setminus j}|$$

注意： A 是一個可逆矩陣，若且唯若 $|A| \neq 0$ 。同樣的， $|AB| = |A||B|$ 且 $|A^T| = |A|$

5.10 矩陣的性質

□ **對稱分解** – 給定一個矩陣 A ，它可以透過其對稱和反對稱的部分表示如下：

$$A = \underbrace{\frac{A + A^T}{2}}_{\text{對稱}} + \underbrace{\frac{A - A^T}{2}}_{\text{反對稱}}$$

□ **範數** – 範數指的是一個函式 $N: V \rightarrow [0, +\infty[$ ，其中 V 是一個向量空間，且對於所有 $x, y \in V$ ，我們有：

- $N(x + y) \leq N(x) + N(y)$
- 對一個純量來說，我們有 $N(ax) = |a|N(x)$
- 若 $N(x) = 0$ 時，則 $x = 0$

對於 $x \in V$ ，最常用的範數總結如下表：

範數	表示法	定義	使用情境
Manhattan, L^1	$\ x\ _1$	$\sum_{i=1}^n x_i $	LASSO regularization
Euclidean, L^2	$\ x\ _2$	$\sqrt{\sum_{i=1}^n x_i^2}$	Ridge regularization
p -norm, L^p	$\ x\ _p$	$\left(\sum_{i=1}^n x_i^p\right)^{\frac{1}{p}}$	Hölder inequality
Infinity, L^∞	$\ x\ _\infty$	$\max_i x_i $	Uniform convergence

□ **線性相關** – 當集合中的一個向量可以用被定義為集合中其他向量的線性組合時，則則稱此集合的向量為線性相關

注意：如果沒有向量可以如上表示時，則稱此集合的向量彼此為線性獨立

□ **矩陣的秩** – 一個矩陣 A 的秩記作 $\text{rank}(A)$ ，指的是其列向量空間所產生的維度，等價於 A 的線性獨立的最大最大行向量

□ **半正定矩陣** – 當以下成立時，一個矩陣 $A \in \mathbb{R}^{n \times n}$ 是半正定矩陣(PSD)，且記作 $A \succeq 0$ ：

$$A = A^T \quad \text{和} \quad \forall x \in \mathbb{R}^n, \quad x^T A x \geq 0$$

注意：同樣的，一個矩陣 A 是一個半正定矩陣(PSD)，且滿足所有非零向量 x ， $x^T A x > 0$ 時，稱之為正定矩陣，記作 $A \succ 0$

□ **特徵值、特徵向量** – 給定一個矩陣 $A \in \mathbb{R}^{n \times n}$ ，當存在一個向量 $z \in \mathbb{R}^n \setminus \{0\}$ 時，此向量被稱為特徵向量， λ 稱之為 A 的特徵值，且滿足：

$$Az = \lambda z$$

□ **譜分解** – 令 $A \in \mathbb{R}^{n \times n}$ ，如果 A 是對稱的，則 A 可以被一個實數正交矩陣 $U \in \mathbb{R}^{n \times n}$ 給對角化。令 $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$ ，我們得到：

$$\exists \Lambda \text{ 對角線}, \quad A = U \Lambda U^T$$

□ **奇異值分解** – 對於給定維度為 $m \times n$ 的矩陣 A ，其奇異值分解指的是一種因子分解技巧，保證存在 $m \times m$ 的單式矩陣 U 、對角線矩陣 Σ $m \times n$ 和 $n \times n$ 的單式矩陣 V ，滿足：

$$A = U \Sigma V^T$$

5.11 矩陣導數

□ **梯度** – 令 $f: \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$ 是一個函式，且 $A \in \mathbb{R}^{m \times n}$ 是一個矩陣。 f 相對於 A 的梯度是一個 $m \times n$ 的矩陣，記作 $\nabla_A f(A)$ ，滿足：

$$\left(\nabla_A f(A) \right)_{i,j} = \frac{\partial f(A)}{\partial A_{i,j}}$$

注意： f 的梯度僅在 f 為一個函數且該函數回傳一個純量時有效

□ **海森** – 令 $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 是一個函式，且 $x \in \mathbb{R}^n$ 是一個向量，則一個 f 的海森對於向量 x 是一個 $n \times n$ 的對稱矩陣，記作 $\nabla_x^2 f(x)$ ，滿足：

$$\left(\nabla_x^2 f(x) \right)_{i,j} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}$$

注意： f 的海森僅在 f 為一個函數且該函數回傳一個純量時有效

□ **梯度運算** – 對於矩陣 A 、 B 、 C ，下列的梯度性質值得牢牢記住：

$$\nabla_A \text{tr}(AB) = B^T \quad \nabla_{A^T} f(A) = (\nabla_A f(A))^T$$

$$\nabla_A \text{tr}(ABA^T C) = CAB + C^T A B^T \quad \nabla_A |A| = |A| (A^{-1})^T$$