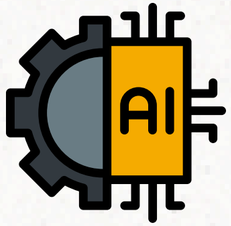
機器學習

作業2
資料迴歸

製作人：B1328023　沈品沅

目錄



機器學習

一、資料介紹

資料集分析

exercise_dataset.csv

從這張圖可以看到：

1.資料非線性

上下起伏、波浪狀的分佈

2.週期性模式

t 可能隨 **x** 呈現某種週期變化

3.雜訊 **(noise)**

點的分佈有一定的隨機性

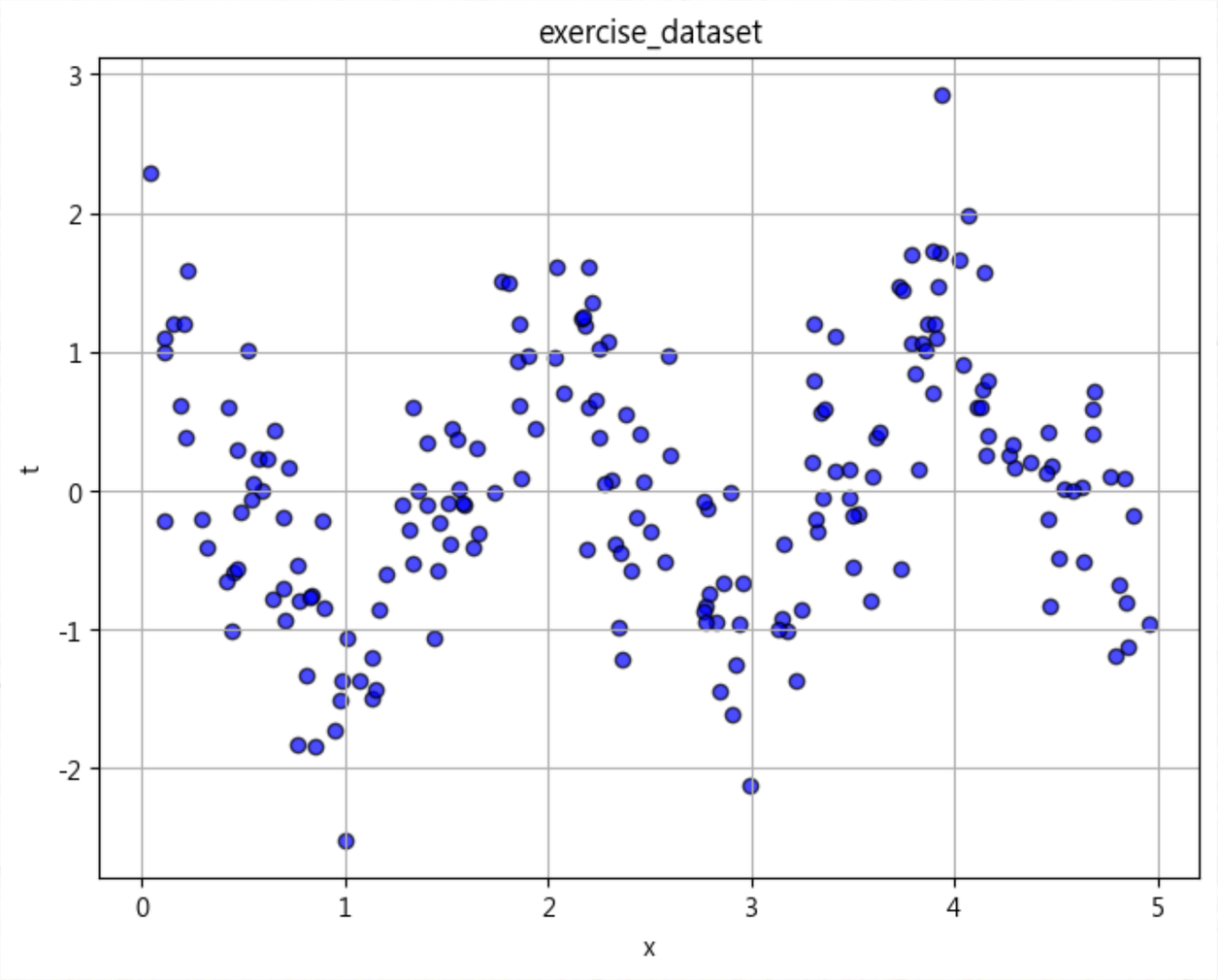
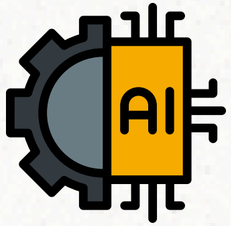


圖 1：資料集散佈圖。藍色點為 200 筆 (x, t) 資料，可以看出 t 與 x 之間並非簡單線性關係，資料呈現多重波動趨勢



機器學習

二、實驗方法

01 資料前處理

在本研究中，資料集包含 200 筆觀測資料，欄位為：

- x：輸入特徵（input feature）
- t：輸出目標（target value）

資料型態均為 float64，沒有缺失值或非數值資料，因此不需額外清理。

由於資料點分佈非線性，推測資料可能來自週期性函數（例如 sin 波），因此後續將比較線性與多項式模型的效果。

```
          x         t
0  3.869780  1.203658
1  2.194392  0.601150
2  4.292990  0.170647
3  3.486840 -0.045119
4  0.470887  0.293785
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   x       200 non-null    float64
 1   t       200 non-null    float64
dtypes: float64(2)
memory usage: 3.3 KB
None
```

無空值

02 訓練與測試資料切分

```python
# 3.2 訓練與測試資料切分（Train-Test Split）
from sklearn.model_selection import train_test_split

X = df[['x']]
y = df['t']
X_train, X_test, y_train, y_test = train_test_split(X, y,
                                                    test_size=0.2,
                                                    random_state=42)
```

為評估模型泛化能力（generalization performance）

將資料隨機切分為：

- 訓練集（Training set）：80%（160 筆）
- 測試集（Test set）：20%（40 筆）

```python
# 3.3 交叉驗證（Cross Validation）

from sklearn.model_selection import cross_val_score
import numpy as np

scores = cross_val_score(model, X, y, cv=5, scoring='r2')
print("平均 R² 分數 =", np.mean(scores))
```

03 交叉驗證

為避免模型表現受單次分割影響，採用 K 折交叉驗證（K-Fold Cross Validation）：

- K = 5
- 每次將資料分為 5 等份，其中 4 份作訓練，1 份作驗證，循環 5 次。
- 評估指標採用 決定係數 R^2。

04 評估指標

指標	說明	sklearn 函式
R^2 score	衡量模型解釋資料變異程度，越接近 1 越好	r2_score
MSE (Mean Squared Error)	測量預測誤差平方平均值，越小越好	mean_squared_error

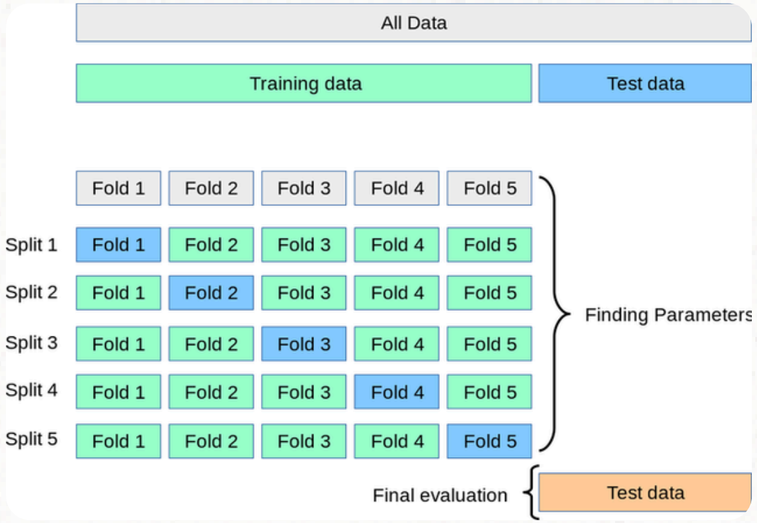
```python
# 3.4 評估指標（Evaluation Metrics）

from sklearn.metrics import mean_squared_error, r2_score

y_pred = model.predict(X_test)
print("MSE =", mean_squared_error(y_test, y_pred))
print("R² =", r2_score(y_test, y_pred))

```

05 模型比較流程

為確保公平比較，各模型皆採相同訓練與驗證流程：

步驟說明：

- 以訓練集訓練模型
- 以測試集評估模型泛化能力
- 使用 5 折交叉驗證取平均 R^2
- 比較各模型的 MSE 與 R^2 分數
- 視覺化預測曲線與原始資料散佈

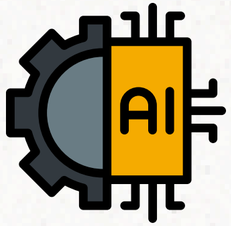


交叉驗證：是一種將資料多次分割成訓練集與測試集、用以評估模型在新資料上表現的穩定性與泛化能力的方法。

06 避免過度擬合

為防止模型學習到資料雜訊而失去泛化能力，採取以下策略：

1. 正規化 (Regularization)
   - 使用 Ridge (L2) 與 Lasso (L1) 模型限制權重大小；
   - 減少模型過度依賴特定資料點。
2. 多項式階數控制
   - 由低階（degree=2）逐步提高；
   - 若測試誤差開始上升，視為過度擬合徵兆。
3. 交叉驗證 (Cross Validation)
   - 用以觀察模型在不同切分下的穩定性；
   - 若訓練表現遠優於驗證表現，即表示模型過擬合。



機器學習

三、模型說明及實作

(1)四模型比較

Linear / Ridge / Lasso /
Polynomial(degree=5)

(2)多階數比較

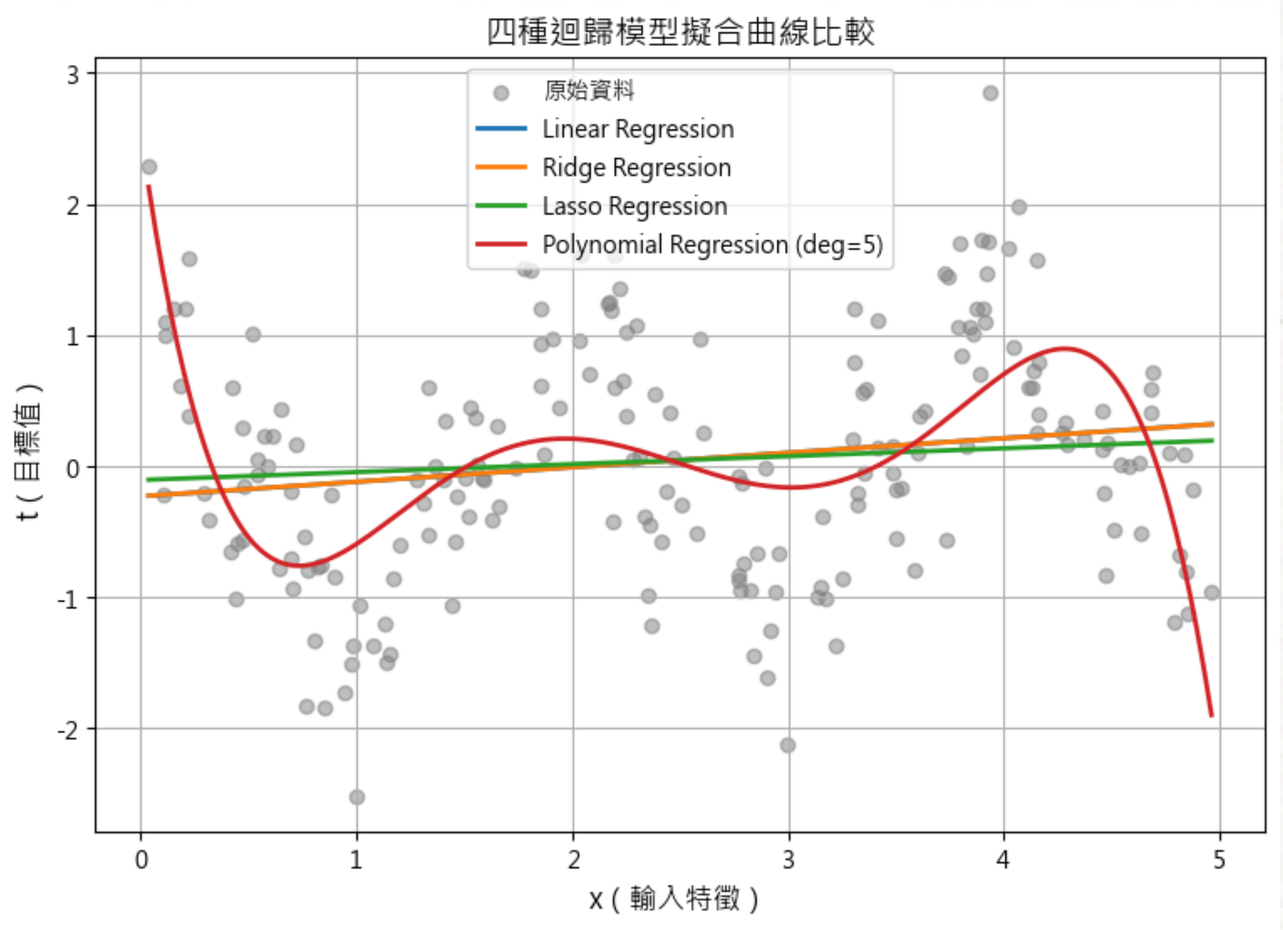
degree = 3, 5, 7, 9, 13, 19

Linear Regression	Ridge Regression	Lasso Regression	Polynomial Regression (degree=3, 5, 9)
$\hat{t} = w_0 + w_1 x$	$\hat{t} = w_0 + w_1 x$ $\text{Loss} = \sum (t_i - \hat{t}_i)^2 + \lambda \sum w_j^2$	$\text{Loss} = \sum (t_i - \hat{t}_i)^2 + \lambda \sum \lvert w_j \rvert$	$\hat{t} = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + \dots$
※ 線性迴歸假設輸入變數 x 與輸出變數 t 之間呈「線性關係」 優點： • 計算簡單、速度快； • 可作為基準模型 缺點： • 無法捕捉非線性關係； • 對異常值敏感	※ 在最小平方法的基礎上，加入 L2 正規化項（懲罰權重過大） 優點： • 減少過度擬合； • 改善模型穩定性 缺點： • 難以自動選特徵； • 仍假設關係是線性的	※ 與 Ridge 類似，但加入的是 L1 正規化項 優點： • 能自動做特徵選擇； • 對高維資料有好處 缺點： • 解可能不唯一； • 適合稀疏特徵，但不適合所有情況	※ 透過建立新的特徵，讓線性模型能擬合非線性曲線 優點： • 能捕捉非線性趨勢； • 延伸性強 缺點： • 高次多項式容易過度擬合； • 需要搭配正規化控制複雜度

(1)
四種主要迴歸模型比較

模型:

- **Linear**
- **Ridge**
- **Lasso**
- **Polynomial(degree=5)**



評估指標

越小越好

MSE（平均平方誤差）

- 定義：預測值與真實值的差距平方平均。
- 數字越小，代表模型預測越精準。

R^2（決定係數）越接近 1 越好

- 定義：模型能解釋多少資料變異。
- R^2 範圍：1 → 完美擬合、0 → 與隨機猜測一樣、負值 → 比平均值預測還差

CV R^2（交叉驗證平均 R^2）

- 定義：在多次不同資料切分（這裡 5 折）下平均 R^2。
- 評估模型的穩定性與泛化能力，避免只看單次測試。

越接近 1 越好

```python
results = []
for name, model in models.items():
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    mse = mean_squared_error(y_test, y_pred)
    r2 = r2_score(y_test, y_pred)
    cv_r2 = cross_val_score(model, X, y, cv=5, scoring='r2').mean()
    results.append([name, mse, r2, cv_r2])

```

計算每個模型的 MSE、R^2、CV R^2

模型效能比較--MSE

📊 四種主要模型比較結果：

	Model	Test MSE	Test R^2	CV R^2
0	Linear Regression	1.068679	-0.130463	0.012431
1	Ridge Regression	1.068541	-0.130318	0.012479
2	Lasso Regression	1.055391	-0.116407	0.009252
3	Polynomial Regression (deg=5)	0.616856	0.347481	0.277461

在MSE的結果裡：

- Polynomial (deg=5) 的 MSE 最小 (0.6168)

→ 表示它在測試資料上誤差最少。

- 其他三個線性模型的 MSE ≈ 1.06

→ 誤差幾乎是 Polynomial 的兩倍。

模型效能比較--R^2

四種主要模型比較結果：

	Model	Test MSE	Test R^2	CV R^2
0	Linear Regression	1.068679	-0.130463	0.012431
1	Ridge Regression	1.068541	-0.130318	0.012479
2	Lasso Regression	1.055391	-0.116407	0.009252
3	Polynomial Regression (deg=5)	0.616856	0.347481	0.277461

在 R^2 的結果中：

- Linear / Ridge / Lasso 的 R^2 都是負值 (-0.13 左右)

→ 代表它們幾乎無法解釋任何變異，甚至比「猜平均」還差。

- Polynomial (deg=5) 的 R^2 為 0.347

→ 表示約能解釋 34.7% 的資料變異。

模型效能比較--CV R^2

四種主要模型比較結果：

	Model	Test MSE	Test R^2	CV R^2
0	Linear Regression	1.068679	-0.130463	0.012431
1	Ridge Regression	1.068541	-0.130318	0.012479
2	Lasso Regression	1.055391	-0.116407	0.009252
3	Polynomial Regression (deg=5)	0.616856	0.347481	0.277461

在 CV R^2 的結果中：

- Polynomial (deg=5) 的 CV R^2 = 0.277 → 雖略低於測試 R^2 (0.347)，但仍表示模型在不同資料切分下穩定。

- 其他三種模型的 CV R^2 約為 0.01，→ 幾乎無法學到任何關係，泛化能力極差。

重點結論

01 MSE（平均平方誤差）

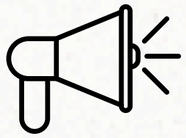
Polynomial 模型明顯能捕捉資料中曲線趨勢，
而線性模型的預測誤差偏大。

02 R^2（決定係數）

只有 **Polynomial** 模型成功捕捉非線性關係；
線性模型幾乎沒有效果。

03 CV R^2（交叉驗證平均 R^2）

Polynomial 模型雖有一點過擬合，但仍能在交叉驗證下維持不錯表現，
是四者中泛化能力最佳的模型。

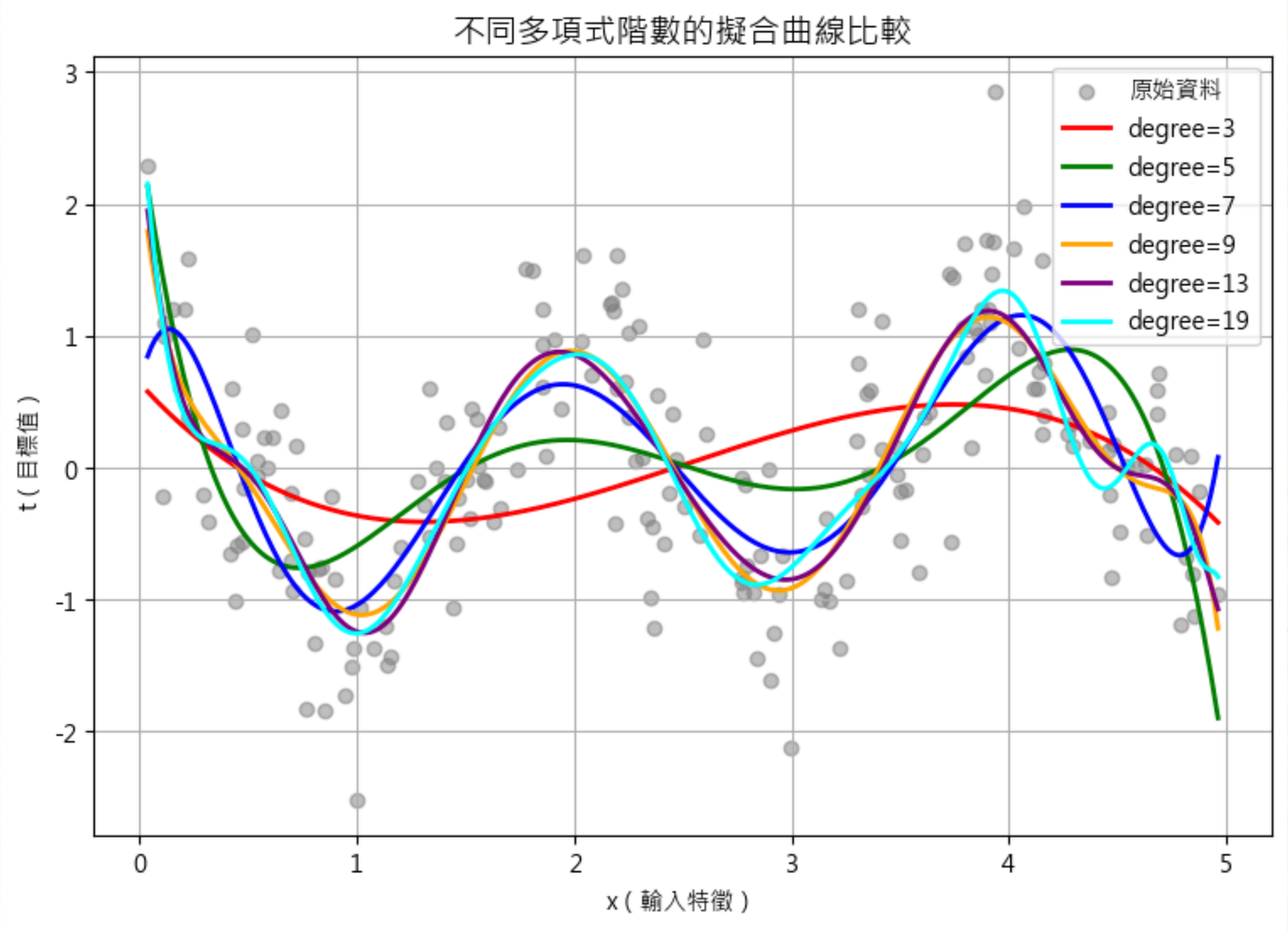


在四種模型中，Polynomial Regression (deg=5) 表現最佳，
其 $R^2 \approx 0.35$，CV $R^2 \approx 0.28$，能有效擬合非線性資料，
而其他線性模型 R^2 皆為負值，顯示無法捕捉資料特徵。

(2)

Polynomial Regression 多階數比較

- **degree = 3, 5, 7, 9, 13, 19**



```python
degrees = [3, 5, 7, 9, 13, 19]
results_poly = []

# 依序測試不同階數的多項式模型
for d in degrees:
    model = make_pipeline(PolynomialFeatures(d), LinearRegression())
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    mse = mean_squared_error(y_test, y_pred)
    r2 = r2_score(y_test, y_pred)
    cv_r2 = cross_val_score(model, X, y, cv=5, scoring='r2').mean()
    results_poly.append([d, mse, r2, cv_r2])

```

- degree=3 → 光滑的曲線（未擬合細節）
- degree=5、7 → 最合理，貼近主要趨勢
- degree=13、19 → 過度扭曲、過擬合雜訊

執行結果

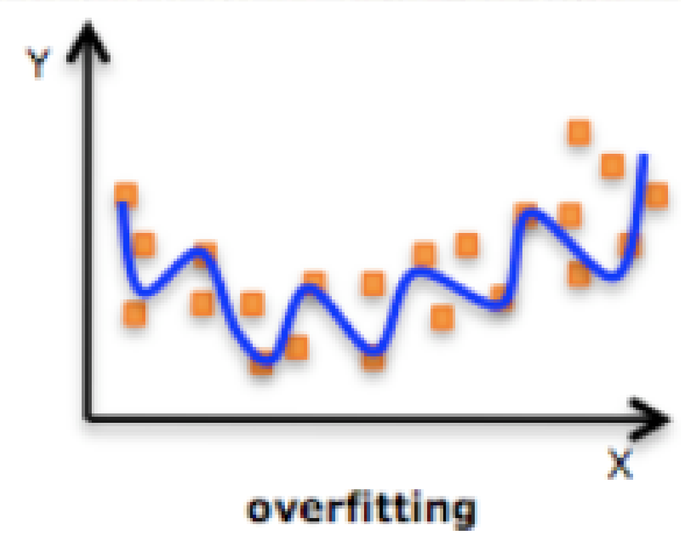
多項式模型不同階數比較結果：

	Degree	Test MSE	Test R²	CV R²
0	3	0.910201	0.037177	0.072177
1	5	0.616856	0.347481	0.277461
2	7	0.415295	0.560695	0.461084
3	9	0.353008	0.626583	0.528400
4	13	0.425718	0.549670	0.479203
5	19	1.416024	-0.497890	0.261102

重點結論

在多項式迴歸模型的階數比較中，隨著 degree 的提升，模型在測試資料上的 R^2 由 0.037 上升至 0.627，顯示模型逐漸能擬合資料中的非線性趨勢。然而，當 degree 超過 9 後，測試 R^2 開始下降，交叉驗證 R^2 也隨之下滑，表示模型開始出現過度擬合 (overfitting)。

綜合 Test R^2 與 CV R^2 的變化，degree = 9 為最佳階數，能在準確度與泛化能力之間取得良好平衡；而 degree ≥ 13 則因模型過度複雜，造成性能退化。



Test R^2 上升後下降 → 擬合曲線現象明顯

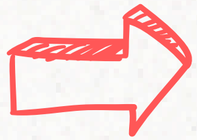
- 從 degree 3 到 9 ：R^2 從 0.04 升至 0.63，模型越來越能擬合資料。
- 超過 degree 9 後：R^2 開始下降甚至變負，代表模型開始學「 雜訊 」。

CV R^2 顯示泛化能力高峰在 **degree 9**

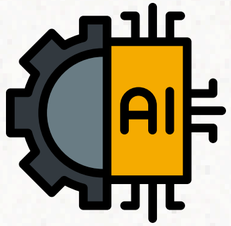
- 交叉驗證 R^2 於 degree 9 達到最高 (0.528)。
- 代表該模型不僅訓練好，也能在新資料上保持穩定。

degree 19 明顯過擬合

- 測試 $R^2 < 0$ 代表模型在新資料上比平均值預測還差。
- 視覺上曲線會劇烈震盪，完全貼合訓練點。



「模型複雜度越高不一定越好」



機器學習

四、結論與發現

一、整體比較結論

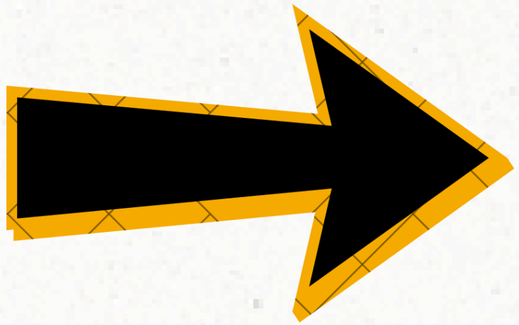
在本研究中，我們以 Linear Regression、Ridge Regression、Lasso Regression 與 Polynomial Regression 四種模型對資料集進行回歸分析。
從結果可觀察到，線性模型（Linear / Ridge / Lasso） 的表現相近，
其 R^2 值皆約為 -0.13，顯示這些模型無法捕捉資料中的非線性關係。
相對地，Polynomial Regression (degree=5) 明顯改善了預測效果，
其測試 R^2 達 0.35、交叉驗證 R^2 約 0.28，
顯示非線性特徵對本資料具有重要影響。

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x^1 + a_0$$

二、多項式階數分析結果

Degree	Test MSE	Test R^2	CV R^2	解釋
3	0.910	0.037	0.072	模型太簡單，僅能捕捉主要趨勢，屬於欠擬合。
5	0.617	0.347	0.277	顯著改善，能擬合主要波形，屬穩定模型。
7	0.415	0.561	0.462	準確度與泛化性兼具。
9	0.353	0.627	0.528	最佳平衡點：精確且穩定。
13	0.426	0.550	0.479	輕度過擬合。
19	1.416	-0.498	0.261	嚴重過擬合。

為進一步探討模型複雜度對性能的影響，
我們測試了多項式模型在不同階數（degree = 3, 5, 7, 9, 13, 19）下的表現。



結果顯示：

- Test R^2 隨階數上升而提升，直到 degree=9 為止達最大值。
- CV R^2 在 degree=9 時也達最高，之後開始下降， 顯示模型開始過度擬合訓練資料。
- 當 degree 過高（如 19），模型對訓練資料幾乎完美貼合， 但在測試資料上表現極差，反映出典型的 overfitting 現象。

三、主要發現

1.非線性模型表現優於線性模型

資料中存在明顯的非線性結構，線性模型無法有效捕捉

2.多項式階數與模型複雜度呈正相關

階數越高，模型越靈活；但靈活度過高將導致過擬合

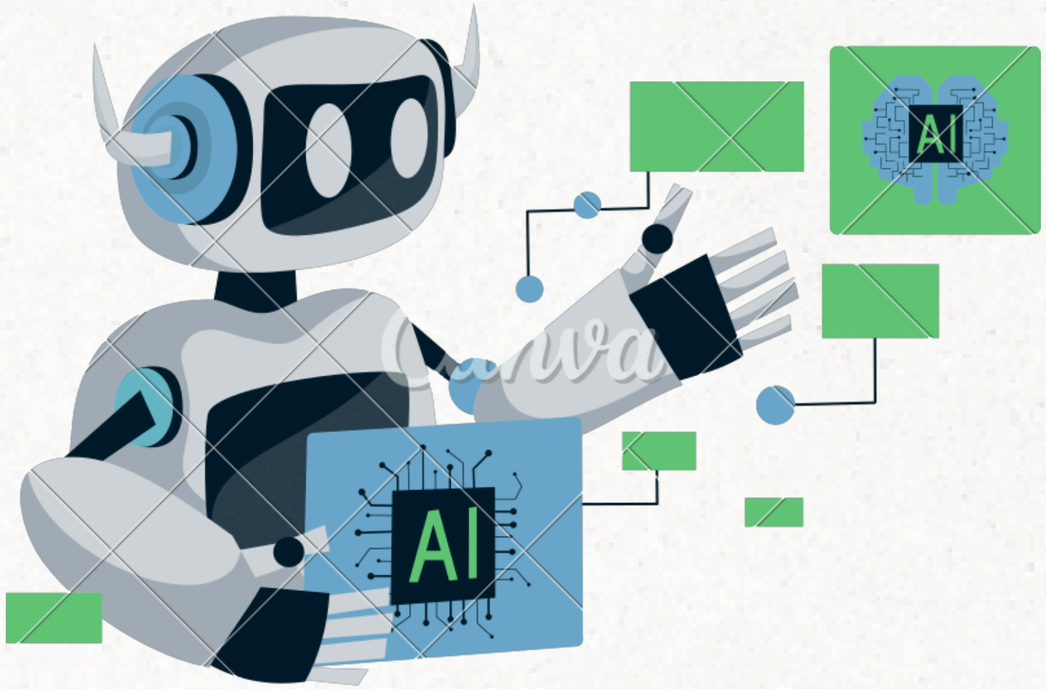
3.最佳階數約為 **degree = 9**

在此階數下，模型在準確度與泛化能力之間取得最優平衡

4.Ridge / Lasso 在單變數資料中影響不顯著

由於只有一個特徵 x，正規化項的效果有限

四、總結



綜合所有結果，本次實作發現：

> Polynomial Regression (degree ≈ 9)
> 是本資料集下表現最佳的模型，
> 其能準確捕捉資料中波浪型的非線性趨勢，
> 並在交叉驗證中展現穩定的泛化能力。

BUT!!!

然而，當多項式階數持續上升（如 degree ≥ 13）時，
模型雖能幾乎完美擬合訓練資料，但在測試集上表現反而下降，
這說明了「擬合得太好反而變壞」的過擬合現象。

參考資料

- 大大通（2020年5月26日）。人工智慧-什麼是擬合過度 (Overfitting)。取自 https://www.wpgdadatong.com/blog/detail/41617(閱覽日期20251016)
- 維基百科（n.d.）。過適 (Overfitting)。取自 https://zh.wikipedia.org/zh-tw/%E9%81%8E%E9%81%A9 (閱覽日期20251016)
- Wizardforcel（n.d.）。十三、交叉验证和得分方法 · SciPyCon 2018 sklearn 教程。取自 https://wizardforcel.gitbooks.io/scipycon-2018-sklearn-tut/content/13.html wizardforcel.gitbooks.io(閱覽日期20251016)
- 楊智淵（n.d.）。機器學習 2025。取自 https://yangchihyuan.github.io/courses/MachineLearning2025(閱覽日期20251016)
- 劉智皓（2021年2月7日）。機器學習_學習筆記系列(13)：交叉驗證(Cross-Validation)和MSE、MAE、R2。取自 https://tomohiroliu22.medium.com/機器學習-學習筆記系列-13-交叉驗證-cross-validation-和mse-mae-r2-bc8fef393f7c (閱覽日期20251016)