基於機器學習實現智慧化水質分析與評估系統

(一) 研究計畫之背景。

1. 研究計畫之背景

本研究旨在開發並實現具機器學習（Machine Learning, ML）功能的智慧化水質分析與評估系統，以解決目前水質資料分析缺乏高效率方法的問題。研究作者曾擔任科技部計畫兼任研究助理，協助執行「運用智慧型綠能永續景觀噴泉促進環境保育效益之研究」（計畫編號：MOST 110-2410-H025-024），該計畫運用物聯網（Internet of Things, IoT）技術蒐集南投麒麟潭的水質資料。藉由該計畫經驗發現自動化運行的結果相較於人工記錄更有效率，但後續資料分析仍需耗費大量工作時間。

2. 研究問題

在水質資料分析方面，除了使用 Microsoft Excel 直接計算外，通常會使用如 Statistical Analysis System（SAS）或 Statistical Product and Service Solutions（SPSS）等統計專有軟體（Proprietary Software）。然而，這些軟體為專有（Proprietary）且非開源，不僅需要額外購買授權，還需投入時間學習才能熟練使用。由於其操作高度依賴使用者的專業知識，分析結果可能因個人經驗而有所差異。由於這些軟體無法修改與物聯網設備直接進行資料串聯傳輸，造成未來計畫整合實現自動化資料收集與資料分析的障礙。

3. 研究原創性

近十年來，人工智慧（Artificial Intelligence, AI）從理論發展到實際應用，各種 AI 等工具也被廣泛運用於日常問題成為解決方案 [1]。本研究運用人工智慧中的機器學習技術，開發一套智慧化水質分析與評估系統，實現更高效且易於操作的創新水質分析方法。不同於傳統統計軟體需要人工輸入與分析，本系統將結合自動化資料處理與機器學習演算法，以提升水質分析的準確性與效率。此外，本研究將探索 AI 在水資源管理與環境保育領域的創新應用，提供創新的智慧化水質資料分析方案。

為了找出最適合水質分析的機器學習模型，本研究針對 XGBoost、LightGBM、支援向量機（Support Vector Machine, SVM）、隨機森林（Random Forest, RF）、多元多項式迴歸（Multiple Polynomial Regression, MPR）以及線性迴歸（Linear Regression, LR）等多種模型進行比較與評估。透過實驗分析與比較不同模型在水質資料分析準確度、計算效能與資料適應性的表現，進一步應用於改良水質分析，使系統能夠有效應對水體環境的分析與評估需求。

實驗設計包括使用先前於「運用智慧型綠能永續景觀噴泉促進環境保育效益之研究」計畫中，使用物聯網設備實際蒐集的水質資料，加上環境部環境資料開放平臺公開資料使用，特徵選擇包括溶解氧（DO）、生化需氧量（BOD）、氨氮（NH3-N）、電導度（EC）和懸浮固體（SS）等關鍵指標。在訓練過程中，使用水體品質指標（Water Quality Index, WQI_5）作為標記資料進行監督式學習。

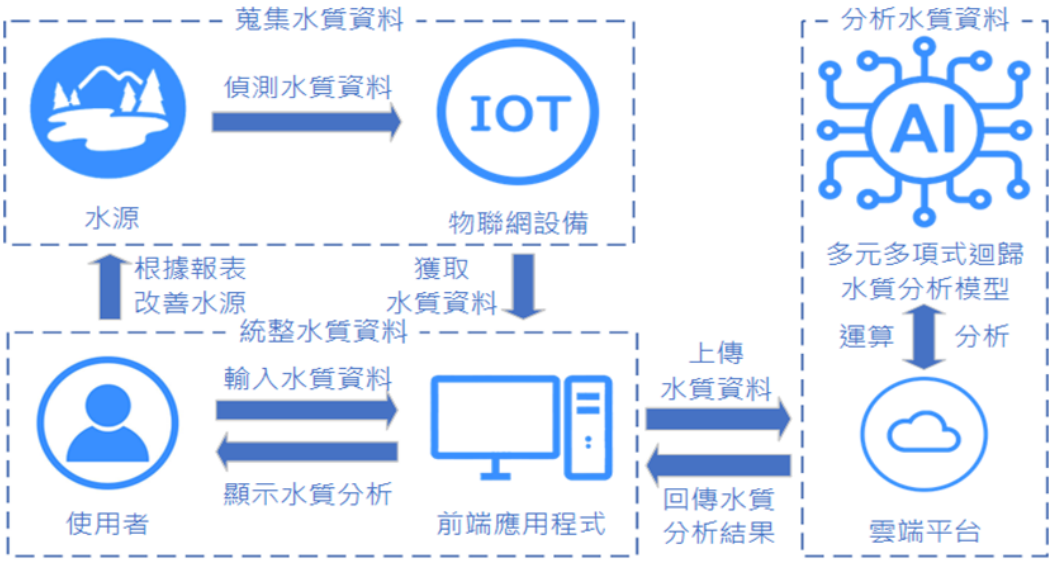

圖 1: 使用情境圖

4. 研究重要性

在過去十年內，人工智慧的研究從長期被忽視的陰霾中走了出來，被人們看見了希望且掀起了新一波風潮 [2]，而水資源管理與環境生態保育是全球關注的議題，而水質分析與評估是確保水資源安全的重要環節。本研究透過機器學習技術，克服傳統方法的效率瓶頸，降低對專業知識依賴，使水質資料分析變得更有效且易於操作，提升水質資料分析技術的普及性與可行性。

本研究針對 XGBoost、LightGBM、支援向量機（SVM）、隨機森林（RF）、多元多項式迴歸（MPR）以及線性迴歸（LR）等機器學習模型進行獨立訓練，並透過決定係數（R^2）、均方根誤差（Root Mean Square Error, RMSE）、平均絕對誤差（Mean Absolute Error, MAE）以及殘差分析等指標評估各模型在水質資料分析上的準確性。

透過實驗分析，我們發現 LightGBM 在大資料量（50000 筆）下表現最佳，其測試集 R^2 達到 0.9991，MAE 為 0.4009，RMSE 為 0.6174。相較之下，XGBoost 的 R^2 為 0.9990，MAE 為 0.4173，RF 的 R^2 為 0.9987，MAE 為 0.4044，顯示 LightGBM 在準確性和誤差控制上均略勝一籌。這些結果不僅驗證了 LightGBM 在處理大規模水質資料時的優異性能，也為未來在類似領域中選擇合適的機器學習模型提供了實證依據。透過本研究的實驗分析，能夠更準確地分析水質指標，從而提升水質分析與評估的效率和準確性。

5. 預期影響性

本研究的智慧化水質分析與評估系統結合了模型比較實驗與系統開發，有望在以下方面產生影響：

1-5-1. 技術創新：人工智慧未來將成為人類不可或缺的夥伴 [3]，本研究在水質資料分析引入機器學習技術，提升分析準確性與自動化程度。

1-5-2. 環境保育：能夠在水資源管理方面準確且有效地發現水質變化，改善水體生態環境。

1-5-3. 跨領域應用：本研究結合了資訊工程、人工智慧、水資源管理和環境生態保育等不同領域，未來也可擴展應用於不同環境監測領域，例如空氣品質監測或污染物分析，促進人工智慧在環境科學中的應用。

6. 國內外研究情況

本研究作者先前參與的計畫已經使用物聯網技術進行水質資料的採集，這種方法在環境監測中已經相當普遍。然而，目前國內外針對水質資料分析的智慧化系統，特別是結合機器學習技術的水質分析系統相對較少。機器學習技術在水質評估中的應用已被證明是有效的，並逐漸成為水資源管理的重要分析工具 [4]，應用各種技術構建學習模型，能夠達成提供即時準確水質資訊的目的，期望得以透過早期診斷水質異常以改善公共健康 [5]。因此，本研究具有創新性和獨特性，有助於填補水質資料分析中高效方法的空白。

7. 重要參考文獻

隨著人工智慧發展，各種機器學習模型也如雨後春筍般出現。模型實作上如開發一種基於多元多項式迴歸（MPR）的機器學習模型，在該研究中以模型與線性迴歸及支援向量機模型進行比較，發現 MPR 在性能評估指標包含 R^2（R-squared）和 RMSE（Root-Mean-Square Error）上皆優於另外兩個模型 [6]，其提出的 MPR 公式如 (1) 所示，其中 $\hat{y}$ 是輸出變數，w_0 是截距，$w_1, w_2, \ldots, w_s$ 為多項式迴歸係數，x 為輸入變數，表示具有 n 個輸入變數和階數（$s > 1$）的 MPR 系統。加上其他研究成果使用 MPR 模型驗證其預測結果 [7]，如圖 2 所示，顯示了前三個模型的殘差和高斯函數，它們都具有正態分佈模式，與 RF 和 LR 演算法相比，MPR 模型似乎具有更高水準的預測準確性，因為 MPR 的 80% 殘差在 [−17.267，17.267] 範圍內。此外，MPR 模型的殘差擬合正態分佈的平均值為 0.703，更為接近 0; 也就是說，MPR 模型具有更好的誤差隨機性。[8]。這些研究成果將有助於本研究的模型選擇與效能評估，也因此本研究前期採用 MPR 模型作為基準模型進行比較。

$$\hat{y} = w_0 + \sum_{i_1=1}^{n} w_{i_1} x_{i_1} + \sum_{i_1=1}^{n} \sum_{i_2=i_1}^{n} w_{i_1 i_2} x_{i_1} x_{i_2} + \cdots + \sum_{i_1=1}^{n} \sum_{i_2=i_1}^{n} \cdots \sum_{i_k=i_{k-1}}^{n} w_{i_1 i_2 \cdots i_k} x_{i_1} x_{i_2} \cdots x_{i_k} \quad (1)$$

另外，隨著近年來 XGBoost 與 LightGBM 等新興模型在機器學習領域中的崛起，本研究在中後期將其納入比較範疇，XGBoost 在資料科學領域中被廣泛使用，也在 Kaggle 許多機器學習及競賽中取得最佳結果，其特點為有非常強的擴展性（Scalibility）與性能，可以使用相對其他模型更少的系統資源，擴充數十億級別的資料 [9]。而 LightGBM 是一種

高效能的梯度提升決策樹，在傳統 Gradient Boosting Decision Tree（GBDT）演算法上加入使用 Gradient-based One-Side Sampling（GOSS）和 Exclusive Feature Bunding（EFB），並顯示出 LightGBM 相較傳統的 GBDT 加快了 20 倍以上的速度，同時也並未降低準確性 [10]。

綜合以上重要參考文獻，本研究對 XGBoost、LightGBM、支援向量機（Support Vector Machine, SVM）、隨機森林（Random Forest, RF）、多元多項式迴歸（Multiple Polynomial Regression, MPR）以及線性迴歸（Linear Regression, LR）等多種模型進行比較與評估，期望找出最適合水質分析的機器學習模型，並提升水質分析的準確性與效率。

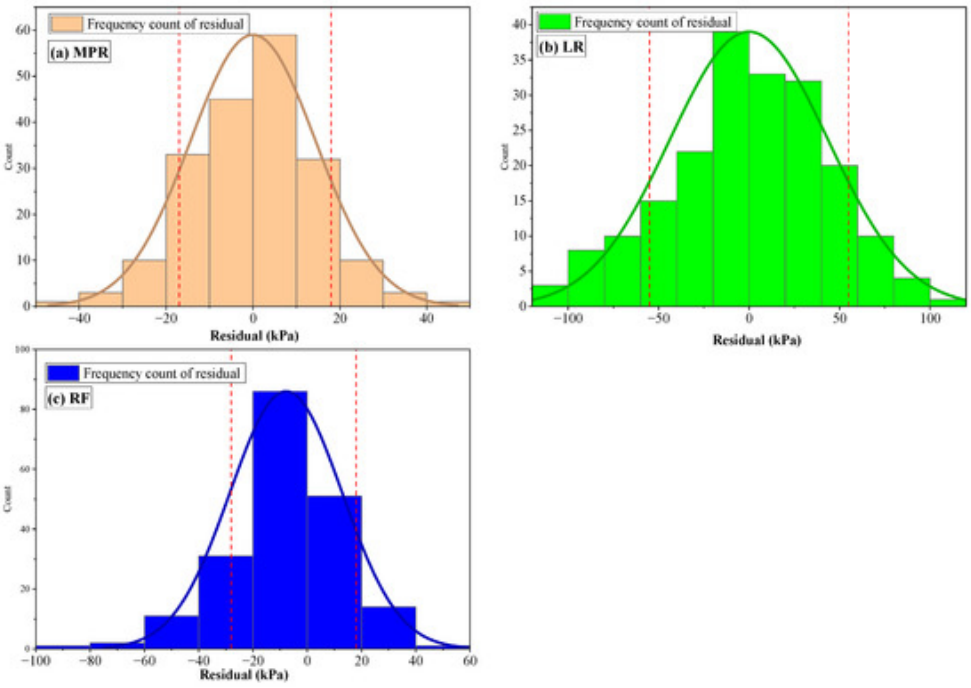

圖 2: Zeini 等人的 5 倍交叉驗證訓練數據集殘差的正態分佈

(二) 研究方法、進行步驟及執行進度。

1. 研究方法

本研究開發之智慧化水質分析與評估系統採用前後端分離架構，透過應用程式介面（Application Programming Interface, API）進行資料傳遞與呼叫。後端（Back-end）水質資料分析模型以 Python 程式語言開發，運用 NumPy、Pandas 及 Scikit-learn 等工具進行水質資料的處理和分析。前端（Front-end）則以 JavaScript 語言的 React Native 框架開發行動裝置應用程式，實現跨系統平臺操作。此架構的水質資料分析模型開發可歸納為四個步驟：匯入資料、資料處理、特徵工程及模型訓練與評估。系統建置完成後，使用者在前端行動裝置應用程式中只需輸入欲分析的水質資料，即可經由已訓練完成的機器學習模型進行分析，再於使用者介面（UI）將產生一份分析報表，包括水質狀況的綜合評分以及狀態說明。

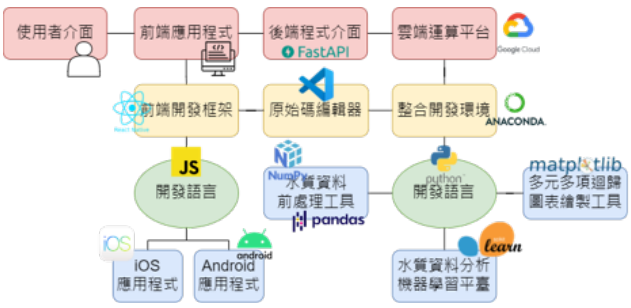

圖 3: 系統程式開發架構圖

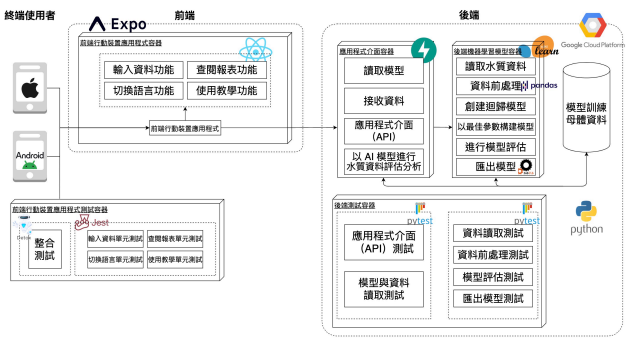

圖 4: 系統部署圖

在模型開發過程中，我們認為機器學習技術不僅適用於水質資料分析，也能拓展到其他相關領域。除了前期使用的 MPR 模型外，我們還加入 XGBoost 和 LightGBM 等新興優秀模型進行比較，以尋找最適合解決問題的方案。研究工具方面涵蓋了機器學習程式語言 (2-1-1)、開發環境 (2-1-2)、功能函式庫 (2-1-3) 及前端開發工具 (2-1-4)，用以構建一套完整的水質資料分析與評估系統，並確保能挑選出最佳模型。各項工具分述如下：

2-1-1. 機器學習語言

近年來 Python 在人工智慧領域中具有顯著影響力，因其優秀的資料處理與分析能力，支援各式開源函式庫使得 Python 成為資料分析的熱門選擇 [11]，且能夠有效處理即時、大型和非結構化資料 [12]，對於本研究進行各式水質資料分析的研究具有明顯的優勢。

2-1-2. 機器學習環境

本研究機器學習環境主要分為 (A) 雲端運算平臺及 (B) 開發環境。正如所言「工欲善其事，必先利其器」，為了能夠有效率地進行整體系統的各項開發，並在完成後部署於雲端運算平臺。

A. 雲端運算平臺

本研究運用機器學習框架－ Scikit-learn 進行功能開發，Scikit-learn 著重高效能的運算資源利用，符合智慧化水質資料分析的應用需求。此外，亦將建置雲端平臺－開發與部署所建置之 MPR 模型於 Google Cloud Platform（GCP），GCP 為一種基礎結構即服務（IaaS）平臺，提供於雲端部署機器模型之能力。

B. 開發環境

本研究採用 Visual Studio Code 作為整合開發環境（Integrated Development Environment, IDE），並透過 Docker 實現容器化開發。可以直接將所開發之機器學習模型與應用程式部署至雲端平臺，運用分散式架構使得本研究具備處理和運算大量水質資料的能力。

2-1-3. 功能函式庫

程式開發所需之功能函式庫分為：(A) 資料處理工具、(B) 機器學習框架、(C) 圖表繪製工具及 (D) 應用程式介面。實作上以模組化方式結合各項工具能有助於完成水質分析模型的建構，相關說明如下：

A. 資料處理工具

本研究以 NumPy 及 Pandas 作為資料處理工具，NumPy 是 Python 程式語言中支援科學和數值計算的函式庫，也相容於 Pandas、Matplotlib 和 Scikit-learn 等工具 [13]。NumPy 作為資料處理的主要工具是因為它提供了科學計算所需的基本資料結構 [14]。同時 NumPy 也提供豐富的數學運算、資料讀寫、線性代數、傅立葉變換和隨機數生成等功能函式 [15]。使用 NumPy 得以更有效率地處理原始資料，進一步提升程式的開發效率。而 Pandas 建立於 NumPy 的基礎之上，是一種用於格式化資料分析的開源 Python 函式庫，它能夠處理類似電子試算表格式的輸入資料，可以快速地進行資料載入、操作、對齊和合併等功能 [16]。

B. 機器學習框架

本研究採用了 Scikit-learn 作為機器學習框架，以建構研究所需的各種機器學習模型模型 [17]。支援 Python 程式的機器學習框架之中，Scikit-learn 提供多樣的演算法，其特色包括具有良好的應用程式介面（Application Programming Interface, API）、詳細文檔說明及技術支援 [18]。由於 Scikit-learn 具有輕量而高效的特性，可使本研究實作能夠有效率地達到智慧化水質資料分析的應用需求。

C. 圖表繪製工具

Matplotlib 是一個多功能的 Python 繪圖函式庫，具有產生各種類型和格式圖表的功能，包括折線圖、散布圖、熱圖、直方圖、圓餅圖和立體圖形，同時也支援動畫和互動顯示 [19]。運用 Matplotlib 能夠以視覺化方式呈現多元多項式迴歸運算後的水質資料分析結果，使模型訓練結果呈現更生動以提升可讀性。

D. 應用程式介面

本研究以 FastAPI 作為應用程式界面（API）框架，FastAPI 是一個新興但可靠的 API 框架 [20]，能夠快速且有效地於 Python 建立 RESTful API 伺服器。機器學習模型與使用 FastAPI 服務的前端整合，能夠相較使用如 Flask 等其他 API 框架

的效能提高了 45% [21]。使用 FastAPI 建立 API，有助於前後端之間的通訊與資料傳遞，實現智慧化水質分析與評估系統的系統化架構。

2-1-4. 前端開發工具

在進行前端應用程式開發的過程中，本研究採用 React Native 框架。此框架由 Meta 公司基於 React 框架與 JavaScript 語言設計開發，重點在於能夠開發跨操作系統運行的行動裝置應用程式，作業系統支援包括 Android 及 iOS [22]。使用 React Native 不僅能夠直接進行跨系統平臺開發，還能顯著提升行動裝置應用程式開發的速度。使得本研究在開發具機器學習功能之智慧化水質分析與評估系統時，前端開發的效率明顯優於單獨針對 Android 或 iOS 進行開發的方法。

2. 研究步驟

本研究所設計之多元多項式迴歸模型開發與實驗步驟流程如圖所示，可分為 (2-2-1) 資料取得、(2-2-2) 資料前處理、(2-2-3) 模型選擇與比較、(2-2-4) 超參數調整、(2-2-5) 改良與測試及 (2-2-6) 實現系統化，整體系統化作業流程包含起始於資料取得、模型完成與最終實際應用於水質分析，以實現「基於機器學習實現智慧化水質分析與評估系統」為目標。

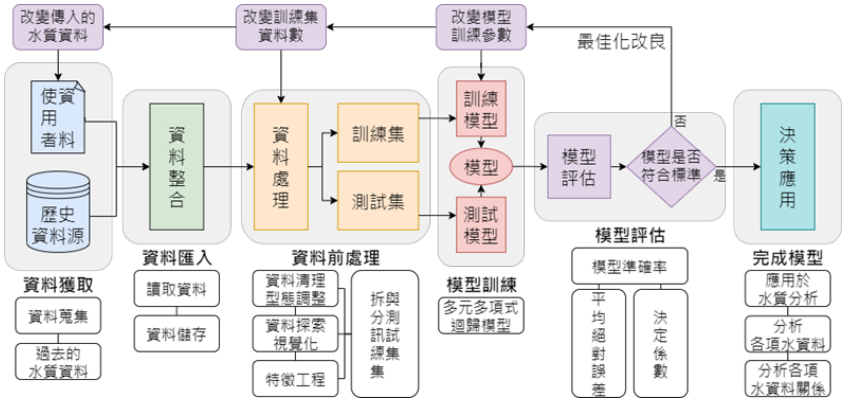

圖 5: 模型建構流程圖

2-2-1. 資料取得

本研究將參考「運用智慧型綠能永續景觀噴泉促進環境保育效益之研究」計畫（編號：MOST110-2410-H025-024）所蒐集的水質資料，並結合中華民國環境部全國環境水質監測資訊網的公開資料，共計蒐集了 87,005 筆水質資料。本研究所採用的水質參數包括溶氧量（DO）、生物需氧量（BOD）、懸浮固體（SS）、氨氮（NH3-N）及導電度（EC）。研究中將使用多元多項式迴歸（MPR）模型對各項水質資料進行機器學習分析。如圖所示，不同資料之間存在交互影響水質健康的關係，例如：DO 濃度越高，NH3-N 及 EC 濃度越低；NH3-N 濃度越高，EC 越高。此外，DO 與水質綜合評分之關係顯示其為重要指標之一，而 BOD、NH3-N、EC 及 SS 與水質綜合評分之關係則顯示其為水質污染的指標。

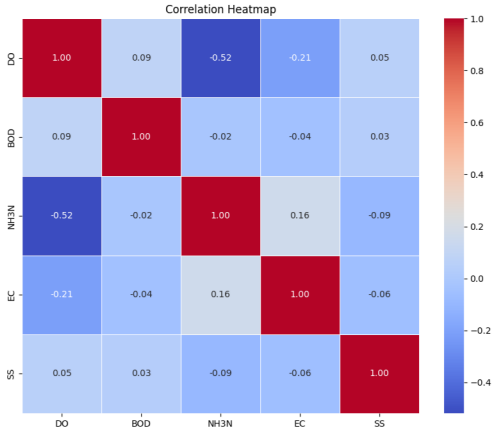

圖 6: 相關性分析矩陣

2-2-2. 資料前處理

在進行機器學習模型訓練前，需對蒐集的原始資料進行前處理，以減少母體資料中的不確定性。此過程可以透過 Pandas 來完成，主要步驟包括：(A) 遺失值處理、(B) 異常值處理、(C) 標準化、(D) 資料拆分及 (E) 特徵工程等。各項操作重點說明如下：

A. 遺失值處理

在機器學習資料前處理階段，需要處理原始水質資料中的遺失值（Missing Value），處理方式通常有兩種：直接刪除或進行填補。雖然直接刪除較為快速，但可能會對模型建構的完整性產生隱患，考慮到模型訓練需要連續分佈的資料，實作上可選擇以填補的方式保持水質資料的完整性，方式包括使用常見值填補、平均值填補、模型預測填補、相似案例填補或使用特殊標記表示遺失值 [23]。本研究將以 NumPy 及 Pandas 對水質資料進行遺失值處理，首先使用 NumPy 的 numpy.nan 表示缺失值，接著以 Pandas 提供的 isnull() 函式尋找資料中是否存在缺失值，最後使用 fillna() 函式進行填補。

B. 異常值處理

除了遺失值的處理外，異常值的檢測與處理也是資料清理中重要的一環。本研究使用修剪法（Trimming）排除機器學習模型資料集中的異常極端值，於整體水質資料集中，刪除中位數 1% 及 99% 之極端值，提升模型的性能 [24]。在經過資料清理排除水質資料集中的異常值後，最後剩餘 60,714 組水質資料作為模型資料集。

C. 標準化

正規化（Normalization）將數值資料縮放到 [0, 1] 的範圍，可能會導致異常值造成的資料遺失，因此使用標準化（Standardization）更佳。標準化將輸入參數的均值（$\mu = 0$）和方差（$\sigma = 1$）進行調整，使資料更符合常態分佈 [25]。本研究使用 Scikit-learn 的 StandardScaler 函式進行標準化，將水質資料集中的數值資料進行標準化處理。

D. 資料拆分

為了有效的建構研究所需的多元多項式迴歸模型，本研究實作上以 Scikit-learn 提供的 `train_test_split()` 函式，將整體水質資料集分為訓練集和測試集，分別占總體資料的 80% 和 20%，訓練集用於模型建構，測試集用於驗證模型 [26]。透過此方法拆分資料，能夠在訓練模型時適度避免過度擬合（Overfltting）現象的發生。

E. 特徵工程

特徵工程（Feature Engineering）在機器學習過程中至關重要，能將資料轉換成模型可以理解的特徵，或將變數處理成特徵 [27]。在水質分析的過程中，實作上將以 Scikit-learn 提供的 PolynomialFeatures 模組將原始資料的特徵轉換為多項式的形式。這個步驟能使模型觀察到各項特徵在非線性關係之間的相互影響，進而提升模型的擬合能力。藉由特徵工程的處理，本研究能夠更清楚地呈現各項水質資料之間的關係，以提升模型判斷的準確性。

2-2-3. 模型選擇與比較

除了提案時使用的多元多項式迴歸（Multiple Polynomial Regression, MPR）模型外，能夠引入 eXtreme Gradient Boosting（XGBoost）及 Light Gradient Boosting Machine（LightGBM）等在結構化資料處理上表現優異的模型進行比較，以能找出更適合解決問題的模型。各模型詳細說明如下：

A. 線性迴歸（Linear Regression, LR）

線性迴歸（LR）是一種簡單但基本的監督式學習模型，透過線性方程式來預測目標變數的值，雖然 LR 與現代統計與機器學習技術方法相比顯得簡單，但仍具價值且為現代模型提供了基礎 [28]。

B. 支援向量機（Support Vector Machine, SVM）

支援向量機（SVM）主要描述支援向量方法的分類 [29]。SVM 模型的目標是找到一個最佳的超平面，使得資料點與超平面之間的邊界最大化，進而達到最佳分類效果 [30]。

C. 隨機森林樹（Random Forest, RF）

隨機森林是一種結合多個樹狀預測器的機器學習模型，每個樹基於獨立抽樣的隨機向量生成，且森林中所有樹的隨機向量具有相同的分佈特性。隨著森林中樹的數量增加，其泛化誤差幾乎確定會收斂至一個有限值，這反映了隨機森林在統計上的穩定性 [31]。

D. 多元多項式迴歸（Multiple Polynomial Regression, MPR）

線性迴歸（Linear Regression, LR）在使用非完全線性分布的資料時可能無法進行分析。相比之下，多元多項式迴歸（Multiple Polynomial Regression, MPR）則適用分析在有多變數之間存在複雜關係的情況。以公式 (2) 為例，其表示一個二階多元多項式 (Sinha, 2013) [32]。在此方程式中，β_1、β_2 代表線性效應參數，β_{11}、β_{22} 是二次效應參數，β_{12} 是交互作用效應參數。

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2 \tag{2}$$

E. eXtreme Gradient Boosting（XGBoost）

XGBoost 在資料科學領域中被廣泛使用，也在 Kaggle 許多機器學習及競賽中取得最佳結果，其特點為有非常強的擴展性（Scalibility）與性能，可以使用相對其他模型更少的系統資源，擴充數十億級別的資料 [9]。

F. Light Gradient Boosting Machine（LightGBM）

LightGBM 是一種高效能的梯度提升決策樹，在傳統 Gradient Boosting Decision Tree（GBDT）演算法上加入使用 Gradient-based One-Side Sampling（GOSS）和 Exclusive Feature Bunding（EFB），並顯示出 LightGBM 相較傳統的 GBDT 加快了 20 倍以上的速度，同時也並未降低準確性 [10]。

2-2-4. 超參數調整

交叉驗證（Cross Validation）能夠用於超參數（Hyperparameter）調整，能夠防止過擬合的發生，並進行模型泛化的評估 [33]。而透過超參數調整，能針對各模型調整參數。

2-2-5. 改良與測試

本研究將以 (A) 決定係數（R^2）、(B) 均方根誤差（RMSE）及 (C) 平均絕對誤差（MAE）對多元多項式迴歸（MPR）模型進行評估測試。R^2 可衡量模型對資料變異性的解釋程度，適合於模型建構階段，用以評估模型對水質資料的擬合程度和解釋能力。RMSE 用以衡量預測值與實際值之間的平均差異，反映模型的精準度和誤差分布。MAE 則著重於分析預測的準確性，適合於模型建構完成後用以評估模型整體擬合的誤差。透過評估測試，能夠判斷所訓練之模型是否能夠有效達成水質分析的需求，三項評估方法重點說明如下：

A. 決定係數 R^2（Coefficient of Determination R-squared）

R^2 可用於評估模型對新觀測值反應分析結果對實際值變異性的解釋程度，R^2 介於 0 至 1 之間，越接近 1 代表 MPR 水質分析模型越準確，R^2 公式如下所示。

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(X_i - Y_i)^2}{\sum_{i=1}^{n}(\overline{Y} - Y_i)^2} \tag{3}$$

B. 均方根誤差（Root-Mean-Square Error, RMSE）

RMSE 是衡量殘差值分散程度的指標，能夠評估資料點在最佳擬合線附近的集中程度，RMSE 衡量預測值與實際值之間的平均差異，並評估 MPR 水質分析模型的精準度。較低的 RMSE 表示模型預測更準確。RMSE 的公式如下所示。

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_i - Y_i)^2} \tag{4}$$

C. 平均絕對誤差（Mean Absolute Error, MAE）

MAE 的計算為目標值與測量值之差的絕對值之和再取平均值，用於判斷 MPR 水質分析模型的準確性，MAE 的數值越低模型所測量出的誤差越小，反之 MAE 的數值越高測量出的誤差越大，MAE 公式如下所示。

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|X_i - Y_i| = \frac{1}{n}\sum_{i=1}^{n}e_i \tag{5}$$

2-2-6. 實現系統化

完成模型建構後，本研究應用訓練完成的模型進行水質資料分析。並進一步系統化機器學習模型，進行以下幾項工作：(A) 序列化模型、(B) 架設應用程式介面（API）、(C) 將模型雲端部署至雲端運算平台及 (D) 完成前端（Front-End）行動裝置應用程式（Mobile Application, APP）的開發。這些步驟旨在實現具機器學習功能之智慧化水質分析與評估系統、多元多項式迴歸模型及前後端程式開發，使其成為一套完整的智慧化水質分析與評估系統。最終成果將包括能夠對水質狀態進行綜合評分，並針對各項水質問題提供改善建議。透過機器學習技術進行水質分析並提供實用建議，使本研究成為解決水質資料分析問題的智慧化解決方案。

A. 模型序列化

模型序列化（Serialization）是模型部署前的第一步，本研究能夠使用 Joblib 將建立好的模型序列化，能夠避免每次使用皆重新訓練的需求。序列化後的模型能夠直接部署於雲端運算平台中，供應用程式介面（API）呼叫使用。

B. 應用程式介面架設

在進行應用程式介面（API）架設時，FastAPI 能夠快速且有效地於 Python 建立 RESTful API 伺服器，本研究將開發完成的水質資料分析模型分別設計需要的 GET 與 POST API，並部署於連接埠 8000。在防火牆中允許該連接埠的通訊後，即可從 API 伺服器外部連線至後端（Back-End）上的 MPR 模型程式進行水質資料分析，同時，也能使用 FastAPI 在 /docs 路徑內建的 Swagger UI 進行 API 測試。

C. 雲端部署

本研究使用 Google Cloud Platform（GCP）基礎架構即服務（IaaS）位於臺灣彰化縣的 asia-east1-b 之 e2-small 等級雲端平臺進行機器學習模型部署，如此可以為所完成的多元多項式迴歸（MPR）水質分析模型設定遠端應用程式介面（API）伺服器，以提供物聯網裝置呼叫 MPR 模型進行分析運算。

D. 前端行動裝置應用程式開發

在本研究中，使用 JavaScript 語言和 React Native 進行前端（Front-End）行動裝置應用程式（Mobile Application, APP）的開發，目的是實現具機器學習功能的智慧化水質分析與評估系統、以及多元多項式迴歸模型的前端部分，如圖 9 所示。React Native 的使用使本研究無需借助 Android 專用的 Android Studio 或蘋果公司的 Xcode 等編輯環境。相反，採用 Expo 和 Visual Studio Code 作為前端應用程式的整合開發環境，這樣可以在開發過程中使用 Expo Go 在 Android 和 iOS 裝置上即時預覽應用程式的效果。

在開發過程中，本研究主要透過 FETCH 呼叫後端（Back-End）上的 MPR 模型程式，將使用者的水質資料 POST 至 API 伺服器，並接收回傳結果以顯示報表資訊，從而為使用者提供改善水質的建議。

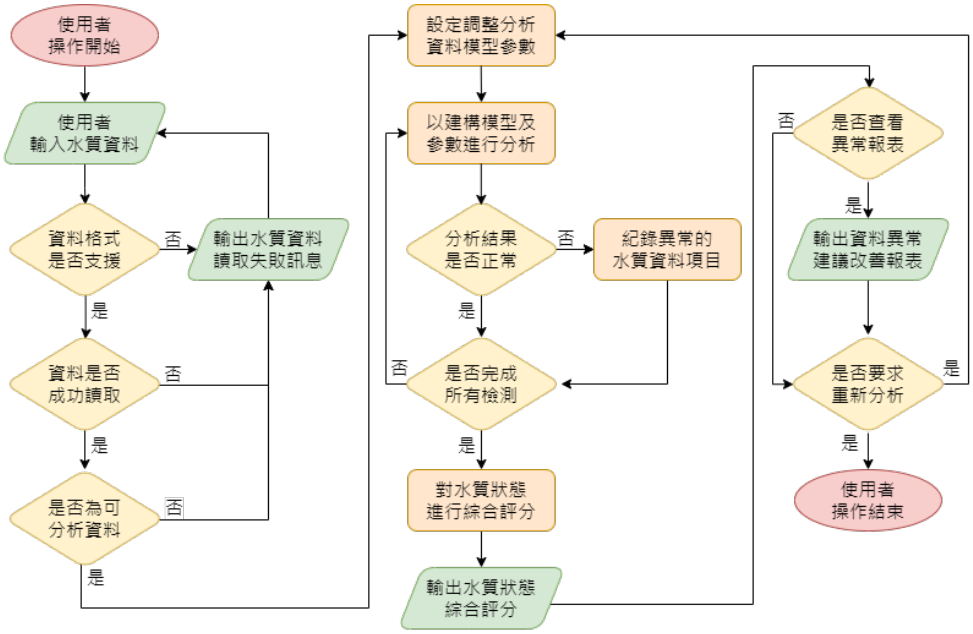

圖 7: 使用者流程圖

3. 執行進度

本研究自 2024 年 7 月 1 日起開始執行，以甘特圖（Gantt Chart）進行嚴謹的規劃和執行，於 2025 年 2 月 28 日結案，共計執行約 8 個月。在 8 個月內完成了實作目標與相關內容，包括水質資料分析模型的比較、選擇與建構，以及智慧化水質分析與評估系統的開發與部署。研究主軸為找出最適合進行水質資料分析的模型並實際應用。各項步驟方向確認至最終完成如下表所示。

4. 遭遇之困難與解決途徑

[內容 TODO]

(三) 完成之工作項目及成果。

1. 完成之工作項目

本研究執行期間完成之工作項目包括文獻回顧與探討、AI 水質分析機器學習模型的訓練、進行實驗以比較不同機器學習模型對水質分析的效能差異、雲端伺服器架設與部署及智慧化水質分析與評估系統 APP 開發..... 等。

2. 完成之研究成果

本研究計畫完成之研究成果大致可分為 (3-2-1) AI 水質分析機器學習模型比較及 (3-2-2) 智慧化水質分析與評估系統兩部分。兩部分詳細說明如下：

3-2-1. AI 水質分析機器學習模型比較

為能有效進行水質資料的分析與評估，研究重點為設計一個能達成計畫目標之機器學習模型。模型選擇如「(二) 研究步驟」中「(2-2-3) 模型選擇與比較」之內容所述。評估以線性迴歸（LR）、支援向量機（SVM）、隨機森林樹（RF）、多元多項式迴歸（MPR）、XGBoost 及 LightGBM 進行比較，並選擇最適合進行水質資料分析的模型。最終成果將以模型的決定係數（R^2）、均方根誤差（RMSE）及平均絕對誤差（MAE）等指標進行評估，並選擇最適合進行水質資料分析的模型。

模型開發亦如「(二) 研究步驟」之內容所述，以前期「運用智慧型綠能永續景觀噴泉促進環境保育效益之研究」計畫所蒐集之水質資料結合中華民國環境部全國環境水質監測資訊網的公開資料，共計 87,005 筆水質資料進行運用。並將開發之機器學習模型將部署於雲端平臺，提供大量水質資料處理所需的運算能力。

為能有效評估水質資料，本研究使用水體品質指標（Water Quality Index, WQI_5）WQI 能夠將大量水質資料轉換如等級一樣的數值，比起各種參數的一長算數值更容易理解與運用，使人們更容易理解水質資料呈現的結果 [34]。WQI 的公式如公式 (6) 所示，同時，本研究在訓練過程中依照 WQI_5 版本為各項水質資料項目加上權重如表 2

所示，再依分析結果依表 3 進行評級，使分析結果具有客觀的可信度。

$$WQI_5 = \frac{\sum_{i=1}^{5}(W_i Q_i)^{1.5}}{10} \tag{6}$$

表 1: WQI_5 水質參數項目與權重

WQI_5 水質參數項目	權重
溶氧（DO）	0.31
生化需氧量（BOD5）	0.26
氨氮（NH3-N）	0.19
懸浮固體（SS）	0.17
電導率（EC）	0.07

表 2: WQI_5 水質指數範圍與評級

指數範圍	評級
$WQI_5 > 100$	N/A
$85 < WQI_5 \leq 100$	優良
$70 < WQI_5 \leq 85$	良好
$50 < WQI_5 \leq 70$	中等
$30 < WQI_5 \leq 50$	不良
$15 < WQI_5 \leq 30$	糟糕
$0 < WQI_5 \leq 15$	惡劣

A. AI 水質分析機器學習模型殘差

從各模型殘差分布中，可以明顯看出 XGBoost 模型在擬合度上的優勢。如圖所示，XGBoost 模型的殘差分布相對均勻，峰值皆在 0 附近。曲線斜率平緩，說明 XGBoost 模型的預測值與實際值之間的差異相對較小，具有較好的擬合度。相比之下，LightGBM 模型的殘差分布雖也接近 0，但曲線略微陡峭，顯示其在某些情況下預測誤差稍大。RF 模型的殘差分布峰值同樣在 0 附近，但曲線較為分散，顯示其預測穩定性略遜於 XGBoost。SVM 模型的殘差分布則呈現較大的波動，峰值偏離 0 較遠，顯示其擬合能力相對較差。MPR 和 RL 模型的殘差分布更為分散，曲線斜率陡峭，顯示其預測誤差較大，擬合度不佳。

綜合以上，XGBoost 模型的擬合度更佳於其他模型，能更好地捕捉水質資料中的非線性關係，從而進行更準確的分析。在本研究開發的智慧化水質分析與評估系統的實際應用中，XGBoost 模型具有捕捉水質資料中非線性關係、擬合度較好以及準確性高的優點，因此在解決非線性迴歸問題、分析具有非線性關係的資料及建立更複雜的模型以進行描述方面，具有更廣泛的應用。由此可見就殘差分布而言，選擇 XGBoost 模型在應對如水質資料等具有非線性關係的資料時，相較於其餘模型而言，是更為理想的選擇。

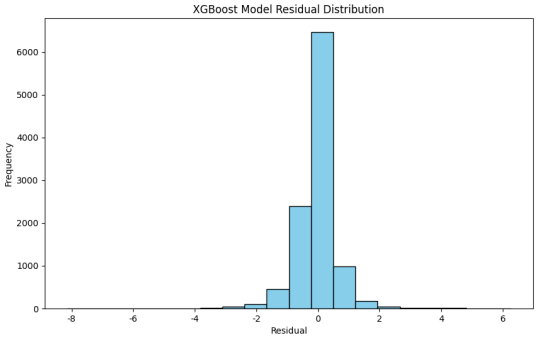

圖 8: XGBoost 殘差圖

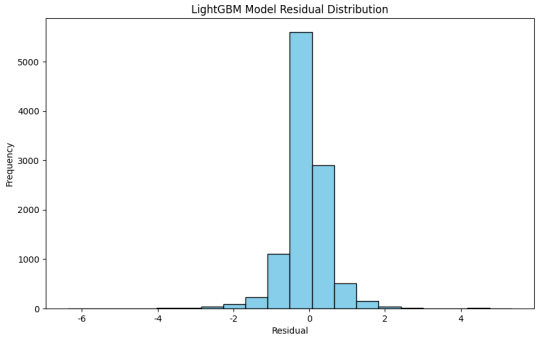

圖 9: LightGBM 殘差圖

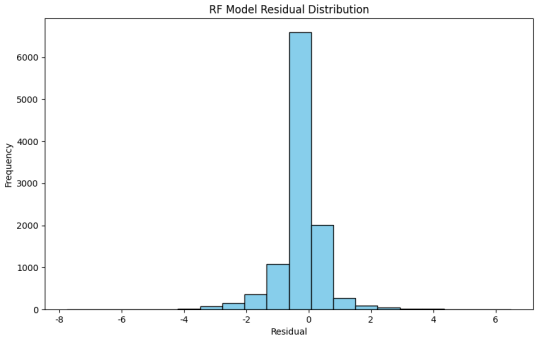

圖 10: RF 殘差圖

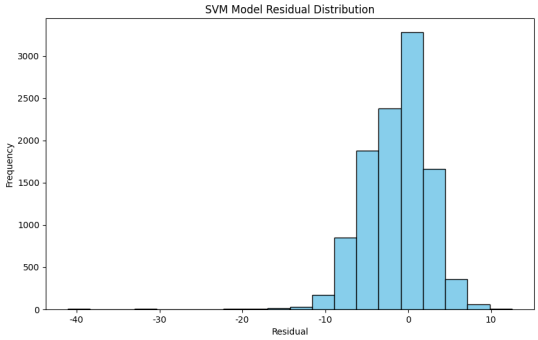

圖 11: SVM 殘差圖

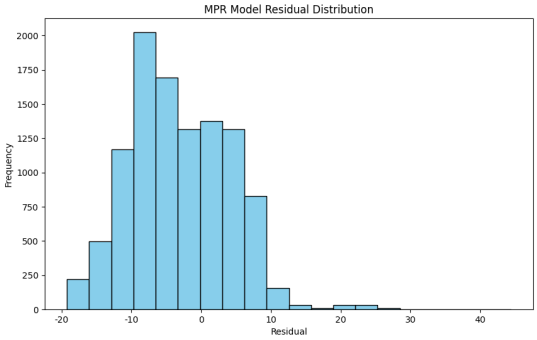

圖 12: MPR 殘差圖

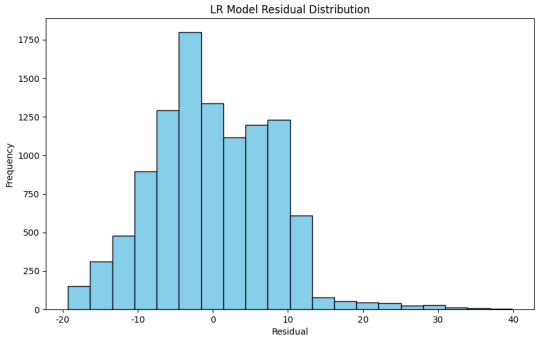

圖 13: LR 殘差圖

B. AI 水質分析機器學習模型指標

在本研究中，我們對多種機器學習模型進行了訓練與評估，包括 XGBoost、LightGBM、隨機森林（RF）、支持向量機（SVM）、多元多項式迴歸（MPR）及線性迴歸（LR），以探討其在水質分析中的表現。為全面評估各模型在不同資料量下的相對優劣，我們使用了多種訓練集大小進行實驗：100 筆、1000 筆、5000 筆、10000 筆、20000 筆及 50000 筆資料，並最終在剩餘的 10714 筆資料上進行測試。評估指標包括決定係數（R^2）、均方根誤差（RMSE）及平均絕對誤差（MAE），並對訓練集與測試集的結果進行了分析。以下將分段呈現各模型在不同資料量下的表現，並在最後總結出相對而言最適合水質資料分析的模型。

在小資料量（100 筆樣本）下，各模型的表現如表 4 所示。相較之下，樹模型（XGBoost、LightGBM、RF）的表現顯著優於其他模型，其中隨機森林（RF）在測試集上的 R^2 達到 0.9377，MAE 為 3.0579，RMSE 為 4.2815，顯示出最佳的預測能力。XGBoost 和 LightGBM 的測試集 R^2 分別為 0.9328 和 0.9223，誤差指標略高於 RF，但仍明顯優於非樹模型。相比之下，支持向量機（SVM）、多元多項式迴歸（MPR）和線性迴歸（LR）的表現較差，尤其是 MPR 的測試集 R^2 為 -164.1864，MAE 為 8.6939，RMSE 為 19.0324，顯示其完全不適合小規模水質資料分析。LR 的測試集 R^2 為 0.6704，雖優於 MPR 和 SVM，但與樹模型相比仍有

較大差距。

表 3: 小資料量（100 筆樣本）下各模型的表現

模型	訓練集 R^2	測試集 R^2	訓練集 MAE	測試集 MAE	訓練集 RMSE	測試集 RMSE
XGBoost	1.0000	0.9328	0.0005	3.1106	0.0007	4.4192
LightGBM	0.9809	0.9223	1.5883	3.4851	2.3875	4.7502
RF	0.9924	0.9377	1.0995	3.0579	1.5840	4.2815
SVM	0.6035	0.5494	8.8048	9.5318	11.7208	12.3027
MPR	0.8717	-164.1864	5.2704	8.6939	6.4709	19.0324
LR	0.8387	0.6704	5.9214	6.8328	7.3687	9.2888

當資料量增加至 1000 筆（中資料量）時，各模型的表現如表 5 所示。相對而言，XGBoost 的測試集 R^2 達到 0.9883，MAE 為 1.3775，RMSE 為 2.0547，略優於 LightGBM（測試集 $R^2 = 0.9917$，MAE = 1.0994，RMSE = 1.7218）和 RF（測試集 $R^2 = 0.9862$，MAE = 1.5021，RMSE = 2.2285）。與此相比，SVM 的測試集 R^2 提升至 0.8789，但誤差指標（MAE = 4.7232，RMSE = 6.6197）仍高於樹模型。MPR 和 LR 的測試集 R^2 分別為 -14.1201 和 0.7580，誤差指標較高（MAE 分別為 7.0965 和 7.0607，RMSE 分別為 18.4625 和 9.3533），顯示其在中等資料量下仍無法有效擬合水質資料的複雜關係。

表 4: 中資料量（1000 筆樣本）下各模型的表現

模型	訓練集 R^2	測試集 R^2	訓練集 MAE	測試集 MAE	訓練集 RMSE	測試集 RMSE
XGBoost	1.0000	0.9883	0.0170	1.3775	0.0243	2.0547
LightGBM	0.9991	0.9917	0.3333	1.0994	0.5557	1.7218
RF	0.9981	0.9862	0.5606	1.5021	0.8473	2.2285
SVM	0.8904	0.8789	4.5071	4.7232	6.3828	6.6197
MPR	0.8444	-14.1201	6.1403	7.0965	7.5740	18.4625
LR	0.7935	0.7580	6.8997	7.0607	8.8015	9.3533

在訓練資料量為 5000 筆時，各模型的表現如表 6 所示。LightGBM 的測試集 R^2 達到 0.9974，MAE 為 0.6258，RMSE 為 0.9886，優於 XGBoost（測試集 $R^2 = 0.9963$，MAE = 0.7599，RMSE = 1.1921）和 RF（測試集 $R^2 = 0.9944$，MAE = 0.9241，RMSE = 1.4664）。SVM 的測試集 R^2 為 0.9424，MAE 為 3.2896，RMSE 為 4.6757；MPR 和 LR 的測試集 R^2 分別為 0.8218 和 0.7806，誤差指標較高（MAE 分別為 6.5485 和 7.1890，RMSE 分別為 8.2400 和 9.2001）。

表 5: 中資料量（5000 筆樣本）下各模型的表現

模型	訓練集 R^2	測試集 R^2	訓練集 MAE	測試集 MAE	訓練集 RMSE	測試集 RMSE
XGBoost	0.9999	0.9963	0.1558	0.7599	0.2193	1.1921
LightGBM	0.9994	0.9974	0.3459	0.6258	0.4934	0.9886
RF	0.9992	0.9944	0.3466	0.9241	0.5607	1.4664
SVM	0.9466	0.9424	3.1858	3.2896	4.5160	4.6757
MPR	0.8304	0.8218	6.4774	6.5485	8.0557	8.2400
LR	0.7802	0.7806	7.1636	7.1890	9.2141	9.2001

在 10000 筆資料下，各模型的表現如表 7 所示。LightGBM 的測試集 R^2 為 0.9985，MAE 為 0.4873，RMSE 為 0.7748，略優於 XGBoost（$R^2 = 0.9978$，MAE = 0.5927，RMSE = 0.9371）和 RF（$R^2 = 0.9965$，MAE = 0.7070，RMSE = 1.1795）。

SVM 的測試集 R^2 為 0.9585，MAE 為 2.8561，RMSE 為 4.0287；MPR 和 LR 的測試集 R^2 分別為 0.8225 和 0.7747，MAE 分別為 6.6178 和 7.2827，RMSE 分別為 8.3381 和 9.3961。

表 6: 中資料量（10000 筆樣本）下各模型的表現

模型	訓練集 R^2	測試集 R^2	訓練集 MAE	測試集 MAE	訓練集 RMSE	測試集 RMSE
XGBoost	0.9997	0.9978	0.2462	0.5927	0.3527	0.9371
LightGBM	0.9994	0.9985	0.3528	0.4873	0.5070	0.7748
RF	0.9995	0.9965	0.2729	0.7070	0.4593	1.1795
SVM	0.9603	0.9585	2.8249	2.8561	3.9624	4.0287
MPR	0.8297	0.8225	6.5971	6.6178	8.2159	8.3381
LR	0.7777	0.7747	7.2795	7.2827	9.3904	9.3961

在 20000 筆資料下，各模型的表現如表 8 所示。LightGBM 的測試集 R^2 為 0.9989，MAE 為 0.4302，RMSE 為 0.6599，優於 XGBoost（$R^2 = 0.9985$，MAE = 0.4856，RMSE = 0.7632）和 RF（$R^2 = 0.9978$，MAE = 0.5321，RMSE = 0.9283）。SVM 的測試集 R^2 為 0.9655，MAE 為 2.6660，RMSE 為 3.6989；MPR 和 LR 的測試集 R^2 分別為 0.8267 和 0.7767，MAE 分別為 6.6856 和 7.3141，RMSE 分別為 8.3275 和 9.4486。

表 7: 中資料量（20000 筆樣本）下各模型的表現

模型	訓練集 R^2	測試集 R^2	訓練集 MAE	測試集 MAE	訓練集 RMSE	測試集 RMSE
XGBoost	0.9995	0.9985	0.3001	0.4856	0.4408	0.7632
LightGBM	0.9993	0.9989	0.3538	0.4302	0.5124	0.6599
RF	0.9997	0.9978	0.2069	0.5321	0.3592	0.9283
SVM	0.9673	0.9655	2.6020	2.6660	3.5852	3.6989
MPR	0.8223	0.8267	6.7467	6.6856	8.3976	8.3275
LR	0.7732	0.7767	7.3626	7.3141	9.4847	9.4486

在大資料量（50000 筆樣本）下，各模型的表現如表 9 所示。LightGBM 的測試集 R^2 達到 0.9991，MAE 為 0.4009，RMSE 為 0.6174，略優於 XGBoost（測試集 $R^2 = 0.9990$，MAE = 0.4173，RMSE = 0.6492）和 RF（測試集 $R^2 = 0.9987$，MAE = 0.4044，RMSE = 0.7367）。SVM 的測試集 R^2 為 0.9660，MAE 為 2.7122，RMSE 為 3.7104；MPR 和 LR 的測試集 R^2 僅分別為 0.7943 和 0.7519，誤差指標高達 MAE = 7.3408 和 7.8900，RMSE = 9.1307 和 10.0296，顯示其在大資料量下相對不具競爭力。

表 8: 大資料量（50000 筆樣本）下各模型的表現

模型	訓練集 R^2	測試集 R^2	測試集 MAE	測試集 RMSE
XGBoost	0.9994	0.9990	0.4173	0.6492
LightGBM	0.9993	0.9991	0.4009	0.6174
RF	0.9998	0.9987	0.4044	0.7367
SVM	0.9671	0.9660	2.7122	3.7104
MPR	0.7952	0.7943	7.3408	9.1307
LR	0.7570	0.7519	7.8900	10.0296

在 10714 筆驗證資料上，各模型的具體表現如表 10 所示。LightGBM 的 R^2 達到 0.9983，MAE 為 0.3836，RMSE 為 0.6152，優於 XGBoost（$R^2 = 0.9982$，

MAE = 0.3940，RMSE = 0.6387）和 RF（$R^2 = 0.9976$，MAE = 0.4160，RMSE = 0.7462）。SVM 的 R^2 為 0.9273，誤差指標顯著高於樹模型（MAE = 3.0782，RMSE = 4.0677）。MPR 和 LR 的表現最差，R^2 分別為 0.7369 和 0.7169，MAE 和 RMSE 均遠高於其他模型。

表 9: 驗證集（10714 筆資料）下各模型的表現

模型	R^2	MAE	RMSE
XGBoost	0.9982	0.3940	0.6387
LightGBM	0.9983	0.3836	0.6152
RF	0.9976	0.4160	0.7462
SVM	0.9273	3.0782	4.0677
MPR	0.7369	6.4667	7.7388
LR	0.7169	6.4676	8.0281

為簡潔呈現 10714 筆驗證資料的整體結果，我們計算了各模型的平均準確率、準確率標準差及總運算時間，如表 11 所示。LightGBM 的平均準確率最高，為 99.0868%，標準差為 1.3615%，顯示其預測精度高且穩定；總運算時間為 0.0943 秒，效率優異。XGBoost 的平均準確率為 99.0826%，標準差為 1.3181%，總運算時間僅 0.0100 秒，表現與 LightGBM 相近但速度更快。RF 的平均準確率為 98.9237%，標準差為 1.9407%，總運算時間為 0.1731 秒，略遜於前二者。SVM 的平均準確率為 92.9570%，標準差高達 10.5956%，總運算時間長達 41.8753 秒，顯示其穩定性和效率不足。MPR 和 LR 的平均準確率分別為 84.6088% 和 84.5803%，標準差分別為 16.0906% 和 17.8491%，總運算時間雖短（分別為 0.0599 秒和 0.0020 秒），但準確性與穩定性均不佳。

表 10: 驗證集（10714 筆資料）下各模型的準確率統計

模型	平均準確率 (%)	準確率標準差 (%)	總運算時間 (秒)
XGBoost	99.0826	1.3181	0.0100
LightGBM	99.0868	1.3615	0.0943
RF	98.9237	1.9407	0.1731
SVM	92.9570	10.5956	41.8753
MPR	84.6088	16.0906	0.0599
LR	84.5803	17.8491	0.0020

綜合以上結果，樹模型（XGBoost、LightGBM、RF）在 10714 筆驗證資料上的表現顯著優於 SVM、MPR 和 LR。其中，LightGBM 以最高的平均準確率（99.0868%）、較低的誤差指標（MAE = 0.3836，RMSE = 0.6152）及適中的運算時間（0.0943 秒）脫穎而出，顯示其在水質資料分析中具有最高的準確性和良好的效率，是最理想的選擇。XGBoost 作為次佳選擇，平均準確率（99.0826%）與 LightGBM 接近，且運算速度最快（0.0100 秒），適合需要快速處理的情境。RF 雖然準確性稍低（98.9237%），且運算時間較長（0.1731 秒），但仍優於非樹模型。SVM、MPR 和 LR 在準確性與穩定性上均不具競爭力，尤其 SVM 的運算時間過長（41.8753 秒），MPR 和 LR 的平均準確率低於 85%，不適合水質資料分析。

值得注意的是，在提案與研究前期所使用的 MPR 模型表現不如文獻預期，顯示其適用性受限於資料特性。因此，對於水質資料分析，LightGBM 是最佳模型，XGBoost 為次佳選擇，兩者在準確性、穩定性和效率上均能滿足需求。

3-2-2. 智慧化水質分析與評估系統

3. 學術研究、國家發展及其他應用方面預期之貢獻

本計畫旨在提供一個快速且準確，基於機器學習的智慧化水質評估方案，期望開發成果能成為解決水質問題的實用工具。本計畫在研究上需要結合跨領域的專業知識與技術，

因此相關成果將在整理分析後進行論文發表，預計能對資訊工程、人工智慧、水資源管理和環境生態保育等領域產生有意義的價值與貢獻。

3-3-1. 參加 PyCon Taiwan 2024 臺灣 Python 年會 Poster Session

3-3-2. 參加 2024 國際民生電子研討會並獲得論文佳作獎

3-3-3. 前期計畫相關研究成果發表期刊並接受

References

[1] Shao, Z., Zhao, R., Yuan, S., Ding, M., & Wang, Y. (2022). Tracing the evolution of AI in the past decade and forecasting the emerging trends. Expert Systems with Applications, 209, 118221.

[2] Cowls, J. (2021). 'AI for Social Good': Whose Good and Who's Good? Introduction to the Special Issue on Artificial Intelligence for Social Good. Philosophy & Technology, 34(Suppl 1), 1-5.

[3] Aggarwal, K., Mijwil, M. M., Al-Mistarehi, A. H., Alomari, S., Gök, M., Alaabdin, A. M. Z., & Abdulrhman, S. H. (2022). Has the future started? The current growth of artificial intelligence, machine learning, and deep learning. Iraqi Journal for Computer Science and Mathematics, 3(1), 115-123.

[4] Zhou, Y., Wang, X., Li, W., Zhou, S., & Jiang, L. (2023). Water Quality Evaluation and Pollution Source Apportionment of Surface Water in a Major City in Southeast China Using Multi-Statistical Analyses and Machine Learning Models. International Journal of Environmental Research and Public Health, 20(1).

[5] Im, Y., Song, G., Lee, J., & Cho, M. (2022). Deep Learning Methods for Predicting Tap-Water Quality Time Series in South Korea. Water (20734441), 14(22), 3766.

[6] Imran, H., Al-Abdaly, N. M., Shamsa, M. H., Shatnawi, A., Ibrahim, M., & Ostrowski, K. A. (2022). Development of prediction model to predict the compressive strength of eco-friendly concrete using multivariate polynomial regression combined with stepwise method. Materials, 15(1), 317.

[7] Narayan, V., & Daniel, A. K. (2022). Energy Efficient Protocol for Lifetime Prediction of Wireless Sensor Network using Multivariate Polynomial Regression Model.

[8] Zeini, H. A., Lwti, N. K., Imran, H., Henedy, S. N., Bernardo, L. F. A., & Al-Khafaji, Z. (2023). Prediction of the Bearing Capacity of Composite Grounds Made of Geogrid-Reinforced Sand over Encased Stone Columns Floating in Soft Soil Using a White-Box Machine Learning Model. Applied Sciences, 13(8), 513

[9] Chen, T., & Guestrin, C. (2016, August). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785-794). New York, NY, United States.

[10] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. Advances in Neural Information Processing Systems, 30.

[11] McKinney W. (2022). Python for Data Analysis: Vol. Third edition. O' Reilly Media.

[12] Harnowo, A. (2022). Blending a MOOC course into a Business School's Course to Introduce Python for Data Analytics. Business Education Innovation Journal, 14(2), 31-35.

[13] Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... & Oliphant, T. E. (2020). Array programming with NumPy. Nature, 585(7825), 357-362.

[14] Nishino, R. O. Y. U. D., & Loomis, S. H. C. (2017). Cupy: A numpy-compatible library for nvidia gpu calculations. 31st confernce on neural information processing systems, 151(7).

[15] McKinney, W. (2012). Python for data analysis: Data wrangling with Pandas, NumPy, and IPython. " O'Reilly Media, Inc.".

[16] Chen, D. Y. (2017). Pandas for everyone: Python data analysis. Addison-Wesley Professional.

[17] Aurélien Géron. (2022). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 3rd Edition. O' Reilly Media, Inc.

[18] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine Learn-ing in Python, Journal of Machine Learning Re-search, 12.

[19] Hunt, J. (2023). Introduction to Matplotlib. In Advanced Guide to Python 3 Programming, Cham: Springer International Publishing, 121-128.

[20] Bill Lubanovic. (2024). FastAPI. O' Reilly Media.

[21] Bansal, P., & Ouda, A. (2022, July). Study on integration of fastapi and machine learning for continuous authentication of behavioral biometrics. In 2022 International Symposium on Networks, Computers and Communications (ISNCC) (pp. 1-6). IEEE.

[22] Native, R. (2020). React native. línea]. Disponible en: https://reactnative. dev/.[Último acceso: 2 de noviembre 2019].

[23] Kotsiantis, S. B., Kanellopoulos, D., & Pintelas, P. E. (2006). Data preprocessing for supervised leaning. International journal of computer science, 1(2), 111-117.

[24] Burlig, F., Knittel, C., Rapson, D., Reguant, M., & Wolfram, C. (2020). Machine learning from schools about energy efficiency. Journal of the Association of Environmental and Resource Economists, 7(6), 1181-1217.

[25] Sharma, P., & Singh, J. (2018, September). Machine learning based effort estimation using standardization. In 2018 International Conference on Computing, Power and Communication Technologies (GUCON) (pp. 716-720). Greater Noida, NCR New Delhi, India.

[26] Ajona, M., Vasanthi, P., & Vijayan, D. S. (2022). Application of multiple linear and polynomial regression in the sustainable biodegradation process of crude oil. Sustainable Energy Technologies and Assessments, 54, 102797.

[27] Verdonck, T., Baesens, B., Óskarsdóttir, M., & vanden Broucke, S. (2024). Special issue on feature engineering editorial. Machine Learning, 113(7), 3917-3928.

[28] James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). Linear regression. In An introduction to statistical learning: With applications in python (pp. 69-134). Cham: Springer international publishing.

[29] Vapnik, V. (2013). The nature of statistical learning theory. Springer science & business media.

[30] Roy, A., & Chakraborty, S. (2023). Support vector machine in structural reliability analysis: A review. Reliability Engineering & System Safety, 233, 109126.

[31] Breiman, L. (2001). Random forests. Machine learning, 45, 5-32.

[32] Sinha, P. (2013). Multivariate polynomial regression in data mining: methodology, problems and solutions. Int. J. Sci. Eng. Res, 4(12), 962-965.

[33] Berrar, D. (2019) Cross-Validation. In: Ranganathan, S., Gribskov, M., Nakai, K. and Christian Schönbach, C., Eds., Reference Module in Life Sciences Encyclopedia of Bioinformatics and Computational Biology, Vol. 1, Elsevier, Amsterdam, 542-545.

[34] Abdul Hameed M Jawad, A., Bahram K, M., & Abass J, K. (2010). Evaluating raw and treated water quality of Tigris River within Baghdad by index analysis. journal of water resource and protection, 2010.