



預測性維護資料集 - 空氣壓縮機

組員：張承軒、陳鐸嘉、魏上凱

目錄

- | | |
|--|--|
| <ul style="list-style-type: none">• 空氣壓縮機簡介• 目的、定義問題與目標• 資料說明• 資料前處理• 預計處理變數• 變數分布圖• 迴歸模型分析• 分類模型分析• 工業價值和實用建議 | <ul style="list-style-type: none">• 模型及應用限制• 未來延伸方向 |
|--|--|

空氣壓縮機

- 空氣壓縮機是一種將動力（使用電動馬達、柴油引擎或汽油引擎等）轉化為儲存在壓縮空氣中的位能的裝置。壓縮空氣可以用來驅動氣動工具、充氣輪胎、操作各種其他機械設備。
- 空氣壓縮機主要可分為兩大類：
 - 容積式
 - 活塞式
 - 螺旋式
 - 葉片式
 - 動力式
 - 離心式
 - 軸流式
- 以冷卻方式可分為水冷和氣冷



目的、定義問題與目標

- 本次研究以『水冷往復式空氣壓縮機』為主
- 為提升機械系統的效能與可靠度，分析結構參數與性能之間的關係。
- 運用多種回歸模型和分類模型進行預測，透過視覺化進行結果比較與解釋。

任務目標	<ul style="list-style-type: none">• 判斷零件是否異常（如 bearings, wpump）• 辨識是否需維護清潔（如 radiator, exvalve）• 確認設備是否運作穩定（如 acmotor）
預測類型	<ul style="list-style-type: none">• bearings: "Ok" vs "Noisy"• radiator: "Clean" vs "Dirty"• acmotor: "Stable" （無異常）

數據欄位說明與類型統整

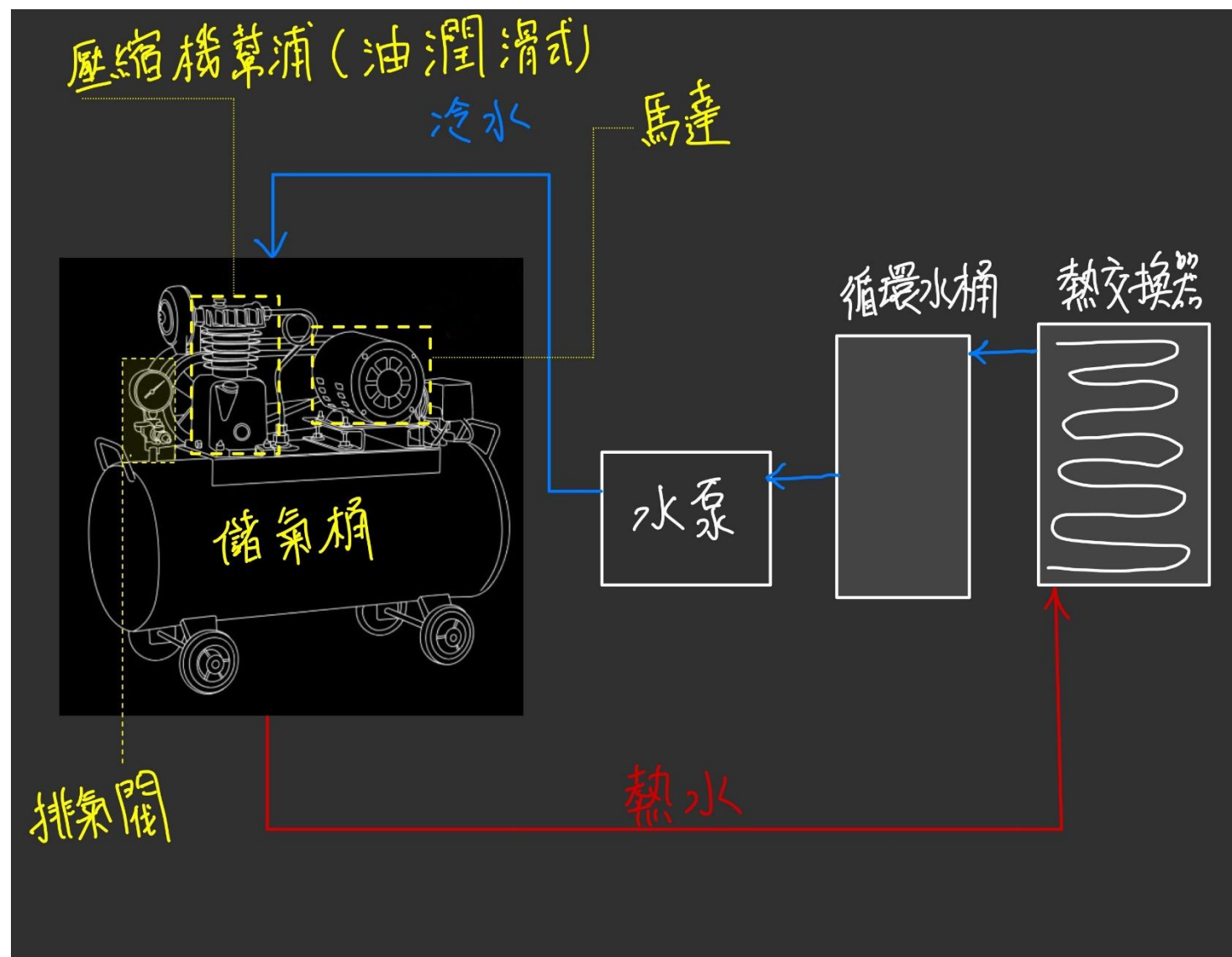
類別	欄位名稱		
識別欄位	id		
數值輸入	rpm(馬達轉速)	motor_power(馬達功率)	torque(扭力)
	outlet_pressure_bar(出口壓力)	air_flow(空氣流量)	noise_db(噪音分貝)
	outlet_temp(出口溫度)	wpump_outlet_press(水泵出口壓力)	water_inlet_temp(冷卻水入口溫度)
	water_outlet_temp (冷卻水出口溫度)	wpump_power(水泵功率)	water_flow(水流量)
	oilpump_power(油泵功率)	oil_tank_temp(油箱溫度)	
	gaccx gaccy gaccz (G 感測器加速度 XYZ 軸, m/s')		
	haccx haccy haccz (H 感測器加速度 XYZ 軸, m/s')		
分類輸出	Bearings(軸承狀態), wpump(水泵狀態), radiator(散熱器狀態), exvalve(排氣閥狀態), acmotor(是否運作穩定)		

數據實際位置及說明

gaccx gaccy gaccz (G 感測器加速度 XYZ 軸, m/s')

haccx haccy haccz (H 感測器加速度 XYZ 軸, m/s')

- **Head Acceleration**是從壓縮機頭部螺栓或上部散熱片測量的值。每個部件都有一個固有頻率，並且由於傳入的驅動而以一定的頻率振盪。這種振動與結構的剛度有關。
- **Ground Acceleration**是從壓縮機連接到剛性部件的位置測量的加速度值。加速度的大小與測量點顯示的位移量有關。



資料前處理

- 移除無意義欄位
(移除 id, acmotor) : acmotor 標籤為單一值 “Stable”，無法進行分類，應視為無效標籤
- 特徵標準化 (Standardization) :
使用 StandardScaler 對數值欄位進行 Z-score 標準化，使平均數為 0，標準差為 1。
- 類別標籤編碼 (Label Encoding) :
使用 LabelEncoder 將 4 個標籤欄位轉為數值類別格式 (例如 OK \rightarrow 0, NG \rightarrow 1)。

5 移除無意義欄位 (視情況而定)

```
if "id" in df.columns:  
    df = df.drop(columns=["id"])  
  
if "acmotor" in df.columns:  
    df = df.drop(columns=["acmotor"])
```

7 特徵標準化

```
scaler = StandardScaler()  
X_scaled = scaler.fit_transform(X)
```

8 類別編碼

```
label_encoders = {}  
y_encoded = pd.DataFrame()  
for col in y.columns:  
    le = LabelEncoder()  
    y_encoded[col] = le.fit_transform(y[col])  
    label_encoders[col] = le # 儲存編碼器以備反解
```

🔥 標籤前五筆 (已編碼) :


	bearings	wpump	radiator	exvalve
0	1	1	0	0
1	1	1	0	0
2	1	1	0	0
3	1	1	0	0
4	1	1	0	0

- 遺失值與重複值處理
- 檢查結果顯示：無明顯遺失值和重複資料。

! 遺失值概況：

id	0
rpm	0
motor_power	0
torque	0
outlet_pressure_bar	0
air_flow	0
noise_db	0
outlet_temp	0
wpump_outlet_press	0
water_inlet_temp	0
water_outlet_temp	0
wpump_power	0
water_flow	0
oilpump_power	0
oil_tank_temp	0

gaccx	0
gaccy	0
gaccz	0
haccx	0
haccy	0
haccz	0
bearings	0
wpump	0
radiator	0
exvalve	0
acmotor	0
dtype: int64	

 重複值筆數： 0

程式碼

```
# 3 檢查重複值與遺失值
print("\n🔍 重複值筆數：", df.duplicated().sum())
print("\n! 遺失值概況：")
print(df.isnull().sum())
```


訓練與測試資料切分

- 使用 `train_test_split` 將資料以 8:2 切分為訓練集與測試集
- 可做為後續機器學習建模之用
- 訓練集：800 筆
- 測試集：200 筆
- 切分比例：80/20

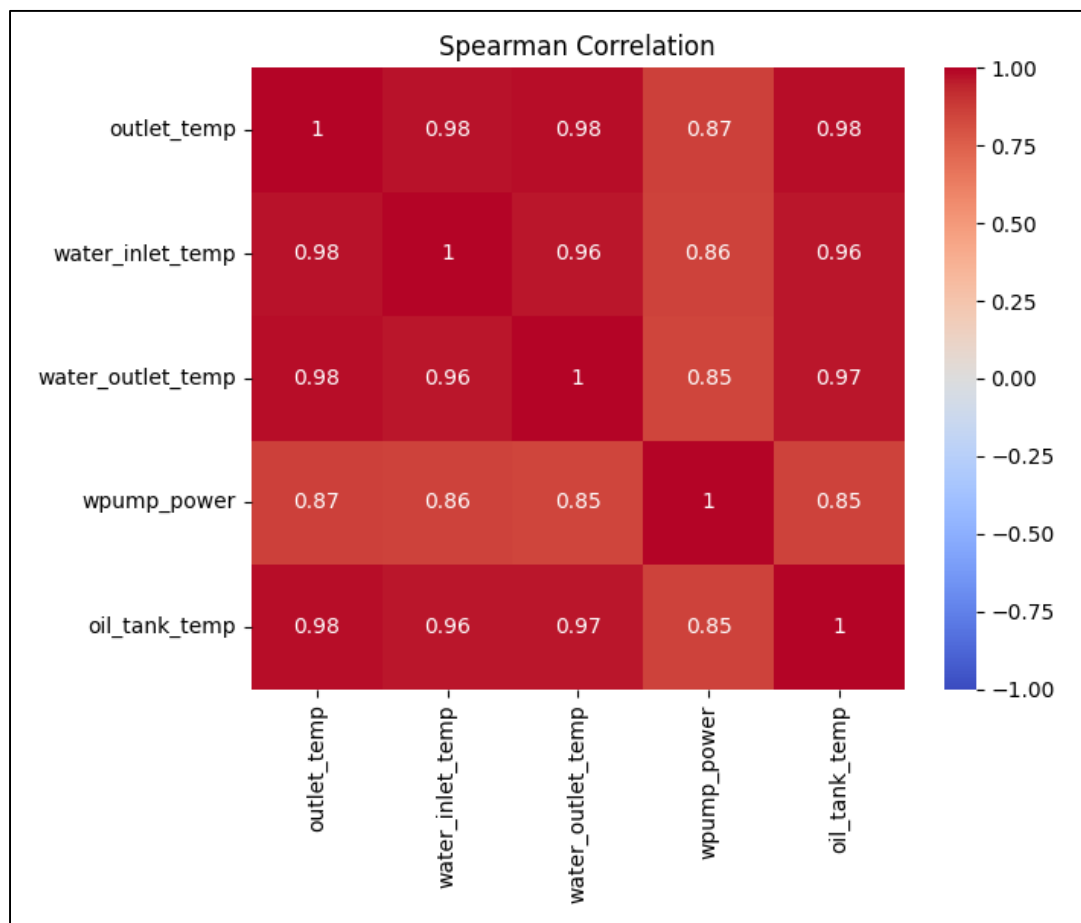
```
# 9 切分訓練集與測試集 (80/20)
X_train, X_test, y_train, y_test = train_test_split(
    *arrays: X_scaled, y_encoded, test_size=0.2, random_state=42
)
```

✅ 資料切分完成

訓練資料維度 (X, y) : (800, 20) (800, 4)

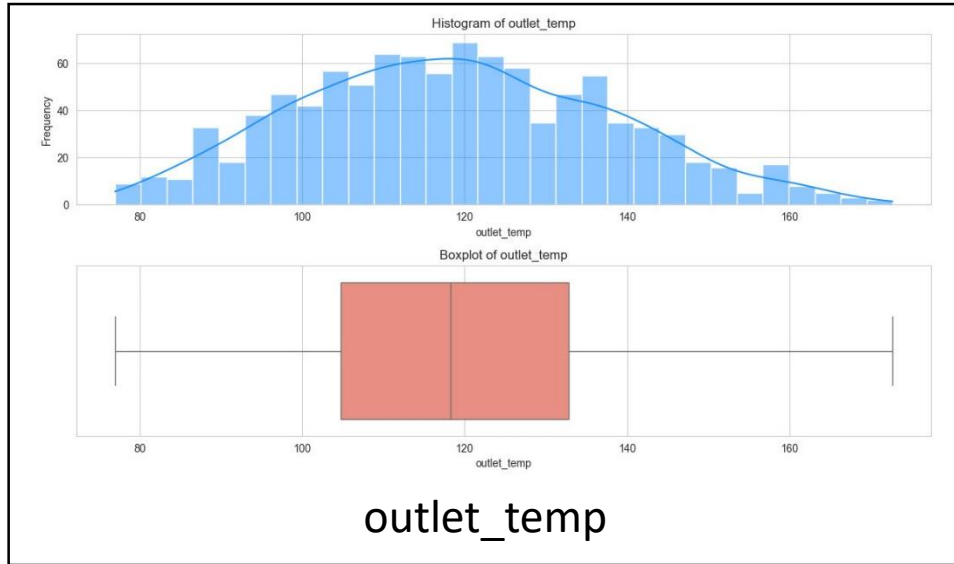
測試資料維度 (X, y) : (200, 20) (200, 4)

預計處理變數



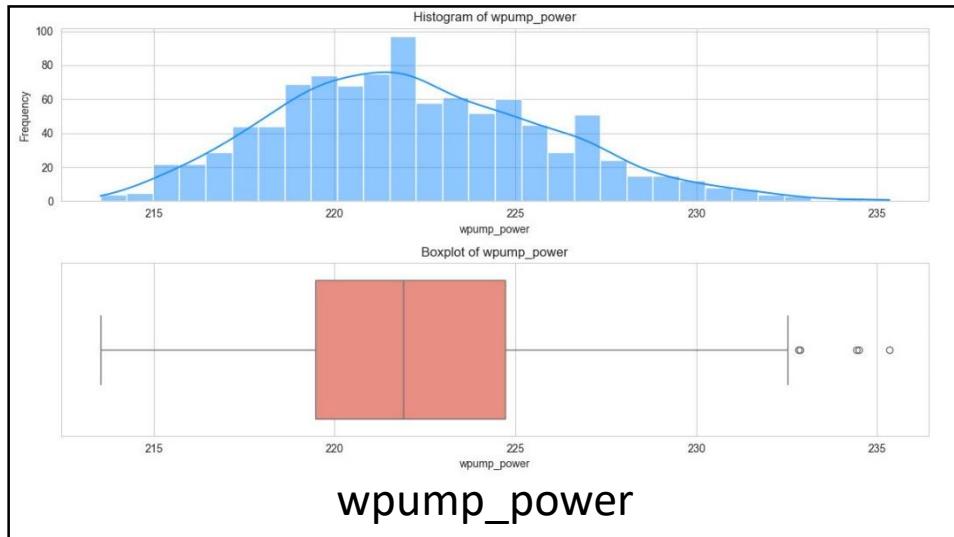
- 變數：
outlet_temp、water_inlet_temp、
water_outlet_temp、wpump_power、
oil_tank_temp
- 原因：
由Spearman熱力圖所示，相關係數值都非常高（多數在 0.85 ~ 0.98），顯示變數之間有非常強的線性相關。

變數分布圖



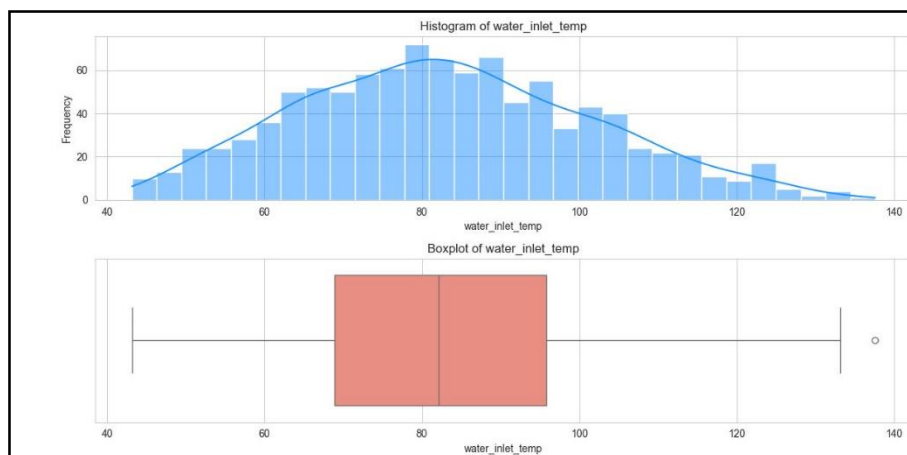
outlet_temp（出口溫度）

- 分布：稍偏右
- 平均值：118.86，標準差：19.12
- 箱型圖：有部分離群值
- 結論：高溫值得注意，但整體變異性在可控範圍

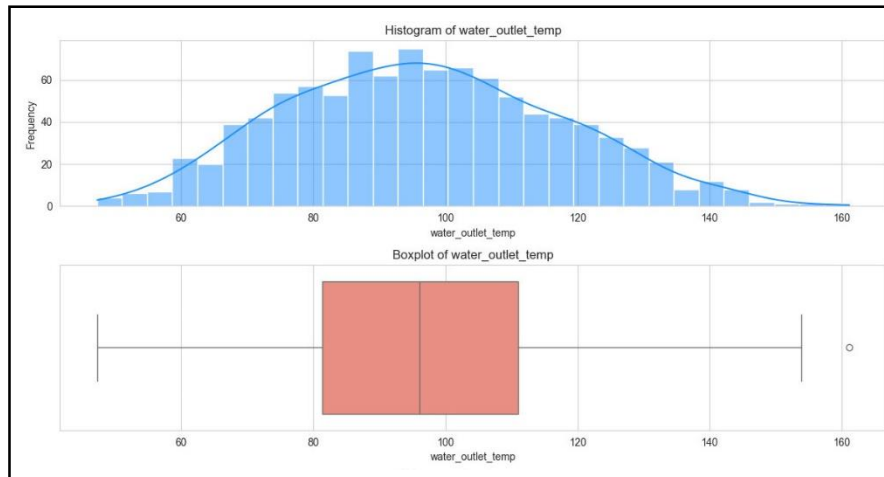


wpump_power（水泵功率）

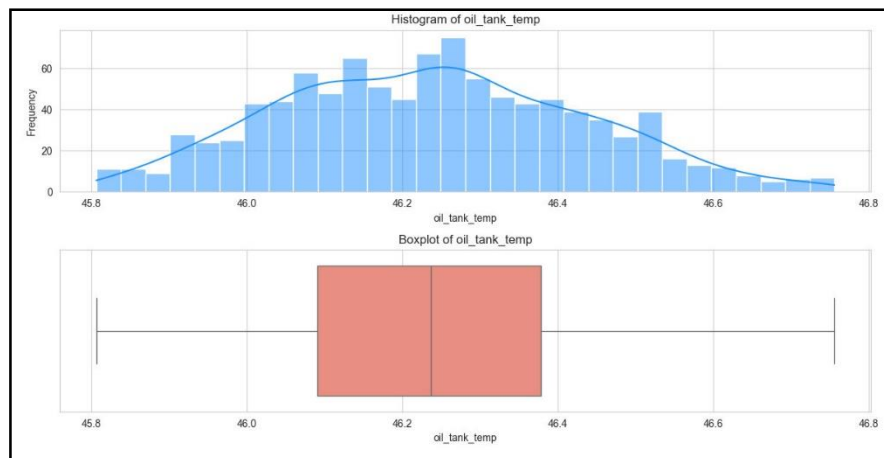
- 分布：集中且偏正態
- 標準差：小（3.77）
- 結論：穩定輸出，無異常點



water_inlet_temp



water_outlet_temp



oil_tank_temp

water_inlet_temp / water_outlet_temp(進水出水溫度)

- 分布：對稱但略偏右
- 平均值：進水83.02，出水 96.64，標準差均在20 左右
- 結論：變異性中等，可能與運作狀態相關

oil_tank_temp（油槽溫度）

- 分布：極度集中 標準差：0.2
- 結論：也可視為常數欄位，變異極小

迴歸模型分析

- 回歸目的：
 - 預測目的
 - 在未知 **outlet_temp** 的狀況下，只要輸入當前的水溫與泵功率，就可快速預測出口油溫，節省實際感測或避免延遲回報。
 - 系統監控與異常預警
 - 若預測值與實際 **outlet_temp** 落差過大，可視為異常現象，早期警示熱交換效率或泵異常。
 - 參數最佳化建議
 - 可反推出哪些參數變動（如提升泵功率或增加水溫差）能有效提升出口油溫，有助於節能與效率優化。

變數簡介

- 使用特徵欄位（X）：
- **water_inlet_temp**（冷卻或加熱水的進水溫度，反映進入熱交換器的水源溫度）
- **water_outlet_temp**（水流離開熱交換器後的溫度，間接表示熱傳效率）
- **wpump_power**（水泵的功率輸出，代表液體循環的能量供應情況）
- **oil_tank_temp**（油槽內部的溫度，與熱能儲存及輸出狀況有關）
- 目標欄位（Y）：**outlet_temp**（最終的加熱油出口溫度，是整個系統效能的重要指標）

使用模型

- **Linear Regression**（線性回歸）：

- 基礎線性模型，假設目標值與特徵之間為線性關係
- 對異常值敏感，無法處理共線性問題

- **Ridge Regression**（嶺回歸）：

- 在線性回歸基礎上加上 **L2** 正則化
- 抑制模型對共線性特徵的過度擬合
- 適合特徵數量較多、可能存在多重共線性的情況

- **Lasso Regression**

- 加入 **L1** 正則化項，可將部分特徵係數收斂為 0
- 具備特徵選擇功能（有助於模型簡化）
- 適合資料稀疏或需進行特徵選擇時使用

- **KNN Regression**（ $K=5$ ）

- 非參數模型，依據距離找出最接近的 K 筆資料（此例為 5 筆）
- 較能擬合非線性資料
- 對資料規模敏感，預測速度慢，對高維資料效果差

- **SVR**（**Support Vector Regression**）

- 可透過 **kernel**（核函數）處理非線性關係
- 適合中小型資料集，有強健的邊界控制能力

模型評估

模型評估結果：

	Model	R2	RMSE	MAE	MAPE (%)
1	Ridge Regression	0.979398	2.700282	2.114425	1.816375
0	Linear Regression	0.979388	2.700932	2.116104	1.817521
2	Lasso Regression	0.979300	2.706733	2.121825	1.822536
4	SVR	0.973830	3.043396	2.345916	1.992057
3	KNN Regression	0.971844	3.156760	2.462632	2.090320

- **R²**：越接近 1 表示模型解釋力越強
- **RMSE**：越小表示預測誤差越小 【單位：和目標變數 y (outlet_temp) 的單位相同】
- **MAE**：平均絕對誤差 【單位：和目標變數 y(outlet_temp) 的單位相同】
- **MAPE**：平均百分比誤差，便於跨尺度比較

可視化分析

- 預測值 vs 真實值 散點圖

Linear/Ridge/Lasso Regression

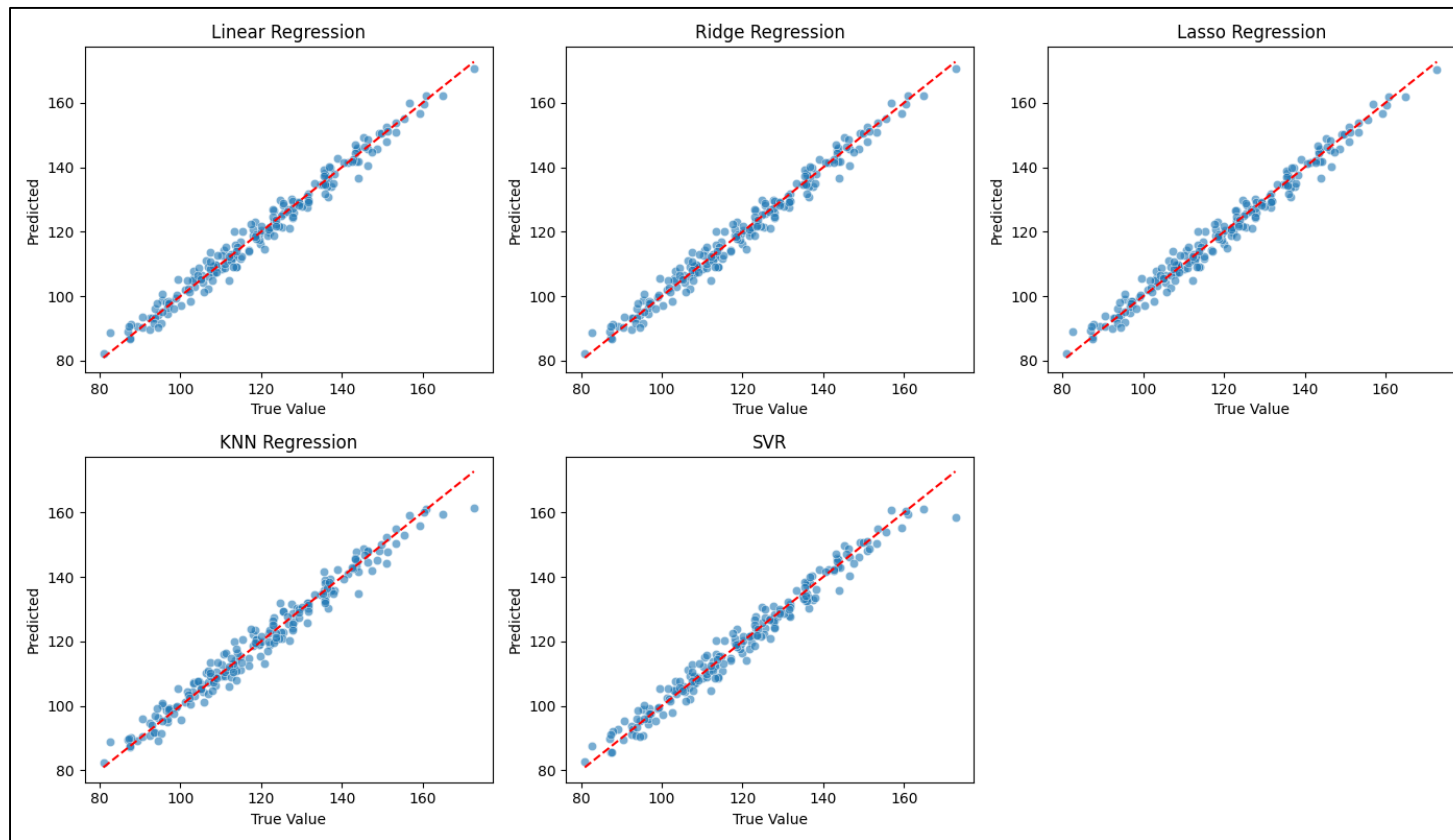
- 預測點與紅線密集貼合，誤差分布均勻。說明線性模型已很好地擬合資料。

KNN Regression

- 表現也很好，但邊緣點較稀疏，代表在極端值可能表現較差（非線性模型常見）

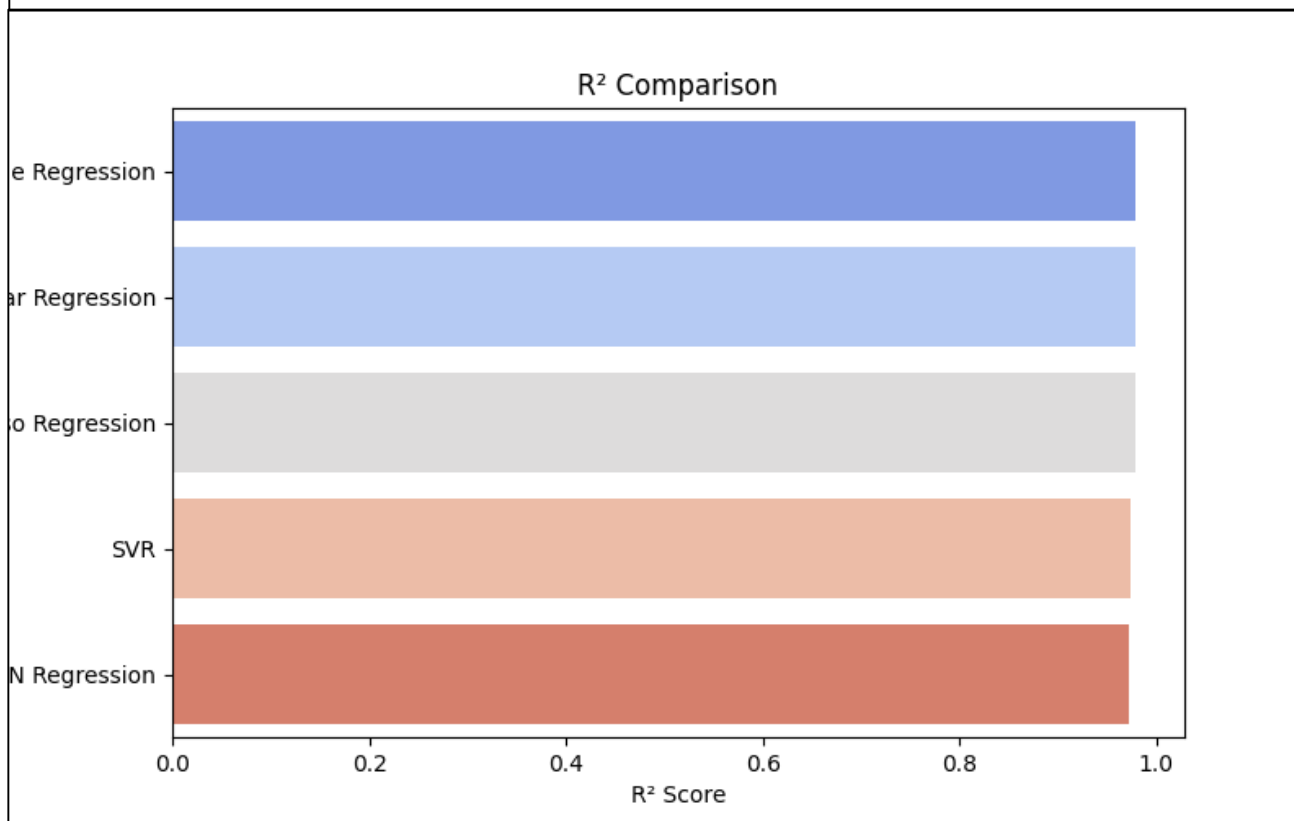
SVR

- 整體準確，但有部分偏離紅線，尤其在高值區域，有些低估情況。表示 SVR 的泛化能力稍遜。

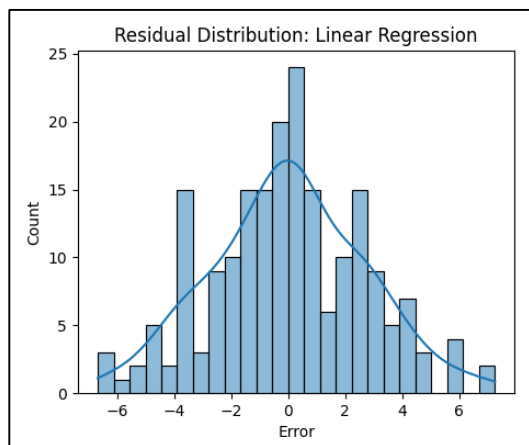


R² 比較圖

- 所有模型的 R² 分數都非常接近 1
- 分數視覺上幾乎一樣長，表示這些模型差異不大。



誤差分布圖（每個子圖顯示了模型的「預測誤差 = 真實值 - 預測值」的分布）



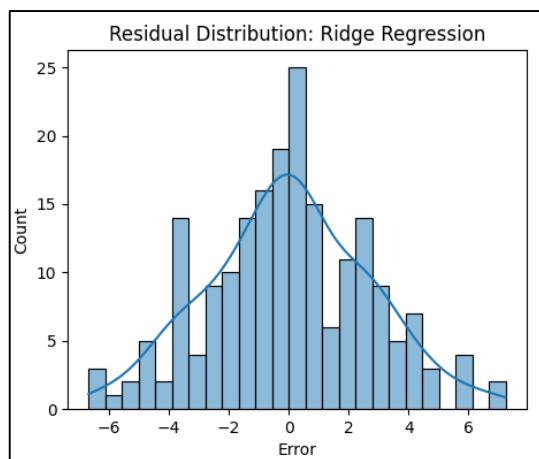
1. Linear Regression

分布形狀：呈現對稱鐘型，類似常態分布

偏態：約 0（近似常態分佈）

極值與尾部：尾部適中，無異常值密集

結論：符合誤差獨立同分布的假設，模型非常穩定且可靠。



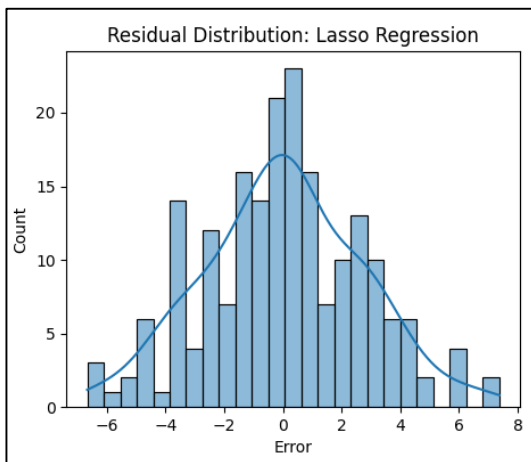
2. Ridge Regression

分布形狀：與 Linear Regression 非常類似

偏態：+0.1 到 +0.3（輕微右偏）

尾部情形：尾部略短，誤差更集中於 0 附近

結論：正則化使模型更穩定，泛化能力略優於 Linear Regression。



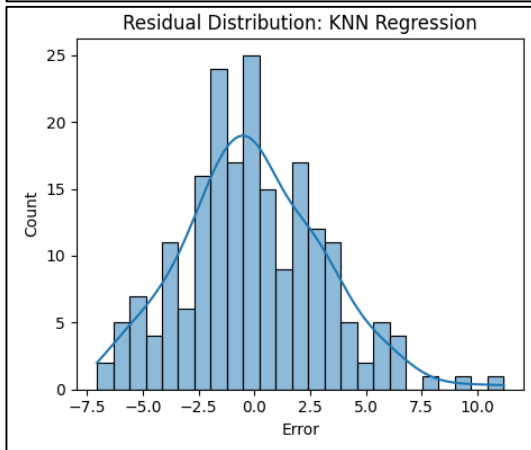
3. Lasso Regression

分布形狀：大致對稱，但相較 Ridge 更扁平

偏態： $+0.3$ 左右，輕微偏右（右側誤差略多）

尾部情形：尾部稍長，有幾個較大的正誤差

結論：稀疏特徵選擇讓模型簡化，略微犧牲穩定性。仍屬穩健模型。



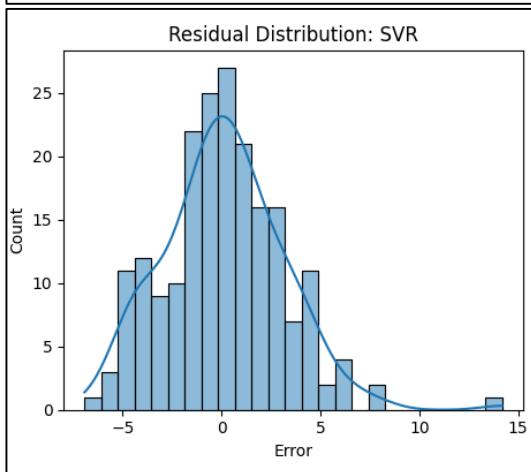
4. KNN Regression

分布形狀：不完全對稱，明顯偏左

偏態： -0.6 到 -0.8 （中度左偏），左偏（負誤差多）

尾部情形：右尾明顯較長（極端值存在）

結論：KNN 對局部樣本敏感，造成部分過擬合或不擬合。誤差集中度差，分布不理想。



5. SVR (Support Vector Regression)

分布形狀：明顯右偏

偏態： $+1.0$ 以上（重度右偏，正誤差多）

尾部情形：右尾長，有明顯異常值（最大誤差接近 $+15$ ）

結論：SVR 在高值預測下表現不佳，泛化性較差。

結論

- 最佳模型：Ridge Regression
 - 其 R^2 值最高（0.9794），同時 RMSE 與 MAE 最低，代表整體預測誤差最小。
 - Ridge 模型適合於特徵多且可能具相關性的資料，可防止過擬合並保持線性可解釋性。
- 其次為 Linear Regression 與 Lasso Regression，兩者與 Ridge 相近，但 Lasso 在某些情況下略受正則化影響造成部分特徵權重減少，影響預測力。
- KNN 與 SVR 模型在此資料表現較差，推測因該資料整體線性關係明確，複雜模型反而增加噪音與運算負擔。

分類模型分析

- 分類目的：
 - 即時預警
 - 提早辨識異常（如高溫），降低設備風險。
 - 優化控制策略
 - 根據預測結果調整冷卻水參數與泵浦功率。
 - 數據驅動維護
 - 協助工程師掌握運行狀況，安排檢修或冷卻優化。

問題設定

- 目標：預測 油箱溫度（oil_tank_temp）的分類（低溫 / 中溫 / 高溫）
- 類別數：3 類（分類標籤為 0、1、2）
- 目標變數（Target）：
 - oil_tank_temp（經過分類處理）
 - 類別 0：低溫
 - 類別 1：中溫
 - 類別 2：高溫
- 特徵變數（Features）：
 - outlet_temp
 - water_inlet_temp
 - water_outlet_temp
 - wpump_power

使用模型

- 羅吉斯迴歸 (Logistic Regression) :
 - 傳統線性分類模型基準模型
 - 適合特徵與目標關係近似線性的情況
- 支持向量機 (Support Vector Machine, SVM) :
 - 可處理非線性問題，有強大的邊界區分能力
- 隨機森林 (Random Forest) :
 - 集成式決策樹，能捕捉非線性與特徵交互關係，並提供特徵重要性解釋

模型效能比較

```
==== Logistic Regression ====
```

```
Accuracy: 0.8750
```

```
Precision: 0.8798
```

```
Recall: 0.8743
```

```
F1-score: 0.8751
```

```
AUC-ROC: 0.9790
```

```
Confusion Matrix:
```

```
[[54 12  0]
```

```
 [ 4 57  5]
```

```
 [ 0  4 64]]
```

```
==== SVM ====
```

```
Accuracy: 0.8750
```

```
Precision: 0.8843
```

```
Recall: 0.8743
```

```
F1-score: 0.8753
```

```
AUC-ROC: 0.9772
```

```
Confusion Matrix:
```

```
[[52 14  0]
```

```
 [ 3 59  4]
```

```
 [ 0  4 64]]
```

```
==== Random Forest ====
```

```
Accuracy: 0.8850
```

```
Precision: 0.8873
```

```
Recall: 0.8844
```

```
F1-score: 0.8851
```

```
AUC-ROC: 0.9712
```

```
Confusion Matrix:
```

```
[[56 10  0]
```

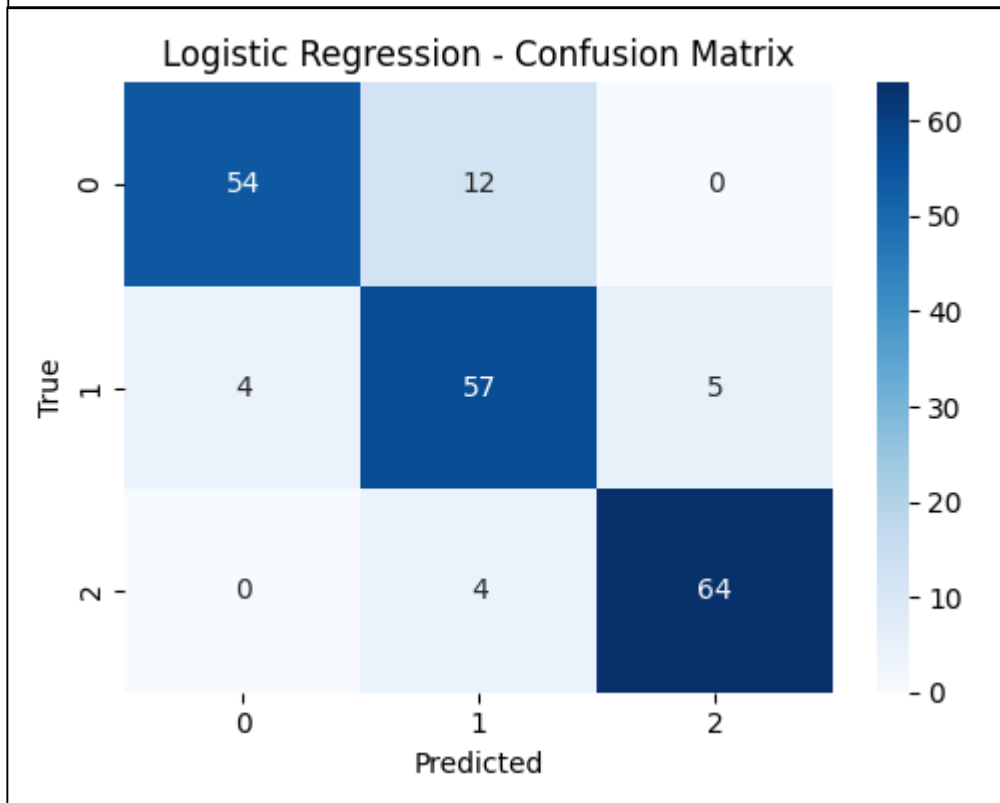
```
 [ 5 57  4]
```

```
 [ 0  4 64]]
```

- Accuracy：預測正確率
- Precision：預測正確的陽性樣本比例（每一類別分開計算）
- Recall：所有實際陽性中預測正確的比例
- F1-score Precision 與 Recall 的調和平均數
- AUC-ROC：針對多分類，採 **One-vs-Rest** 方法計算平均 AUC
- Confusion Matrix：顯示每一類別的預測正確與錯誤分布

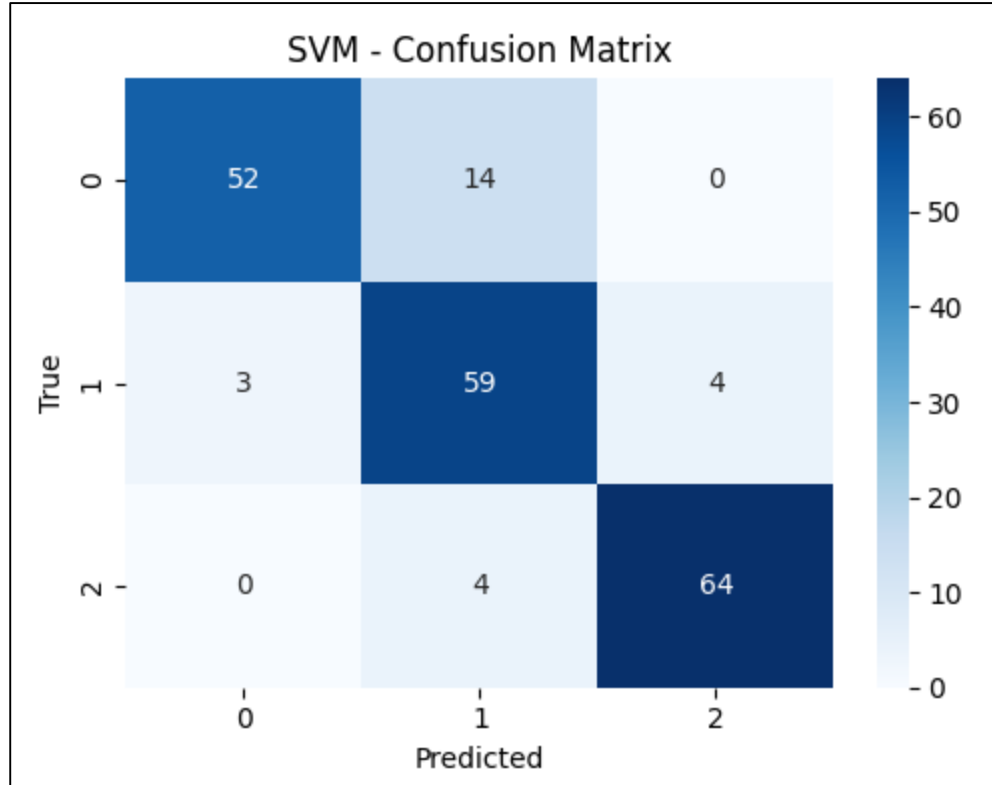
可視化分析

混淆矩陣（用來比較模型預測結果與真實標籤的表格，每個欄位代表預測類別，每個列代表真實類別。）



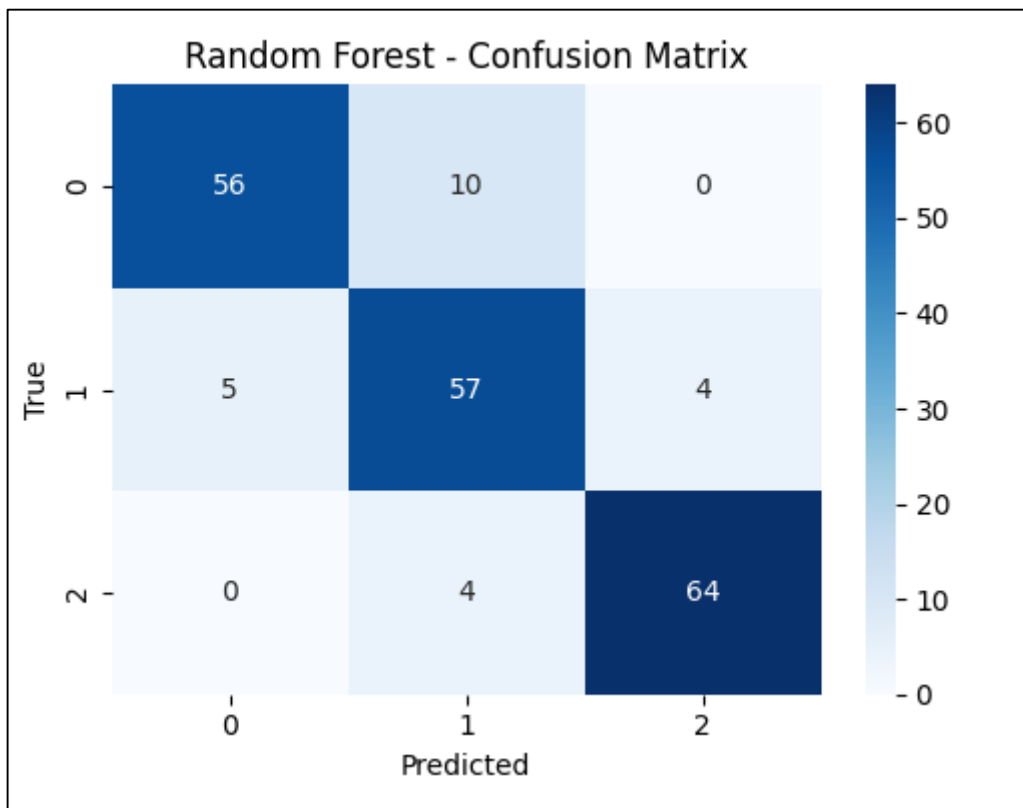
1. Logistic Regression

- True Class 0：共 66 筆樣本
 - 正確預測為 0：54
 - 錯誤預測為 1：12
- True Class 1：共 66 筆樣本
 - 正確預測為 1：57
 - 錯誤預測為 0：4，預測為 2：5
- True Class 2：共 68 筆樣本
 - 正確預測為 2：64
 - 錯誤預測為 1：4
- 表現亮點：
 - 類別 2 的預測非常精準（64/68 筆準確，誤判僅 4 筆）
- 潛在問題：
 - 類別 0 有 12 筆被錯預測為 1，明顯偏差



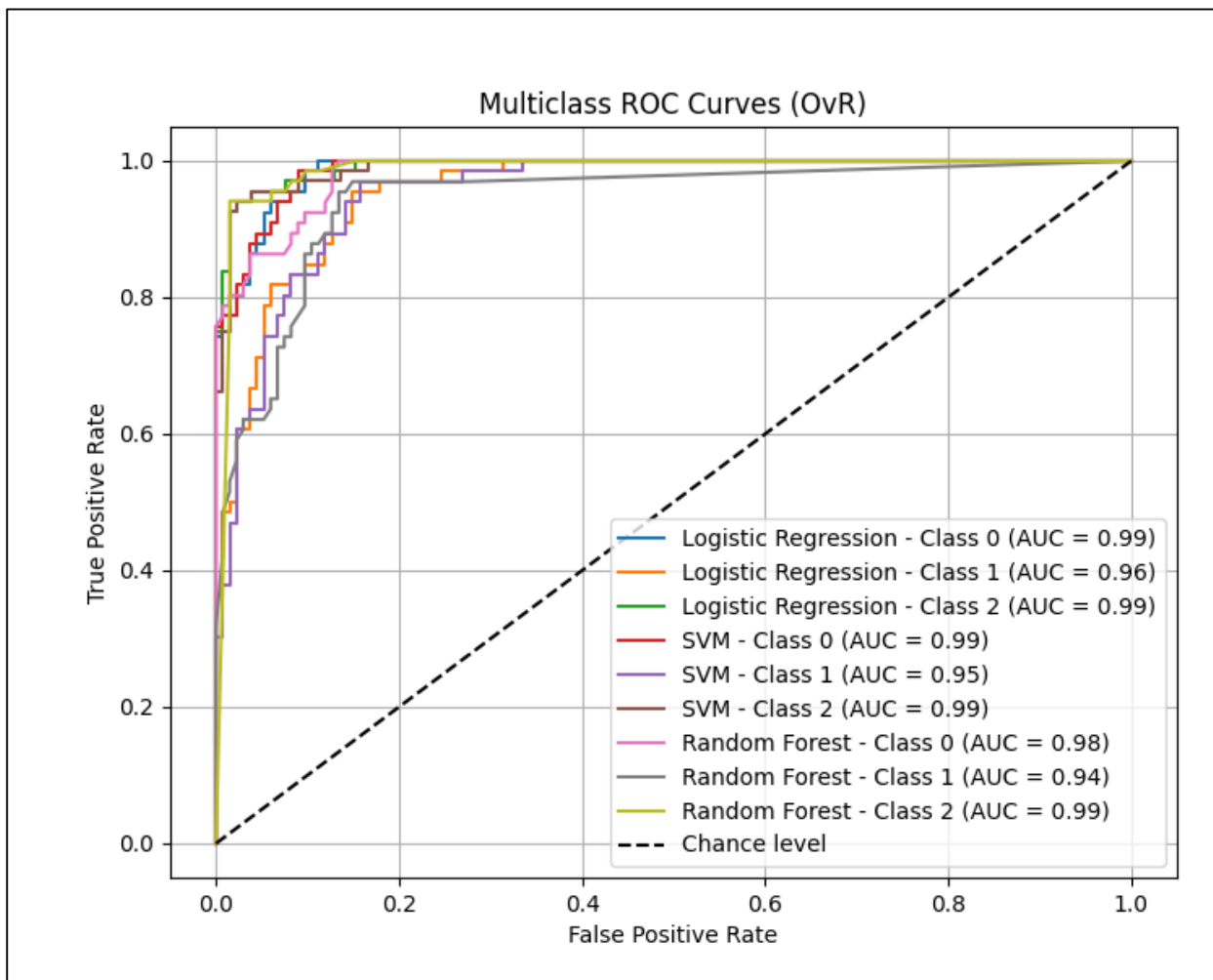
2. SVM

- True Class 0：共 66 筆樣本
 - 正確預測為 0：52
 - 錯誤預測為 1：14
- True Class 1：共 66 筆樣本
 - 正確預測為 1：59
 - 錯誤預測為 0：3，預測為 2：4
- True Class 2：共 68 筆樣本
 - 正確預測為 2：64
 - 錯誤預測為 1：4
- 表現亮點：
 - 類別 2 預測表現穩定（僅 4 筆誤判）
- 潛在問題：
 - 類別 0 有 14 筆誤判為 1，類別 0 與 1 間界線可能不清晰，出現互相混淆的情況



3. Random Forest

- True Class 0：共 66 筆樣本
 - 正確預測為 0：56
 - 錯誤預測為 1：10
- True Class 1：共 66 筆樣本
 - 正確預測為 1：57
 - 錯誤預測為 0：5，預測為 2：4
- True Class 2：共 68 筆樣本
 - 正確預測為 2：64
 - 錯誤預測為 1：4
- 表現亮點：
 - 類別 2 一致穩定。類別 0 預測表現提升，錯誤減少至 10 筆
- 潛在問題：
 - 類別 1 仍有被錯預測為 0 與 2 的混淆現象（共 9 筆）



ROC 曲線（OvR）

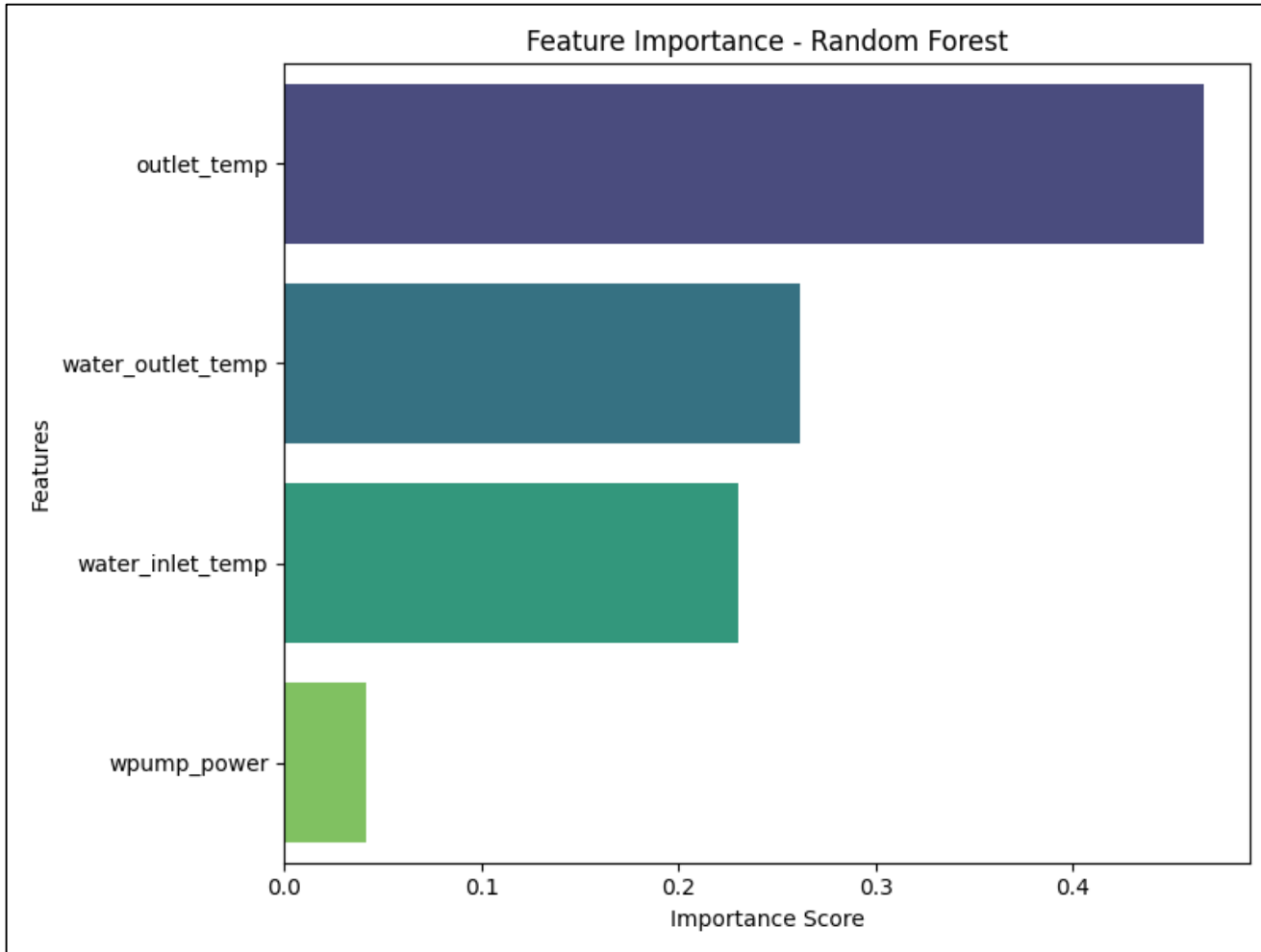
橫軸（X）：False Positive Rate（假陽性率）

縱軸（Y）：True Positive Rate（真正率）

虛線：Chance level（隨機預測參考線）

解讀與觀察：

- 整體而言，三種模型（Logistic Regression, SVM, Random Forest）在 Class 0 和 Class 2 的辨識表現都非常優異（ $AUC \geq 0.98$ ）。
- Class 1 的預測相對較弱，尤其是 Random Forest（ $AUC = 0.94$ ）和 SVM（ $AUC = 0.95$ ）
- Logistic Regression 在三個類別上表現最穩定。所有模型的曲線幾乎都貼近左上角，說明分類效果極佳。



特徵重要性圖

解讀與觀察：

- **outlet_temp** 是最重要的變數，遠高於其他特徵，這表示它對預測結果的影響最大。
- **wpump_power** 幾乎沒有貢獻，可能可以考慮剔除以簡化模型。
- **water_outlet_temp** 與 **water_inlet_temp** 也具有中度貢獻，可能和系統熱力性能高度相關。

結論

- 最佳模型：Random Forest
 - 多數類別準確分類：準確率與召回率皆為 85% 左右，表現穩定
 - 能處理非線性與特徵交互：較其他模型能捕捉複雜的變數關係
 - 提供特徵重要性分析能力：有助於了解預測背後的物理邏輯
 - 抗雜訊與過擬合能力強：使用多棵樹降低過擬合機率

工業價值

- 預測與控制能力：
 - 可提前預測熱交換系統的出口溫度，協助操作者進行主動調整，提升能源使用效率與產品品質穩定性。
- 異常偵測與預警機制：
 - 透過 **Random Forest** 的分類結果，快速辨識異常工作狀態，避免設備過熱、冷卻效率不佳等問題，提高設備使用壽命。
- 參數敏感性分析：
 - 特徵重要性排序（**Feature Importance**）揭示 **wpump_power**、**water_outlet_temp** 等對系統輸出的主導影響，可作為操作參數優化與節能調控的依據。

實用建議

- 將模型嵌入現場 IoT 系統中，進行即時預測與警示。
- 結合模型輸出與操作指引，建立智能化調控決策系統。
- 擴充模型以支援多機台或不同操作條件的泛化應用。

模型限制

- Ridge Regression 的線性假設限制：
 - Ridge Regression 雖具備正則化能力，可避免過度擬合，但本質上仍是線性模型，對於輸出變化與輸入變數間存在非線性關係的情況，預測精度有限。
- 過擬合風險：
 - 若訓練資料樣本數不多、或資料含雜訊，Random Forest 容易在訓練集上表現良好、但在實際應用時準確率下降。
- 特徵維度有限：
 - 僅考慮溫度與泵浦功率等參數，尚未納入壓力、振動、流量等多源感測資訊，限制模型精度。
- 缺乏時間考量（靜態模型）：
 - 所使用模型為靜態學習法，無法考慮時間序列變化（如逐步升溫、機台老化），限制其在動態系統中的應用。

應用限制

- 感測器誤差與資料異常：
 - 工業場域中感測器可能出現飄移或讀值錯誤，會直接影響模型輸入品質，導致預測或分類錯誤。
- 即時性應用需運算效能支援：
 - 雖然 **Random Forest** 相較於深度學習演算法較輕量，但若要進行即時監控或邊緣運算部署，仍需考量模型複雜度與計算資源限制。

未來延伸方向

- 整合時間序列模型：
 - 探索結合 LSTM、GRU 等 RNN 架構，處理連續狀態變化與多步預測問題。
- 感測器資料擴增與融合：
 - 將更多物理量（如振動、聲音、壓差）納入模型訓練，提升系統整體可判別性。
- 引入 SHAP / LIME 等可解釋 AI 技術：
 - 強化模型透明性與可解釋性，讓現場工程師能明確理解模型推論邏輯。

THANK YOU