



HW2：多元線性回歸分析 — AI 協作開發紀錄(整理完版)

1 找資料集

指令：

幫我從 Kaggle 上找 3 個符合條件的資料集（10–20 個特徵，適合做多元線性回歸），並幫我挑一個最適合的，附上連結與理由。

回覆內容：

這是在 Kaggle 上找到的三個適合進行多元線性回歸分析，且特徵數量（欄位）在 10–20 個之間的資料集。

✅ 選項 1：葡萄酒品質 (Wine Quality)

- 簡介：約 1600 筆葡萄牙紅酒的物理化學特性數據 + 一個 0–10 的品質評分
- 特徵數量：11 個自變數 + 1 應變數，共 12 欄位
- 適合性：非常適合。數據乾淨、目標明確、格式統一，是經典教學案例。

🚗 選項 2：汽車價格預測 (Car Price Prediction)

- 簡介：約 200 筆汽車資料，包含多種屬性與價格。
- 特徵數量：20+，可挑選 15–20 個核心特徵。
- 適合性：適合，但需做 One-Hot Encoding 與特徵篩選。

🎓 選項 3：學生表現 (Student Performance)

- 簡介：包含中學生個人、家庭與學校數據，用於預測期末成績 G3。
- 特徵數量：超過 30，可篩出 10–15 個。
- 適合性：主題明確，需適度清理。

★ 最佳推薦：葡萄酒品質 (Wine Quality)

Kaggle 連結：

 [Red Wine Quality Dataset](#)

推薦理由：

1. 特徵數完美落在 10-20 之間
2. 數據乾淨、可直接建模
3. 目標明確 (quality 為連續變數)
4. 經典且容易找到參考資料

2 CRISP-DM — Business & Data Understanding

指令：

幫我針對這個資料集撰寫 CRISP-DM 中的『Business Understanding』與『Data Understanding』部分說明，包括變數解釋與問題定義。

回覆內容：

完整撰寫如下 

葡萄酒品質預測專案：CRISP-DM 流程說明

1. 商業理解 (Business Understanding)

1.1 背景

品質是葡萄酒價值的關鍵。傳統評估仰賴品酒師，主觀且昂貴。若能利用化學特徵建立品質預測模型，將能提高效率與客觀性。

1.2 商業目標

- 主要目標：找出影響品質的關鍵化學因子
- 次要目標：
 - 品質控管：自動偵測低品質批次
 - 定價參考：預測分數可輔助新酒定價
 - 行銷應用：了解高分酒的關鍵特徵

1.3 問題定義

能否利用 11 個物理化學特徵建立一個多元線性回歸模型，以預測葡萄酒品質分數？

1.4 成功標準

- 技術面： $R^2 > 0.35$ 、RMSE 低
- 商業面：模型具可解釋性，能提供可行改善建議

2. 資料理解 (Data Understanding)

2.1 資料來源

來源：Kaggle「Red Wine Quality Dataset」

2.2 資料描述

- 1599 筆樣本
- 12 欄位 (11 特徵 + 1 品質分數)
- 無遺失值，資料乾淨

2.3 變數解釋

變數	說明
fixed acidity	不易揮發的主要酸
volatile acidity	揮發性酸（過高會產生醋味）
citric acid	提升清新感
residual sugar	剩餘糖分，影響甜度
chlorides	鹽分含量
free sulfur dioxide	抗氧化抗菌成分
total sulfur dioxide	SO ₂ 總量
density	與酒精、糖分有關
pH	酸鹼度
sulphates	抗氧化助劑
alcohol	酒精濃度
quality	感官評分（0–10）

3 建立 app.py — 資料準備階段

指令：

幫我新增一個 app.py 檔案，寫出 Data Preparation 的 Python 程式碼（含 Streamlit、pandas、numpy、seaborn、matplotlib、scikit-learn），包含缺失值處理、離群值偵測、特徵標準化與訓練/測試切分。

AI 動作：

- ✓ 建立 app.py
 - ✓ 完成缺失值檢查、離群值處理、特徵標準化與 8:2 資料切分
 - ✓ 使用 Streamlit 視覺化結果
-

4 模型訓練階段

指令：

幫我對 app.py 撰寫多元線性回歸模型的訓練程式碼，並輸出模型係數、截距與回歸方程式。

AI 動作：

- ✓ 新增 `from sklearn.linear_model import LinearRegression`
 - ✓ 在頁面輸出模型係數與截距
 - ✓ 顯示完整回歸方程式
-

5 特徵選擇

指令：

幫我對 app.py 使用 RFE 或 SelectKBest 做特徵選擇，列出被選出的重要特徵與其顯著性分析。

AI 動作：

- ✓ 使用 SelectKBest + f_regression
 - ✓ 顯示每個特徵的 F-score 與 p-value
 - ✓ 加入互動式滑桿選取最佳 k 值
 - ✓ 重新訓練模型並更新結果
-

6 模型評估

指令：

幫我計算模型評估指標（MAE、MSE、RMSE、 R^2 ），並撰寫一段文字說明模型好壞與可能改進方向。

AI 動作：

- ✓ 新增模型評估區塊
 - ✓ 計算並顯示 MAE、MSE、RMSE、 R^2
 - ✓ 撰寫分析說明與改進建議
-

7 模型視覺化

指令：

幫我產生預測 vs 實際值的散佈圖、殘差圖，並加上 95% 信賴區間或預測區間的圖。

AI 動作：

- ✓ 新增預測值 vs 實際值散佈圖（含 45° 參考線）
 - ✓ 新增殘差圖
 - ✓ 加入 $\pm 1.96\sigma$ 信賴區間
-

8 新增 CRISP-DM.txt

指令：

幫我新增 CRISP-DM.txt，並撰寫完整的六大步驟報告草稿。

AI 動作：

- ✓ 生成完整報告（Business → Deployment）
 - ✓ 儲存為 CRISP-DM.txt
-

9 新增 NotebookLM.txt

指令：

幫我新增 NotebookLM.txt，撰寫 100 字以上摘要，整理網路上多元線性回歸的主流方法與優化建議。

AI 動作：

- ✓ 生成 NotebookLM.txt
 - ✓ 內容包含 Ridge、Lasso、ElasticNet、特徵工程與非線性轉換
-

10 撰寫 Gemini 輔助內容章節

指令：

請幫我撰寫報告中『Gemini 輔助內容』章節，包括這次的對話過程摘要、我如何利用 Gemini 完成本次作業，以及與 NotebookLM 摘要的整合建議。

AI 動作：

- ✓ 根據 hw2-log.txt 紀錄撰寫完整章節
 - ✓ 詳述開發過程、策略與 NotebookLM 整合方向
-

11 啟動應用程式

指令：

`streamlit run app.py`

錯誤處理：

`ModuleNotFoundError: No module named 'seaborn'`

- ✓ 解決方案：`pip install seaborn`
-

12 再次執行

指令：

`streamlit run app.py`

- ✓ 執行成功，專案完整！
-

13 生成 README.md

指令：

請幫我生成 README.md

AI 動作：

✓ 建立 README.md，包含：

- 專案概述與功能特色
 - 檔案結構
 - 安裝與使用教學
 - 分析流程
 - 授權與作者資訊
-



備註

請幫我整理成 Notion 格式，這是在紀錄我與 AI 對話的過程，內容請不要做任何更動，格式可以修改。