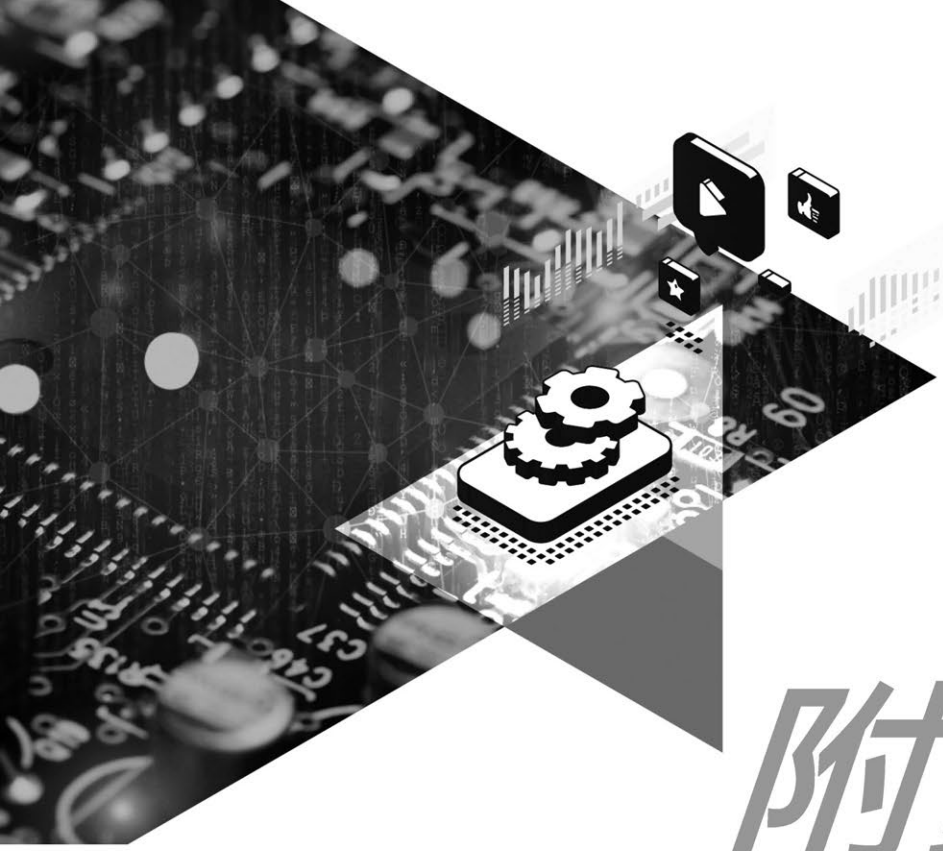

附錄D

超參數最佳化

李家岩、洪佑鑫（2022），製造數據科學：邁向智慧製造與數位決策，前程文化。

「**超參數**」(hyperparameter) 在數據科學的模型中十分常見。例如決策樹的**深度與葉節點最小樣本數**，深度學習的**隱藏層與節點數**，或集成學習等複雜模型中幾十甚至是幾百個超參數。在建模過程中，針對所選定模型，期望能找出使得模型效果最佳的超參數組合，這樣的過程稱之為「**超參數最佳化**」(hyperparameter optimization)，有時也被稱為「**超參數調整**」(tuning hyperparameter)。

D.1 超參數最佳化基礎

D.1.1 超參數與模型參數的關係

首先，我們先釐清**超參數**與**模型參數** (model parameter) 之間的關係與差異。如圖 D.1 所示，下方區塊的**模型參數**是在我們**訓練模型時（最小化損失函數）以迭代形式所更新的參數**，例如類神經網路全連結中的**權重**或是套索迴歸的**迴歸係數**，演算法本身會提供更新模型參數的機制。然而，上方區塊的**超參數**則是在**訓練模型前我們需要提前設定的參數**，有些超參數決定了模型架構與複雜度（偏誤與變異的權衡），例如類神經網路的層數、每層的節點數等；有些則決定了模型的學習過程，例如學習率或套索迴歸的懲罰權重。相對於隨著迭代更新的模型參數，超參數與模型是一對一的關係，也就是若我們要最佳化超參數時，要重複訓練多個模型逐步找出最好的，而每個模型在給定超參數的情況下來訓練出最佳的模型參數。因此超參數是架構於模型參數之上的，這也是取名為**超** (hyper) 參數的原因。

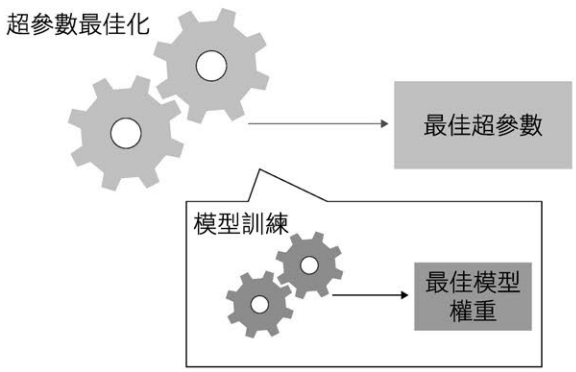

圖 D.1 超參數與模型參數的關係

李家岩、洪佑鑫（2022），製造數據科學：邁向智慧製造與數位決策，前程文化。

以圖 D.2 為例。若我們選定了一個類神經網路模型，我們分別調整層數、每層的節點數與學習率等三個超參數。每當測試一組超參數時，最佳化了模型參數（權重），並得到預測準確度。接著基於選用特定的超參數最佳化方法，依照每組對應的模型評估結果進行超參數最佳化。

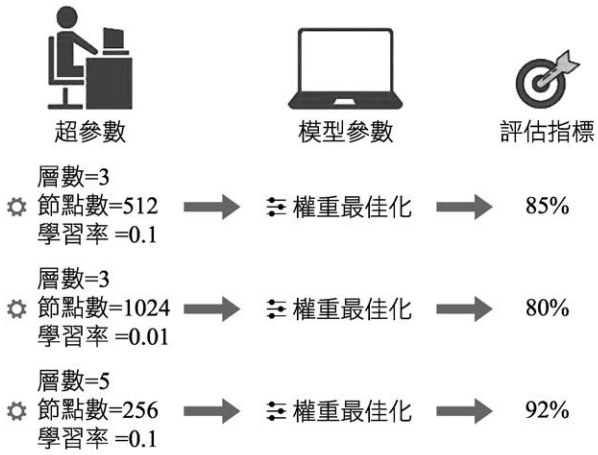

圖 D.2　超參數最佳化於類神經網路模型

D.1.2　超參數最佳化的困難與挑戰

事實上，超參數最佳化是個複雜且困難的問題。在訓練模型時，模型參數會依照我們給定與損失函數的關係進行更新，例如類神經網路使用微分關係的**梯度下降法**或羅吉斯迴歸的**牛頓法**。然而，這樣的關係並不存在於超參數與模型之間。在學理上，超參數與損失的函數關係是不容易建構出的，唯有直接對每組超參數進行訓練後才能得到其優劣。因此超參數最佳化是個「**黑盒子**」（blackbox）的問題，而且通常超參數與損失之間又為非線性的關係，再加上交叉驗證的測試集具有隨機的特性。**黑盒子**、**非線性**以及**隨機性**三項因素使得超參數最佳化成為一個複雜的問題。黑盒子使得這樣的非線性關係無法直接被觀察到，非線性則又使得最佳化函數中存在多個區域最佳解，並且實際問題中存在隨機性。更困難的是，當模型有越多的超參數時，會遇到**維度詛咒**的問題，使得這個最佳化的過程彷彿大海撈針極度困難。此外，要留意的是在實際執行超參數最佳化時，還需注意**每當給定一組超參數以訓練一個模型，就必須耗費一定的時間與資源，這是我們在實務上必須權衡的**。

以隨機森林為例，我們需要調整的超參數就包含了決策樹的**棵樹**以及每棵決策樹的**深度**、**特徵數量**、**葉節點最小樣本數**、**樣本抽樣比例**等，約莫有五個超參數。我們可能分別知道每個超參數對模型的影響，然而要同時調整時彼此間又存在非線性的關係，究竟該如何進行超參數最佳化呢？

D.2 超參數最佳化的方法

超參數最佳化方法有兩大類型，一是「**窮舉搜尋**」（exhaustive search of the space）（也就是暴力搜尋），另一為「**代理模型**」（surrogate model）。以下我們將分別介紹常使用的三種方法，「**網格搜尋法**」（grid search）、「**隨機搜尋法**」（random search）屬於**窮舉搜尋**的方法，另一種則是在理論效果最好但也較複雜與進階的「**貝氏最佳化**」（Bayesian optimization）則是屬於**代理模型**的方法。

D.2.1 網格搜尋法

首先，簡單易懂的方法就是**網格搜尋法**。此方法透過在超參數空間中指定子集進行窮舉搜尋，並通過交叉驗證對模型進行評估來衡量不同超參數組合的優劣。理想上，若能窮盡所有的超參數組合便能直接選出最好的那一個。但由於不同模型的超參數空間可能包括有實數值範圍（例如上界下界）或無限制值空間（infinite space），因此在進行網格搜尋之前，需事先手動設置**邊界**（boundary）與**實數離散化**（discretization）。

以一個典型具有「**徑向基函數核**」（radial basis function kernel, RBF）的支持向量機為例，需要最佳化的兩個超參數分別為正則化的常數C與核（kernel）的超參數γ，這兩個超參數皆為連續的正實數。因此要進行網格搜尋法前，需對此二超參數選擇適當的範圍與間隔，例如$C \in [10,1000]$。為簡化說明，先假設$C \in \{10,100,1000\}$與$\gamma \in \{0.1,0.5,1\}$，接著「網格搜尋」對兩個「超參數」的 3×3=9 種組合訓練支持向量機，在這九種組合中分別使用交叉驗證進行多次的建模（依照不同的切割數決定各別組合需訓練的模型數，例如 10-Fold 則訓練十個模型）並計算各別的效果，最後從選擇這九種組合中最好的（建議同時參考**平均數**與**變異數**）。

網格搜尋法有三項缺點描述如下：

- 超參數的**間隔不易給定**：給得過於密集可能導致運算量大幅增

加，相反的過於鬆散時找出的最佳組合不夠理想，在某些情況下，是可以基於領域知識、工程特性或商業理解去給定不等距離的間隔以減少不必要的資源投入，然而這樣事先給定的間隔也難以確認其好壞。

- **資訊的獨立性**：每組超參數計算的資訊彼此獨立，**沒有善加利用每組運算結果所提供的資訊**（一般來說相近的組合有相近的解），而是採取一種盲目猜測的策略，並未試著建構出超參數與損失函數間的關係；雖可進行平行運算，然而卻也非常耗費資源。
- 有**維數詛咒**的情形發生：在模型預測效果有一定的要求下，我們通常會使用較為複雜的模型，例如隨機森林、梯度提升機或深度學習等，然而這些模型具備較多的超參數，當超參數維度越多時耗費的時間與資源也會指數地遞增。

在實務上，網格搜尋法因簡單易懂的特性因而被廣為使用，但綜合上述缺點有時也不易達到較為精準且有效率的超參數最佳化。

D.2.2 隨機搜尋法

隨機搜尋法是以隨機方式選擇來搜尋出最佳超參數組合可能。在離散的空間中可簡單地應用，同時也可以推廣到連續空間和混合空間。我們無需提供要每個超參數離散化的間隔，而是基於經驗提供每個超參數的一個先驗機率（例如機率分配如均勻或指數分配），並從該分配中隨機抽出多個組合，再從中挑出最佳的那一個。以決策樹為例，基於過去的經驗，可設定**葉節點最小樣本數**一個平均數為 20 標準差為 5 的**連續常態分配**，並設定**深度**一個範圍為 1 到 6 的**離散均勻分配**。隨機搜尋法在給定的機率分配下隨機抽出多個超參數組合，並使用模型評估指標來決定最佳組合。在理論與實務上，隨機搜尋法相對於網格搜尋法更有效率，主要原因在於「隨機搜尋法」有效地搜索了更大的超參數空間，如圖 D.3 兩個方法的比較（Feurer and Hutter, 2019）。其次，通常在給定的數據下，並非所有超參數對於預測準確度都是等同的重要（如同在前述章節「**特徵挑選與維度縮減**」中不同特徵有著不同重要性），相較於找出**所有**超參數組合，找出**重要**超參數（圖 D.3 圖中的 x 軸）的最佳解帶來的效益更大。

「**元啟發式演算法**」（metaheurisitc algorithm）為隨機搜尋法當中廣為使用的方法之一。將超參數求解問題當作一最佳化問題來看，元啟發式演

算法常用的包含例如「**禁忌搜尋法**」（Tabu search）、「**模擬退火法**」（simulated annealing, SA）、「**基因演算法**」（genetic algorithm, GA）、「**粒子群最佳化**」（particle swarm optimization, PSO）、「**基因規劃**」（genetic programming, GP）、「**猴群搜尋演算法**」（monkey search algorithm, MSA）等。這些方法概念上透過**試誤**（trial and error）來窮舉搜尋，其優點在於會試著妥善使用每個超參數組合的資訊來找出可能的最佳解，然而依然會用大量的樣本去探索超參數的解空間，這種做法需要訓練較多的模型個數。

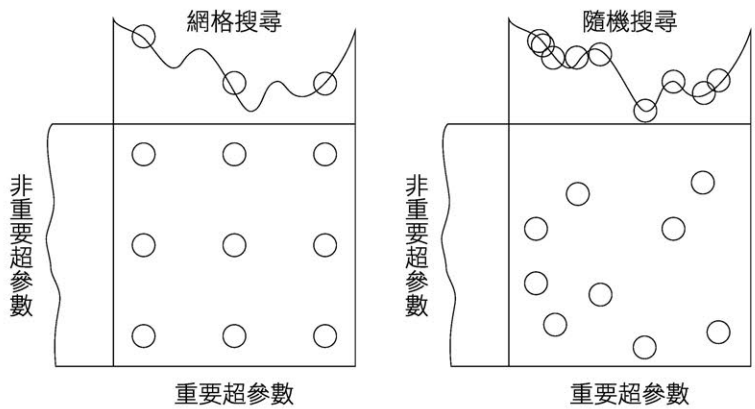

圖 D.3　網格搜尋法與隨機搜尋法（Feurer and Hutter, 2019）

隨機搜尋法與**網格搜尋法**都是實務上相對簡單易懂的方法，且均屬於窮舉搜尋類型的方法，同時也都能進行平行運算。然而它們具有同樣的缺點，包含了前述提及的**間隔或機率分配不易給定**、**資訊的獨立性**以及**維度詛咒**。如果模型的運算速度快且超參數的數量少，實務上使用不會有太大的問題；但**如果模型超參數的數量多，且需要大量時間與資源進行模型訓練，則不論是隨機搜尋法或網格搜尋法的效率與效果均受到限制**。因而後續我們將介紹屬於代理模型的**貝氏最佳化**方法，並說明此方法在有限資源下相較優於窮舉搜尋方法。

李家岩、洪佑鑫（2022），製造數據科學：邁向智慧製造與數位決策，前程文化。

D.3 貝氏最佳化

貝氏最佳化為目前在超參數最佳化最合適的方法之一。在 Google DeepMind 開發的 AlphaGo 強化學習模型、Google 的雲端機器學習模型、Facebook 的後端機器學習模型等皆使用了貝氏最佳化進行超參數優化。貝氏最佳化基於貝氏定理建構及更新超參數與損失函數間的**近似函數關係**，在給定的時間與資源下於超參數空間中逐步搜尋潛在的最佳解，其中包含了四個步驟：

步驟 1：建構或更新「**代理模型**」（surrogate function），為一個高斯過程的機率分配，以近似「超參數」與損失函數的實際函數關係。

步驟 2：最佳化「**收穫函數**」（acquisition function），以決定下個抽樣的超參數組合。

步驟 3：基於決定出的超參數組合進行建模，收集該模型的評估結果，並返回步驟 1。

步驟 4：於給定的時間與資源下所建構的超參數與損失函數關係找出最佳解。

由於貝氏最佳化具備大量統計與最佳化的思維與技巧，包含了核心思維「**反應曲面法**」（response surface method, RSM）、作為**代理模型**（surrogate function）的「**高斯過程**」（gaussian process, GP）、模型更新的「**貝氏推論**」（Bayesian inference）以及作為抽樣依據的**收穫函數**（acquisition function），以下各別講述這些重要的統計與最佳化思維，並說明貝氏最佳化如何將整合這些思維進行運作的。

D.3.1 反應曲面法

貝氏最佳化的核心思維是基於**反應曲面法**所發展出的方法。在統計學的**實驗設計**（design of experiments, DOE）領域中，期望藉由特定的實驗設計下以**最少的實驗次數**從變異數分析中想找出影響反應變數的因子以及它們彼此間關係。反應曲面法在相同的精神下，更進一步想完整地**建構反應變數與因子的關係**，並**依特定方向搜索**找出最佳反應變數**值**，屬於一種全域最佳化的方法。以圖 D.4 簡單說明反應曲面法的主要步驟，首先（右下），在某個特定的實驗設計下，實驗並收集了多個樣本，並依照這些實驗數據建構出因子與反應變數一次的**線性迴歸模型**。其次（中間箭頭），

依照迴歸係數的顯著性、大小以及方向決定下一個實驗樣本點，並逐步實驗直到反應變數不再往最佳解的方向走。最後（左上），在潛在最佳解的附近配適二次方（或多項式）的模型，最後利用這個模型找出的最佳解。在機械與電機領域中，牽涉到許多最佳化的問題，如馬達與機械機構的設計，可藉由反應曲面法作為分析。

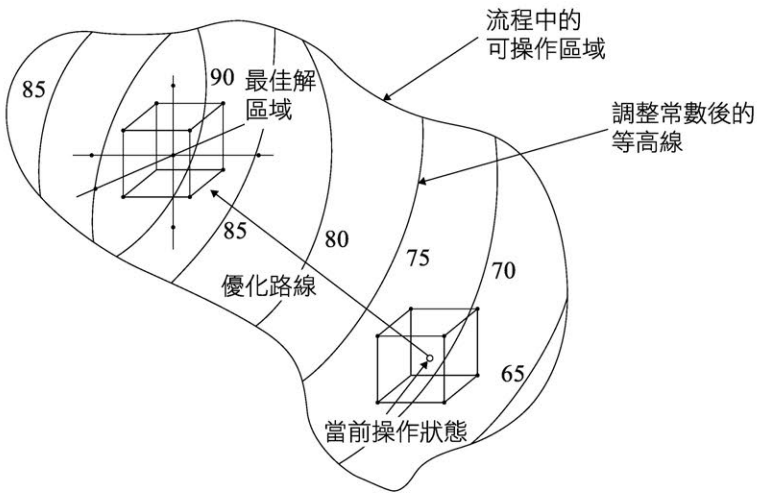

圖 D.4　反應曲面法

反應曲面法是貝氏最佳化中重要的觀念之一，以**最少實驗次數**的精神實踐最佳化的兩個步驟，分別為**建構反應變數與因子的關係**與**依特定方向搜索**。若套用在超參數最佳化中，反應曲面相當於我們想要找出的超參數與損失函數之間的函數關係，每次實驗則相當於每個超參數組合的模型訓練。相較於窮舉搜尋法（隨機搜尋法與網格搜尋法）的缺點（間隔與分配不易給定、資訊的獨立、維度詛咒、時間與資源的耗費），以反應曲面法為核心思維的代理模型，則替超參數與損失函數建構關係模型，充分運用每個超參數解的資訊，並遵循實驗設計最小化實驗成本的思維，期望以最少次數的模型訓練中逐步找出理想的超參數組合。

然而，反應曲面法有著兩個可改進的點。第一，反應曲面通常是由一次與二次迴歸所建構而成，但超參數與損失函數實際的關係（代理模型）可能非常複雜與非線性（如圖 D.3 所示）。第二，反應曲面法的搜尋方向是依照迴歸的線性方向，也就是目前潛在可能為最佳解的方向，與梯度下

降的概念相同，但仍有可能最佳解並非在這個方向上，反而落在一些難以探索的位置。綜合前述的兩點，反應曲面法有可能若入**區域最佳解**。若線性迴歸的模型難以描述真實的反應曲面，要使用什麼樣的代理模型更為合適呢？若使用其他的代理模型，那麼每得到一組超參數解如何更新呢？搜尋方向該依據什麼標準呢？以下將針對這三點依序說明。

D.3.2 高斯過程

有了反應曲面法中**建構與更新反應曲面**與**以最少次數逐步實驗**的概念後，接著我們深入了解貝氏最佳化中超參數與損失函數之間建構所選用的代理模型－**高斯過程**（Snoek et al., 2012）。相較於一次或二次的線性迴歸模型，我們想找到一個模型或函數具備足夠的**彈性**來描述超參數與損失的函數關係，甚至是具備**可更新**的特性，而高斯過程就是同時兼具這兩個特性的模型。

高斯過程是**多變量常態分布**擴展到無限維度的「**隨機過程**」（stochastic process），相對於我們熟知的**常態分布**是以機率分布來描述（隨機）**變數**，**高斯過程**則是以機率分布來描述（隨機）**函數**。如圖 D.5 所示，左圖是由一個平均數均為零的高斯過程隨機生成的五條函數，作為**先驗**（prior）；右圖則是由增加給定五點函數關係的高斯過程隨機生成的五條函數，作為**後驗**（posterior）。從兩圖中可知，在我們未知所有函數其(x, y)所對應的關係前，高斯過程均為一個隨機函數，這也是其具有彈性的原因。更具體地說，高斯過程與一般函數不同的點在於當我們任意給定一點x時，返回的函數值$f(x)$並非一個數值，而是一個服從常態分配的隨機變數。以圖 D.6 為例（Brochu et al., 2010），我們可以從一個給定三個樣本的高斯過程下，以函數中的任意點x_1, x_2, x_3為例觀察到它們各自在函數上的**平均數**$\mu(x)$與**變異數**$\sigma(x)$，而變異數也代表著對於此點與其函數關係的**不確定性**。此外，由於x在實數中有無限多個點，因此高斯過程是一個**無限維度常態分配的聯合函數**。

我們可將**高斯過程**表示如公式(D.1)所示。

$$f(x) \sim GP\big(m(x), k(x, x')\big) \tag{D.1}$$

如同前述所述，對於每個函數值$f(x)$而言，均服從**平均數函數**$m(x)$與**共變異函數**（covariance function）$k(x, x')$的**多變量常態分布**。一般平均數函

數會設為零，而共變異函數則是連結每個函數值的關鍵，它決定了高斯過程上點與點之間的相關性。共變異函數通常選用「**平方指數函數**」（squared exponential function）如公式(D.2)所示。

$$k(x_i, x_j) = \exp\left(-\frac{1}{2l^2}\|x_i - x_j\|^2\right) \tag{D.2}$$

其中l是平滑程度的參數，當兩個點(x_i, x_j)很靠近時會趨近於一，越遠則會趨近於零，這表示當兩點越近時影響彼此越大。

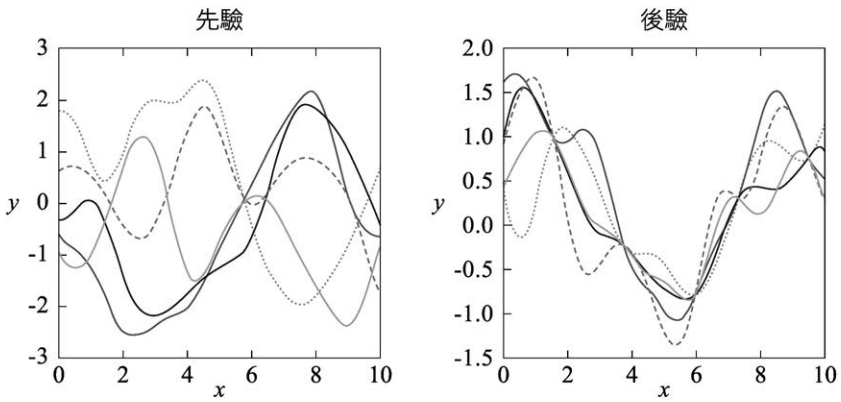

圖 D.5　高斯過程的先驗與給定五點的後驗

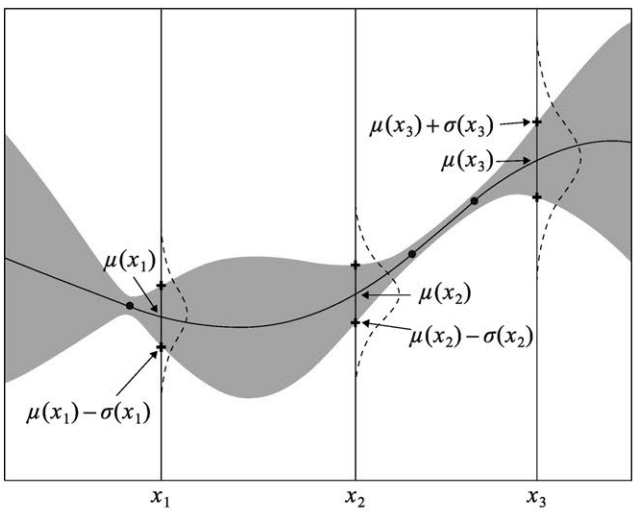

圖 D.6　高斯過程的函數平均數與變異數（Brochu et al., 2010）

李家岩、洪佑鑫（2022），製造數據科學：邁向智慧製造與數位決策，前程文化。

由於高斯過程對不確定性具有高度彈性，並且是一個平滑且非線性的隨機函數，對於作為超參數與損失函數之間的代理模型（好比前述的反應曲面）相當合適。然而在我們收集到新的函數關係（例如樣本點）後，如何更新高斯過程，降低已知點的不確定性呢？

D.3.3 貝氏推論

貝氏推論此處用於**模型更新**，是一種基於「**貝氏定理**」（Bayes' theorem）的推論統計方法，傳統「**頻率論**」（frequentist）的統計視角是以多次試驗觀測所得到的結果作為參考依據，例如母體平均數μ（參數）是一個由收集到的樣本集n估計出的樣本平均數$\bar{x}$（統計量），將母體參數視為一個定值加以估計。然而，貝氏推論的思想則是將母體特性視為一種**信念**（belief）或者又被稱為**先驗知識**（prior knowledge），為一個機率分配，而在我們收集到新證據後加以**更新**這個信念，以更新信念為核心的貝氏推論則在貝氏最佳化中扮演著**更新**代理模型的角色。

貝氏定理的公式如(D.3)所示，

$$P(\theta|X) = \frac{P(X|\theta)P(\theta)}{P(X)} \tag{D.3}$$

是由看到證據前的**先驗機率**（prior）$P(\theta)$（原有信念），乘上所收集到**新證據**X在原有信念下所出現的可能，也就是**概似函數**（likelihood）$P(X|\theta)$（可參閱機率論或統計學相關書籍）（可參閱機率論或統計學相關書籍），並接著除去新證據基於**所有可能的機率**（margin）$P(X)$後，便可得到更新後的**後驗機率**（posterior）$P(\theta|X)$（新信念）。若將貝氏推論以分配的形式表示，並以視覺化呈現如圖 D.7 所示，貝氏推論將信念與證據串連成新的信念。左邊的分配為**先驗機率**（prior），且這個分配具有一定的**不確定性**（如高斯過程的隨機性），這取決於對於先驗的信念有多強烈，當我們的經驗很有把握時可以給予相對較小的變異；相反地，若把握不足則變異較大。而右邊的分配為**概似函數**（likelihood），是由我們收集到的證據所形成的分配，且這個分配具有一定的噪音（樣本的隨機性），它是真實資料的分布。而中間的分配則是**後驗機率**（posterior），是綜合我們的先驗機率（信念）與概似函數（證據）所得到的估計結果。

這裡舉一個例子來說明傳統統計**頻率論**與**貝氏推論**的差異。倘若今天我們想檢驗一個錯誤的假設，這個假設宣稱不良品是由 B 製程機台造成

的，並且不良品表面上有一刮痕是該 B 製程機台所使用的刀具以作為證據。在傳統的推論中，我們會先基於假設是正確的前提加以解釋這項證據，顯然該證據的正確性將導致我們可能做出誤判。而在貝氏推論中，我們基於經驗（或領域知識）對於不良品的發現是由於 A 機台先使厚度變厚超出規格，而造成後續 B 製程機台使用刀具加工後在產品表面有所刮傷（先驗機率）。因此原先對於這個假設的信念是低的，即便再看到新的證據說明不良品刮痕與 B 機台刀具相符（概似函數），在更新後對於這個假設的認同度依舊很低（後驗機率）。

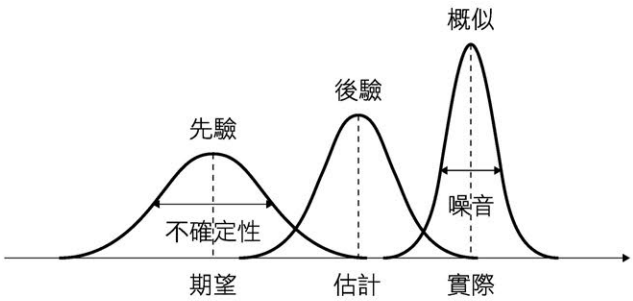

圖 D.7　先驗機率、概似函數與後驗機率

回到貝氏最佳化中，究竟要如何將收集到**新函數關係**（證據）更新**高斯過程**（先驗分配）以近似實際的**目標函式**（後驗分配）呢？首先，我們將貝氏定理的公式(D.3)轉成貝氏最佳化的形式如公式(D.4)所示，

$$P(f|\mathcal{D}_{1:t}) = \frac{P(\mathcal{D}_{1:t}|f)P(f)}{P(\mathcal{D}_{1:t})} \tag{D.4}$$

其中我們定義先驗分配為一個高斯過程$P(f)$，並期望透過新證據的收集能降低這個隨機函數的不確定性；概似函數$P(\mathcal{D}_{1:t}|f)$（新證據）則在給定由t組超參數$x_{1:t}$以及其對應到的模型訓練結果$f(x_{1:t})$所形成的配對$\mathcal{D}_{1:t} = (x_{1:t}, f(x_{1:t}))$下可能的函數；而邊際$P(\mathcal{D}_{1:t})$是一個常數（其效果用於將分子標準化）。因此，最後我們可得到的後驗分配為$P(f|\mathcal{D}_{1:t})$，且同樣為一個高斯過程。

然而，我們更好奇的是基於這個得到的後驗分配在任意x上對應函數值可能的分布，也就是當有任意一個新的x_{t+1}時，其$f(x_{t+1})$可能會落在哪

邊，這時則需仰賴「**預測性後驗分配**」（predictive posterior distribution）的輔助。首先，我們可將已有的證據$\mathcal{D}_{1:t}$與好奇的新樣本x_{t+1}所形成的**聯合多變量常態分配**表示如公式(D.5)所示，

$$\begin{bmatrix} f_{1:t} \\ f_{t+1} \end{bmatrix} \sim N\left(0, \begin{bmatrix} \mathbf{K} & \mathbf{k} \\ \mathbf{k^T} & k(x_{t+1}, x_{t+1}) \end{bmatrix}\right)$$
$$\mathbf{k} = [k(x_{t+1}, x_1), k(x_{t+1}, x_2), \ldots, k(x_{t+1}, x_t)] \tag{D.5}$$

其中**原有的共變異函數**（矩陣）為K，**新共變異函數**的元素則包含了新樣本x_{t+1}與其他已知樣本以及自身的共變異$\mathbf{k}$與$k(x_{t+1}, x_{t+1})$。因此基於常態分配的條件機率下，**預測性後驗分配**$P(f_{t+1}|\mathcal{D}_{1:t}, x_{t+1})$推導後的結果如公式(D.6)所示。

$$P(f_{t+1}|\mathcal{D}_{1:t}, x_{t+1}) = N\left(\mu_t(x_{t+1}), \sigma_t^2(x_{t+1})\right)$$
$$\text{where} \begin{cases} \mu_t(x_{t+1}) = \mathbf{k^T K^{-1}} f_{t+1} \\ \sigma_t^2(x_{t+1}) = k(x_{t+1}, x_{t+1}) - \mathbf{k^T K^{-1} k} \end{cases} \tag{D.6}$$

對於任意的每個點而言都服從一個平均數為$\mu_t(x_{t+1})$而變異數為$\sigma_t^2(x_{t+1})$的常態分配。而這個預測性後驗分配的結果可對應到上述的圖 D.6，圖中三個黑點與對應的函數為$\mathcal{D}_{1:3}$，而任意給定的新樣本x_{t+1}（例如圖中x_1, x_2, x_3樣本）。我們可以觀察在這個此分配中，收集到證據的已知函數點具有較小的變異，而與已知點越遠的地方則是不確定性越大，這樣的關係對於我們追求的真實代理模型（反應曲面）來說相當合理。從最原先的高斯過程是在某個函數範圍下隨機的函數分布，當我們進一步收集到真正的函數關係後，將該點的不確定性壓縮，而離證據靠近的地方則相信他們之間的關係較為緊密。

緊接著回顧超參數最佳化的主要目標，即是找出使得損失函數最小的超參數組合。前述說明了以**高斯過程**為核心的**代理模型**以及**貝氏推論**在新證據上的**更新機制**，我們期望有越多精確的證據使得能近似真實超參數與損失函數的關係，於是接下來的重要議題，在於該在哪裡收集新證據？並且收集證據時還要同時兼具效率呢？更具體的說，超參數的**搜尋方向**該依據什麼呢？例如圖 D.6 的高斯過程中，我們應該直接往**最小的目標函數值**去搜尋，還是應該往**不確定性最大的空間**去搜尋呢？

D.3.4 收穫函數

上述提到兩個搜尋的方向，一是往**目標函數最小**的方向，二是**不確定性最大**的方向，而我們應該如何選擇呢？在最佳化領域中，並非是個二擇一的問題，而是在這兩者之間取得一個平衡：往最佳解方向的稱之為**開採**（exploitation），往不確定性的方向的稱之為**探索**（exploration）。這樣的平衡稱之為**開採與探索的權衡**（exploration-exploration trade-off），許多最佳化的方法皆架構在這個觀念上（參閱章節「**元啟發式演算法**」）。舉一個簡單的案例，假設今天我們出社會工作後，已經花了 10 年分別曾任職於三家公司，開始思考一個決策問題，是否應該從這三家公司中挑最合適的公司，重新對該公司進行**開採**？又或還是去找尋其他新的機會，**探索**別的公司呢？這便是開採與探索的權衡。換言之，於超參數最佳化中，在我們得到具有不確定性（變異數）的預測性後驗分配後，如何取得開採與探索的權衡呢？

收穫函數則是作為引導最佳解**搜尋方向**的工具。高的收穫函數值對應到的是潛在最佳解的抽樣位置，也就是**最小損失函數值**（開採）與**最大不確定性**（探索）的權衡，因此在超參數最佳化中，以**最大化收穫函數**後的超參數組合作為下一個模型訓練與評估的樣本。收穫函數的選擇，我們介紹以下三種方法分別是：「**機率進步函數**」（probability improvement, PI）、「**期望進步函數**」（expected improvement, EI）以及「**上信賴邊際**」（upper confidence bound, UCB），以下均以最大化目標函數為例（實際可能是最小化損失函數，加上負號替換方向即可）。

D.3.4.1 機率進步函數

機率進步函數$\mathrm{PI}(x)$是以**開採**為主要核心的方法，如公式(D.7)所示，

$$\begin{aligned}\mathrm{PI}(x) &= P\big(f(x) \geq f(x^+)\big) \\ &= \Phi\left(\frac{\mu(x) - f(x^+)}{\sigma(x)}\right)\end{aligned} \tag{D.7}$$

其中$x^+ = \mathrm{argmax}_{x_i \in x_{1:t}} f(x_i)$是目前觀測值當中具有最佳目標函數值的，而$\Phi$代表常態的累計分配函數。搭配圖 D.8 的示意圖（Brochu et al., 2010），若以x_3做為範例，首先找出x^+與其對應到的$f(x^+)$，接著計算x_3的機率進步值$\mathrm{PI}(x_3)$，可看出相當於計算右圖常態分配區域的面積。若我們

接著觀察x的變化，當x_3往左邊移動時，平均數越小，而變異也跟著變小；往右時平均數與變異數變大，而發生最大面積的值大約落於x_3往右移動的位置。然而，在開採與探索的權衡視角上，當區域解出現時，容易導致整體以開採為主要方向，這是由於平均數對於機率進步值的影響力遠遠凌駕於變異數之上。雖學者後續提出增加一個權衡的參數ξ以降低平均數的影響力，如公式(D.8)所示。

$$\begin{aligned} \mathrm{PI}^*(x) &= P(f(x) \geq f(x^+) + \xi) \\ &= \Phi\left(\frac{\mu(x) - f(x^+) - \xi}{\sigma(x)}\right) \end{aligned} \tag{D.8}$$

但在本質上此函數依舊以開採為主，探索的能力有限，因此**期望進步**接著被提出以達到更好的平衡。

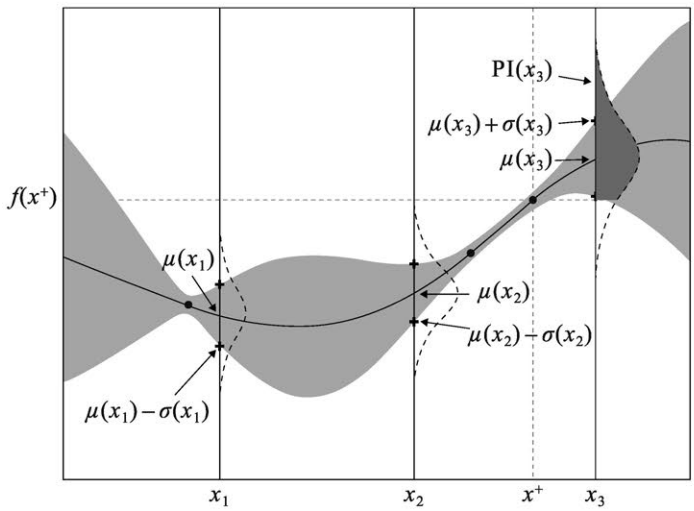

圖 D.8　機率進步的收穫函數（Brochu et al., 2010）

D.3.4.2　期望進步函數

基於前述的**機率進步函數**，若我們除了考慮進步的累積機率，也希望同時考慮到每個機率的函數值。舉例來說，若我們找到兩個x所對應到的機率進步函數值相等，在這樣的情況下，我們應該認為變異較大的那一個更具有潛在的可能越大。因此，若我們進一步去計算進步的期望值（期望

值為函數值乘上發生的機率$\mathrm{E}(x) = x \cdot P(x)$，變異較大的其期望值會較大）是不是更加合理呢？於是乎**期望進步函數**$\mathrm{EI}(x)$就在這樣的思維下發展而來。首先，**進步**可被定義如公式(D.9)所示，

$$\mathrm{I}(x) = \max\{0, f_{t+1}(x) - f(x^+)\} \tag{D.9}$$

其中只取正的進步值是為了減少不必要的搜尋，而我們的目標則是求出最大化期望進步則如公式(D.10)所示。

$$x = \underset{x}{\operatorname{argmax}}\, \mathrm{E}(\max\{0, f_{t+1}(x) - f(x^+)\} \,|\mathcal{D}_{1:t}) \tag{D.10}$$

接著，我們先列出進步函數$\mathrm{I}(x)$的**常態後驗機率分配**如公式(D.11)所示。

$$\mathrm{I} \sim \frac{1}{\sqrt{2\pi}\sigma(x)} \exp\left(-\frac{(\mu(x) - f(x^+) - \mathrm{I})^2}{2\sigma^2(x)}\right) \tag{D.11}$$

將此機率分配經積分後即可計算出**期望值**如公式(D.12)所示，

$$\begin{aligned} \mathrm{EI}(x) &= \int_{\mathrm{I}=0}^{\mathrm{I}=\infty} \mathrm{I} \frac{1}{\sqrt{2\pi}\sigma(x)} \exp\left(-\frac{(\mu(x) - f(x^+) - \mathrm{I})^2}{2\sigma^2(x)}\right) d\mathrm{I} \\ &= \sigma(x)\left[\frac{\mu(x) - f(x^+)}{\sigma(x)} \Phi\left(\frac{\mu(x) - f(x^+)}{\sigma(x)}\right) + \phi\left(\frac{\mu(x) - f(x^+)}{\sigma(x)}\right)\right] \end{aligned} \tag{D.12}$$

其中$\phi(\cdot)$與$\Phi(\cdot)$分別為標準常態分配的**機率密度函數**（pdf）與**累積密度函數**（cdf）。經整理後，我們可以得到**期望進步函數**如公式(D.13)所示，

$$\begin{aligned} \mathrm{EI}(x) &= \begin{cases} (\mu(x) - f(x^+))\Phi(Z) + \sigma(x)\phi(Z) & \text{if } \sigma(x) > 0 \\ 0 & \text{if } \sigma(x) = 0 \end{cases} \\ Z &= \frac{\mu(x) - f(x^+)}{\sigma(x)} \end{aligned} \tag{D.13}$$

事實上，我們可將公式(D.13)拆解成兩部分。若**以開採與探索的**視角來看，前半部的**進步**$\mu(x) - f(x^+)$乘上累積密度函數代表**開採**，後半部的**變異數**$\sigma(x)$乘上機率密度函數則代表**探索**。因此，**開採與探索的權衡**則仰賴**機率密度函數**與**累積密度函數**的加權，我們分別討論兩種情形加以說明上述的這項宣稱。第一，若**平均數有進步**而**變異數維持不變**，此情形下，前項的進步與其累積密度函數均會增加，然而後項變異數不變而機率密度函

數變小，這樣的關係說明了在開採上有明顯的增進，但相對由於較少的探索因而減低了期望進步函數（可視為某種懲罰）。第二，相反的狀況，若**平均數沒有進步**而**變異數變大**，此情形下，後項的變異與其累積密度函數均會增加，然而前項平均數不變而機率密度函數變小，因此與第一種情形相反，雖在探索增加了期望進步函數，但在開採則部分降低了此函數值。綜合上述兩種情形，說明了期望進步函數在開採與探索達到更好的權衡，即便如此，如同機率進步函數(D.8)可增加一個參數ξ調整開採與探索的權衡，如公式(D.14)所示。

$$\mathrm{EI}^*(x) = \begin{cases} (\mu(x) - f(x^+) - \xi)\Phi(Z) + \sigma(x)\phi(Z) & \text{if } \sigma(x) > 0 \\ 0 & \text{if } \sigma(x) = 0 \end{cases}$$
$$Z = \frac{\mu(x) - f(x^+) - \xi}{\sigma(x)} \tag{D.14}$$

D.3.4.3 上信賴界限

上信賴界限（UCB）是另一個也考慮開採與探索權衡的收穫函數。信賴邊界限的思維更加簡單，既然我們得到了**預測性後驗分配**，若我們直接使用其信賴區間的**上界**（upper bound），是不是同樣的也能往潛在最佳解的方向，如(D.15)所示，

$$\mathrm{UCB}(x) = \mu(x) + \kappa\sigma(x) \tag{D.15}$$

其中κ如同前述兩種收穫函數中的參數ξ，也扮演開採與探索的權衡。若給定較大的κ，則更趨向**探索**，反之則趨向**開採**。

D.3.4.4 收穫函數最大化

接著說明**收穫函數最佳化**，其可視為將貝氏最佳化拆解成多個子最佳化問題，與原始最佳化的黑盒子問題不同的在於收穫函數是已知的，因此一般會使用確定性且不須微分的「DIRECT **最佳化**」方法（將可行解域切割成多個矩形）。

我們比較上述介紹的三種收穫函數：**機率進步函數**（PI）、**期望進步函數**（EI）以及**上信賴界限**（UCB）。如圖 D.9 所示（Brochu et al., 2010），最上方的圖爲收集到四個樣本（圓點）的**預測性後驗分配**，接著探討使用不同收穫函數（其他三張圖，三角形為最大的收獲函數）以及不

同的參數對於搜尋方向的**開採與探索權衡**。在機率進步函數（PI）的圖中，較小的參數ξ使得搜尋方向更傾向開採（探索能力較弱）；在期望進步函數（EI）的圖中，與 PI 使用相同的參數ξ時，搜尋方向傾向探索多一些；在上信賴界限（UCB）的圖中，參數κ的特性與參數ξ相同，越大越傾向探索。從不同收穫函數的比較中，可發現在 EI 與 UCB 的結果在開採與探索權衡的表現以及可調整的彈性均較佳。

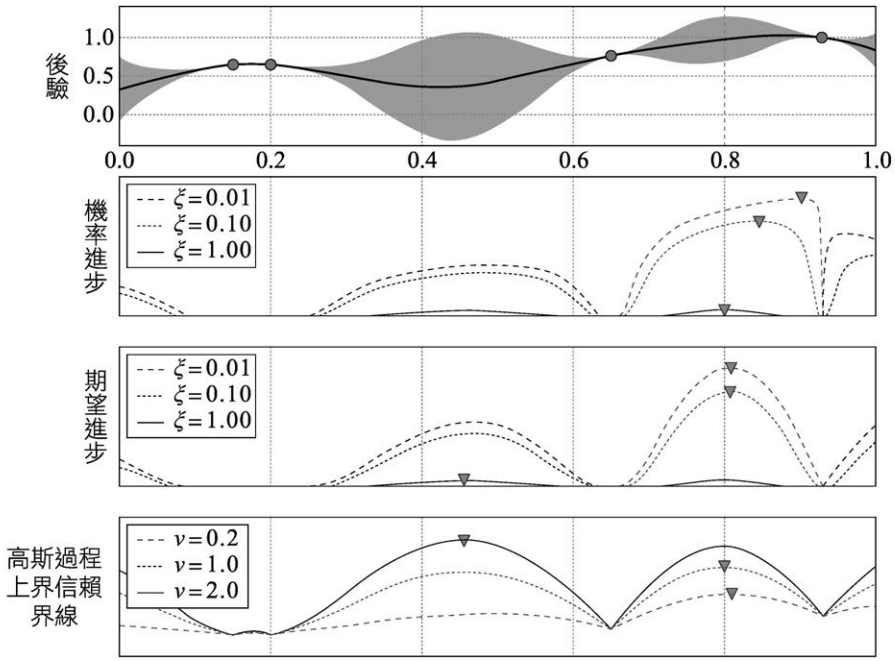

圖 D.9　機率進步、期望進步與信賴界限的比較（Brochu et al., 2010）

D.3.5　貝氏最佳化演算法

最後，整理貝氏最佳化完整的運作流程。將前述提及的**高斯過程**（代理模型）、**貝氏推論**（模型更新）以及**收穫函數**（搜尋方向）串連起來，如圖 D.10 所示（Brochu et al., 2010）。虛線為真實的函數，實線為**高斯過程後驗分配**的平均數，機率密度區域則為其平均數加減一倍標準差，下方區域則是**收穫函數**。令t為所收集的觀測值，在$t = 2$時（第一張子圖）收集兩個觀測值，基於這兩點可計算出預測性後驗分配，接著依照最大的收穫函數（使用期望進步函數）選擇下一個樣本點（三角形位置，為第三個觀測值）。接著在$t = 3$時藉由三個已知觀測值與其函數值，並陸續更新高

斯過程，對應到的步驟分別為：**建構與更新代理模型**、**收穫函數最大化**、**模型訓練與結果收集**，依此類推直到滿足停止準則，並得到目前最佳的超參數組合。

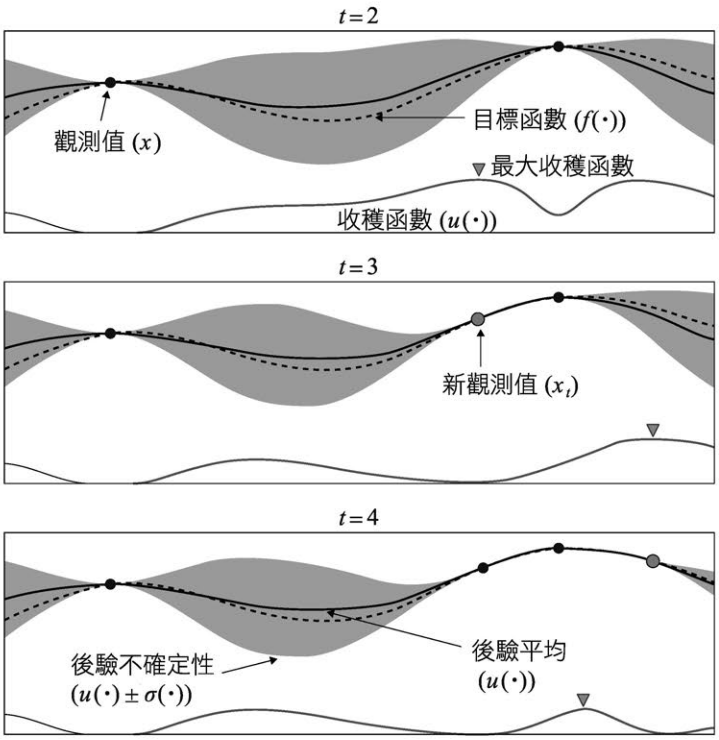

圖 D.10　貝氏最佳化的運作流程（Brochu et al., 2010）

貝氏最佳化演算法的虛擬碼如表 D.1 所示，如同前述提到的四個步驟（建構與更新代理模型、收穫函數最大化、模型訓練與結果收集）。貝氏最佳化涵蓋了大量統計與最佳化的思維，本小節首先從**反應曲面法**的最佳化思維說明了**代理模型**相較於**窮舉搜尋**的差異，並介紹了具有高度彈性的**高斯過程**（隨機過程）代理模型，以及高斯過程如何藉由**貝氏推論**加以更新，接著介紹**收穫函數**使新樣本點能取得**開採與探索的權衡**。前述的步驟串連一起就是完整的貝氏最佳化。

表 D.1　貝氏最佳化的虛擬碼

演算法 貝氏最佳化
1. For $t = 1,2, \ldots$:
2.　　基於高斯過程的預測性後驗分配，找出使得最大化收穫函數的新樣本點x_t;
3.　　計算出新樣本點的目標函數（模型訓練後的結果）$y_t = f(x_t)$;
4.　　增加新資料$\mathcal{D}_{1:t} = \{\mathcal{D}_{1:t-1}, (x_t, y_t)\}$，並更新高斯過程;
5. End
6. Return 最佳解x^*

D.4　結語

本章節分別介紹了三種超參數方法分別為屬於**窮舉搜尋**類型的**網格搜尋法**、**隨機搜尋法**以及屬於**代理模型**類型的**貝氏最佳化**等方法。表 D.2 整理出各方法的假設、使用時機與優劣。若建構模型的超參數個數少，較為建議使用隨機搜尋法即可；當模型複雜度高以及時間與資源受限時，則會建議使用貝氏最佳化，而開採與探索的權衡決定了解的速度與品質。一般在貝氏最佳化中，當平衡傾向於**開採**時，則解收斂的速度較快（所需訓練的模型也較少），但**品質差**較容易產生**區域最佳解**；另一方面，當平衡傾向於**探索**時，則解收斂速度慢，但更充分探索解空間所得到的解**品質較高**。因此當我們資源足夠時，則建議在貝氏最佳化中可試著**調升探索**的權重。

表 D.2　超參數最佳化方法比較

方法		假設與使用時機	優點	缺點
窮舉搜尋	網格搜尋法	超參數個數少	1. 方法實作容易 2. 可平行運算	1. 間隔不易給定 2. 資訊的獨立性 3. 受維度詛咒影響
	隨機搜尋法	超參數個數少	1. 方法實作容易 2. 可平行運算 3. 搜尋的解空間較大	1. 分配不易給定 2. 資訊的獨立性 3. 受維度詛咒影響
代理模型	貝氏最佳化	超參數個數多 時間與資源有限	1. 得到的超參數組合最為精確 2. 善用每一個樣本資訊 3. 可同時得到反應曲面	1. 不可平行運算 2. 開採與探索的權衡需參考資源的限制

參考文獻

[1] Brochu, E., Cora, V. M., and De Freitas, N. (2010). A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *ArXiv Preprint* https://arxiv.org/abs/1012.2599v1.

[2] Feurer, M., and Hutter, F. (2019). Hyperparameter optimization. In: Hutter F., Kotthoff L., Vanschoren J. (eds) *Automated Machine Learning*, pp. 3-33. The Springer Series on Challenges in Machine Learning. Springer, Cham.

[3] Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 25, 2960-2968.

問題與討論

1. (a)試說明超參數與模型參數的差異。(b)試舉例用一個模型說明之。
2. (a)試說明為什麼要進行超參數最佳化呢？(b)其挑戰與困難之處為何？
3. (a)超參數最佳化主要包含了哪些方法？(b)這些方法分別的使用時機與優劣為何？
4. (a)試說明貝氏最佳化的流程（提示：繪製流程圖）？(b)其代理模型與收穫函數分別的功用為何？(c)為什麼這樣的代理模型方法比窮舉搜尋更精準或更有效率？
5. (a)試說明高斯過程與常態（高斯）分配的差異與關係為何？(b)為什麼貝氏最佳化主要以高斯過程為代理模型？
6. (a)試簡述貝氏推論的核心觀念（提示：試畫圖或舉案例說明）。(b)試說明貝氏最佳化是如何使用貝氏推論來更新高斯過程（代理模型）的（提示：試畫圖輔助說明）？
7. 貝氏最佳化以收穫函數作為評估新超參數組合的搜尋方向，試問(a)為什麼不直接以代理模型的最小值作為新超參數組合？(b)試列舉一收穫函數並說明其函數設計。
8. 試應用貝氏最佳化對以下問題進行超參數最佳化，並與網格搜尋法、隨機搜尋法進行比較：

   (a) 第 8 章「**決策樹與集成學習**」的問題 11：以 UCI Machine Learning

Repository 開放數據中的半導體製造數據（semiconductor manufacturing dataset，https://archive.ics.uci.edu/ml/datasets/SECOM）的分類問題建構決策樹、隨機森林以及梯度提升機。

(b) 第 10 章「**類神經網路與深度學習**」的問題 12：以 Kaggle 開放數據中的半導體晶圓圖數據（WM-811K wafer map dataset， https://www.kaggle.com/qingyi/wm811k-wafer-map）的分類問題建構類神經網路模型。