模型診斷與驗證

├─ 檢查殘差常態性、變異數齊一性

├─ 檢查多重共線性 (VIF)

├─ 若為預測 → 做交叉驗證

└─ 若為解釋 → 檢查 η², R², Adj-R²

**建議流程**

1. 檢查 X.isna().sum() → 確定沒有 NaN / inf
2. 檢查每個 dummy 的分佈（例如 value\_counts()）→ 找出幾乎全 0 的欄位
3. 跑 VIF，挑掉 VIF > 10 的變數，再重新建模

import numpy as np

import pandas as pd

from statsmodels.stats.outliers\_influence import variance\_inflation\_factor

from statsmodels.tools.tools import add\_constant

def clean\_and\_check\_X(X, vif\_thresh=10, sparse\_thresh=0.99):

"""

自動化清理自變數矩陣 X

1. 移除 NaN / inf

2. 移除 dummy 幾乎全 0 或全 1 的稀疏變數

3. 移除 VIF > vif\_thresh 的變數

參數:

X : DataFrame

vif\_thresh : float, 預設 10，超過代表共線性過高

sparse\_thresh : float, 預設 0.99，若 99% 都是同一值則視為稀疏

回傳:

X\_clean : 清理後的 DataFrame

report : 各步驟清理紀錄

"""

report = {}

# Step 1: NaN / inf

n\_before = X.shape[1]

X = X.replace([np.inf, -np.inf], np.nan).dropna(axis=1, how="any")

report["移除 NaN/inf 變數數量"] = n\_before - X.shape[1]

# Step 2: 稀疏 dummy

n\_before = X.shape[1]

sparse\_cols = [col for col in X.columns

if (X[col].value\_counts(normalize=True).max() > sparse\_thresh)]

X = X.drop(columns=sparse\_cols)

report["移除稀疏變數數量"] = len(sparse\_cols)

# Step 3: VIF

n\_before = X.shape[1]

X\_const = add\_constant(X)

vif = pd.DataFrame()

vif["變數"] = X\_const.columns

vif["VIF"] = [variance\_inflation\_factor(X\_const.values, i)

for i in range(X\_const.shape[1])]

vif = vif[vif["變數"] != "const"]

high\_vif = vif[vif["VIF"] > vif\_thresh]["變數"].tolist()

X = X.drop(columns=high\_vif, errors="ignore")

report["移除高 VIF 變數數量"] = len(high\_vif)

return X, report, vif

X\_clean, report, vif\_result = clean\_and\_check\_X(X, vif\_thresh=10, sparse\_thresh=0.99)

print("=== 清理報告 ===")

print(report)

print("\n=== VIF 結果 (前 10) ===")

print(vif\_result.sort\_values("VIF", ascending=False).head(10))